# Disentangling and machine learning the many-fermion problem

Peter Bröcker

Dissertation

# Disentangling and machine learning the many-fermion problem

## Inaugural-Dissertation

zur
Erlangung des Doktorgrades
der Mathematisch-Naturwissenschaftlichen Fakultät der Universität
zu Köln
vorgelegt von

**Peter Bröcker geb. Boertz**

aus Münster

Köln, 2018

| | |
|---|---|
| **Berichterstatter** | Prof. Dr. Simon Trebst |
| | Prof. Dr. David Gross |

Tag der mündlichen Prüfung: 16.4.2018

# Contents

# Introduction

One of the most intriguing areas of physics is the study of strongly correlated many-body systems. In these kinds of systems, a large number of particles interacts via interactions that are strong enough to play a major role in determining the properties of the system, which is particularly interesting when multiple interaction terms compete and cannot be satisfied simultaneously. Examples of phases they realize range from simple magnetism to puzzling phenomena such as superconductivity or topological order. Of special interest in condensed matter are fermionic models which have an extra layer of complexity added by the possibly intricate sign structure of the wave function.

However, exactly solvable models that exhibit the sought-after phenomena are scarce and Hamiltonians that are supposed to model experiments often elude analytical solutions precisely due to the strong interactions. Numerical methods are thus essential to gain insight into these systems and provide an important link between theoretical models and real world experiments. There are many different methods that in principle allow solving a problem numerically exact, most prominently exact diagonalization, tensor network based methods such as DMRG and (quantum) Monte Carlo. Exact diagonalization is certainly the most straightforward technique which allows measuring any observable of interest but its application to interacting models is severely limited by the exponential growth of the Hilbert space with the system size which only allows studying rather small systems. Tensor network methods excel in the study of gapped, one-dimensional systems and while they have shown some impressive results for two-dimensional systems, they need a lot of fine tuning and careful analysis to be considered reliable in this very important domain. Quantum Monte Carlo, on the other hand, is in principle neither limited by the size of the Hilbert space nor by the strength of the interactions and has thus been a tremendously important tool for studying condensed matter systems which is why it is the focus of this thesis.

It is necessary to continuously advance existing and develop new techniques to keep up with theoretical as well as with experimental progress. In this thesis, new methods to identify and characterize conventional and novel phases of matter within quantum Monte Carlo are developed and tested on archetypical models of condensed matter.

In the first part of this thesis, a method to study entanglement properties of strongly interacting fermionic models in quantum Monte Carlo is presented. Measurements of this kind are needed because some characteristics of a system remain hidden from conventional approaches based on the calculation of correlation functions. Prime examples are systems that realize so-called topological order entailing long-range entanglement that can be positively diagnosed using entanglement techniques. Realizing such measurements in a concrete algorithm is not straightforward for fermions. A large section of this part is thus devoted to developing a solution to the problem of implementing entanglement measurements and benchmarking the results against known data.

The second part is concerned with a truly novel and recent approach to the many-body problem, namely the symbiosis of machine learning and quantum Monte Carlo techniques. Out of the many different possibilities to combine the two, this part is concerned with exploring ways in which machine learning can help to effortlessly explore phase diagrams of hitherto unknown Hamiltonians.

Despite the power and versatility of quantum Monte Carlo techniques, it does suffer from one significant weakness called the fermion sign problem. At its core related to the peculiar exchange statistics of fermions, it is in principle merely a technical problem of the Monte Carlo procedure but one that actually prohibits the use of Monte Carlo methods for a large number of interesting problems. The sign problem has been known for a long time but remains unsolved to this day. Thus, it is time to approach the problem from a different perspective with the hope of learning something new that can either help to better understand the problem itself or the properties of the models affected by it. Both of the two major research areas, entanglement measures and machine learning, are capable of providing such new perspectives and the sign problem will be revisited in more detail in the context of each of them.

# Kurzzusammenfassung

Stark korrelierte Vielteilchensysteme gehören zu den faszinierendsten Bereichen der Physik. Für sie charakteristisch ist, dass eine große Anzahl Teilchen miteinander wechselwirkt und diese Wechselwirkung so stark ist, dass sie die physikalischen Eigenschaften entscheidend beeinflusst. Besonders interessant sind diese Systeme, wenn mehrere Wechselwirkungen konkurrieren und deren Energien nicht gleichzeitig minimiert werden können. Beispiele von Phasen, die in wechselwirkenden Systemen realisiert werden, reichen von einfachen magnetischen Phasen bis hin zu noch immer nicht vollständig verstandenen Phasen wie Supraleitung oder topologisch geordneten Phasen. Von besonderem Interesse in der kondensierten Materie sind fermionische Modelle, die aufgrund der möglicherweise komplizierten Vorzeichenstruktur der Wellenfunktion eine weitere Komplexititätsebene aufweisen.

Exakt lösbare Modelle sind allerdings schwer zu finden und Hamiltonians, die experimentell untersuchte Materialien modellieren entziehen sich oft analytischen Studien genau wegen der starken Wechselwirkung, die sie erst interessant macht. Numerische Methoden sind deswegen essentiell, um Erkenntnisse über diese Art von Systemen zu gewinnen und stellen eine wichtige Verbindung zwischen theoretischen Modellen und tatsächlichen Experimenten dar. Es gibt eine Reihe von verschiedenen Methoden, die prinzipiell erlauben ein Vielteilchenproblem numerisch exakt zu lösen, wie zum Beispiel exakte Diagonalisierung, Tensornetzwerkmethoden wie DMRG oder (Quanten) Monte Carlo. Exakte Diagonalisierung wäre sicherlich die Methode der Wahl, da sie erlaubt alle gewünschten Observablen zu bestimmt. Allerdings ist ihre Anwendung deutlich limitiert, weil mit dem Hilbertraum auch der numerische Aufwand in der Systemgröße exponentiell wächst und man daher nur relativ kleine Systeme untersuchen kann. Tensornetzwerkmethoden wie DMRG sind hervorragend geeignet, um eindimensionale, gegappte Systeme zu studieren. Obwohl es einige vielversprechende Resultate für zweidimensionale Systeme gibt, benötigt diese Art von Simulationen weiterhin eine genaue Überwachung und viel Feintuning, um die Konvergenz zum korrekten Ergebnis zu gewährleisten. Quanten Monte Carlo Algorithmen haben den Vorteil, dass sie im Prinzip weder durch die Größe des Hilbertraums, noch durch die Stärke der Wechselwirkung limitiert sind, weshalb sie sich zu einem der wichtigsten Werkzeuge in der kondensierten Materie entwickelt haben und auch Hauptforschungsgegenstand dieser Arbeit sind.

Es ist überaus wichtig fortwährend bestehende Techniken zu verbessern sowie neue Ansätze zu entwickeln, um mit den aktuellen theoretischen und experimentellen Entwicklungen Schritt zu halten. Dies wird im Rahmen dieser Arbeit getan, indem neue Methoden entwickelt werden um konventionelle und neue, exotische Phasen mit Quanten Monte Carlo zu identifizieren und zu charakterisieren, was anhand von archetypischen Beispielen demonstriert wird.

Im ersten Teil dieser Arbeit wird dabei eine Methode präsentiert, mithilfe derer man die Verschränkungseigenschaften von wechselwirkenden fermionischen Modellen studieren kann. Messungen dieser Art sind notwendig, weil einige Eigenschaften eines Systems sich nicht über konventionelle Messungen mit Korrelationsfunktionen bestimmen lassen. Eine Paradebeispiel ist die sogenannte topologische Ordnung, die mit langreichweitiger Verschränkung einher geht und durch Messung der Verschränkungseigenschaften einwandfrei diagnostiziert werden kann. Wie man einen Algorithmus fermionisches System implementiert, ist allerdings nicht offensichtlich. Ein großer Teil des entsprechenden Kapitels ist der Entwicklung einer Lösung dieses Problems sowie dem Vergleich mit bekannten Referenzdaten gewidmet.

Der zweite Teil befasst sich mit einem neuartigen und erst kürzlich entwickelten Ansatz, nämlich der Symbiose von Machine Learning und Quanten Monte Carlo Techniken. Aus der großen Anzahl möglicher Varianten diese beiden Themen zu kombinieren, wurde für dieses Kapitel die Frage gewählt, inwiefern Machine Learning helfen kann das Phasendiagramm eines unbekannten Hamiltonians mit möglichst geringem Aufwand zu bestimmen.

Trotz der vielfältigen Anwendbarkeit von Quanten Monte Carlo Techniken besitzen sie eine große Schwäche, nämlich das sogenannte Vorzeichenproblem. Der Ursprung des Vorzeichenproblems liegt in der speziellen Austauschstatistik von Fermionen und ist im Grunde nur ein technisches Problem des Monte Carlo Verfahrens. Allerdings sind die Auswirkungen so signifikant, dass es die Anwendung von Monte Carlo Methoden für eine große Anzahl interessanter Probleme verhindert. Obwohl das Vorzeichenproblem bereits seit Jahrzehnten bekannt ist, wurde bislang noch keine vollständige Lösung gefunden. Es ist deshalb an der Zeit das Problem von einer anderen Perspektive zu betrachten – in der Hoffnung so zu neuen Erkenntnissen zu kommen, die helfen entweder etwas über das Vorzeichenproblem selbst oder zumindest über die betroffenen Modelle und Algorithmen zu lernen. Beide der oben kurz dargestellten Fragestellungen erlauben es eine solch neue Perspektive auf das Vorzeichenproblem zu gewinnen, weshalb es im jeweiligen thematischen Kontext detailliert diskutiert wird.

# Part I.

# Monte Carlo methods for the many-body problem

# 1. The Monte Carlo method

Monte Carlo methods comprise a class of algorithms which use random numbers and statistical sampling to solve problems that are in principle deterministic in nature. It was originally conceived by Stanislaw Ulam and put to practice in collaboration with John von Neumann at Los Alamos laboratories in the 1940s [1]. Since then, it has found many applications in diverse contexts from virtually all of the natural sciences to finance and even some of the humanities. Monte Carlo methods often excel in many-dimensional problems that cannot be factored into simple products of lower dimensional ones. An obvious drawback is that because a problem is solved by statistical sampling, a statistical error in introduced which can, however, be systematically reduced.

This chapter briefly explores the fundamentals of Monte Carlo simulations and how they are employed in the context of condensed matter physics. Detailed explanations of the particular Monte Carlo algorithms used in this thesis can be found in chapter 2 and chapter 3. Even more information about the Monte Carlo technique is available in the respective literature [2, 3, 4, 5] on which this chapter relies as well.

## 1.1. Fundamentals of the Monte Carlo method in physics

The Monte Carlo algorithms used in this thesis aim at drawing random samples from a given probability distribution whose normalization is unknown. These random samples are then used to calculate other quantities, such as the value of an integral (whose integrand defines the probability distribution) or the expectation values of physical observables, when the probability distribution is given by a partition function or the overlap of two trial wavefunctions. In practice, this is achieved by setting up a Markov Chain on the configuration space of possible states that converges in distribution to the desired probability distribution, which is why this procedure is also often referred to as Markov Chain Monte Carlo (MCMC).

### 1.1.1. Markov chains

Markov chains are stochastic models that generate sequences of events, called states and denoted by $\sigma$, with the important property that whatever state is generated only depends on the current state and is independent of any earlier states. This is precisely what is done when sampling the configuration space of a physical model. New configurations

$\sigma'$ are iteratively proposed and, depending only on the currently active configuration $\sigma$, accepted or rejected as the next state of the Markov chain. The decision to accept or reject a proposed state is made based on the elements of a transition matrix $W_{\sigma,\sigma'}$ that enumerates all the probabilities of moving from one configuration to any other possible configuration.

In order for the Markov chain to properly model the desired probability distribution, it can be shown that three conditions have to be imposed on the transition probability from one state $\sigma$ to another $\sigma'$:

1. The probability of moving from a configuration $\sigma$ to a configuration $\sigma'$ is positive.

   $$W_{\sigma,\sigma'} \geq 0 \tag{1.1}$$

2. It has to be possible to move from any given state $\sigma$ to any other possible state $\sigma'$ in a finite number of steps $n$.

   $$\left(W^n\right)_{\sigma,\sigma'} \neq 0 \tag{1.2}$$

   If this condition is fulfilled, the process is called ergodic.

3. The sum probabilities of moving from a configuration $\sigma$ to any configuration add up to 1.

   $$\sum_{\sigma} W_{\sigma,\sigma'} = 1 \tag{1.3}$$

4. The probability $P_{\sigma'}$ to realize a state $\sigma'$ in the target probability distribution $P$ is obtained when all transition probabilities to $\sigma'$ from a state $\sigma$ are multiplied by the probability to reach the respective initial state $P_{\sigma}$.

   $$\sum_{\sigma} P_{\sigma} W_{\sigma,\sigma'} = P_{\sigma'} \tag{1.4}$$

   This condition can also be written as $\mathbf{P} \cdot \mathbf{W} = \mathbf{P}$ which emphasizes that the target probability distribution $\mathbf{P}$ is an eigenvector with eigenvalue 1 of the transition probability matrix $W$. Alternatively, the condition can also be stated as:

   $$P_{\sigma} W_{\sigma,\sigma'} = P_{\sigma'} W_{\sigma',\sigma} \tag{1.5}$$

   In this form, the condition is also known as detailed balance. It can be shown to be equivalent to the previous form by summing over $\sigma$ on both sides and evoking the second condition.

The Markov chain represents the probability distribution in the sense that the frequency with which a state is visited is proportional to the probability of realizing it in the actual distribution.

### 1.1.2. Moving along Markov chains

The rules laid out above make no precise statement about how to propose new states and decide whether they should be added to the Markov chain or not. This is done by employing the Metropolis-Hastings algorithm [6, 7] which states that the decision should be made by considering the ratio $r$ of the weights of the proposed configuration and the current configuration and compare this to a random $p$. The proposed configuration is then accepted if

$$p \leq \min(1, r) . \tag{1.6}$$

In practice, the more interesting question is how a new configuration $\sigma$ is proposed. Broadly speaking, there are two ways of doing so: One may either propose a small perturbation, often a local change, to the configuration such as flipping a single spin. Another way is to attempt a non-local change that affects many degrees of freedom at the same time. The latter variant is typically the more effective as it allows exploring a "larger" area of the configuration space in the same time because the configurations are less correlated. Such algorithms are also called cluster algorithms. They are also a highly effective in overcoming a problem called critical slowing [8] which appears in the vicinity of second order phase transitions. In such a setting, the correlation length diverges and it typically becomes highly unlikely to change a sufficient number of individual degrees of freedom using only local updates to achieve a significant change in the overall configuration. As a result, the Monte Carlo simulation has to be run for longer and longer times to generate independent samples. Unfortunately, cluster algorithms are not available for all models and they usually have to be constructed with a specific model or at least Monte Carlo flavor in mind [9].

## 1.2. Analysis of Monte Carlo data

### 1.2.1. Calculating mean values and their error bars

In the course of a Monte Carlo calculation, a sequence of $N$ configurations $\{c_1, \ldots c_N\}$ is generated which can be used to calculate the values of observables of interest for each configuration. If it is possible to evaluate the full probability distribution $p(c)$, the expectation value of an observable $X$ is estimated via

$$\langle X \rangle = \sum_{c \in \mathcal{C}} X(c) p(c) . \tag{1.7}$$

In a Monte Carlo simulation on the other hand, this true mean is approximated by mean values obtained from averaging observable values along the Markov chain:

$$\bar{x} = \frac{1}{N} \sum_{i}^{N} x_i . \tag{1.8}$$

As was already mentioned previously, the role of the probability of a configuration $p(c)$ is played by the frequency that a particular configuration is visited relative to the other configurations. It is important to distinguish the true but unknown expectation value $\langle X \rangle$ and the mean value $\bar{x}$ obtained from a single Monte Carlo run. To assess the quality of a Monte Carlo simulation, it is necessary to quantify the distance between the true expectation value and the Monte Carlo mean value. At first sight he standard deviation $\sigma$, defined as the square root of the variance $\sigma_X^2$

$$\sigma_X^2 := \sum_{c \in \mathcal{C}} (X(c) - \langle X(c) \rangle)^2 p(c) , \tag{1.9}$$

appears to be a suitable candidate. However, the variance only makes a statement about the spread of $X$ relative to its mean which does not decrease as one increases the number of samples $N$ and does not indicate how close the calculated mean is from the true mean. Another possibility is to consider the variance of multiple instances of the mean value $\sigma_{\overline{x}}^2$, given by

$$\begin{aligned}\sigma_{\overline{x}}^2 &= \frac{1}{N^2}\sum_{i,j=1}^{N}\langle x_i x_j\rangle - \frac{1}{N^2}\sum_{i,j=1}^{N}\langle x_i\rangle\langle x_j\rangle \\ &= \frac{1}{N^2}\sum_{i=1}^{N}\left(\langle x_i^2\rangle - \langle x_i\rangle^2\right) + \frac{1}{N^2}\sum_{i\neq j}^{N}\left(\langle x_i x_j\rangle - \langle x_i\rangle\langle x_j\rangle\right) . \qquad (1.10)\end{aligned}$$

Assuming at first that the measurements $x_i$ and $x_j$ are uncorrelated, the second term in (1.10) vanishes, which leads to

$$\sigma_{\overline{x}}^2 = \frac{1}{N^2}\sum_{i=1}^{N}\left(\langle x_i^2\rangle - \langle x_i\rangle^2\right) = \frac{\sigma_X^2}{N} . \qquad (1.11)$$

The variance of the means of the Monte Carlo measurements is proportional to the variance of the underlying probability distribution but inversely proportional to the number of measurements in the Markov chain. It thus fulfills the requirement of becoming smaller with increasing number of Monte Carlo samples. Another aspect that qualifies this quantity for estimating the Monte Carlo error, is that the probability distribution of the means $\overline{x}$ is fully defined by its mean and its variance. This fact is rooted in the central limit theorem, which states that the distribution of the properly normalized sum of independent identically distributed random variables tends towards a normal distribution regardless of the underlying distribution of the original random variables. The conditions for the application of the central limit theorem are fulfilled by the Monte Carlo procedure if one splits the sequence of measurements into several subsequences, called bins, that each have their own mean value. For sufficiently many subsequences, each of sufficient length, the central limit theorem applies and the distribution is solely governed by the mean value of the means $\langle\overline{x}\rangle$ and the variance $\sigma_{\overline{x}}^2$. By splitting a given sequence of measurements, one is not required to average over several distinct Markov chains. The subsequences can be used to calculate another estimate for the error via a resampling technique called jackknife [4, 5].

### 1.2.2. Autocorrelation

In the derivation of (1.11), off-diagonal terms were ignored on the assumption that the samples are uncorrelated. In general, this is, however, not true and the term may not be discarded. The expression can be simplified by exchanging the indices $i$ and $j$ and exploiting the supposed invariance in simulation time

$$\sigma_{\overline{x}}^2 = \frac{1}{N}\sigma_X^2\left[1 + 2\sum_{k=1}^{N}A(k)\left(1 - \frac{k}{N}\right)\right] , \qquad (1.12)$$

with the normalized autocorrelation function

$$A(k) = \frac{\langle x_1 x_k\rangle - \langle x_1\rangle\langle x_k\rangle}{\sigma_X^2} . \qquad (1.13)$$

The resulting correction factor to (1.11)

$$\tau_{X,\mathrm{int}} \equiv 1 + 2\sum_{k=1}^{N} A(k)\left(1 - \frac{k}{N}\right) \tag{1.14}$$

is called the integrated autocorrelation time. It enlarges the variance and thus the error of the observable of interest. Because consecutive configurations are correlated and thus not independent, they cannot reduce the error as much as truly independent samples would.

In principle, the autocorrelation time can be estimated from the above equation. Another way to determine the autocorrelation time is to use the fact that in the limit of $k \to \infty$ the autocorrelation function is expected to decay exponentially

$$A(k) \propto \exp\left(-\frac{k}{\tau_{X,exp}}\right), \tag{1.15}$$

with the exponential autocorrelation time $\tau_{X,exp}$ setting the time scale of the decay. After performing $\tau_{X,exp}$ Monte Carlo steps, one can speak about truly independent samples amenable to ordinary statistical analysis. Only if the subsequent error analysis is performed with these truly independent samples is it possible to obtain sensible error bars.

### 1.2.3. Functions of observables

Often, one is interested in combinations of more fundamental observables such as Binder cumulants of the form

$$g(A,B) \equiv g(X^2,X^4) \propto \frac{\langle X^4\rangle}{2\langle X^2\rangle} \tag{1.16}$$

Such combinations are more difficult to evaluate correctly because the expressions in the numerator and the denominator are calculated from the same time series and are thus correlated. Working out the correct formula for the error in a similar fashion to the previous calculation is a tedious task involving the covariance and derivatives of $g$ [5]. Instead, it is possible to use the aforementioned jackknife technique even for a combination of observables [4, 5] which allows for an easy estimation of the error.

# 2. Determinant Quantum Monte Carlo

The Determinant Quantum Monte Carlo (DQMC) method was first conceived to study field theories with bosonic and fermionic degrees of freedom at finite temperatures [10, 11]. As will be shown, the origin of the name is rooted in the fact an interacting fermion system is transformed into a non-interacting system such that fermionic degrees of freedom can be integrated out from the partition sum ultimately generating the eponymous determinant. Following a slight modification, the algorithm can also be turned into a projective method for the study of ground state properties [12, 13]. The details of both variants are described in this chapter, which first discusses the basic idea of how the algorithm is set up and how updates are performed. Next, the Hubbard-Stratonovich transformation used to decouple the system is discussed, including some numerical aspects. The remaining part of the chapter is devoted to the numerical implementation of the algorithm. The description of the algorithm in this chapter is based on a number of general introductions to quantum Monte Carlo algorithms and DQMC in particular [14, 15, 5].

## 2.1. Theoretical formulation

For the sake of concreteness the well known one band Hubbard model [16]

$$H = -\sum_{i,j\in\mathcal{L},\,\sigma=\pm1} t_{ij}\, c^{\dagger}_{i,\sigma} c_{j,\sigma} + U\sum_{i} n_{i,\uparrow} n_{j,\downarrow} \tag{2.1}$$

is used as a model system for the discussion of the algorithm. The Hubbard model describes spinful fermions on an arbitrary lattice $\mathcal{L}$ that are allowed to hop between sites $i$ and $j$ with an amplitude $t_{ij}$ and which interact with an interaction strength of $U$ when present on the same site. The DQMC method has played a big role in elucidating many properties of this model and its many variants, which is why it is well suited to be used in the following discussion.

In the finite temperature simulation of DQMC, the goal is to sample from the partition function

$$Z = \mathrm{Tr}\, e^{-\beta\hat{H}} \tag{2.2}$$

while the ground state algorithm employs a projective procedure to sample from the ground state wave function:

$$|\psi\rangle = e^{-\theta\hat{H}}\,|\psi_T\rangle\ . \tag{2.3}$$

In the latter approach, a large number of exponentials of the Hamiltonian is applied to a trial wave function which projects out all excited states and leaves only the ground state.

Technically, the only condition a trial wave function must fulfill is that it is non-orthogonal to the true ground state wave function. However, the details of how the trial wave function is set up may greatly influence certain properties of the simulation, in particular how long it takes to converge [17, 18].

Both variants of the DQMC algorithm have in common that the exponential of the Hamiltonian operator is a central object. Regarding the concrete implementation, this means that some of the core concepts can be applied to both variants and thus the following discussion is valid for both approaches unless indicated otherwise. The prefactor of the Hamiltonian operator in the exponential represents the inverse temperature in the finite temperature approach and the projection time in the ground state approach, respectively. In the following, it will be denoted as $\beta$ and referred to as imaginary time in both cases.

The exponential cannot be readily evaluated when the Hamiltonian is not quadratic as is the case for the Hubbard Hamiltonian (2.1) because of the quartic interaction term. To tackle this problem, the first step is to perform a Trotter-Suzuki decomposition, discretizing imaginary time into

$$N_\tau = \beta/\Delta\tau \tag{2.4}$$

time steps of width $\Delta\tau$. The Hubbard Hamiltonian is of the form $\hat{H} = \hat{K} + \hat{V}$ with the quadratic hopping part $\hat{K}$ and the quartic interaction term $\hat{V}$, respectively. Applying the Baker-Campbell-Hausdorff formula, the exponential of the sum is split into a product of exponentials keeping only the lowest order term:

$$e^{-\Delta\tau\hat{H}} = e^{-\Delta\tau\hat{K}}e^{-\Delta\tau\hat{V}} + \mathcal{O}(\Delta\tau^2)\,. \tag{2.5}$$

Ignoring higher order terms introduces an error which can be considered negligible if $\Delta\tau$ is chosen sufficiently small. What exactly constitutes sufficiently small depends on the model and the choice of parameters of the Hamiltonian but can be found out by performing simulations on small lattices for several values of $\Delta\tau$ and observing the convergence behavior of the observables of interest.

In a second step, the quartic term is transformed into a quadratic term by virtue of a Hubbard-Stratonovich transformation [19, 20] at the cost of introducing an auxiliary field. This auxiliary field is of dimension $(N_i, N_\tau)$ where $N_i$ is the number of quartic terms in the interaction operator and thus the transformed interaction term $\hat{V}(s,\tau)$ depends on the auxiliary field configuration $s$ and the imaginary time slice $\tau$. Both the kinetic operator $\hat{K}$ and the decoupled interaction operator $\hat{V}(s,\tau)$ are now quartic and can therefore be represented as matrices in the single particle basis which are denoted by $\mathsf{K}$ and $\mathsf{V}(s,\tau)$, respectively:

$$e^{-\Delta\tau\hat{H}(s,\tau)} = e^{-\Delta\tau\, c_i^\dagger \mathsf{K}_{ij} c_j} e^{-\Delta\tau\, c_i^\dagger \mathsf{V}_{ij}(s,\tau) c_j}\,. \tag{2.6}$$

This transformed expression can now be used to evaluate the partition sum or to calculate the projected wave function, either case requiring the additional sum over all possible auxiliary field configurations. The fermion states in the trace of the partition sum and the trial wave function are represented by a Slater determinant and thus one has to determine how (2.6) acts on such Slater determinants. It turns out that applying one of the discretized exponential operators to the Slater determinant results in another Slater determinant with the matrix exponential of the single particle operators,

$$e^{-\Delta\tau\,\mathsf{K}_{ij}} e^{-\Delta\tau\,\mathsf{V}_{ij}(s,\tau)}\,,$$

as its prefactor. Due to the length of the calculation, the derivation is deferred to the appendix.

As a shorthand, the product of matrix representations for a given imaginary time slice will be denoted by

$$\mathsf{B}(s,\tau) = e^{-\Delta\tau\,\mathsf{K}_{ij}} e^{-\Delta\tau\,\mathsf{V}_{ij}(s,\tau)}\,. \tag{2.7}$$

Further simplifying the notation, the auxiliary field argument $s$ is dropped unless explicitly needed and it shall be understood that each instance of $\mathsf{B}(\tau)$ also depends on a particular auxiliary field configuration. A partial product of slice matrices starting at time slice $\tau$ and ending at time slice $\tau'$ will be denoted as

$$B(\tau',\tau) = \prod_{i=\tau}^{\tau'} \mathsf{B}(i)\,. \tag{2.8}$$

Finally, the full product of all time slices is denoted as

$$\mathcal{B}(\tau=1) = \prod_{i=1}^{N_\tau} \mathsf{B}(i)\,. \tag{2.9}$$

The label $\tau$ in $\mathcal{B}(\tau=1)$ indicates the starting point for the matrix product, i.e. which time slice the leftmost matrix represents. If $\tau \neq 1$, the corresponding matrix product should simply be viewed as cyclically permuted by $\tau-1$ elements. Objects like that are required later when accessing equal-time Green's functions at different imaginary time slices.

### 2.1.1. Finite temperature simulations

Continuing with the specifics of the finite temperature approach, the partition sum can now be rewritten in the following concise way

$$Z = \sum_{\{s(\tau,j)\}} \mathrm{Tr}\,\mathcal{B}(\tau=1)\,. \tag{2.10}$$

At this point, there is a sum over all auxiliary field configurations $s(\tau,j)$ *and* the trace over all fermion states and the problem is apparently more complex than it was initially. However, after the Hubbard-Stratonovich transformation, the fermions are non-interacting and thus the expectation value can be evaluated for each of the Hubbard-Stratonovich field configurations and each of the fermion configurations, resulting in a determinant for each pair. In fact, it is shown in the appendix that the sum over all of these determinants, i.e. the trace over all fermion states, results in yet another determinant which depends solely on the auxiliary field configuration and eliminates the need to evaluate the fermion trace in a Monte Carlo calculation. This core identity not only gives the algorithm its name but is also a crucial ingredient for the algorithm because the evaluation of the fermionic trace would be prohibitively slow. Finally, the partition sum takes the form

$$Z = \sum_{\{s(\tau,j)\}} \det\left(1+\mathcal{B}(\tau)\right), \tag{2.11}$$

where only the auxiliary field configurations have to be sampled via a Monte Carlo calculation.

### Sampling auxiliary field configurations

The sampling of the auxiliary fields configurations is performed within a Metropolis-Hastings scheme, as discussed in the previous chapter. A central question is how to calculate the ratio of the statistical weights $r$ of two configurations, denoted as $w$ and $w'$, respectively, in order to accept or reject proposed updates:

$$r = \frac{w'}{w} = \frac{\det\left(1 + \mathcal{B}'(\tau)\right)}{\det\left(1 + \mathcal{B}(\tau)\right)}\,. \tag{2.12}$$

Again, the explicit evaluation of the determinant at each step would be too costly and thus one has to find an alternative way. In the appendix, it is shown that the matrices whose determinants are the weights of the corresponding auxiliary field configurations are related to the equal times Green's function given by

$$G \equiv \left(1 + \mathcal{B}(\tau)\right)^{-1}\,. \tag{2.13}$$

If an auxiliary spin is updated on imaginary time slice $\tau$, the product of time slices with the updated spin, $\mathcal{B}'(\tau)$, can be expressed in terms of the original product $\mathcal{B}(\tau)$ as

$$\mathcal{B}'(\tau) = \mathcal{B}(\tau)\,\mathbf{B}^{-1}(t)\,\mathbf{B}'(t)\,. \tag{2.14}$$

Using both of these shorthand notation, (2.12) can be rewritten as

$$\begin{aligned} r &= \det(G)\ \det\big((\mathbb{1} + \mathcal{B}'(\tau))\big) \\ &= \det\big(G\,(\mathbb{1} + \mathcal{B}'(\tau))\big) \\ &= \det\Big(G\,(\mathbb{1} + \mathcal{B}'(\tau)\,\mathbf{B}^{-1}(t)\,\mathbf{B}'(t)\Big) \\ &= \det\Big(G\,(\mathbb{1} + (G^{-1} - \mathbb{1})\,\mathbf{B}^{-1}(t)\,\mathbf{B}'(t)\Big) \\ &= \det\Big((G + (\mathbb{1} - G)\,\mathbf{B}^{-1}(t)\,\mathbf{B}'(t)\Big) \\ &= \det\Big((\mathbb{1} + (\mathbb{1} - G)\,(\mathbf{B}^{-1}(t)\,\mathbf{B}'(t) - \mathbb{1})\Big) \\ &= \det\left((\mathbb{1} + (\mathbb{1} - G)\,\mathbf{\Delta}\right)\,. \end{aligned} \tag{2.15}$$

The matrix $\mathbf{\Delta}$ is often sparse or can at least be approximated as being so which simplifies the calculation tremendously and allows quickly evaluating the remaining determinant. Its exact form always depends on the model at hand and the choice of Hubbard-Stratonovich transformation. Some concrete examples are provided later when the Hubbard-Stratonovich transformation is discussed in detail. It has to be emphasized that the update ratio depends solely on the matrix $\mathbf{\Delta}$, which is calculated for a single time slice, and the Green's function $G$ which depends on the entire product of slice matrices.

### Updating the Green's function

Upon changing one of the auxiliary spins, the Green's function $G = \left(1 + \mathcal{B}(\tau)\right)^{-1}$ has to be updated, since it depends on the product of slice matrices $\mathcal{B}$ which changes with each update. Instead of recalculating $\mathcal{B}$ and $G$ from scratch, the Shermann-Morrison-Woodbury formula can be used to express the change in $G$ as:

$$\begin{aligned}
G' &= \left(1+\mathcal{B}'(\tau)\right)^{-1} \\
&= \left(1+\mathcal{B}(\tau)\mathbf{B}^{-1}\mathbf{B}'\right)^{-1} \\
&= \left(1+(G^{-1}-1)\mathbf{B}^{-1}\mathbf{B}'\right)^{-1} \\
&= \left(1+(1-G)(\mathbf{B}^{-1}\mathbf{B}'-1)\right)^{-1} G \\
&= (1+(1-G)\boldsymbol{\Delta})^{-1} G \\
&= \left(1+\boldsymbol{u}\boldsymbol{v}^T\right)^{-1} G \\
&= \left(1+\boldsymbol{u}\left(1+\boldsymbol{v}^T\boldsymbol{u}\right)^{-1}\boldsymbol{v}^T\right) G
\end{aligned} \tag{2.16}$$

Again, the concrete form of the row and column matrices $\boldsymbol{u}$ and $\boldsymbol{v}^T$ depends on the choice of Hubbard-Stratonovich transformation.

### 2.1.2. Projective formulation

For the projective formulation, it is not immediately obvious how the sampling procedure should be carried out. The objective is to evaluate observables $\mathcal{A}$ using the projected trial wave function:

$$\begin{aligned}
&\frac{\langle \psi_T | e^{-\theta H} \mathcal{A} e^{-\theta H} | \psi_T \rangle}{\langle \psi_T | e^{-\theta H} e^{-\theta H} | \psi_T \rangle} \\
&= \sum_{\{s_i\}} \frac{\det\left(P^\dagger \mathbf{B}_l(2\theta,\tau)\mathbf{B}_r(\tau,0)P\right)}{\sum_{\{s_i\}} \det\left(P^\dagger \mathbf{B}_l(2\theta,\tau)\mathbf{B}_r(\tau,0)P\right)} \cdot \frac{\langle \psi_T | e^{-\theta H} \mathcal{A} e^{-\theta H} | \psi_T \rangle}{\langle \psi_T | e^{-\theta H} e^{-\theta H} | \psi_T \rangle} \\
&= \sum_{\{s_i\}} P_{\mathbf{s}} \langle \mathcal{A} \rangle_s \,.
\end{aligned} \tag{2.17}$$

In going from the first to the second line, the summation over auxiliary fields in the numerator and denominator was written out explicitly ($\mathbf{B}$ depends on it) and the fraction was expanded by another overlap of the function evaluated for the same auxiliary field configuration as the numerator. Note that each projected wave function, the bra $\langle \psi |$ and the ket $| \psi \rangle$, contribute one projection of magnitude $\theta$, and therefore each come with their own product of slice matrices. Writing the expectation value of an arbitrary operator in this way allows separating the measurement of said operator from the statistical weight factor and thus provides a prescription for the Monte Carlo procedure which samples auxiliary field configurations based on the strength of the overlap between the two projected wave functions.

#### Sampling auxiliary field configurations

Starting from the ratio of two determinants with matrices $\mathbf{B}$ that differ in their auxiliary field configuration on time slice $\tau$ (which belongs to the bra), the weight ratio $r$ can be written as

$$\begin{aligned}
r &= \frac{\det\left(P^{\dagger}\mathsf{B}'_l(2\theta,\tau)\mathsf{B}_r(\tau,0)P\right)}{\det\left(P^{\dagger}\mathsf{B}_l(2\theta,\tau)\mathsf{B}_r(\tau,0)P\right)} \\
&= \frac{\det\left(P^{\dagger}\mathsf{B}_l(2\theta,\tau)\left(\mathbb{1}+\boldsymbol{\Delta}\right)\mathsf{B}_r(\tau,0)P\right)}{\det\left(P^{\dagger}\mathsf{B}_l(2\theta,\tau)\mathsf{B}_r(\tau,0)P\right)} \\
&= \det\left(\boldsymbol{\Psi}_l\left(\mathbb{1}+\boldsymbol{\Delta}\right)\boldsymbol{\Psi}_r\left(\boldsymbol{\Psi}_l\boldsymbol{\Psi}_r\right)^{-1}\right) \\
&= \det\left(\mathbb{1}+\boldsymbol{\Delta}\boldsymbol{\Psi}_r\left(\boldsymbol{\Psi}_l\boldsymbol{\Psi}_r\right)^{-1}\boldsymbol{\Psi}_l\right) \\
&= \det\left(\mathbb{1}+\boldsymbol{\Delta}\left(\mathbb{1}-\boldsymbol{G}\right)\right) \\
&= \det\left(\mathbb{1}+\left(\mathbb{1}-\boldsymbol{G}\right)\boldsymbol{\Delta}\right),
\end{aligned} \tag{2.18}$$

which turns out to be exactly the same expression as in the finite temperature case, except that the equal time Green's function is defined differently in the context of the ground state algorithm:

$$G = \mathbb{1} - \boldsymbol{\Psi}_r\left(\boldsymbol{\Psi}_l\boldsymbol{\Psi}_r\right)^{-1}\boldsymbol{\Psi}_l\,. \tag{2.19}$$

A proof of this statement can also be found in the appendix.

### Updating the Green's function

Although the Green's function is computed from completely different formulas in the finite temperature and the projective variant, the resulting equation will turn out to be the same which is very convenient for actual simulations. The derivation the ground state case goes as follows:

$$\begin{aligned}
\boldsymbol{G}' &= \mathbb{1} - \boldsymbol{\Psi}'_r\left(\boldsymbol{\Psi}_l\boldsymbol{\Psi}'_r\right)^{-1}\boldsymbol{\Psi}_l \\
&= \mathbb{1} - (\mathbb{1}+\boldsymbol{\Delta})\boldsymbol{\Psi}_r\left(\boldsymbol{\Psi}_l(\mathbb{1}+\boldsymbol{\Delta})\boldsymbol{\Psi}_r\right)^{-1}\boldsymbol{\Psi}_l \\
&= \mathbb{1} - (\mathbb{1}+\boldsymbol{\Delta})\boldsymbol{\Psi}_r\left(\boldsymbol{\Psi}_l\boldsymbol{\Psi}_r+\boldsymbol{\Psi}_l\boldsymbol{\Delta}\boldsymbol{\Psi}_r\right)^{-1}\boldsymbol{\Psi}_l \\
&= \mathbb{1} - (\mathbb{1}+\boldsymbol{\Delta})\boldsymbol{\Psi}_r\left[\left(\boldsymbol{\Psi}_l\boldsymbol{\Psi}_r\right)^{-1} - \left(\boldsymbol{\Psi}_l\boldsymbol{\Psi}_r\right)^{-1}\boldsymbol{u}\left(\mathbb{1}+\boldsymbol{v}^T\left(\boldsymbol{\Psi}_l\boldsymbol{\Psi}_r\right)^{-1}\boldsymbol{u}\right)^{-1}\boldsymbol{v}^T\left(\boldsymbol{\Psi}_l\boldsymbol{\Psi}_r\right)^{-1}\right]\boldsymbol{\Psi}_l \\
&= \boldsymbol{G} - \boldsymbol{G}\left(\mathbb{1}+\boldsymbol{v}^T\left(\boldsymbol{\Psi}_l\boldsymbol{\Psi}_r\right)^{-1}\boldsymbol{u}\right)^{-1}\boldsymbol{\Delta}(\mathbb{1}-\boldsymbol{G})
\end{aligned} \tag{2.20}$$

The calculation also uses the Sherman-Morrison-Woodbury formula to transform the expression in line three. Again, the precise form of the matrices $\boldsymbol{u}$ and $\boldsymbol{v}^T$ depends on the specific Hubbard-Stratonovich transformation employed.

## 2.2. The Hubbard-Stratonovich transformation

The preceding section showed that the Hubbard-Stratonovich transformation is an integral part of the determinant quantum Monte Carlo method, allowing to replace the exponential of a squared operator $A$ by an integral over the exponential of the operator coupled to a newly introduced auxiliary field and the exponential of the square of the field variable:

$$e^{\frac{1}{2}A^2} = \int \mathrm{d}s\, e^{-sA} e^{-\frac{1}{2}s^2}\,. \tag{2.21}$$

In the context of DQMC calculations, $A$ typically corresponds to the quartic interaction operator $V$ that couples, for example, spins of opposite flavor on the same site for the case of the spinful Hubbard model or electrons on neighboring sites as is the case for the spinless $t-V$ model. In both cases, one particular term of the interaction between two particles takes the form

$$V_{\alpha\beta} = \left(n_\alpha - \frac{1}{2}\right)\left(n_\beta - \frac{1}{2}\right) \tag{2.22}$$

where the index represents either a spin index in the spinful case or a site index in the spinless case, respectively. In DQMC calculations, such a term appears in the exponential as

$$e^{-\frac{1}{2}\Delta\tau\left(n_\alpha-\frac{1}{2}\right)\left(n_\beta-\frac{1}{2}\right)}\,, \tag{2.23}$$

which is not in the quadratic form required to apply the transformation (2.21). In practice, one completes the square in one of the following ways:

$$n_\alpha n_\beta = -\frac{1}{2}(n_\alpha - n_\beta)^2 + \frac{1}{2}(n_\alpha + n_\beta) \tag{2.24}$$

$$n_\alpha n_\beta = \frac{1}{2}(n_\alpha + n_\beta)^2 - \frac{1}{2}(n_\alpha + n_\beta)\,. \tag{2.25}$$

Named after the quantity that the squared term represents, these two possibilities are said to decouple in the magnetic and charge channel, respectively.

Applying the Hubbard-Stratonovich transformation (2.21) to the quadratic part yields

$$\exp\left[\frac{1}{2}\Delta\tau U(n_\alpha \pm n_\beta)^2\right] = C\int \mathrm{d}s\, \exp\left[-s\sqrt{\pm\Delta\tau U}(n_\alpha \pm n_\beta)\right]\exp\left(-\frac{1}{2}s^2\right). \tag{2.26}$$

It turns out that the continuous, auxiliary variable $s$ can be replaced by a discrete variable [21], simplifying the simulation. For the repulsive Hubbard model ($U < 0$) of spinful fermions one then proceeds as follows: Choosing the magnetic channel to decouple using (2.24), yields

$$V = -\frac{1}{2}\Delta\tau U(n_\uparrow - n_\downarrow)^2 + U\frac{1}{4}\,, \tag{2.27}$$

and thus $A = \Delta\tau U(n_\uparrow - n_\downarrow)^2$. To find a discretized version of the Hubbard-Stratonovich transformation, one uses the *ansatz*

$$\exp(-\Delta\tau V) = C\sum_{s=\pm 1} \exp\left[\alpha s(n_\uparrow - n_\downarrow)\right], \tag{2.28}$$

where the constants $C$ and $\alpha$ have to be determined such that the equation becomes a true identity. This is done by evaluating the original operator $n_\uparrow n_\downarrow = n_\uparrow \otimes n_\downarrow$ on its four-state Hilbert space

$$\mathcal{H} = \{|00\rangle, |01\rangle, |10\rangle, |11\rangle\}, \tag{2.29}$$

which results in the expectation values listed in table 2.1. The parameters $C$ and $\alpha$ are

| state | $e^{-\Delta\tau V}$ | $C \sum_{s=\pm 1} e^{\alpha s(n_\uparrow - n_\downarrow)}$ |
|---|---|---|
| $\|00\rangle$ | $\exp(-\Delta\tau U/4)$ | $2C$ |
| $\|01\rangle$ | $\exp(\Delta\tau U/4)$ | $2C\cosh\alpha$ |
| $\|10\rangle$ | $\exp(\Delta\tau U/4)$ | $2C\cosh\alpha$ |
| $\|11\rangle$ | $\exp(-\Delta\tau U/4)$ | $2C$ |

**Table 2.1.:** Determining the prefactor and the coupling constant of the Hubbard-Stratonovich transformation.

thus given by

$$C = \frac{1}{2} e^{-\Delta\tau U/4} \quad \text{and} \quad \cosh\alpha = e^{\Delta\tau U/2} \tag{2.30}$$

Using (2.33), the product $\mathbf{B}^{-1}(t)\,\mathbf{B}'(t)$ which is relevant for determining the matrix $\boldsymbol{\Delta}$ becomes

$$\mathbf{B}^{-1}(t)\,\mathbf{B}'(t) = e^{\Delta\tau\mathbf{V}} e^{\Delta\tau\mathbf{K}} e^{-\Delta\tau\mathbf{K}} e^{-\Delta\tau\mathbf{V}'} = e^{\Delta\tau(\mathbf{V}-\mathbf{V}')}, \tag{2.31}$$

and thus, the matrix $e^{\Delta\tau(\mathbf{V}-\mathbf{V}')} - \mathbb{1}$ is non-zero only on the site index, $k$, of the auxiliary spin $s$:

$$\left(e^{\Delta\tau(\mathbf{V}-\mathbf{V}')} - \mathbb{1}\right)_{i,j} = e^{\alpha(s-s')}\,\delta_{i,j}\,\delta_{i,k}\,. \tag{2.32}$$

In this case, the ratio of determinants in (2.15) boils down to one number that can easily be read off from the Green's matrix.

Alternatively, (2.25) can be used to decouple in the charge channel. Proceeding with the same ansatz as before gives

$$\exp(-\Delta\tau V) = C \sum_{s=\pm 1} \exp\left[\mathrm{i}\alpha s(n_\alpha + n_\beta - 1)\right], \tag{2.33}$$

where the constants are now determined by

$$C = \frac{1}{2} e^{\Delta\tau U/4} \quad \text{and} \quad \cos\alpha = e^{\Delta\tau U/2}\,. \tag{2.34}$$

This decoupling scheme has the advantage of not breaking $SU(2)$ symmetry, which was shown to lead to shorter autocorrelation and thus short simulation times despite the fact that one now has to deal with complex numbers because $\alpha$ is complex [22].

The Hubbard-Stratonovich transformations presented so far are exact transformations of the quartic term. Since the algorithm itself suffers from a Trotter error, it is also permissible to approximate the exponential's Taylor expansion as long as the additional error is of equal or smaller magnitude than the Trotter error. Technically, transformations of this type are not Hubbard-Stratonovich transformations anymore because they are not exact. They do, however, also introduce additional degrees of freedom that couple linearly to the desired operator and thus look and work just like an actual transformation would. One example of such an approximate transformation is the following one that preserves $SU(N)$ symmetry which can be used to simulate Hubbard models of $SU(N)$ fermions:

$$e^{\Delta\tau\lambda A^2} = \sum_{s=\pm 1,\pm 2} \gamma(s) e^{\sqrt{\Delta\tau\lambda}\,\eta(s) A} + \mathcal{O}(\Delta\tau^4) \tag{2.35}$$

Note that there are now two fields $\gamma$ and $\eta$, which take values

$$\gamma(\pm 1) = 1 + \sqrt{6}/3, \quad \gamma(\pm 2) = 1 - \sqrt{6}/3 \tag{2.36}$$

$$\eta(\pm 1) = \pm\sqrt{2\left(3 - \sqrt{6}\right)}, \quad \eta(\pm 2) = \pm\sqrt{2\left(3 + \sqrt{6}\right)} \tag{2.37}$$

This more general scheme is not only used for the Hubbard model but also in a model that features a transition from an antiferromagnet to a superconducting state [23]. The validity of this approach can be verified by expanding both sides of (2.35) which agree up to $\mathcal{O}(\Delta\tau^4)$.

These are the most common versions of Hubbard-Stratonovich transformations used for Hubbard models. Although they all transform the same problem, each results in very different simulations. Which transformation is best suited depends on the problem at hand: On an algorithmic level, they may differ in autocorrelation times [22], data type required, i.e. real or complex double, and they also play a major role in the stability of the algorithm, in particular when calculating entanglement entropies, which is discussed in chapter 5.

## 2.3. Numerical implementation

Any practical implementation of the DQMC algorithm faces the problem that calculating the product of slice matrices is numerically unstable and produces an ill-conditioned matrix [24]. This section describes how to circumvent this problem and how the resulting product can be used to calculate the Monte Carlo weight and the Green's function after all.

### Stabilizing the product of slice matrices

The key idea for stabilization is to perform only a certain number of direct multiplications $m$ followed by either a rank revealing QR or a singular value decomposition (SVD). This number $m$ is typically chosen small enough such that the product of $m$ matrices remains exact up to machine precision. The decomposition recasts a given matrix $M$ into a product $M = UDT$, where $U$ is typically unitary, $D$ is a diagonal matrix that contains the information about the inherent scales of the matrix (e.g. in the form of its singular value spectrum), and $T$ is either unitary (SVD) or triangular (QR decomposition). Independent of the specific decomposition algorithm, the values of the diagonal matrix $D$ will be called singular values.

Applying this to the matrix product (2.9), the imaginary time interval is divided into $N_m$ groups of $m$ time slices representing a segment of length $\Delta = m \cdot \tau$ in imaginary time. After the multiplication and subsequent decomposition of the first $m$ matrices yielding $U_1 D_1 T_1$, the next set of matrices is then multiplied onto $U_1$, *before* $D_1$ is multiplied from

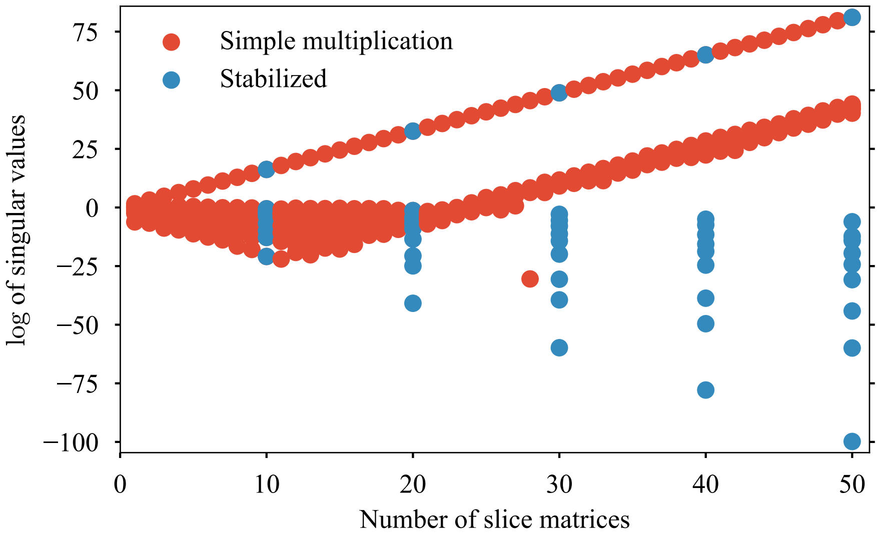


**Figure 2.1.:** Illustrating the inherent instability of a product of many random matrices with entries in the range of $[0,1]$. At each step, a singular value decomposition is performed and the log of the resulting singular values are plotted in red. At every tenth step, a stabilization step is performed which yield the stabilized singular values, plotted in blue.

the right and the resulting product is decomposed again, i.e.

$$\left(\left(\prod_{i=m+1}^{2m} \mathsf{B}(i)\cdot U_1\right) D_1\right) T_1 = U_2 D_2 (T T_1) = U_2 D_2 T_2 \,. \tag{2.38}$$

This procedure is repeated until all slice matrices are incorporated into the $UDT$ decomposition. Note that all the intermediate results will be used at a later point so they are stored in memory. This set of matrices, $\{\{U_i\},\{D_i\},\{T_i\}\}$, will be referred to as the *stack*. The effect of this procedure can be easily illustrated by comparing a long product of random matrices multiplied in a simple manner one after the other to the same product where a stabilization step has been performed at certain intervals. The result of this experiment is shown in Fig. 2.1. As is known from the numerical details of the underlying decomposition algorithms, the range of singular values is limited by the largest singular value and the machine precision of the data type. Therefore, the spread of the singular values of the simple product becomes larger as long as the singular values remain in the range defined by the machine precision. Multiplying more matrices theoretically increases this spread but the decomposition procedure is incapable of resolving the lower lying eigenvalues correctly and, due to how the algorithm is set up, they increase along with the largest eigenvalue. In contrast, when working with a stabilized product, the lower lying eigenvalues are resolved correctly even when the condition number becomes very large.

### Calculation of the weight and the Green's matrix

The weight and the Green's matrix are calculated from the slice matrix product and thus their precision is limited by the stability of slice matrix product. To achieve optimal precision, one thus has to make use of the decomposed slice matrix product, which means that the determinant is given as

$$\det\left(\mathbb{1}+\mathcal{B}(\tau)\right)=\det\left(\mathbb{1}+UDT\right), \tag{2.39}$$

and the equal-time Green's function as

$$G(\tau)=\left(\mathbb{1}+\mathcal{B}(\tau)\right)^{-1}=\left(\mathbb{1}+UDT\right)^{-1}. \tag{2.40}$$

The addition of the identity matrix, although seemingly simple, requires careful attention to ensure an accurate calculation of the determinant and the Green's function. This problem is solved by separating the diagonal matrix $D$ from the auxiliary matrices $U$ and $T$ before the resulting matrix is decomposed yet again:

$$\left(\mathbb{1}+UDT\right)=U\left(U^{-1}T^{-1}+D\right)T=\left(UU'\right)D'\left(T'T\right). \tag{2.41}$$

The weight, i.e. the determinant, can then easily be read off as the product of the diagonal entries of $D'$. It should be noted that in practice one would calculate the logarithm of the determinant, a common procedure to avoid numerical overflow and rounding errors.

### An efficient updating procedure

The Green's matrix at a given imaginary time $\tau$ is a crucial component of the updating procedure of the auxiliary field at time slice $\tau$, which means that it is important to have access to the Green's function at each time slice. One option would be to recalculate the Green's function from scratch at every time slice which would be numerically very expensive. Instead, one may propagate the Green's function along imaginary time by virtue of

$$G(\tau+1)=\mathbf{B}(\tau)G(\tau)\mathbf{B}^{-1}(\tau). \tag{2.42}$$

This equation suggests that it would suffice to calculate the Green's function only once, e.g. at $\tau=1$, and then continue propagating the Green's function along imaginary time updating the appropriate auxiliary field degrees of freedom. However, the propagation suffers from the same problem as the build-up of the matrix product (2.9) itself – it is stable only for a few steps. Just like before, one can assume a number of stable propagation steps $N_{\text{prop}}$ that is equal to the number of safe multiplications of slice matrices $m$. In general, this number can be determined by considering the difference between the propagated Green's function and the recalculated one after $m$ propagations have taken place. One possible metric by which to measure the stability is the element-wise relative deviation

$$\delta_{ij}=2\cdot\frac{\left|G_{ij}^{\text{recalculated}}-G_{ij}^{\text{propagated}}\right|}{\left|G_{ij}^{\text{recalculated}}+G_{ij}^{\text{propagated}}\right|}, \tag{2.43}$$

between the two Green's matrices. It should be small enough such that the update probabilities and measurements which use the entries of the Green's matrix are not seriously affected. Therefore, one typically focuses on comparing only those matrix elements that are used in the updating and measurement procedure.

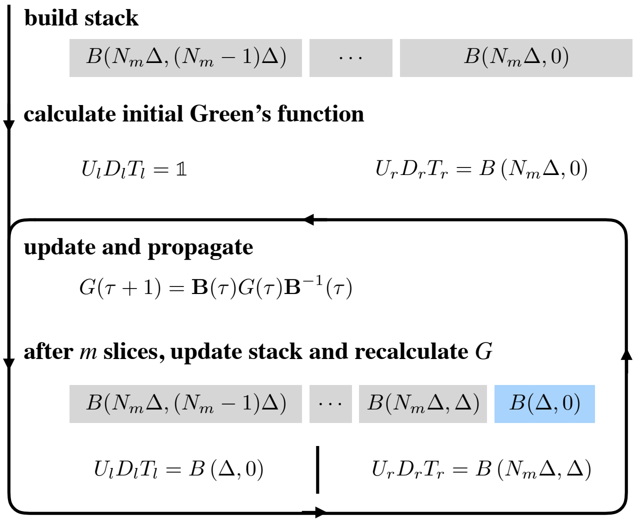

build stack
$B(N_m\Delta,(N_m-1)\Delta)$
$\cdots$
$B(N_m\Delta,0)$
calculate initial Green's function
$U_lD_lT_l = \mathbb{1}$
$U_rD_rT_r = B\,(N_m\Delta,0)$
update and propagate
$G(\tau+1) = \mathbf{B}(\tau)G(\tau)\mathbf{B}^{-1}(\tau)$
after $m$ slices, update stack and recalculate $G$
$B(N_m\Delta,(N_m-1)\Delta)$
$\cdots$
$B(N_m\Delta,\Delta)$
$B(\Delta,0)$
$U_lD_lT_l = B\,(\Delta,0)$
$U_rD_rT_r = B\,(N_m\Delta,\Delta)$

**Figure 2.2.:** Flow diagram of the DQMC algorithm. Shaded in grey is the initial build-up of the stack of decomposed matrix products. This is used to calculate the Green's function which is subsequently updated and propagated in imaginary time before updating the stack and recalculating the Green's function from the stack to ensure stability. The slice product $B(\Delta,1)$ shaded in blue has to be recalculated from scratch because it contains auxiliary field values that were updated in the Monte Carlo procedure. Subsequent slice products will the respective preceding $U$ matrix as a starting point for the recalculation. Figure first appeared in [25].

## Making use of the stack

For those imaginary time slices where the Green's function has to be recalculated from scratch, one can greatly facilitate its calculation by making use of the matrix products saved in the stack. After building up the stack in an initial procedure, the last entry contains the decomposed product over all slice matrices, the previous entry has all matrices but the last $m$ slices and so forth. Updates starting at the last slice, i.e. down from the "top" of the stack. After $m$ updates have been performed, the Green's function has to be recalculated at the same time slice. The stack already contains the decomposed product of all matrices from $\tau = 1$ up until the current time slice, called $U_LD_LT_L$. Thus, only the decomposed product for the sequence starting at the current time slice up until $\tau = \beta$, denoted by $U_RD_RT_R$, is needed. The entire slice product is then given as $U_LD_LT_L\,U_RD_RT_R$, but it is not necessary to actually carry out the recombination. The product of left and right decomposition can be directly used in equations (2.39) and (2.40) by making slight changes to (2.41).

A summary and overview of the DQMC algorithm are presented in Fig. 2.2.

# 3. Stochastic Series Expansion

Stochastic Series expansion (SSE) [26, 27] is a highly efficient Monte Carlo flavor mainly used for the simulation of bosonic quantum mechanical systems. It can be formulated in both a finite temperature and a ground state setting, which work almost identically. This chapter aims to convey the basics of the finite temperature formulation which was employed to collect some of the data presented in the later chapters. Similarly structured as the preceding chapter 2, this chapter first introduces the basic idea of the SSE algorithm and then proceeds to describe update and measurement schemes.

## 3.1. Formulation of the Monte Carlo procedure

The key idea of SSE is to rewrite the partition sum in its power series representation

$$Z = \mathrm{Tr}\exp\left(-\beta H\right) = \sum_{|\psi\rangle} \langle\psi| \sum_{n=0}^{\infty} \frac{\beta^n}{n!} (-H)^n |\psi\rangle \,. \tag{3.1}$$

This approach is based on Handscomb's method [28] but it had to reformulated to be successfully applied to a variety of models [29, 30, 31, 32, 33]. For each power in the expanded series, a resolution of unity $|\phi_i\rangle\langle\phi_i|$ is inserted to give

$$Z = \sum_{|\psi\rangle} \sum_{n=0}^{\infty} \frac{\beta^n}{n!} \langle\psi|(-H)|\phi_1\rangle\langle\phi_1|(-H)|\phi_2\rangle\langle\phi_2|\ldots|\phi_{n-1}\rangle\langle\phi_{n-1}|(-H)|\psi\rangle \,. \tag{3.2}$$

Evaluating all of the expectation values for all combinations is impossible which motivates the use of Monte Carlo to sample the terms that contribute most to the partition sum. Each Hamiltonian operator is then broken up into smaller units, most often into bond operators. In some cases, larger clusters are possible and sometimes advantageous or even necessary to use [34, 35]. In any case, the Hamiltonian is then written as

$$H = -\sum_{t}\sum_{l} H_{t,u} \,. \tag{3.3}$$

Each term is indexed by its operator type $t$, e.g. a diagonal $S_z$ interaction or an off-diagonal spin exchange, and by the index of the lattice unit $u$, e.g. the number of the bond this operator acts on. To shorten the long multiplication that results when (3.3) is inserted into (3.2), the many different combinations of operators are encoded in a new object called operator sequence, denoted by $S_n$. This allows writing the partition function in the following concise way:

$$Z = \sum_{|\psi\rangle} \sum_{n=0}^{\infty} \sum_{S_n} \frac{\beta^n}{n!} \prod_{i=1}^{n} \langle\phi_{i-1}|H_{t_i,u_i}|\phi_i\rangle \,. \tag{3.4}$$

In this expression, the identification $|\psi\rangle \equiv |\phi_0\rangle \equiv |\phi_n\rangle$ has been made in favor of a clearer notation. Although there is a sum over all expansion orders $n \in [0, \infty]$, it turns out that in practice there is an effective maximum expansion order $M$ that suffices to capture *all* relevant sequence lengths. Its precise value is determined dynamically in the course of the simulation. In contrast to DQMC and other worldline techniques, there is, therefore, no additional approximation error other than the statistical error from the Monte Carlo sampling. Any sequence of shorter length $n < M$ can be artificially enlarged to one of size $M$ by considering all possible arrangements of $n$ operators on $M$ places and filling the remaining slots with identity operators. When writing the operator sequence in such a way, the contribution of any shorter sequences their contribution to the final expression has to be corrected by a factor $(M-n)!\,/\,M!$ to avoid overcounting:

$$Z = \sum_{|\psi\rangle} \sum_{S_M} \frac{(M-n)!}{M!} \frac{\beta^n}{n!} \prod_{i=1}^{M} \langle \phi_{i-1} | H_{t_i,u_i} | \phi_i \rangle \tag{3.5}$$

Note that $\beta$ still comes with a power $n$ because it depends on how many non-trivial elements are contained in the extended operator sequence.

## 3.2. Sampling operator sequences

Moving along the Markov chain and proposing new operator sequences follows a set of well defined, general rules which are nevertheless easiest to illustrate on a concrete example. For this purpose, the anisotropic Heisenberg model in a magnetic field is used which will reappear in chapter 6. Its Hamiltonian is

$$H = -\sum_{\langle i,j\rangle} \left( S_i^+ S_j^- + S_i^- S_j^+ \right) + \Delta \sum_{\langle i,j\rangle} S_i^z S_j^z + h \sum_i S_i^z \,, \tag{3.6}$$

and the simulation is carried out in the $S^z$ basis. The first term is a sum of off-diagonal operators responsible for exchanging spins of opposite orientation. The second term is diagonal in the simulation basis and simply measures the relative orientation of two spins, while the third one compares the orientation of each individual spin to the direction of the magnetic field. Following the rules for the formulation of the partition sum laid out previously, each of the sums in the Hamiltonian is rewritten in terms of bond operators. In fact, the first and the second term are already of the correct form, but the coupling to the magnetic field requires a little work. To make it compatible with the bond formulation, its contribution is distributed over the adjacent bonds of each site giving rise to a magnetic field $h_B = h/4$. Ultimately, there are thus only two possible types of operators, which are the diagonal operator

$$H_{\text{diag}} = \Delta S_i^z S_j^z + h_B (S_i^z + S_j^z) \,, \tag{3.7}$$

and the off-diagonal operator

$$H_{\text{exchange}} = -\left( S_i^+ S_j^- + S_i^- S_j^+ \right) \,. \tag{3.8}$$

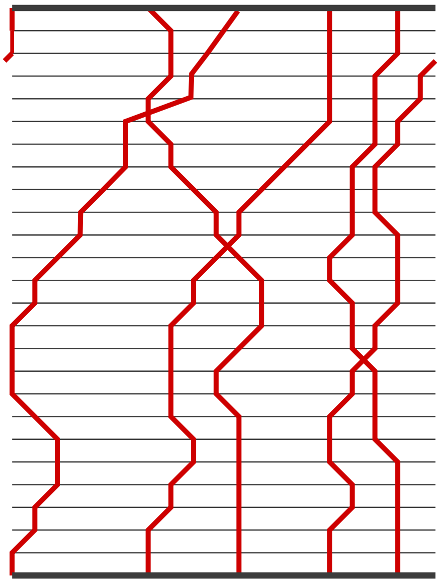


**Figure 3.1.:** Visualization of the propagation of degrees of freedom along operator time as worldlines. worldlines can switch places by virtue of off-diagonal operators. Because of the periodic boundary conditions imposed on the lattice, they can wrap around the simulation cell. In addition, the trace in the partition sum enforces periodic boundary conditions in operator time.

These operators allow for three different non-trivial expectation values (and their respective versions with all spins flipped):

$$H_{\mathrm{fe},\uparrow} = \langle\uparrow\uparrow| H_{\mathrm{diag}} |\uparrow\uparrow\rangle = \Delta/4 + h_B/2 \tag{3.9}$$
$$H_{\mathrm{fe},\downarrow} = \langle\downarrow\downarrow| H_{\mathrm{diag}} |\downarrow\downarrow\rangle = \Delta/4 - h_B/2 \tag{3.10}$$
$$H_{\mathrm{af}} = \langle\uparrow\downarrow| H_{\mathrm{diag}} |\uparrow\downarrow\rangle = -\Delta/4 \tag{3.11}$$
$$H_{\mathrm{od}} = \langle\downarrow\uparrow| H_{\mathrm{exchange}} |\uparrow\downarrow\rangle = -1 \tag{3.12}$$

Because these four expectation values enter with different prefactors, there is a trivial sign problem which can be solved easily by subtracting a constant $\Delta/4 + h_B/2 + \varepsilon$ from the Hamiltonian which leads to the final set of weights

$$H_{\mathrm{fe},\uparrow} = \langle\uparrow\uparrow| H_{\mathrm{diag}} |\uparrow\uparrow\rangle = -\varepsilon \tag{3.13}$$
$$H_{\mathrm{fe},\downarrow} = \langle\downarrow\downarrow| H_{\mathrm{diag}} |\downarrow\downarrow\rangle = -h_B/2 - \varepsilon \tag{3.14}$$
$$H_{\mathrm{af}} = \langle\uparrow\downarrow| H_{\mathrm{diag}} |\uparrow\downarrow\rangle = -\Delta/2 - h_B/2 - \varepsilon \tag{3.15}$$
$$H_{\mathrm{od}} = \langle\downarrow\uparrow| H_{\mathrm{exchange}} |\uparrow\downarrow\rangle = -1 \tag{3.16}$$

The resulting simulation can be visualized as taking place in a two-dimensional plane where one dimension represents the flattened spatial dimension of the lattice and the other one represents the operator time. At each time slice, the current state is visualized in some way, for example by marking sites occupied by a spin up by a black dot and leaving sites occupied by a spin down empty. Along the operator time direction, the dots are then connected to form so-called worldlines. Operators live between time slices and, for the operators in this model, each has two entry legs and two exit legs. The trace in the partition sum enforces the constraint that the spin state is the same on the bottom as it is on the top, meaning that wherever a solid line starts on the bottom there has to be one with the same spin state ending at the top.

Devising a valid update algorithm for moving from operator sequence to operator sequence requires thinking about how to ensure detailed balance and ergodicity, see chapter 1. Historically, this problem was solved in three steps: At first, only local updates were used [26]. They are easy to construct such that detailed balance is respected, but the ergodicity can often not be ensured. Especially moving between sectors of different values of global quantities like winding numbers is often very difficult or even impossible with only local updates. Non-local, so-called loop updates [9, 30, 27] and their more sophisticated version called directed loops [36, 37, 38] are often capable of efficiently solving the problem of ergodicity. In this work, a simple loop update was sufficient to obtain the desired results. For details about the directed loop algorithm, the interested reader is referred to the respective literature.

One sweep using the loop algorithm generally consists of two steps: one diagonal update step where diagonal operators are inserted and removed along the entire operator time dimension, and an off-diagonal update step where a loop is built that changes operator types from diagonal to off-diagonal and vice versa.

### Diagonal update

Any diagonal operator can be inserted or removed at any time with a certain probability. In the diagonal update, this is probed systematically by iterating over all operator time slices from bottom to top. If there is no operator present, a pair of a random bond and a random diagonal operator type is chosen based on their cumulative probability. If the worldline configuration on the selected bond matches the requirements of the chosen operator type, its insertion will be probed using the Metropolis criterion. The required ratio $r$ of weights is the ratio of the weight of the operator sequence before and after inserting the proposed operator. Note that an off-diagonal operator cannot be inserted like this because it requires two worldlines to trade places which is not allowed without an off-diagonal operator present in the first place. This problem will be solved by the off-diagonal loop update. In this model, the selection of bond and operator can be split into two separate operations because the coupling is isotropic. Thus, one first selects one of the $2 \cdot L^2$ bonds at random and then proceeds to choose between an antiferromagnetic or ferromagnetic operator by comparing a random number to

$$\frac{\varepsilon}{\Delta/2 + h_B/2 + 2\varepsilon} \quad \text{or} \quad \frac{\Delta/2 + h_B/2 + \varepsilon}{\Delta/2 + h_B/2 + 2\varepsilon}. \tag{3.17}$$

The numerator is the matrix element of the respective operator calculated earlier in this chapter and the denominator is the sum of all possible operator types.

### Off-diagonal loop update

The off-diagonal update is capable of introducing off-diagonal operators into the operator string while respecting the periodic boundary conditions in operator time. This is accomplished by randomly an operator and one of its legs at random and then flipping the associated spin, creating a defect in the worldline. This defect, also called loop head, is propagated through the space-time lattice by choosing one of the operator's legs as its exit and jumping to the next connected leg until the starting point is reached again and the loop is closed. Which exit leg is chosen depends on the model, the type of operator, the incoming leg and some parameters of the loop algorithm all of which can greatly

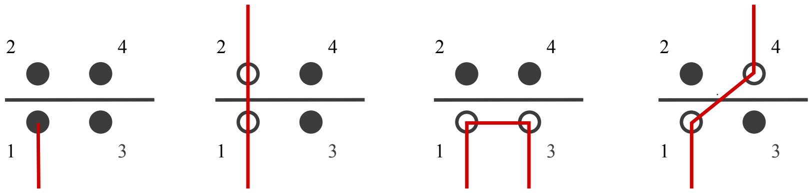


**Figure 3.2.:** The loop head arrives at a ferromagnetic operator and has four possible exits: It can bounce and exit the same leg it entered, leaving the operator type unchanged. By going straight through and exiting on leg 2, the operator is changed to a diagonal, antiferromagnetic operator. Exiting on leg 3 is forbidden in this case because the spin states on the bottom (before) and the top (after) have zero overlap for any of the possible operators. Lastly, the operator is replaced by an off-diagonal operator when the loop head exits on leg 4.

influence the efficiency of the Monte Carlo simulation. For the simplest loop algorithm, the rules are derived as follows: Figure 3.2 shows the situation where a loop head arrives at an off-diagonal operator. Out of the four possible exit legs, only three have non-zero weight because they leave the spins in a valid position. If the loop were to exit at leg 3, flipping the worldline segments connected to leg 1 (the entrance) and 3 (the exit) would result in a state that has zero overlap with the model Hamiltonian (3.6). The choice is between the remaining legs, i.e. exiting leg 1 and thus bouncing, exiting leg 2 and turning the operator into an antiferromagnetic operator or taking exit leg 4 which would turn the operator into a ferromagnetic operator. A random number is drawn and compared to the cumulative probability defined by

$$\frac{H_{\text{fe}}}{H_{\text{fe}}+H_{\text{af}}+H_{\text{od}}}, \quad \frac{H_{\text{fe}}+H_{\text{af}}}{H_{\text{fe}}+H_{\text{af}}+H_{\text{od}}}, \quad \frac{H_{\text{fe}}+H_{\text{af}}+H_{\text{od}}}{H_{\text{fe}}+H_{\text{af}}+H_{\text{od}}}. \tag{3.18}$$

The exit rules for the other types of operators are derived in exactly the same fashion.

For completeness, two properties of loops should be discussed. One is that this particular loop formulation is non-deterministic which simply means that the exit leg is chosen at random for each operator encountered. Only very special cases such as the isotropic Heisenberg model allow the construction of deterministic loops which offer particularly efficient updates. In that case, the bounce and the ferromagnetic operator can be eliminated completely leaving only two types of operators, the antiferromagnetic and the off-diagonal one, which are converted into one another whenever a loop head arrives at a given operator. The second property is that the loop does not branch, which simply means that there is only one active loop head at any given time. However, branching loops with multiple active heads are possible and, in fact, do occur for another seemingly simple model, the transverse field Ising model [30].

What makes the directed loop algorithm [36, 37, 38] stand out in comparison to the one described above is that it does not view each leg and each entrance direction as being equal but instead assigns possibly different exit probabilities depending on the entrance leg, operator type and exit leg. There are many different choices which all fulfill detailed balance but may lead to drastically different autocorrelation times. The optimal set of probabilities is the one that minimizes the autocorrelation time as well as the time needed to build a loop.

## 3.3. Measurements

In general, expressions for measurements have to be carefully derived from the expression

$$\langle A\rangle = \frac{\sum\limits_{|\psi\rangle}\sum\limits_{S_M}\langle A\rangle W_{\psi,S_M}}{\sum\limits_{|\psi\rangle}\sum\limits_{S_M}W_{\psi,S_M}}\,. \tag{3.19}$$

There are some measurements which are particularly easy to make. One of the most fundamental quantities in quantum Monte Carlo simulations is the energy $E = \langle H\rangle$ whose expression is based on (3.5):

$$\begin{aligned}\langle H\rangle &= \frac{1}{Z}\sum_{n=0}^{\infty}\frac{(-\beta)^n}{n!}\sum_{\{\phi\}_n}\langle\psi|H|\phi_1\rangle\ldots\langle\phi_{n-1}|H|\psi\rangle \\ &= \frac{1}{Z}\sum_{n=1}^{\infty}\frac{(-\beta)^n}{n!}\sum_{\{\phi\}_n}\langle\psi|H|\phi_1\rangle\ldots\langle\phi_{n-1}|H|\psi\rangle \\ &= -\frac{\langle n\rangle}{\beta}\,.\end{aligned} \tag{3.20}$$

In going from the first to the second line, one of the expectation values in the product is used to represent the measurement process. The first sum over all expansion orders in the second line can safely be extended to start from $n = 0$ because that term does not contribute at all. The result shows that the energy is directly proportional to the expansion order of the model. The same expression is true for all terms that make up the Hamiltonian, i.e. if the Hamiltonian can be written as a sum of $i$ terms $H_i$, then the expectation value is proportional to the average number of occurrences of this operator in the operator sequence. Another type of operator that can be measured with relative ease are those that are diagonal in the basis used in the simulation. They can be measured on each operator time slice and then averaged over the entire operator time direction. Winding numbers for each worldline are calculated by starting at the bottom and, for each spatial dimension, counting how off-diagonal operators are encountered that move the worldline to a neighboring site. Of course, one has to define a positive and a negative direction for each dimension. Dividing this number by the linear size of the system gives the winding number which is used in the calculation of the helicity modulus or for machine learning purposes as will be detailed later. Measuring off-diagonal operators is more involved and requires more work. The interested reader is referred to [39] for more information.

# 4. The sign problem

The power of Monte Carlo methods for evaluating highly dimensional integrals such as the partition sum lies in the fact that the error of the desired quantity decreases with the square root of Monte Carlo steps regardless of the dimensionality of the underlying integral, see (1.11). Another interpretation, in the spirit of its applications in physics, is to say that Monte Carlo methods allow to explore the relevant parts of the exponentially fast growing Hilbert in polynomial time. Unlike the absolute error, the relative error, i.e. the ratio of the square root of the variance and the mean value, can show dramatically different scaling behavior with the number of Monte Carlo steps which happens for models that suffer from the so-called sign problem [40]. A model with sign problem has statistical weights that are both positive and negative numbers preventing the straightforward application of Monte Carlo techniques because the weights cannot be interpreted as probabilities by simple normalization. This chapter first shows the detrimental effects the sign problem can have on the statistical error of Monte Carlo measurements and then finishes with a discussion of the various approaches to solving and circumventing the sign problem.

## 4.1. Monte Carlo calculations with sign problem

It is in principle known how to deal with Monte Carlo calculations of models with sign problem: The goal is always to evaluate the expectation value of some observable $\mathcal{O}$ along the Markov chain. To be able to propose new states for the Markov Chain, the prefactor of the weight is transferred to the observable, creating an auxiliary ensemble for the configuration space where all weights are the absolute values of the original weights [41]:

$$\langle \mathcal{O} \rangle = \frac{\sum \mathcal{O}(C) w(\mathcal{C})}{\sum w(\mathcal{C})} = \frac{\sum \mathcal{O}(C)\sigma(\mathcal{C})|w(\mathcal{C})|}{\sum \sigma(\mathcal{C})|w(\mathcal{C})|} = \frac{\langle \mathcal{O} \cdot \sigma \rangle_{\text{abs}}}{\langle \sigma \rangle_{\text{abs}}} . \tag{4.1}$$

The subscript "abs" denotes that the averaging process takes place in the auxiliary ensemble. The true expectation value of the observable $\langle \mathcal{O} \rangle$ is thus reconstructed as the ratio between the expectation value $\langle \ldots \rangle_{\text{abs}}$ in the modified ensemble with absolute statistical weights $|w(\mathcal{C})|$ and the expectation value of the sign of the weight, which has now taken on the role of an observable.

While this procedure is theoretically well-defined for any situation, it is often of little use in practical simulations. The reason being that in general, the expectation value of the sign $\langle \sigma \rangle_{\text{abs}}$ decreases exponentially [42] with the number of particles $N$, i.e. the system size, and inverse temperature $\beta$:

$$\langle \sigma \rangle_{\text{abs}} = \exp\left(-\beta N \Delta f\right) . \tag{4.2}$$

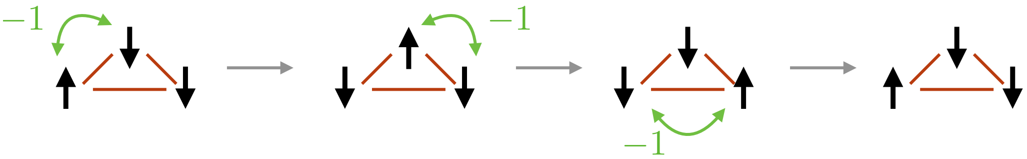


**Figure 4.1.:** Illustration of the origin of the sign problem for a triangular plaquette of spins where each exchange contributes one sign change along with its weight.

The additional factor $\Delta f = f_{\text{fermion}} - f_{\text{abs}}$ is the difference in the free energy densities of the original fermionic system and the one with absolute weights. Exponentially small quantities with large variance (the sign takes values of $\pm 1$) are very difficult to measure. The relative error of the sign can be estimated to be

$$\frac{\Delta \sigma}{\langle \sigma \rangle} \propto \frac{\exp\left(\beta N \Delta f\right)}{\sqrt{M}}, \tag{4.3}$$

which shows that for the relative error to become small enough to make any meaningful statement, the number of sweeps $M$ has to be of the order of the exponential is system size and inverse temperature. For most physical systems of interest, it is unfeasible to be able to realize this number of sweeps for any but the smallest system sizes and / or higher temperatures.

## 4.2. The physical origin of the sign problem

The origin of the sign problem can be traced back to the exchange statistics of fermions which causes a sign change every time two fermions are exchanged. This is very obvious in worldline simulations of fermions but can also be shown to be the underlying cause for bosons on frustrated lattices simulated within worldline techniques [40] and even for Monte Carlo methods based on auxiliary fields [43]. As an example, consider a triangular plaquette of spins that interact via an antiferromagnetic interaction, see Fig. 4.1. In the worldline picture, the spins propagate through imaginary time and may change places. Each of these exchanges is entailed by a change of sign of the weight. On bipartite lattices, the number of exchanges is constrained to by the periodic boundary conditions in imaginary time to be an even integer, whereas on frustrated lattices like the triangular lattice this constraint is lifted. As illustrated in the figure, the periodic boundary conditions in imaginary time can be respected by worldline configurations with either an odd number of exchanges, or, if for example an exchange is undone right away, by worldlines with an even number of exchanges.

Interestingly, this means that simulations of fermions and hardcore bosons are identical up to the presence or absence of a sign change when two word-lines trade places. Here, the fermion sign problem appears even on bipartite lattices because it is truly caused by the exchange statistics of the fermions which requires to calculate the sign of the permutation of the fermion state along operator time in order to distinguish bosons from fermions. Similarly, antiferromagnetic spin systems are simulated by the same rules as ferromagnetic ones and differ only by the fact that a sign change has to be introduced for spin exchanges. Despite giving a good first impression on the origin of the fermion sign problem, this view is only strictly true for worldline approaches as will be discussed in later chapters.

## 4.3. Circumventing the sign problem

Experience shows that many of the systems believed to exhibit some very intriguing phenomena suffer from the sign problem, most prominently fermionic systems on frustrated lattices. One thus has to wonder whether there is some way to circumvent the consequences of the sign problem.

### Ignoring the sign problem

The most attractive "solution" would be to ignore the sign problem altogether, i.e. sample in the auxiliary ensemble but not take the signs into account. However, there is no justification for believing that evaluating observables while ignoring the sign problem should yield valid results. This is due to the fact that the physical information is contained not in the first but in the very last digits of the observables: Considering again (4.1), one can rewrite that equation as

$$\langle \mathcal{O} \rangle = M \frac{\langle \mathcal{O} \rangle^{+} - \langle \mathcal{O} \rangle^{-}}{M^{+} - M^{-}}, \tag{4.4}$$

where $M^{\pm}$ denotes the number of states visited during the Markov procedure with positive and negative prefactor, respectively. It was shown previously that the expectation value of the sign decreases exponentially fast, which implies that the number of configurations with a negative prefactor approaches that with a positive prefactor. Because the expectation value of the operator remains finite on finite lattices, it has to show a similar scaling behavior, which means in particular that the difference of the expectation values in the two respective ensembles also decreases exponentially fast to ensure convergence. Ultimately, the expectation values only differ further and further to the right of the decimal point, which is where the physical information is contained. Therefore, there is no reason to believe that the sign can simply be ignored and that the same information can be read off from the first few digits. This was also famously illustrated in an early study on the sign problem [42], where the superconducting properties of the doped Hubbard model were studied. An adapted plot of the superconducting susceptibility versus the temperature is shown in Fig. 4.2. It illustrates clearly that simply ignoring the sign will lead to wrong results for correlation functions.

### Getting rid of the sign problem

If the sign problem cannot be ignored, the next best possibility of dealing with it is to find a way of getting rid of it. In fact, this is always possible because the sign problem depends on the simulation basis. In the energy eigenbasis, there is no sign problem because the basis states are also eigenstates of the Hamiltonian. However, switching to this basis requires the diagonalization of the full Hamiltonian which in itself is exponentially difficult

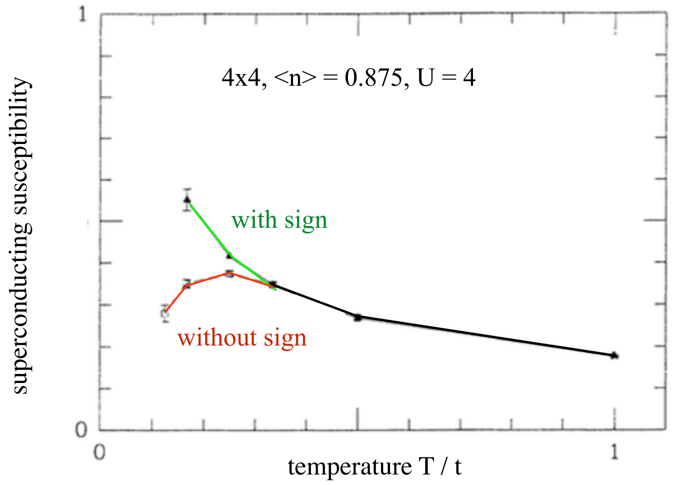


**Figure 4.2.:** The detrimental effect of ignoring the sign problem is illustrated by comparing the measurement of the superconducting susceptibility in a doped Hubbard model with repulsive interaction on a small 4x4 square lattice with and without including the sign problem. Figure adapted from [42]

and would make the use of Monte Carlo methods obsolete anyway. There are other bases which do not have a sign problem, but they are rare and their construction appears simple only in hindsight [44, 45, 46, 47]. It is possible to explain why certain Monte Carlo flavors can simulate certain models without suffering from the sign problem with the help of group theory [48] but currently, this serves mostly as an a posteriori explanation rather than as a helpful tool to construct interesting Hamiltonians. Note that even when a basis without sign problem can be found, the simulation may still take exponentially long to deliver converging results because the sign problem is at worst an NP-hard problem [40].

### Alternative perspectives

Another option that lies somewhere in between the two extremes of ignoring the sign problem and finding a sign problem free basis is to study effective models or designer Hamiltonians that exhibit the desired characteristics of the full model while being sign problem free. This approach has been used to study antiferromagnetism in metals [49], the properties of certain types of phase transitions [50] and many other models with properties of interest [51, 52]. This is a new perspective on the sign problem offering an interesting route towards studying a host of new models. However, there are still open problems that appear to be out of reach, as for example studying a system of fermions with emergent topological properties that are not explicitly built-in to a sign problem free Hamiltonian. These examples show that viewing the sign problem from a different perspective can be a very fruitful endeavor that supplements the exact approach.

This thesis provides two additional, novel perspectives on the sign problem that shed new light on the inner workings of the sign problem: Chapter 5 discusses the sign problem from the entanglement perspective while chapter 6 uses machine learning as its reference point.

# Part II.

# Disentangling and learning

# 5. Entanglement

Entanglement is one of the most intriguing phenomena that separates the classical world from the realm of quantum physics. Through entanglement, quantum mechanical objects like photons, electrons, or spins can be interwoven into a collective state that can no longer be described as a simple product state of single-particle wave functions, and is thus said to be entangled [53].

In this chapter a novel method to quantify the entanglement in many-fermion systems is developed which allows extracting physical properties that remain hidden to conventional, correlator based measurements. The method brings about new numerical difficulties and it is shown how to solve them. Following some benchmark results, the connection to the sign problem described in the preceding chapter is explored. The key results have been published in [54, 55, 25] from which this chapter was compiled.

## 5.1. Entanglement and its use in condensed matter

The concept of entanglement is typically associated with the physics of only a handful of quantum mechanical degrees of freedom as described in the seminal papers of Einstein, Podolsky and Rosen [53] and Bell [56]. Their work challenged the very existence of entanglement in quantum mechanics and paved the way for devising experiments that are capable of answering this question, respectively. Being so crucial for the foundation of quantum mechanics, these papers triggered enormous experimental efforts to realize the so-called Bell experiment that confirmed with ever increasing certainty that entanglement is part of our world [57, 58, 59, 60, 61, 62, 63, 64].

Beyond its conceptual relevance, entanglement has turned into a key resource in various fields of modern physics: In quantum information theory, it is exploited for storing and manipulating information in so-called qubits [65, 66, 67]. In condensed matter physics, entanglement has become increasingly appreciated as a measure to classify different states of quantum matter which cannot be distinguished by any local observable such as topologically ordered states [68, 69, 70, 71]. The probably stunning realization that oftentimes ground states of quantum many-body systems are often only lowly entangled has led to the development of a novel class of variational (tensor network) algorithms[72, 73, 74, 75].

Exploring the significance of entanglement in quantum many-body systems requires novel entanglement measures that allow dealing with an almost arbitrarily large number of interwoven quantum mechanical degrees of freedom instead of just a few. The starting point for such considerations are reduced density matrices, which can are calculated from a bipartition of a quantum many-body system into two complementary parts $A$ and $B$ as

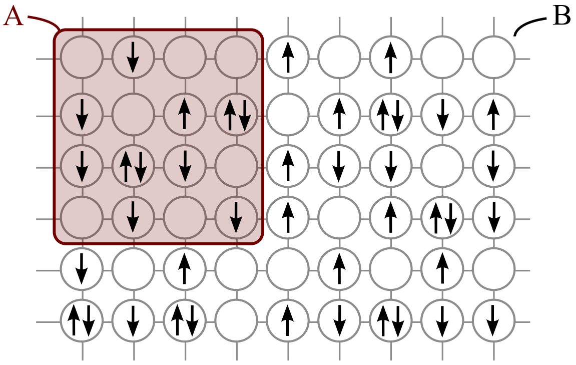


**Figure 5.1.:** The bipartition of a quantum many-body system into part $A$ and its complement $B$. Figure first appeared in [54].

illustrated in Fig. 5.1. Tracing out the degrees of freedom in one subsystem one can calculate a reduced density matrix for the other part, e.g. $\rho_A = \mathrm{Tr}_B(|\psi\rangle\langle\psi|)$. There are two ways in which one can proceed. One is to view the reduced density matrix as stemming from an actual Hamiltonian $\mathcal{H}_A$, the so-called entanglement Hamiltonian to which it is related by [76]

$$\rho_A = e^{-\mathcal{H}_A} . \tag{5.1}$$

However, accessing this spectrum is very difficult in quantum Monte Carlo methods [77, 78, 79, 80], which is why one focusses on another, also powerful quantity called the entanglement entropy. For its calculation, the information in the density matrix is condensed into a single number, for example by calculating the von Neumann entropy [66]

$$S(A) = -\mathrm{Tr}\left[\rho_A \log \rho_A\right] . \tag{5.2}$$

The von Neumann entropy is the most prominent member of a more general family of entanglement entropies, the so-called Renyi entropies [81] which are calculated from the density matrix as

$$S_n(A) = \frac{1}{1-n} \log\left(\mathrm{Tr}\left(\rho_A^n\right)\right) , \tag{5.3}$$

where the limit $n \to 1$ recovers the above von-Neumann entropy.

The strength of these entropic entanglement measures becomes apparent when considering the scaling of the entanglement entropy for varying sizes of the subsystem $A$ [82]. In contrast to conventional thermodynamic entropies the entanglement entropy is not extensive, but rather scales with the length of the boundary between the two partitions – the so-called boundary law [68] (which is often also referred to as area-law). Corrections to this prevalent boundary law have received widespread attention for their ability to classify different states of quantum matter [82]. For instance, it has been shown that the topological character of non-local order present in a quantum ground-state wavefunction can be revealed by a universal $O(1)$ correction $\gamma$ [70, 71] to the boundary law. Numerical simulations of quantum spin systems have subsequently shown that such an identification is indeed feasible and unambiguously revealed the long conjectured topological order present in ground states of certain frustrated quantum magnets [83, 84, 85]. Another important subleading correction that scales with the length of the boundary is a logarithmic term that reveals the presence of Goldstone modes in the system [86] or may allow studying emerging low energy degrees of freedom at quantum critical points [87]. Finally, there are multiplicative corrections to the boundary law, which may indicate the presence of a

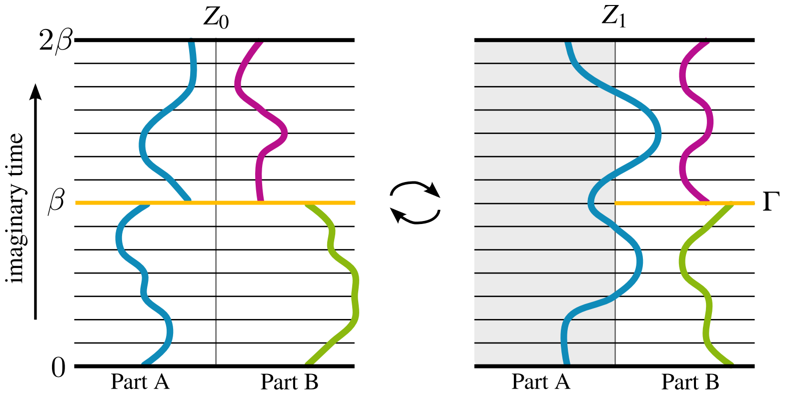


**Figure 5.2.:** Ensemble switching in a worldline picture. The left side shows the square of the regular partition sum $\mathcal{Z}^2$ where all worldlines have to be $\beta$-periodic. The right side shows a configuration of allowed worldlines in the $\mathcal{Z}[A,2,T]$ partition sum, where $2\beta$-periodicity is enforced in part $A$ of the system but part $B$ remains $\beta$-periodic. Figure first appeared in [25].

Fermi surface [88, 89]. Devising a method to determine these corrections in interacting many-fermion systems are the focus of the work presented in this chapter.

## 5.2. Entanglement entropies from Monte Carlo simulations

Monte Carlo methods mostly use an approach called the replica trick[90, 91] to calculate entanglement entropies. One speaks of replicas because the calculation of the Renyi entropy involves taking powers of the reduced density matrices which generates copies of the system as will be demonstrated in the following. The first step is to determine the reduced density matrix $\rho_A$ by tracing out the degrees of freedom in subsystem $B$

$$\rho_A = \mathrm{Tr}_B \rho\,. \tag{5.4}$$

The following deals mostly with calculating the second Renyi entropy as a concrete example but Renyi entropies of higher order are determined in much the same fashion. For $n = 2$, it takes the form

$$S_2(A) = -\log\left(\frac{\mathrm{Tr}_A\left(\rho_A^2\right)}{\mathcal{N}^2}\right). \tag{5.5}$$

Note that in (5.5) the reduced density matrix $\rho_A$ is squared *before* the remaining degrees of freedom of subsystem $A$ are traced out and that the density matrix has a normalization factor $\mathcal{N}$. This normalization factor enters as the square of the usual partition sum:

$$\mathcal{N}^2 = (\mathrm{Tr}\,\rho)^2 = \mathcal{Z}^2\,. \tag{5.6}$$

The numerator can also be interpreted as a partition sum, albeit with a more complex definition:

$$\begin{aligned}\mathrm{Tr}_A\left(\rho_A'^2\right) &= \sum_{\mathcal{A},\mathcal{A}',\mathcal{B},\mathcal{B}'} \langle \mathcal{A}\mathcal{B}'|\,\rho'\,|\mathcal{A}'\mathcal{B}'\rangle \langle \mathcal{A}'\mathcal{B}|\,\rho'\,|\mathcal{A}\mathcal{B}\rangle \\ &\equiv \mathcal{Z}[A,2,T]\,,\end{aligned} \tag{5.7}$$

The additional arguments indicate the subsystem, the order of the Renyi entropy and the temperature for which it shall evaluated. To better understand the nature of the two respective partition sums, $\mathcal{Z}^2$ and $\mathcal{Z}[A,2,T]$, they can be turned into their respective worldline representations as illustrated in Fig. 5.2. On the left side, the worldline representation of $\mathcal{Z}^2$ from the denominator in Eq. (5.5) is depicted with two sets of $\beta$-periodic worldlines extending from 0 to $\beta$ and from $\beta$ to $2\beta$, respectively. These arise from the squaring of the partition sum in the denominator and the $\beta$-periodicity is due to the trace operator applied to both partition sums separately. The right-hand side shows a worldline representation for the partition function $\mathcal{Z}[A,2,T]$ of the numerator of Eq. (5.5), where two replicas of the system are partially connected in imaginary time because only part of the degrees of freedom were traced out before taking the square.

The important question is how to numerically calculate these two partition functions, and the answer depends on the particular flavor of Monte Carlo. In worldline based methods, implementing the respective imaginary time boundary conditions is fairly straightforward [91, 92]. Being based on partial traces of density matrices, the replica trick can also be applied to ground state methods such as variational Monte Carlo which is also relatively easily done [93] because the wave function is a central object in the Monte Carlo procedure. Neither of those statements apply to the finite temperature or the projective formulation of DQMC, respectively, requiring a reformulation of the replica trick as will be shown now.

## 5.3. Determinant QMC and the replica trick

In the finite temperature variant of DQMC, the fermions and their respective worldlines are traced out and the object of the Monte Carlo simulation is to sample auxiliary field configurations. This creates a problem when implementing the replica scheme described above because a crucial ingredient is the propagation of the worldlines in imaginary time and the question of how they can be matched with the boundary conditions imposed by the trace. Before delving into the details of how to modify the replica trick, it should be noted that there is an alternative approach which relies on the fact that entanglement entropies can be readily calculated for free fermions [94, 95]. This can be exploited because after the Hubbard-Stratonovich transformation, the fermions are, in fact, free which allows one to sample powers of the reduced density matrix [96]. It turned out and will be shown later that this approach is unfortunately ridden with large sampling errors for all but the smallest subsystems and weak interactions, severely limiting its applicability.

### 5.3.1. Implementation of the replica trick in DQMC

The implementation of the replica trick in DQMC begins with writing down an expression for the density matrix in both the finite temperature and the projective formulation. At zero temperature, the density matrix is given by

$$\rho = \frac{|\psi\rangle\langle\psi|}{\langle\psi|\psi\rangle} = \frac{|\psi\rangle\langle\psi|}{\mathcal{N}}, \tag{5.8}$$

while at finite temperatures it takes the form

$$\rho = \frac{\exp(-\beta\mathcal{H})}{\mathrm{Tr}(\exp(-\beta\mathcal{H}))} = \frac{\exp(-\beta\mathcal{H})}{\mathcal{N}}. \tag{5.9}$$

Both expressions are explicitly normalized by the normalization constant $\mathcal{N}$ in the denominator. This will play an important role in the following derivation of the algorithm. The notation

$$\rho = \frac{1}{\mathcal{N}}\rho' \tag{5.10}$$

will be used as a unifying notation where the choice for $\rho'$ and $\mathcal{N}$ depends on the context.

### Finite temperature formulation

In the finite temperature algorithm,the statistical weight $W(s,\psi)$ for an arbitrary pair of a fermion state $\psi$ and an auxiliary field configuration $s$ is given as the determinant of a matrix:

$$W(s,\psi) = \langle\psi|\,\boldsymbol{B}(s)\,|\psi\rangle\,. \tag{5.11}$$

A crucial step in the original formulation of the finite temperature algorithm was to realize that the grand-canonical trace over these Slater determinants can be recast as a *single* determinant

$$\mathrm{Tr}\ \langle\psi|\,\boldsymbol{B}(s)\,|\psi\rangle = \det(\mathbb{1} + \boldsymbol{B}(s))\,, \tag{5.12}$$

allowing to sample only configurations of the auxiliary field. The importance of this simplification cannot be overstated: Without rewriting the sum over fermion states in this way, the sampling procedure would be numerically too involved to carry out. However, a straightforward implementation of the partition function $\mathcal{Z}[A,2,T]$ runs exactly into this problem. Due to the two replicas in imaginary time, the weight is given as the product two determinants which is cannot be reduced a single determinant or the product of two determinants except for the special cases where the subsystem B contains no or all sites, respectively:

$$\begin{aligned}\mathcal{Z}[A,2,T] &= \sum_{\{s\}}\sum_{\mathcal{A},\mathcal{A}',\mathcal{B},\mathcal{B}'} \langle\mathcal{A}\mathcal{B}'|\,\rho'\,|\mathcal{A}'\mathcal{B}'\rangle\,\langle\mathcal{A}'\mathcal{B}|\,\rho'\,|\mathcal{A}\mathcal{B}\rangle \\ &= \sum_{\{s\}}\sum_{\mathcal{A},\mathcal{A}',\mathcal{B},\mathcal{B}'} \det\left(s,\mathcal{A}\mathcal{B}'|\mathcal{A}'\mathcal{B}'\right)\det\left(s,\mathcal{A}'\mathcal{B}|\mathcal{A}\mathcal{B}\right). \end{aligned} \tag{5.13}$$

For the sake of brevity the two determinants are denoted as $\det(s,\mathcal{A}\mathcal{B}'|\mathcal{A}'\mathcal{B}')$ and $\det(s,\mathcal{A}'\mathcal{B}|\mathcal{A}\mathcal{B})$.

The key idea is to modify the replica scheme in such a way that the statistical weights can again be simplified to yield only a single Slater determinant thus allowing to take a grand-canonical trace of the form (5.12).

This is achieved by artificially enlarging the system by considering an additional copy $B'$ of subsystem $B$, which will be used to selectively evolve subsystem $B$ in imaginary time. In particular, it will be shown that an imaginary-time dependent Hamiltonian of the form

$$\widetilde{\mathcal{H}}(\tau) = \mathcal{H}_{AB}\,\Theta(\tau)\,\Theta(\beta-\tau) + \mathcal{H}_{AB'}\,\Theta(\tau-\beta)\,\Theta(2\beta-\tau) \tag{5.14}$$

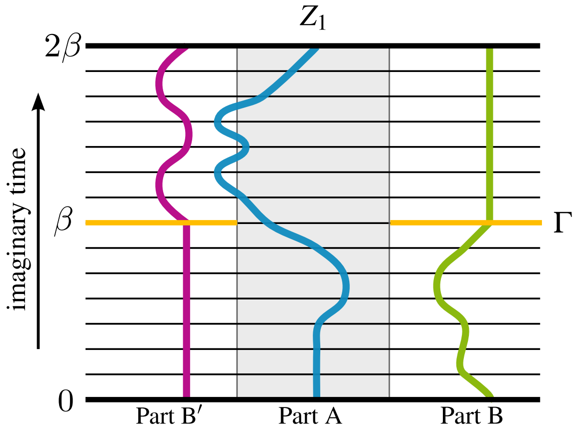


**Figure 5.3.:** The enlarged simulation cell used to port the replica trick to DQMC simulations. Figure first appeared in [25].

will give direct access to the partition sum $\mathcal{Z}[A,2,T]$. If a subsystem does not appear in the index of the two respective Hamiltonians, there are no operators acting on any of the degrees of freedom in that particular subsystem, i.e. the first term contains no operators with basis in $B'$ and the second term contains no operators with basis in $B$. A worldline representation of this Hamiltonian is given in Fig. 5.3.

To proof this statement, (5.14) is regarded as a given model Hamiltonian whose physics should be explored at an arbitrary temperature, suggestively written as $2\beta$. A given basis state is denoted by $|\psi\rangle = |\mathcal{A},\mathcal{B},\mathcal{B}'\rangle$ leading to the partition sum

$$\widetilde{\mathcal{Z}} = \sum_{\{\mathcal{A},\mathcal{B},\mathcal{B}'\}} \left\langle \mathcal{A},\mathcal{B},\mathcal{B}' \right| \exp\left(-\beta\widetilde{\mathcal{H}}\right) \left|\mathcal{A},\mathcal{B},\mathcal{B}'\right\rangle .$$

Although there are two propagation operators, the weight of the system is still be given as a *single* Slater determinant because only one expectation value has to be evaluated. Inserting a resolution of unity in between the two exponential operators gives

$$\widetilde{\mathcal{Z}} = \sum_{\{\mathcal{A},\mathcal{B},\mathcal{B}',\mathcal{C},\mathcal{D},\mathcal{D}'\}} \left\langle \mathcal{A},\mathcal{B},\mathcal{B}' \right| \exp\left(-\beta\mathcal{H}_{AB'}\right) \left|\mathcal{C},\mathcal{D},\mathcal{D}'\right\rangle \left\langle \mathcal{C},\mathcal{D},\mathcal{D}' \right| \exp\left(-\beta\mathcal{H}_{AB}\right) \left|\mathcal{A},\mathcal{B},\mathcal{B}'\right\rangle . \tag{5.15}$$

In the right term, states of $B'$ do not appear in the Hamiltonian which requires $\mathcal{B}' = \mathcal{D}'$ for any non-vanishing term contributing to this partition function. Similarly, the left term enforces the constraint $\mathcal{B} = \mathcal{D}$. Further, if subsystems $B$ and $B'$ do not appear in the Hamiltonian for a given imaginary time interval they not only do not evolve, but also remain completely decoupled from the rest of the system over this interval. As a result, they will also not affect the statistical weight

$$\begin{aligned} &\left\langle \mathcal{C},\mathcal{B},\mathcal{B}' \right| \exp\left(-\beta\mathcal{H}_{AB}\right) \left|\mathcal{A},\mathcal{B},\mathcal{B}'\right\rangle \\ &= \left(\langle\mathcal{C},\mathcal{B}| \otimes \left\langle\mathcal{B}'\right|\right) \exp\left(-\beta\mathcal{H}_{AB}\right) \left(|\mathcal{A},\mathcal{B}\rangle \otimes \left|\mathcal{B}'\right\rangle\right) \\ &= \langle\mathcal{C},\mathcal{B}| \exp\left(-\beta\mathcal{H}_{AB}\right) |\mathcal{A},\mathcal{B}\rangle \end{aligned}$$

and can safely be ignored. Finally, renaming $\mathcal{C}$ to $\mathcal{A}'$ to match earlier notation the following simplified expression is obtained:

$$\widetilde{\mathcal{Z}} = \sum_{\mathcal{A},\mathcal{A}',\mathcal{B},\mathcal{B}'} \left\langle \mathcal{A}\mathcal{B}' \right| \exp\left(-\beta\mathcal{H}_{AB'}\right) \left|\mathcal{A}'\mathcal{B}'\right\rangle \left\langle \mathcal{A}'\mathcal{B} \right| \exp\left(-\beta\mathcal{H}_{AB}\right) |\mathcal{A}\mathcal{B}\rangle = \mathcal{Z}[A,2,T] , \tag{5.16}$$

This is precisely the expression for the sought-after partition function $\mathcal{Z}[A,2,T]$ in a form that relies only on single determinants thus allowing to take the grand-canonical trace (5.12).

### Ground-state formulation

When considering the ground-state DQMC algorithm only minor modifications to the above scheme have to be implemented. The normalization constant $\mathcal{N}$ introduced in (5.10) is now given as

$$\mathcal{N} = \langle\psi|\psi\rangle = \sum_{|\mathcal{A}\rangle} \langle\psi|\mathcal{A}\rangle \langle\mathcal{A}|\psi\rangle = \mathrm{Tr}\left(|\psi\rangle\langle\psi|\right).$$

In chapter 2, it was shown that the ground-state wave function $|\psi\rangle$ is obtained by projection of a trial wave function

$$|\psi\rangle = \lim_{\Theta\to\infty} e^{-\Theta\mathcal{H}} |\psi_T\rangle\,. \tag{5.17}$$

Inserting this projection into the definition of the Renyi entropy in Eq. (5.5), leads to an expression for the canonical $\mathrm{Tr}\,\rho_A'^2$ that looks very similar to the finite temperature expression of the grand-canonical trace for $\widetilde{\mathcal{Z}}$ in Eq. (5.16) discussed above

$$\mathrm{Tr}\,\rho_A'^2 = \lim_{\Theta\to\infty} \sum_{\mathcal{A},\mathcal{A}',\mathcal{B},\mathcal{B}'} \left\langle\mathcal{A}\mathcal{B}'\right| \exp\left(-\Theta\mathcal{H}\right)|\psi_T\rangle\langle\psi_T| \exp\left(-\Theta\mathcal{H}\right)\left|\mathcal{A}'\mathcal{B}'\right\rangle \tag{5.18}$$

$$\times \left\langle\mathcal{A}'\mathcal{B}\right| \exp\left(-\Theta\mathcal{H}\right)|\psi_T\rangle\langle\psi_T| \exp\left(-\Theta\mathcal{H}\right)|\mathcal{A}\mathcal{B}\rangle\,, \tag{5.19}$$

where the only difference is the appearance of the density matrices $|\psi_T\rangle\langle\psi_T|$. However, this comes in handily because it allows taking the grand-canonical trace to make use of Eq. (5.12). The occurrence of these density matrices eliminates all states with an occupation number different from the one of the trial wave function and thus, the sampling is effectively done in the canonical ensemble as desired. This surprising simplification, inserting density matrices at the right place, which turns a finite temperature algorithm into a ground state algorithm comes at the cost of decreased stability of the algorithm. This issue is discussed at length in a later section of this chapter.

### Calculation of higher Renyi entropies

The above discussion is focussed on Renyi entropies of order 2, but the algorithm can easily be extended to also compute higher Renyi entropies. For the calculation of the $n$-th Renyi entropy via the replica trick, imaginary time has to be split into $n$ segments which would contribute one determinant each in Eq. (5.13). It would thus be necessary to introduce $n$ replicas of the subsystem $B$ and work in an overall system of size $N_A + n\cdot N_B$. This scheme was successfully implemented in [80] and used to calculate the entanglement spectrum.

### Ensemble switching

The replica scheme outlined above allows directly sampling partition functions of the form $\mathcal{Z}[A,2,T]$. For the calculation of the Renyi entropy, however, one needs to determine the *ratio* of the partition functions $\mathcal{Z}[A,2,T]$ and $\mathcal{Z}^2$ as given in Eq. (5.5). This ratio can be

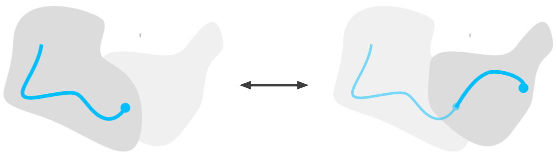


**Figure 5.4.:** Schematic illustration of the ensemble switching method to calculate the Renyi entropy of Eq. (5.5). The random walk is started in one of the configuration spaces corresponding to the ensembles appearing in the numerator and the denominator of Eq. (5.5), denoted by $\mathcal{C}(\mathcal{Z}[A,2,T])$ and $\mathcal{C}(\mathcal{Z}^2)$, respectively. Whenever a configuration that is admissible in both ensembles is encountered, the relative weights are compared and the Metropolis criterion is used to decide in which ensemble the sampling process continues.

directly accessed [97] by cleverly switching between the two ensembles without explicitly calculating the individual partition sums.

Imagine a two-state simulation, where the weight of two states is given by $w_1$ and $w_2$. For any simulation fulfilling detailed balance, the random walk would spend $N_1 = w_1/(w_1+w_2)$ steps in state 1 and $N_2 = w_2/(w_1+w_2)$ steps in state 2. Thus, the ratio of the weights $w_1/w_2$ corresponds precisely to the relative time spent in the two respective states. This statement is readily generalized [97] to a situation where a random walk switching back and forth between two ensembles whose partition functions evaluate to what were previously the weights $w_1 = \mathcal{Z}[A,2,T]$ and $w_2 = \mathcal{Z}^2$, respectively. The ratio of relative time spent sampling each of the two ensembles can therefore be used to calculate the entanglement entropy

$$S_2(A) = -\log\left(\frac{\mathcal{Z}[A,2,T]}{\mathcal{Z}^2}\right) = -\log\left(\frac{N_1}{N_2}\right). \tag{5.20}$$

In practical terms, the simulation is started in one of the two ensembles and after a fixed number of Monte Carlo steps, the weight of the current configuration is calculated in both ensembles and ensembles are switched according to Metropolis rules. When implementing this ensemble switching method, one benefits from an additional advantage of the DQMC framework. The configuration space of $\mathcal{Z}[A,2,T]$ and $\mathcal{Z}^2$ is exactly the same and the transition probabilities $p_{1\to2}$ and $p_{2\to1}$ are typically spread over the entire range $(0,1]$.

To estimate the ratio (5.5) it actually suffices to measure the expectation values of $\langle p_{1\to2}\rangle$ and $\langle p_{2\to1}\rangle$, which are the probabilities to switch ensembles from ensemble 1 to 2 and from ensemble 2 to 1, respectively. Using a Metropolis scheme, the probability to switch ensembles for a given configuration $c$ is given as

$$p_{1\to2} = \min\left(1, \frac{W_2(c)}{W_1(c)}\right),$$

where $W_1(c)$ and $W_2(c)$ are the weights in the respective ensembles. The configurations $c$ are configurations of the auxiliary field which each exist in both configuration spaces.

Writing out the ratio of the expectation values for the switching operation gives

$$\begin{aligned}
\frac{\langle p_{1\to 2}\rangle}{\langle p_{2\to 1}\rangle} &= \frac{\mathcal{Z}_2}{\mathcal{Z}_1} \cdot \frac{\sum\limits_{c\in\mathcal{C}(\mathcal{Z}_1)} \min\left(1, \dfrac{W_2(c)}{W_1(c)}\right) W_1(c)}{\sum\limits_{c\in\mathcal{C}(\mathcal{Z}_2)} \min\left(1, \dfrac{W_1(c)}{W_2(c)}\right) W_2(c)} \\
&= \frac{\mathcal{Z}_2}{\mathcal{Z}_1} \cdot \frac{\sum\limits_{W_2(c)<W_1(c)} W_2(c) + \sum\limits_{W_1(c)<W_2(c)} W_1(c)}{\sum\limits_{W_1(c)<W_2(c)} W_1(c) + \sum\limits_{W_2(c)<W_1(c)} W_2(c)} \\
&= \frac{\mathcal{Z}_2}{\mathcal{Z}_1}. 
\end{aligned} \tag{5.21}$$

Thus, the ratio of partition functions is estimated as the ratio of probabilities to switch from one ensemble to the other. These numbers can be obtained from two separate simulations.

It should further be noted that this approach does not require to iteratively build-up the subsystem $A$ from smaller blocks to achieve convergence [91], as it has been observed in the context of worldline Monte Carlo approaches where the overlap between the ensembles might become rather small. If needed, however, it would still be possible to conduct such an iterative build-up.

## 5.4. The Hubbard chain as a test case

The one-dimensional Hubbard chain provides the perfect testing ground for the applicability and numerical efficiency of the proposed implementation of the replica trick. This model is well suited because besides not suffering from the sign problem, it can be studied using MPS based algorithms which allow to generate high precision data for comparison. Further, some zero temperature results are known exactly from the conformal field theory description of the gapless theory governing the physics of the ground state of the Hubbard chain in the presence of a finite on-site interaction $U$. To test the finite temperature algorithm, the crossover of the Renyi entropy the low-temperature entanglement entropy to the high-temperature thermal entropy is studied.

### 5.4.1. Zero-temperature physics

In the presence of a repulsive on-site interaction $U > 0$ the ground state of the half-filled Hubbard chain is well known to be a Mott insulator exhibiting quasi-long range antiferromagnetic order. This means that at zero temperature charge fluctuations are frozen out entirely for any $U > 0$, while the localized spin degrees of freedom interact via an effective Heisenberg exchange of order $t^2/U$ thereby building up quasi-long range antiferromagnetic order. The system thus remains gapless and can be described in terms of a conformal field theory with a central charge $c = 1$ corresponding to the number of

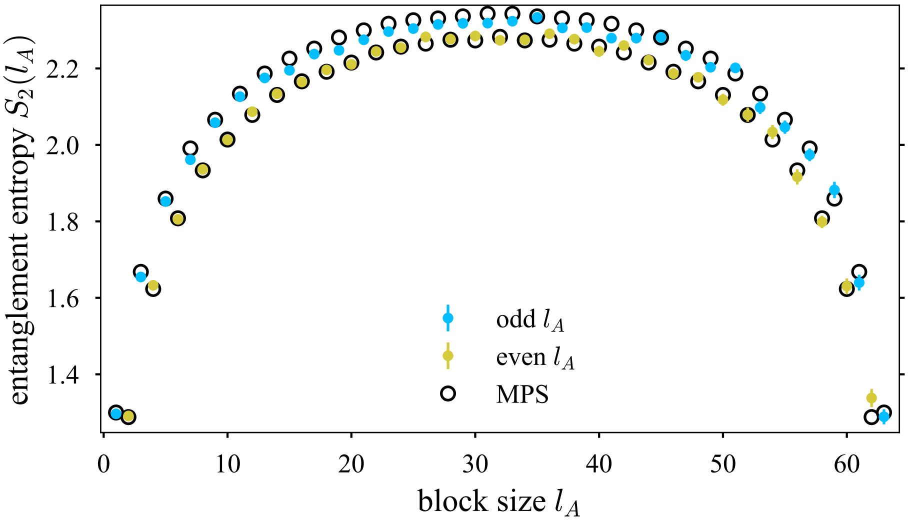


**Figure 5.5.:** The entanglement entropy $S_2$ of a periodic, half-filled Hubbard chain with $L = 64$ sites in the presence of a repulsive on-site interaction $U/t = 2$. Shown is the entanglement entropy versus the length of the subsystem $l_A$. The numerical data obtained with the zero-temperature DQMC algorithm ($\Theta = 10$) is in good agreement with DMRG reference data for the same system (open circles).

gapless modes. The entanglement entropy of such a gapless one-dimensional system is known [98, 99] to exhibit a logarithmic correction to the boundary law, which for a one-dimensional system simply states that the entanglement entropy is a constant for any bipartition. The logarithmic correction, however, does reflect the relative size of the two subsystems in the bipartition and for all Renyi entropies follows the general form [99]

$$S_n(l_A) = \frac{c}{6\eta}\left(1+\frac{1}{n}\right)\ln\left[\frac{\eta L}{\pi}\sin\left(\frac{\pi l_A}{L}\right)\right] + O(1)\,, \tag{5.22}$$

where $c$ is the central charge of the conformal field theory, $L$ is the overall system length and $l_A \leq L$ is the length of subsystem $A$. Open and periodic boundary conditions correspond to $\eta = 2$ and $\eta = 1$, respectively, and further subleading corrections of order $O(1)$ in the system size are neglected. Numerical results obtained with the zero-temperature DQMC algorithm for an open chain of length $L = 64$ are shown in Fig. 5.5. The DQMC data is generally in good agreement with quasi-exact results obtained using density matrix renormalization group (DMRG) simulations. The slight underestimation of the DQMC data hints at the fact that the projection time might not have been chosen sufficiently long. Choosing it to be longer, however, proves to be difficult due to stability issues that will be discussed later in more detail.

## 5.4.2. Thermal crossover of the entanglement

At finite temperatures, both quantum and thermal fluctuations contribute to all entropies including the Renyi entropy of interest here. Upon increasing the temperature, the relative contributions of these two types of fluctuations of course change. As a result, the Renyi entropy shows a crossover from a boundary law (with logarithmic corrections) at zero temperature to a more conventional extensive behavior (i.e. a volume law) at high

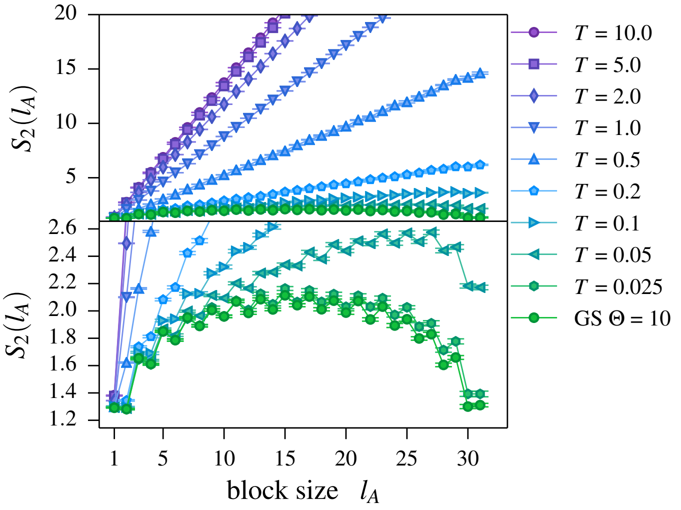


**Figure 5.6.:** The thermal crossover of the entanglement entropy for a $L = 32$ site half-filled Hubbard chain with $U/t = 2$. While at high temperatures the entanglement entropy exhibits a volume law $S = \log 4 \cdot L$, it crosses over to a boundary law at low temperatures with a characteristic arc-like structure. Figure first appeared in [54].

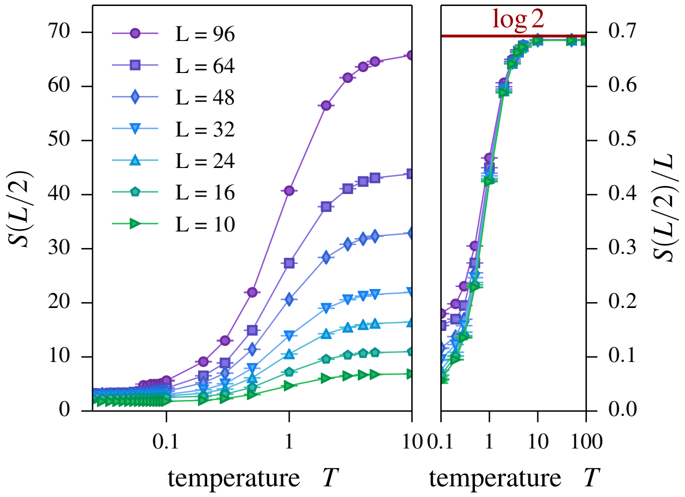


**Figure 5.7.:** Thermal crossover of the Renyi entropy for a half-filled Hubbard chain with $U/t = 1$. Shown is $S_2(L/2,T)$ (left panel) and a rescaled $S_2(L/2,T)/L$ (right panel), which at high temperatures converges to $\log 2$ for different system sizes. Figure first appeared in [54].

temperatures of the form

$$S(l_A) = l_A \cdot \log 4 \,, \tag{5.23}$$

simply counting the number of possible states in the subsystem.

This thermal crossover of the Renyi entropy from the regime dominated by quantum fluctuations to a thermal entropy at high temperatures can easily be observed in DQMC simulations. It is illustrated in Fig. 5.6 for a half-filled Hubbard chain of length $L = 32$ with intermediate on-site interaction $U/t = 2$ in a temperature range $0.025 \leq T \leq 5$ (for $t = 1$). With increasing temperature the arc-like structure of the low-temperature entanglement entropy disappears and gives way to the simple linear form of an extensive thermal entropy. This thermal crossover is also reflected in Fig. 5.7 where the Renyi entropy $S_2(L/2)$ of an equal-size bipartition of the chain for different system sizes is plotted versus the temperature. In particular, it is possible to collapse high temperature data when rescaling the calculated Renyi entropies by the respective system size, see the right panel of Fig. 5.7.

A suitable measure to quantitatively determine the crossover temperature $T^*$, below which a finite-sized system is effectively in its ground state is the so-called purity

$$\mathcal{P} = \exp\left(-S_2(L)\right), \tag{5.24}$$

which becomes 1 for a quantum mechanical ground state, since the entropy $S_2(L)$ needs

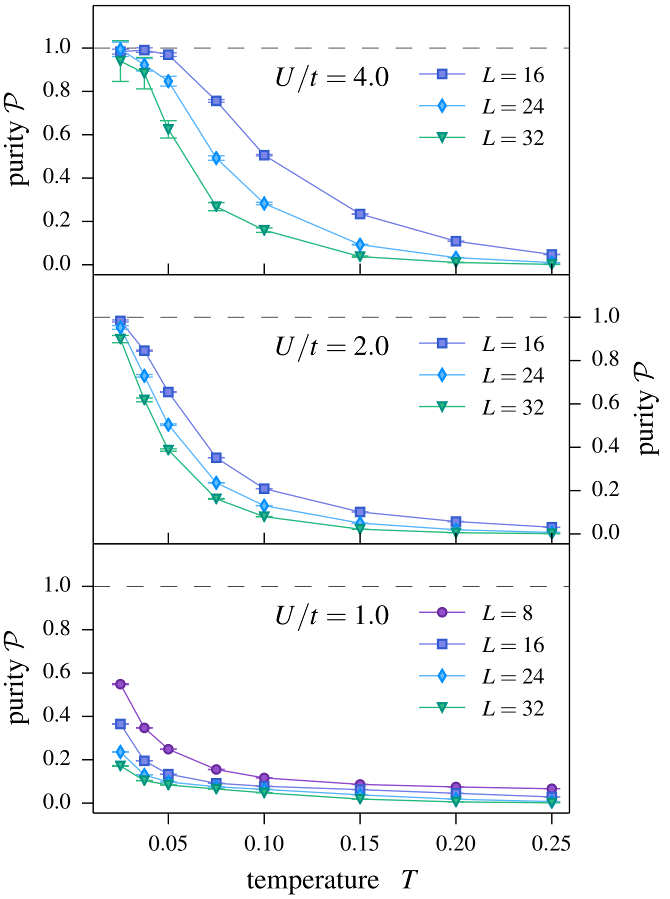


**Figure 5.8.:** The purity $\mathcal{P}$ for a grand-canonical DMQC simulation of a half-filled Hubbard chain versus temperature for varying on-site interactions $U/t$ and chains of varying length $L$. Figure first appeared in [54].

to equal its complement $S_2(\emptyset)$ and thus must vanish for any quantum mechanical ground state. Indeed the purity sharply rises towards 1 as the temperature is lowered, see Fig. 5.8 which shows the purity as a function of temperature for different system sizes and a sequence of on-site interactions. On the one hand, for a fixed value of the on-site interaction the crossover temperature decreases with system size in accordance with the fact that the finite-size gap of the system also decreases with increasing system size. On the other hand, for fixed system size the transition temperature $T^*$ decreases as the on-site interactions $U$ is reduced reflecting the enhancement of charge fluctuations in this weakly coupled regime.

## 5.5. Comparison to the free fermion decomposition method

As mentioned earlier, there is an alternative method of calculating Renyi entropies from DQMC simulations, which makes use of the fact that after the application of the Hubbard-Stratonovich transformation, one deals with free fermions moving in an external potential [100].

For a fair comparison, the free fermion decomposition method was implemented using the same algorithmic optimizations as for the replica switching technique whenever possible. Both codes were run on identical parameter sets logging the exact same CPU time.

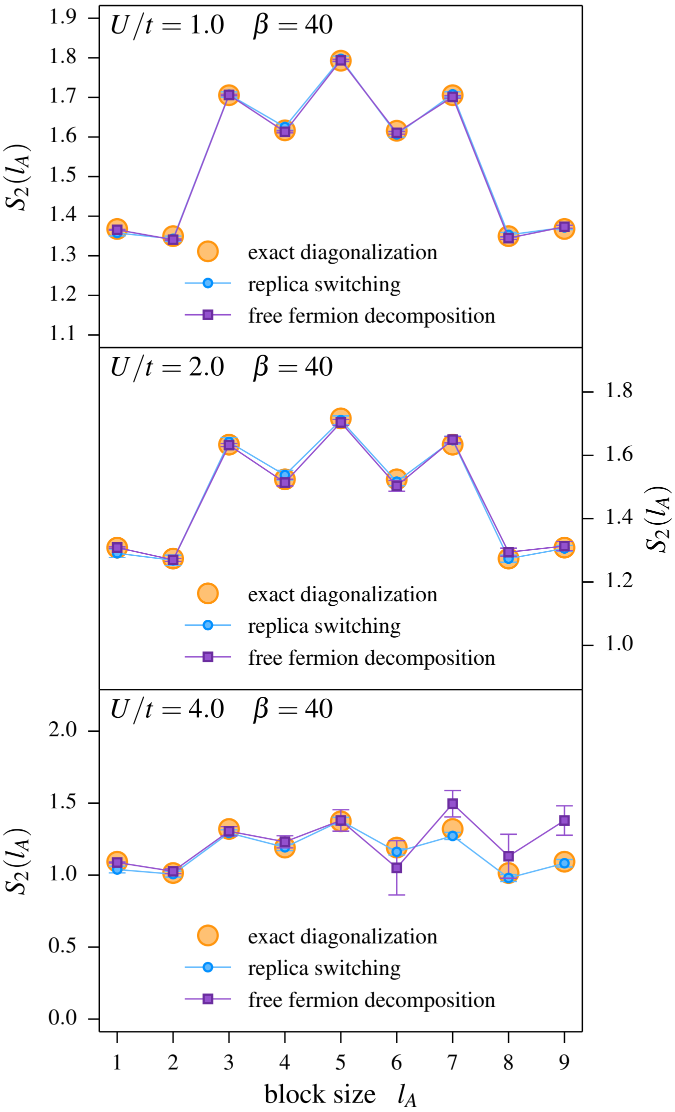


**Figure 5.9.:** Comparison of the replica switching (squares) and free fermion decomposition (circles) DQMC algorithms for the Renyi entropy of a half-filled Hubbard chain with varying on-site interactions $U/t$ at temperature $T = 0.025$ ($\beta = 40$). For comparison exact diagonalization data (for $\beta = \infty$) is indicated by the open circles. With both codes allocating the exact same amount of CPU time, a much faster convergence of the replica switching method is found. Figure first appeared in [54].

The replica method takes considerably fewer measurements because the matrices are of size $N_A + 2 \cdot N_B$ instead of $N_A + N_B$. For the largest cut, i.e. $l_A = 10$, some 1300 measurements were recorded, while for the smallest cut, i.e. $l_A = 1$ there are only around 100 measurements.

Results from this comparison are summarized in Fig. 5.9 which shows results of both approaches for the entanglement entropy of a half-filled 10-site Hubbard chain at fixed temperature $T = 0.025$ ($\beta = 40$) for different values of the on-site interaction $U/t \in \{1,2,4\}$. While the free fermion decomposition method reproduces the arc-like structure of the entanglement entropy for small on-site interaction $U/t = 1$, it shows deviations from this behavior already for moderate values of the on-site interaction $U/t = 2$. In contrast, the replica switching method nicely reproduces the exact diagonalization data up to strong on-site interaction $U/t = 4$. The deviations increase with increasing subsystem size and increasing on-site interaction $U$. A more detailed study for a fixed temperature $\beta = 40$ is shown in Fig. 5.10.

In conclusion, the replica approach is significantly more efficient in capturing the entanglement properties in the interaction dominated regime of the Hubbard model.

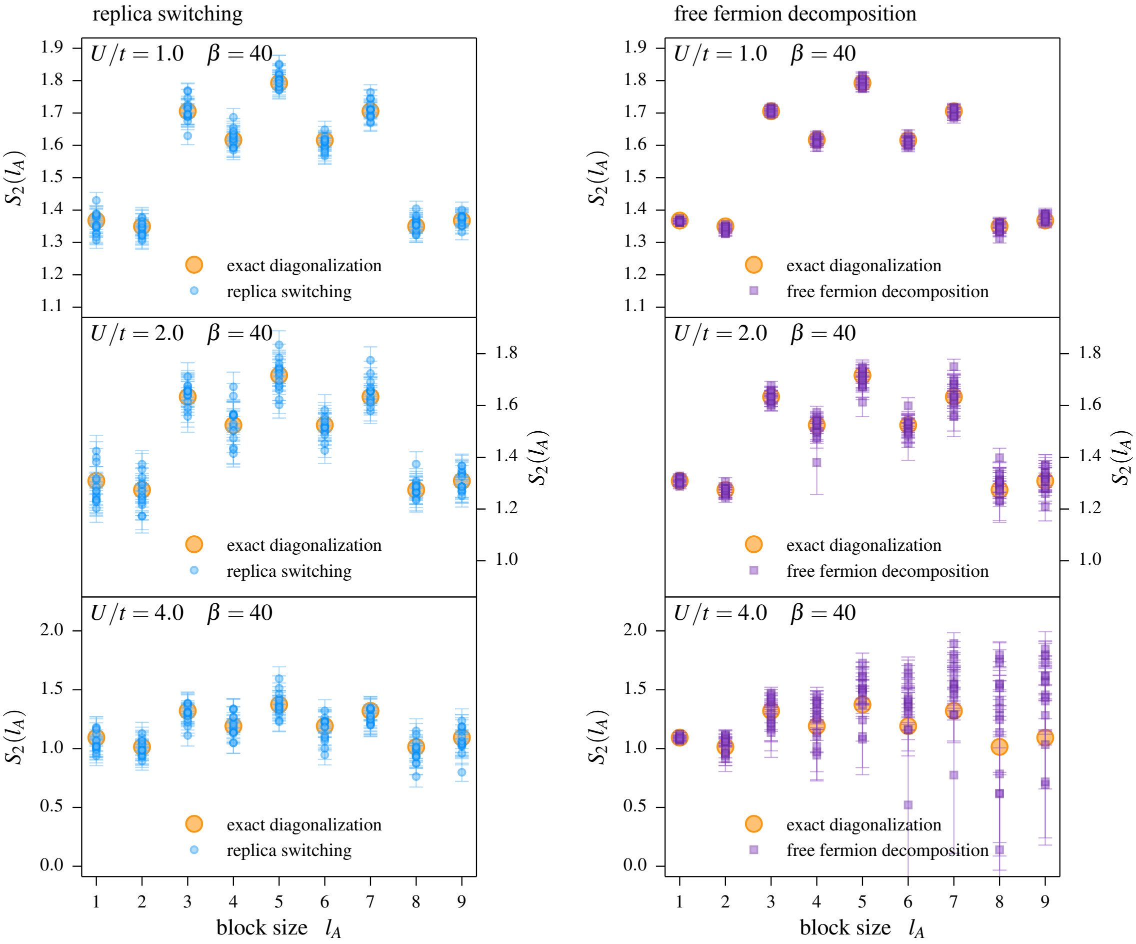


**Figure 5.10.:** Comparison of the spread of the raw data of DQMC runs for 16 independent runs using the replica switching (squares) and free fermion decomposition (circles) algorithms. Shown is the Renyi entropy of a half-filled Hubbard chain with varying on-site interactions $U/t$ at temperature $T = 0.025$ ($\beta = 40$). Figure first appeared in [54].

## 5.6. Stabilization of the ground state algorithm

The ground state algorithm to calculate Renyi entropies suffers from more instabilities than the conventional ground state algorithm. To overcome these difficulties, the numerical stabilization requires a variety of algorithmic steps ranging from linear algebra aspects such as the inversion of seemingly singular matrices and technical tricks such the inclusion of an artificial chemical potential, which is shown to control the condition numbers of the underlying linear algebra algorithms, to more DQMC specific aspects such as the optimal choice of Hubbard-Stratonovich transformation. These details are discussed here including a detailed discussion of the convergence behavior. Finally, these technical improvements are used to provide an entanglement perspective on the quantum phase transition between an antiferromagnetically ordered Mott insulator and a featureless band insulator in the Hubbard model on the bilayer square lattice.

### 5.6.1. Invertibility of the matrix products in the replica scheme

A fundamental problem in the adoption of the finite-temperature algorithm for ground-state properties is that the density matrix is typically singular, i.e. it is in general not invertible. Numerically, this issue can be traced back to the fact that a trial wave function is represented by a rectangular matrix which is only of full rank if the particle number equals the number of total sites on the lattice which is the least interesting case. The matrix is typically constructed by diagonalizing the quadratic part of the Hamiltonian and then keeping only the first $N_p$ eigenvectors with the lowest eigenvalues. The trial wave function is thus represented as a matrix $\psi_T$ of dimension $N \times N_p$ with orthogonal columns, whereas the associated density matrix $\rho$ of size $N \times N$ is constructed as $\rho = \psi_T \psi_T^\dagger$. It is thus apparent that the density matrix as the product of two non-square matrices must be rank deficient and as such singular. As a consequence, the entire matrix product $\mathcal{B}(\tau)$ and its decomposed form (2.40) are not invertible, which ultimately prevents the computation of the equal-time Green's function and the associated Monte Carlo weights.

The central idea to elude this problem is to modify the matrices $U, D$, and $T$ of the singular value decomposition (2.39) such that they become invertible but leave the original matrix unchanged up to machine precision, denoted by $\varepsilon$. Figure 5.11 illustrates how to do this in practice: Any non-square matrix with orthogonal (unitary) columns (or rows) can be extended to a square, fully orthogonal (unitary) matrix by applying a Gram-Schmidt process. To ensure that the original matrix, i.e. the one that results from remultiplying $UDT$, remains unchanged, the diagonal matrix is extended by values that are at least $1/\varepsilon$ smaller in magnitude than the smallest singular value found in the original $D$ matrix. Numerically, this means that none of the additional columns or rows actually contribute because they are weighted below numerical precision by the diagonal matrix. By multiplying the modified matrices $U \times D \times T$, the original, singular square matrix $B$ is recovered.

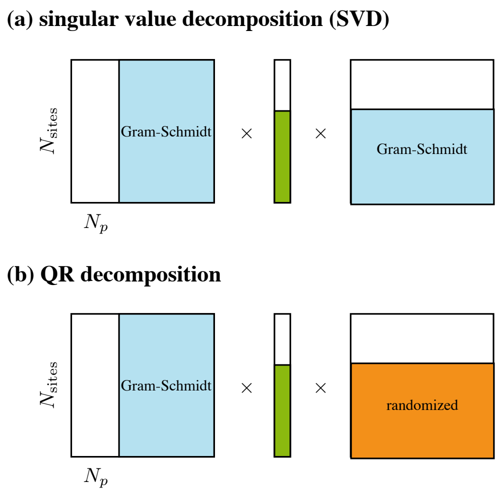


**Figure 5.11.:** Inversion scheme as applicable to SVD (top) and QR (bottom) based Green's function calculations. Shown here is the example of the projected trial density matrix $B(\tau,\tau/2)\,|\psi_T\rangle\,\langle\psi_T|\,B(\tau/2,0)$. Matrices with orthogonal columns or rows that form an incomplete basis are extended to a full basis using a Gram-Schmidt process, denoted here in blue. The singular value vectors are extended by a vector of decreasing numbers clearly separated from the smallest physical singular value by the singular value gap $\Delta$ here pictured in green. For the case of the QR decomposition matrices without orthogonal columns or rows appear naturally which are extended by a random matrix to match dimensions and ensure invertibility (for example by LU decomposition), pictured in orange. Figure first appeared in [25].

## 5.6.2. Stable calculation of the Green's function

The calculation of the equal-time Green's function requires the inversion of the matrix product $\mathcal{B}(\tau)$, see Eq. (2.40), and therefore also sensitively depends on the matrix decomposition discussed in the previous section. Despite having circumvented the problem of singularity, the calculation of the Green's functions remains difficult because the matrices of the decomposition typically remain ill-conditioned, i.e. they retain an extremely broad singular value spectrum resulting in a high condition number [101]. The problem of calculating the Green's function from ill-conditioned matrices in the finite-temperature algorithm has long been known and was solved by Hirsch and Fye [102] by using *multiple* consecutive matrix decompositions (instead of just one as given in Eq. (2.40)). These multiple decompositions can be arranged in an enlarged matrix of size $(N_{\text{sites}} \cdot N_{\text{decompositions}}) \times (N_{\text{sites}} \cdot N_{\text{decompositions}})$, such that the determinant of this enlarged matrix remains equal to that of the original one, but the equal-time Green's function can now be read off as a submatrix of the inverse of the enlarged matrix. The deeper reason that this approach allows avoiding the ill-conditioned matrix problem above is found in a considerably narrower singular value spectrum of the enlarged matrix. A maximum matrix size for the enlarged matrix is reached when each of the slice matrices is used as an input decomposition, i.e. $N_{\text{decompositions}} = N_\tau$. However, it is prohibitively expensive to invert such a large matrix [103]. Fortunately, for the entanglement computations at hand it typically suffices to choose just two or three consecutive decompositions for the calculation of the Green's function for a simulation cell with a complete cut and up to five for the Green's function for a simulation cell with a partial cut.

One more technical caveat in the entanglement computation that warrants attention arises when inserting the density matrices $|\psi_T\rangle\langle\psi_T|$ in the calculation of the trace over the reduced density matrix in Eq. (5.18). When building a matrix product like

$$\mathcal{B} = B(\theta,\theta/2)|\psi_T\rangle\langle\psi_t|B(\theta/2,0)\,,$$

one usually starts from the right and multiplies slice matrices $\mathbf{B}(i)$, applying the stabilization procedure using successive matrix decompositions, until the decomposition for the slice matrix group $B(\theta/2,0)$ is built up. Up until this point in imaginary time, these matrices are square and invertible but the insertion of the singular density matrix turns the entire product into a singular matrix as well. Continuing to multiply slice matrices on the left and decomposing the resulting matrices as was done before, would result in $N_p$ non-zero singular values (corresponding to the particle number of the trial wave function) and $N - N_p$ zero singular values – at least in theory. In practice, however, a decomposition algorithm like SVD or QR will typically find only the $N_p$ non-zero singular values to high precision but the $N - N_p$ remaining singular values (again, strictly zero in theory) are found to be zero only relative to the actual non-zero singular values [103, 101] with machine precision. These inaccuracies will accumulate as more slice groups are multiplied which ultimately results in incorrect Green's matrices.

This problem can be overcome by a modification of the stack structure. For the Green's function of the completely cut system, the modification looks as follows: There will be *three* stacks to keep track of: one which is built from the bra version of the wave function and includes decompositions obtained from matrices of the form $\langle\psi| B(\tau,\tau')$, a second one which is simply the ket version based on $B(\tau,\tau')\,|\psi\rangle$ and finally a third one that is used as temporary storage and includes decompositions of the full slice matrix groups $B(\tau,\tau')$ without an added wave function. In combination, these three stacks allow calculating the Green's function for a given imaginary time slice. An example configuration of the three stacks for imaginary time $\tau = n\cdot\Delta$ looks as follows:

**stack 1**

$$\langle\psi|\, B_m \qquad \ldots \quad \langle\psi|\, B_m B_{m-1}\cdots B_i$$

**stack 2**

$$B_{2m}\cdots B_{m+2}B_{m+1}\,|\psi\rangle \qquad \cdots \qquad B_{m+1}\,|\psi\rangle$$

**stack 3**

$$B_{i-1}\cdots B_1 \qquad \cdots \qquad B_1$$

These three stacks can be used to set up a $3\cdot N_{\text{sites}} \times 3\cdot N_{\text{sites}}$ matrix and to subsequently calculate the Green's function and the corresponding Monte Carlo weight. Alternatively, if the condition numbers of the matrices allow it, it is possible to contract two or even all of them for a faster calculation. In that case, one has to think about the order of the matrix contractions. Naturally, the optimal choice is the one that keeps the condition numbers for the resulting matrices as small as possible, thus ensuring the highest stability of the following operations. Note that the number of contractions can vary for each recalculation and should be chosen according to the magnitude of the relative error between propagated and recalculated Green's function as defined in Eq. (2.43).

These numerical refinements are incorporated in the modified flow diagram of the DQMC algorithm in Fig. 5.12. As for the calculation of the Green's function in the partially cut system, one has to keep track of five individual stacks because the replicas are now connected in imaginary time. The rest of the algorithm works exactly as just described for $G_0$.

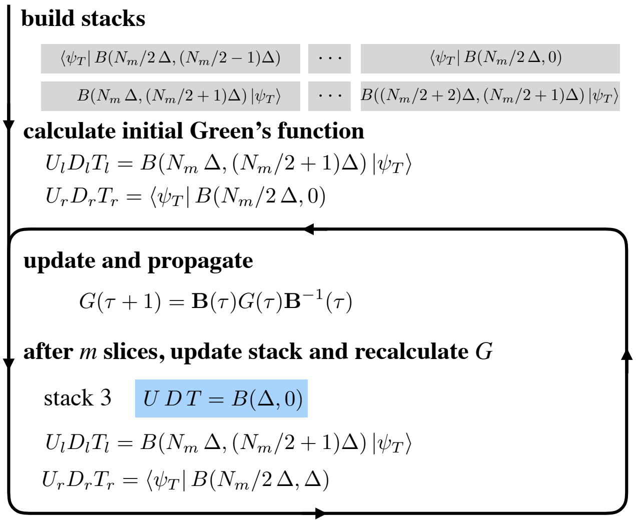

build stacks
$\langle\psi_T|\, B(N_m/2\,\Delta,(N_m/2-1)\Delta)$
$\cdots$
$\langle\psi_T|\, B(N_m/2\,\Delta,0)$
$B(N_m\,\Delta,(N_m/2+1)\Delta)\,|\psi_T\rangle$
$\cdots$
$B((N_m/2+2)\Delta,(N_m/2+1)\Delta)\,|\psi_T\rangle$
calculate initial Green's function
$U_l D_l T_l = B(N_m\,\Delta,(N_m/2+1)\Delta)\,|\psi_T\rangle$
$U_r D_r T_r = \langle\psi_T|\, B(N_m/2\,\Delta,0)$
update and propagate
$G(\tau+1) = \mathbf{B}(\tau)G(\tau)\mathbf{B}^{-1}(\tau)$
after $m$ slices, update stack and recalculate $G$
stack 3
$U\,D\,T = B(\Delta,0)$
$U_l D_l T_l = B(N_m\,\Delta,(N_m/2+1)\Delta)\,|\psi_T\rangle$
$U_r D_r T_r = \langle\psi_T|\, B(N_m/2\,\Delta,\Delta)$

**Figure 5.12.:** Flow diagram of the modified DQMC algorithm for the simulation of the partition function $Z_0$ with complete cut in imaginary time. Stacks are initialized starting from the bra and ket version of the wave function and subsequently used to calculate an initial Green's function. This Green's function is then used to update the auxiliary field time slice by time slice and recalculated after $m$ time steps to retain numerical stability. Intermediate decompositions of slice matrix groups $U, D, T$ without wave function are stored in a third stack, here highlighted by shading. Figure first appeared in [25].

### 5.6.3. Choice of Hubbard-Stratonovich transformation

At the heart of the DQMC approach is the decoupling of quartic terms in the Hamiltonian using a Hubbard-Stratonovich transformation. The particular choice of Hubbard-Stratonovich transformation can greatly affect the numerical stability and convergence of the DQMC approach. This can easily be illustrated by considering the example of a density-density interaction of the general form

$$\hat{U}_{\alpha\beta} = U_{\alpha\beta}\, n_\alpha n_\beta\,, \tag{5.25}$$

where the indices $\alpha$ and $\beta$ may represent adjacent sites for a nearest-neighbor interaction or different spin species for a given site, respectively. It was already shown that this type of interaction can be decoupled in two different ways, leading to either real or complex valued matrices.

At first look, one might expect that the real-valued Hubbard-Stratonovich transformation leads to lower computational cost. However, in computations of the entanglement entropy, it is the complex-valued Hubbard-Stratonovich transformation that should be used preferably. The reason for this preference is that the Hubbard-Stratonovich transformation also greatly affects the singular value spectrum of the matrix decompositions discussed in the previous subsection. To illustrate this point, Fig. 5.13 shows a comparison of the singular value spectrum of the matrix $B(\Theta/2,0)\,|\psi_T\rangle$, i.e. the projected wave function, obtained for real and complex Hubbard-Stratonovich transformations. The data is calculated for a bilayer Hubbard model on a square lattice of size $4\times4\times2$ at half filling and equal hopping within and between the layers, i.e. $t=t'$, but varying on-site interaction $U$, see also section 5.7 for a more detailed discussion of this model. In the complex case, the range of the singular values is seen to decrease as the interaction increases, hence reducing the condition number and stabilizing the algorithm. In the real case, the behavior is found to be exactly opposite with the condition number becoming worse as the interaction strength increases.

As a consequence, the following simulations employed a complex-valued Hubbard-Stratonovich transformation in the computation of Renyi entanglement entropies.

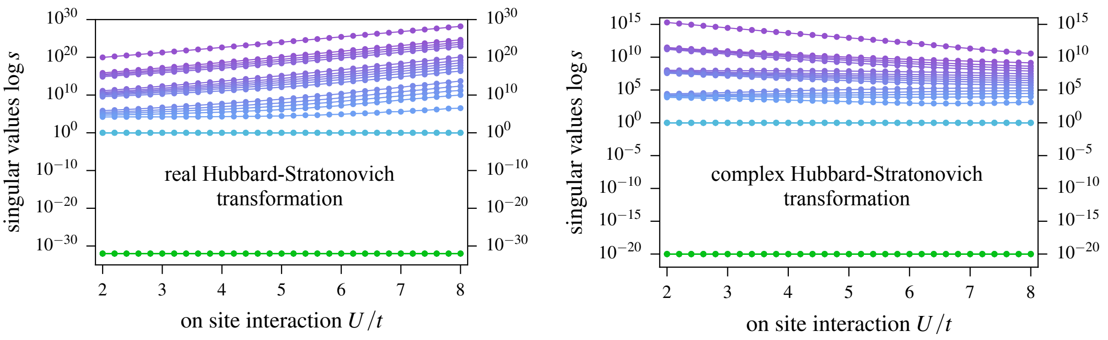


**Figure 5.13.:** The singular value spectrum of the matrix $B(\Theta/2,0)\,|\psi_T\rangle$ i.e. the projected wave function, on a logarithmic scale of base 10 for a real (left) and complex (right) Hubbard-Stratonovich transformation.The projection time is $\theta = 10$. Figure first appeared in [25].

### 5.6.4. Convergence

One key distinction between the zero-temperature entanglement calculations in the projective DQMC approach and the closely related finite-temperature algorithm is in the type of simulated ensemble. While the finite-temperature algorithm samples states from a grand canonical ensemble, the projective scheme samples a canonical ensemble of *fixed* particle number (encoded in the trial wave function). This seemingly small modification of the ensemble is found to have a rather large impact on the convergence properties of the projective algorithm, in that it does not converge readily without additional work. To ensure convergence, one has to resort to a technical trick by introducing an artificial chemical potential in the Hamiltonian

$$H' = H + \mu_a \sum_i n_i\,. \tag{5.26}$$

Note that the inclusion of this artificial chemical potential $\mu_a$ does not alter the physics of the original problem because it does not affect the particle number which is fixed by the choice of the trial wave function.

On a technical level, such a term is represented by a diagonal matrix and has the ability to significantly shift the singular value spectrum. This is displayed in Fig. 5.14, which plots the smallest singular value of the unmodified matrix $B(\theta/2,0)\,|\psi\rangle$ versus the projection time $\theta$. Clearly visible is the increase of the magnitude of the smallest singular value with increasing $\theta$ and increasing chemical potential $\mu_a$. The convergence of the entanglement entropy upon inclusion of this artificial chemical potential term is dramatically improved even by a small additional potential dramatically as shown in Fig. 5.15. The convergence rate eventually saturates when further increasing the artificial chemical potential. In general, the optimal value of $\mu_a$ depends rather sensitively on the parameters of the Hamiltonian and needs to be chosen with great care.

Closer inspection of the convergence properties of the entanglement DQMC algorithm reveals that the added chemical potential (5.26) only affects the calculation of the Green's function of the partially cut system. This observation already hints at the origin of the improved convergence. For the respective partition sum, all slice matrices are obtained from the “unfolded" Hamiltonian (5.14), where subsystem $B$ is duplicated as illustrated in Fig. 5.3. The slice matrices include blocks of identity matrices and zero matrices to

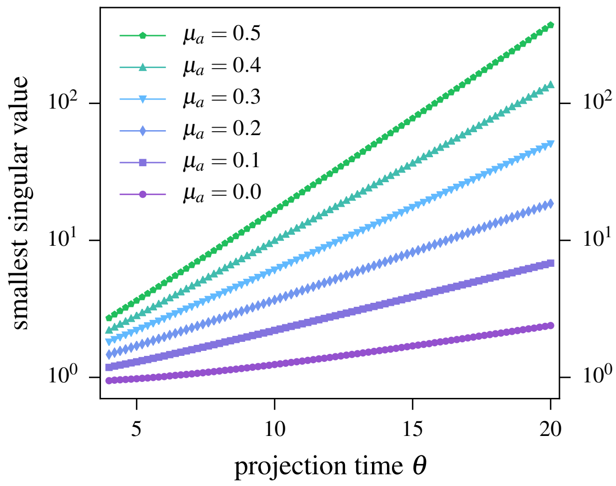


**Figure 5.14.:** Evolution of the smallest singular value with projection time $\Theta$ for various values for the artificial chemical potential $\mu_a$ shown on a logarithmic scale with base 10. The value of the smallest singular value increases with increasing artificial chemical potential $\mu_a$. Figure first appeared in [25].

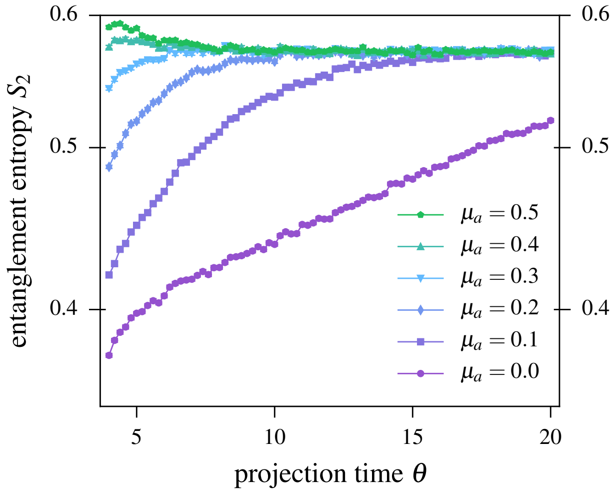


**Figure 5.15.:** Convergence of the Renyi entanglement entropy $S_2$ in the projective DQMC algorithm versus projection time for varying values of the artificial chemical potential $\mu_a$. The Renyi entropy is calculated for a half-filled Hubbard model on the square lattice for $U/t = 4$ with an equal-size bipartition of the lattice. For all non-zero artificial chemical potentials $\mu_a$, the entanglement entropy $S_2$ converges to the same value with rate of convergence increasing from the smallest $\mu_a$ at the bottom to the largest $\mu_a$ at the top. Figure first appeared in [25].

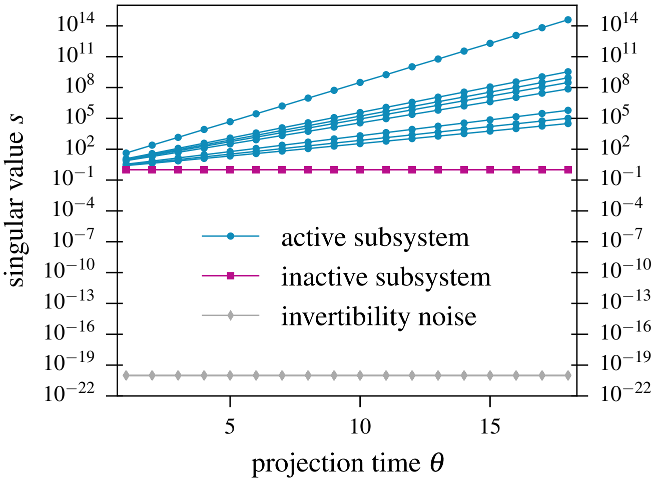


**Figure 5.16.:** Magnitude of the singular values of the Green's function in the replicated system for a Hubbard model on a square lattice at half filling, an on-site interaction of $U/t = 4$ and $\mu_a = 0.5$ on a logarithmic scale with base 10. The inactive subsystem in the middle and the invertibility noise on the bottom remain constant while the singular values of the active subsystems increase as projection time is increased. Figure first appeared in [25].

accommodate the currently “inactive" subsystem $B$ or $B'$, i.e. the one which is deactivated by the $\Theta$-function in the Hamiltonian (5.14). As demonstrated in Fig. 5.16 this greatly affects the singular value spectrum by introducing $N_B$ singular values of unit value (indicated by the magenta dataset) representing the said subsystem. A comparison of Figs. 5.14 and 5.16 shows that once the difference between the smallest singular value of the active subsystem becomes separated from the unit singular values corresponding to the inactive subsystem, the simulation has a chance to converge. Thus, the requirement is that this gap $\Delta_s$ in the singular value spectrum has to be sufficiently large. It is empirically found that $\Delta_s \approx 10^2$ turns out to be a good choice which can be used to determine a value for the artificial chemical potential $\mu_a$.

## 5.7. Application to the bilayer Hubbard model

The discussed technical improvements now allow for stable computation of many models such as the bilayer Hubbard model – a paradigmatic model that allows studying the transition between a Mott insulator and a band insulator. Whether these two insulating states are fundamentally distinct or can in general be adiabatically connected into one another [104], has been a question of debate [105, 106, 107, 108, 109, 110, 111, 112, 113, 114, 115]. This discussion mostly preceded the days of the topological insulator [116, 117] – a second type of band insulator that can be clearly distinguished from the conventional “trivial" band insulator by certain topological invariants [118, 119], while its fundamental distinction to correlated Mott insulators (possibly exhibiting topological order as well) is an open question of much current interest [120]. This section adds the entanglement perspective on the elementary phase diagram of the bilayer Hubbard model, whose Hamiltonian is given as:

$$H = -t \sum_{\langle i,j \rangle, \sigma} c^{\dagger}_{i,\sigma} c_{j,\sigma} - t' \sum_{\langle i,j \rangle', \sigma} c^{\dagger}_{i,\sigma} c_{j,\sigma} + U \sum_{i} n_{i,\uparrow} n_{i,\downarrow} \,. \tag{5.27}$$

At its core, the model describes the competition between conventional (free-fermion) band structures arising from hopping within and between the layer of a double-layer square lattice (parametrized by hopping amplitudes $t$ and $t'$, respectively) and Mott physics arising from an on-site Coulomb repulsion $U$. A schematic phase diagram for this model in terms of the on-site interaction $U$ and the interlayer hopping $t'$ is given in Fig. 5.17. For sufficiently large interlayer hopping strength $t'$ and any value of $U$ the system is a featureless band insulator, while for sufficiently large $U$ and small interlayer hopping $t'$ the system forms a Mott insulator with antiferromagnetic order. The phase transition from Mott to band insulator has been studied using a variety of methods in the past [123, 124, 125, 126, 127, 128, 129, 121, 122]. In the absence of a Hubbard interaction $U$ and small interlayer hopping it is a metal. Whether this metallic state survives for small Hubbard $U$ (and small interlayer hopping $t'$ ) or immediately gives way to Mott physics is still under debate, with dynamical mean field theory (DMFT) [113] and finite-temperature DQMC simulations [126, 121] pointing to an extended metallic phase, while a functional renormalization group (FRG) analysis in combination with a ground state DQMC simulation [122] and a variational Monte Carlo study [129] find an immediate breakdown of the metallic phase upon inclusion of the Hubbard term.

With the help of the stabilization procedure presented in the previous section, the DQMC

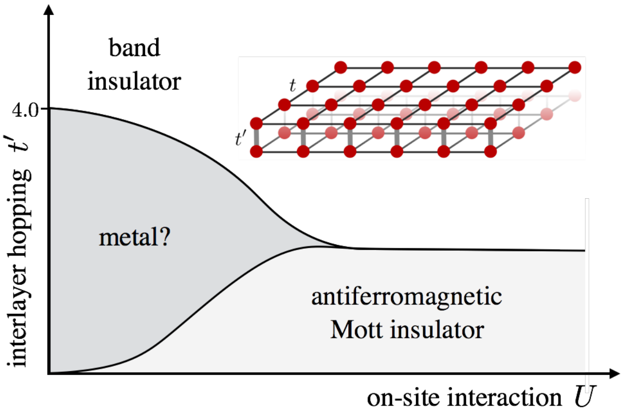


**Figure 5.17.:** Schematic phase diagram for the bilayer Hubbard model at half filling, adapted from dynamical mean field theory (DMFT) [113], DQMC simulations [121], and functional renormalization group (FRG) calculations [122]. It comprises two insulating phases, a band insulator for sufficiently strong interlayer hopping $t' \gtrsim 4$, an antiferromagnetic insulator for weak interlayer hopping strength and large on-site Coulomb repulsion $U$. There possibly is an extended metallic phase in the regime of small Coulomb repulsion and interlayer hopping as indicated in the phase diagram, with DMFT and DQMC simulations in favor of it [113, 121], while FRG calculations point to an absence of a metallic phase for any finite Coulomb repulsion [122]. The inset depicts of the bilayer lattice. The intralayer hoppings are indicated by $t$, while $t'$ denotes the interlayer hoppings. Figure first appeared in [25].

replica switching approach is ideally suited to compute Renyi entanglement entropies for all parameters of interest, in particular for large values of the coupling $U$ and the large hopping amplitudes $t, t'$. In the simulation, the system is partitioned into a strip of width $L/4 \times L \times 2$ and its complement, with upper and lower layer belonging to the same subsystem. For this type of cut, the entanglement signature of the band insulator is expected to be rather trivial: A singlet dimers forms on the rungs between the layers so that any cut between the rungs will effectively see no significant entanglement contribution, similar to the entanglement signature of an unentangled product state. Deep in the Mott regime the prevalent entanglement signature of any entangled quantum many-body state – a finite entanglement entropy that is subject to the famous boundary-law scaling is expected [130]. For any metallic state, an even stronger entanglement signature in the form of a logarithmic violation of the aforementioned boundary law [131, 132] should be observed.

The first step towards revealing the phase diagram is to scan parameter space for $t' \in [0.5, 5.3]$ and $U in [1, 10]$ and plot the *absolute* value of the Renyi entropy for a system of $4 \times 4 \times 2$ sites. In this phase diagram, several distinct regimes of almost constant amount of entanglement can be readily distinguished. As argued above, the band insulator is realized in the regime of almost vanishing entanglement for large interlayer coupling $t'$. The lower part of the phase diagram is a region of finite entanglement entropy pointing to at least one separate regime. The Renyi entropy along a vertical cut through the data of Fig. 5.18 is shown in Fig. 5.19 for $U/t = 4$ and $L = 4$ and $L = 8$. Another possibility to assess whether this third regime is a finite size effect or a genuine third phase is to study the scaling of the Renyi entropy with system size. This is shown in Fig. 5.20 for different values of the on-site Coulomb repulsion $U/t = 2, 4, 8$ and 16 and fixed interlayer hopping $t'/t = 1$. With the entropy renormalized by the linear system size $L$, the boundary law scaling expected for the Mott insulator corresponds to a flat line. This is precisely what is found for moderate to large Hubbard interaction $U/t = 4, 8, 16$. In contrast, for small Hubbard interaction $U/t = 2$ the clearly noticeable slope of the data points suggests an additional logarithmic contribution to the entanglement entropy. Precisely

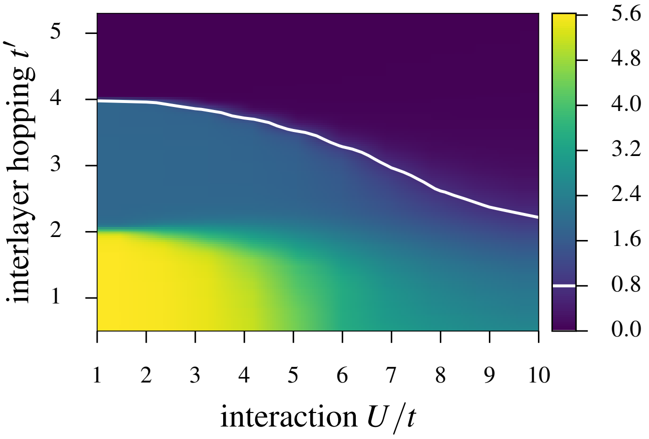


**Figure 5.18.:** Color-coded Renyi entanglement entropy $S_2$ for the bilayer Hubbard model calculated for a bipartition of a $4 \times 4 \times 2$ system (with the system divided into a strip of extent $1 \times 4 \times 2$ and its complement). The entanglement entropy was calculated for integer values of $U$ ranging from 1 to 10 and with $t'$ in the range of $[0.5, 5.3]$ with a step size of 0.05. The different entanglement regimes hint at general characteristics of the underlying phase diagram. For dominant interlayer coupling $t' \gtrsim 4$ the vanishingly small entanglement points to the featureless band insulator with singlet formation on the interlayer rungs. For moderate interlayer coupling, the finite entanglement reveals the Mott insulator. The phase boundary separating the band insulator is indicated by an equipotential line of $S_2 = 0.8$. The relevant color values corresponding to these values of the entanglement entropy are also marked on the colorbar by white lines. Figure first appeared in [25].

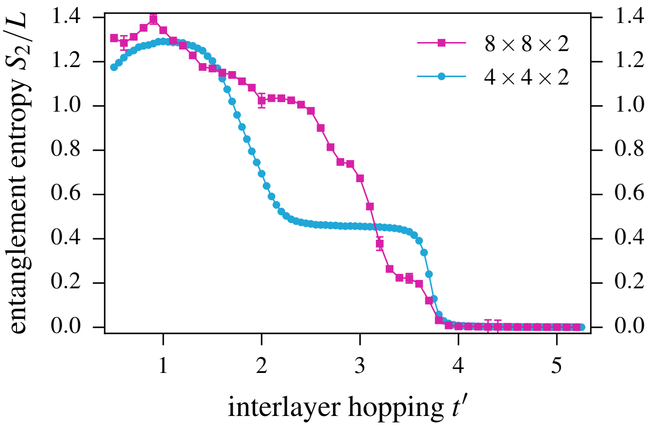


**Figure 5.19.:** Renyi entanglement entropy $S_2$ for the bilayer Hubbard model along a vertical cut through the phase diagram of Fig. 5.18 at $U/t = 4$. Data for different system sizes is shown. Figure first appeared in [25].

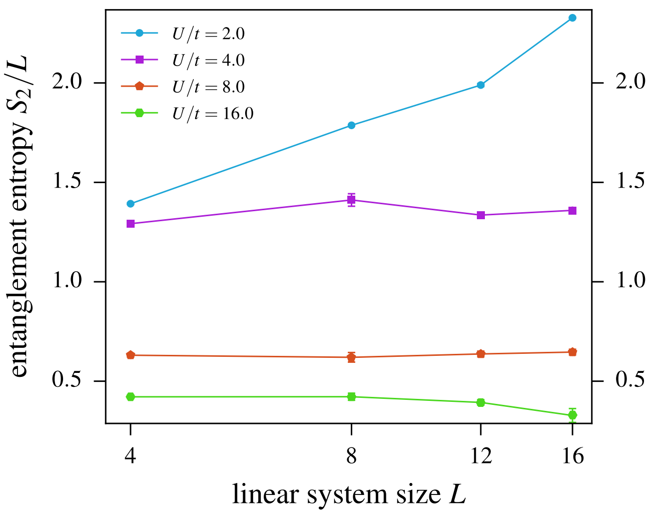


**Figure 5.20.:** Finite-size scaling of the entanglement entropy $S_2$ renormalized by the linear system size $L$ for the bilayer Hubbard model at different values of the on-site interaction $U/t$ and fixed interlayer coupling $t'/t = 1$. Note the logarithmic scale with base 10 chosen for the abscissa. Figure first appeared in [25].

such an $S \propto L \log L$ scaling is expected for a two-dimensional metal with a Fermi surface [131, 132]. These results are consistent with finite-temperature data given for smaller lattices in Ref. [121]. Note, however, that the system sizes at hand are still relatively small when inferring this logarithmic contribution and it would be highly desirable to go to substantially larger system sizes to exclude possible finite-size effects, see Ref. [122] for a more in-depth discussion and analysis from the point of view of correlation functions. Unfortunately, such a substantial increase in system sizes is difficult with the current algorithms and state-of-the-art computational resources.

## 5.8. Entanglement and the sign problem

In this section, an alternative perspective on the sign problem from the viewpoint of entanglement is offered. The guiding idea is to explore whether extracting *global* information about the ground state of a many-fermion system via its entanglement properties – such as the general classification whether it exhibits gapless modes, conventional or topological order – is possibly not as strongly affected by the sign problem as are measurements of expectation values of observables aimed at providing a full ground-state characterization such as order parameters or two-point correlation functions. This information theoretical angle on the sign problem is originally rooted in the observation that in the Monte Carlo calculation of entanglement entropies the sign problem enters in an *additive* instead of an multiplicative way as for conventional observables, as will be explained below. It was thus interesting to study whether this observation might point a way to an amenable probe of global ground-state properties of interacting many-fermion systems even in the presence of a strong sign problem. It will be shown that auxiliary-field quantum Monte Carlo approaches appear to be more immune to the sign problem than others using the example of spinless fermion models on the honeycomb lattice at and below half filling.

### 5.8.1. Entanglement entropies for models with sign problem

It was shown earlier that the entanglement entropy is calculated as the ratio of two partition sums

$$S_2(A) = -\log \frac{Z[A,2,T]}{Z^2} \equiv -\log \frac{Z_1}{Z_0}\,, \tag{5.28}$$

where $Z_1 = Z[A,2,T]$ is the partition sum of the replicated system and $Z_0 = Z^2$ is the square of the partition function of the original system. If the system under consideration exhibits a sign problem, its partition sum $Z$ can be split into a product of a partition sum $Z^{\text{abs}} = \sum_{\mathcal{C}} |w(\mathcal{C})|$, where all weights $w(\mathcal{C})$ of the original partition sum have been taken as their absolute values, and the expectation value of the sign

$$Z = \sum_{\mathcal{C}} w(\mathcal{C}) = \sum_{\mathcal{C}} \sigma(\mathcal{C})\,|w(\mathcal{C})| = Z^{\text{abs}} \cdot \langle \sigma \rangle_{\text{abs}}\,. \tag{5.29}$$

Inserting this expression in the replica representation of the Renyi entropy (5.28) one readily obtains

$$\begin{aligned} S_2(A) &= -\log \left( \frac{Z_1^{\text{abs}}}{Z_0^{\text{abs}}} \cdot \frac{\langle \sigma_1 \rangle_{\text{abs}}}{\langle \sigma_0 \rangle_{\text{abs}}} \right) \\ &= -\log \frac{Z_1^{\text{abs}}}{Z_0^{\text{abs}}} - \log \frac{\langle \sigma_1 \rangle_{\text{abs}}}{\langle \sigma_0 \rangle_{\text{abs}}} \\ &= S_2^{\text{abs}}(A) + S_2^{\sigma}(A)\,. \end{aligned} \tag{5.30}$$

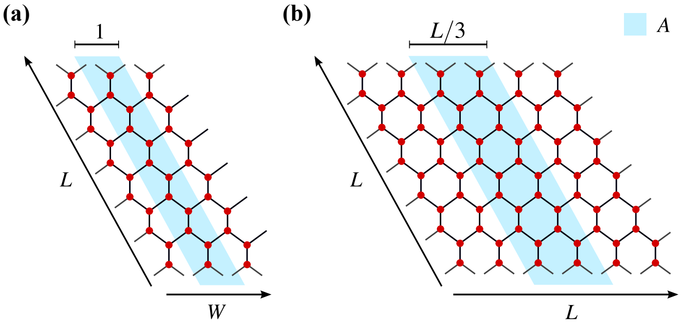
Thus, the Renyi entropy for a system with a sign problem separates into two *additive* contributions, one coming from the partition sum with absolute weights and one arising from the sign. This additive behavior, which has earlier been discussed in the context of variational Monte Carlo approaches [133], should be contrasted to the *multiplicative* contribution of the sign to the calculation of conventional observables (4.1). This additive splitting also raises the immediate question of how contributions to the scaling behavior of the Renyi entropy $S_2(A)$ are split among the two terms, in particular whether the boundary law scaling or any of its subleading contributions can arise solely from the sign contribution $S_2^{\sigma}(A)$. Could it be sufficient to only consider the absolute partition sum and its associated entanglement entropy $S_2^{\text{abs}}(A)$, which can be computed from a QMC simulation in a straight-forward manner? The simulations of a system of spinless fermions on the honeycomb lattice presented in the following do not give a general answer to this question but nevertheless give some numerical evidence that the leading behavior can be reproduced and thus show that it is worthwhile pondering this question.

### 5.8.2. Spinless Dirac fermions on the honeycomb lattice

In this model, spinless fermions on the honeycomb lattice are subjected to a nearest-neighbor repulsive interaction $V$ described by a Hamiltonian

$$H = -t \sum_{\langle i,j \rangle} \left( c_i^{\dagger} c_j + c_j^{\dagger} c_i \right) + V \sum_{\langle i,j \rangle} n_i n_j\,. \tag{5.31}$$

**Figure 5.21.:** Geometries of the honeycomb lattice and its bipartitions into subsystems $A$ (shaded in blue) and $B$ used in (a) Fig. 1 and (b) Fig. 2 of the main text, respectively. Figure first appeared in [55].



At half-filling $\langle n \rangle = 1/2$ this model exhibits a fermionic quantum phase transition, likely in the Gross-Neveu universality class [134, 135], from a Dirac semimetal for small repulsion $V$ to a charge density wave state for large $V$, as recently discussed in various numerical works [136, 46, 137, 138]. What makes this spinless fermion system particularly interesting in this context is that it exhibits a severe sign-problem in the complex-fermion typically used in DQMC, but can be recast in a Majorana fermion representation without sign problem [46]. This allows benchmarking results for the sign-ignoring entanglement entropies $S_2^{\text{abs}}(A)$ obtained for the complex-fermion case with numerically exact data for $S_2(A)$ from the Majorana fermion approach and in particular distill the sign contribution $S_2^{\sigma}(A)$ to the Renyi entropy (5.30).

For the simulation, it is of great importance to choose lattice sizes with a particular length because the singular points of the two Dirac cones should be included in the discretized Brillouin zone. This leads to the constraint that the linear system size has to be a multiple of three. The lattice is then partitioned into two subsystems made up of two thirds and one-third of the sites, respectively, see Fig. 5.21 for a visualization. The cut is chosen to be free of corners to avoid potential contributions from subleading terms to the area law and so that the aspect ratio remains constant when the lattice size is increased.

Results for simulations on these lattices of various sizes are presented in Fig. 5.22. While the entanglement entropies show almost perfect agreement in the charge density wave for $V > V_c \approx 1.356(1)$ [136, 46, 137, 138], there is a noticeable difference for the Dirac semimetal phase (for $V < V_c$). This discrepancy partially reflects the strength of the sign problem in the two phases indicated in the lower panel of Fig. 5.22, with the average sign $\langle \sigma \rangle_{\text{abs}}$ almost dropping to zero in the Dirac semimetal phase (indicating a strong sign problem), while the average sign quickly recovers (approaching unity) beyond the phase transition into the charge density wave. Note, however, that the deviation of the sign-ignoring entanglement entropy $S_2^{\text{abs}}(A)$ from the correct entanglement entropy $S_2(A)$ (calculated in the sign-free Majorana representation) remains almost constant with increasing system size.

This immediately raises the question how the expected scaling behavior of the entanglement entropy $S_2(A)$ is split into contributions from the sign-ignoring entanglement entropy $S_2^{\text{abs}}(A)$ and the sign entropy $S_2^{\sigma}(A)$. To probe in particular the scaling behavior of $S_2^{\text{abs}}(A)$ the spinless fermion system of Eq. (5.31) is studied at two different fillings for which one expects two different scaling laws. At half-filling $\langle n \rangle = 1/2$ (and small $V$) the system exhibits a Dirac cone with the entanglement entropy expected to follow a boundary law $S_2(L) = a\ell + \dots$ with the length of the boundary, in our case $\ell = L$ (and $a$ some non-universal prefactor). At one-third filling $\langle n \rangle = 1/3$ (and small $V$) the partially filled

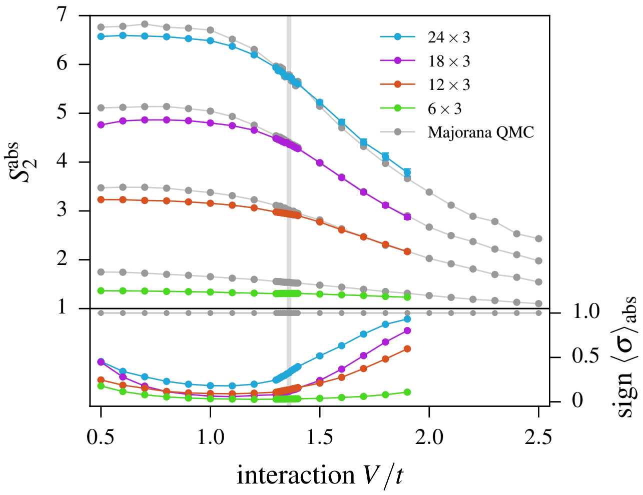


**Figure 5.22.:** Upper panel: Sign-ignoring entanglement entropy $S_2^{\rm abs}(L)$ for spinless fermions on the honeycomb lattice at half-filling subject to a nearest-neighbor repulsive interaction $V$. The latter drives a transition from a Dirac semimetal for $V < V_c \approx 1.356(1)$ (indicated by the vertical grey line) to a charge density wave for $V > V_c$. The sign-ignoring data is from projector auxiliary-field QMC calculations using a real-valued Hubbard-Stratonovich transformation at $\theta = 40$. Sign-free reference data from Majorana-QMC simulations at $\theta = 20$ is indicated by the grey dots. Lower panel: The average sign $\langle\sigma\rangle_{\rm abs}$ indicating the strength of the sign-problem. Figure first appeared in [55].

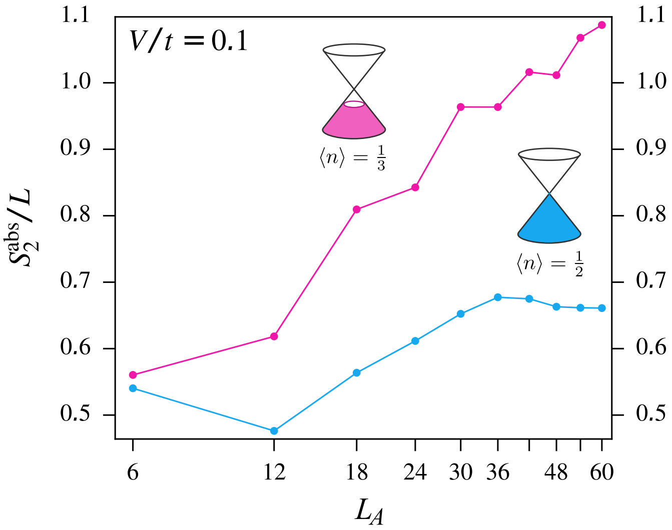


**Figure 5.23.:** Scaling of the sign-ignoring entanglement entropy with system size of a spinless fermion system on the honeycomb lattice with a small nearest-neighbor repulsive interaction $V/t = 0.1$. The system exhibits semimetallic states with a Dirac cone at half-filling and a nodal line at one-third filling. Data is from a projector auxiliary-field QMC calculations using a complex-valued Hubbard-Stratonovich transformation at $\theta = 1$, for which there is a severe sign problem with $\langle\sigma\rangle_{\rm abs} \approx 0$ for any finite $V$ and all, but the smallest system sizes. Error bars are smaller than the symbol sizes. Figure first appeared in [55].

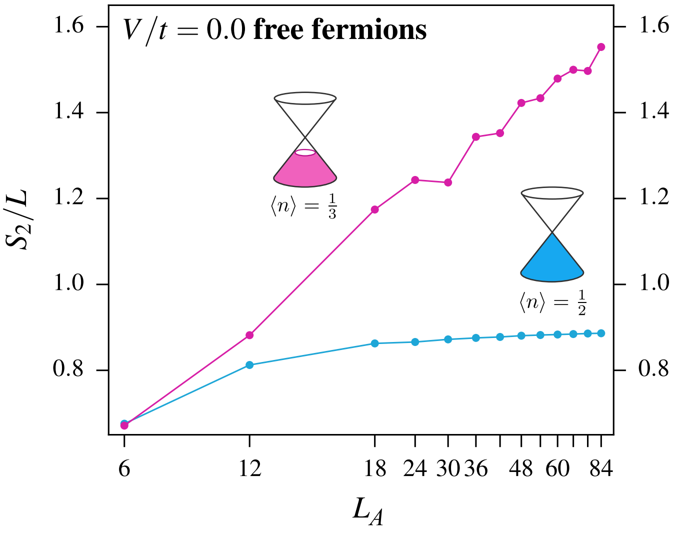


**Figure 5.24.:** Scaling of free spinless fermions on the honeycomb lattice at half and one-third filling using the same geometries as described in the previous section. Figure first appeared in [55].

band structure exhibits a Fermi surface with a nodal line of gapless modes, which leads to a violation of the boundary law with a multiplicative, logarithmic correction arising in the entanglement entropy [139, 140], i.e. $S_2(L) = a\ell \log \ell + \ldots$, where $a$ is again some non-universal prefactor.

Numerical results for the sign-ignoring entanglement entropies $S_2^{\text{abs}}(A)$ calculated for cuts $A$ of dimension $2 \cdot L \times L/3$ in a system of $2 \cdot L \times L$ sites are shown in Fig. 5.23. The data in this plot is obtained from DQMC calculations using the complex-valued Hubbard-Stratonovich transformation, in which $\langle \sigma \rangle_{\text{abs}}$ is suppressed to zero for any finite $V$ and all but the smallest system sizes. Remarkably, the numerical results of Fig. 5.23 suggest that the sign-ignoring entanglement entropies $S_2^{\text{abs}}(A)$ completely reflect the scaling behavior of the full entanglement entropy $S_2(A)$, with the data for the half-filled case clearly seen to flatten out for large system sizes indicative of the boundary-law scaling, while the data for the one-third filled case nicely follows the logarithmic scaling behavior expected for the total entanglement entropy $S_2(L)$. This behavior should be contrasted to the results of a *variational* Monte Carlo study [141] of model wavefunctions for gapless spin liquids obtained from a Gutzwiller projection of a Fermi sea. In this study, the authors found that even the leading order term is not generically robust against the sign problem as the multiplicative logarithmic contribution to the entanglement entropy for a system with gapless nodal lines was solely arising in the sign contribution $S_2^{\sigma}(A)$.

The results of Fig. 5.23 can be compared with entanglement data obtained via the correlation matrix method [95], which are shown in Fig. 5.24. The two figures are strikingly similar, with the sign-ignoring simulation reproducing the general form of the scaling relation including the oscillatory even-odd behavior. Although the data was obtained for two different values of the nearest-neighbor repulsion $V$, namely $V = 0.0$ in 5.24 and $V = 0.1$ in Fig. 2 of the main text, the results shown in Fig. 1 and Fig. 3 of the main text, suggest that the free fermion scaling persists for finite values of $V$.

A similar quantum phase transition can be observed when doping the spinless fermion model (5.31) below half filling. Figure 5.25 shows results for the sign-ignoring entanglement entropy $S_2^{\text{abs}}(A)/L$ for system sizes up to $15 \times 30 = 450$ sites (and 150 fermions), well beyond the system size limit of any other numerical fermion technique. With increasing $V$ a clear transition in the scaling behavior is seen, going from the logarithmic

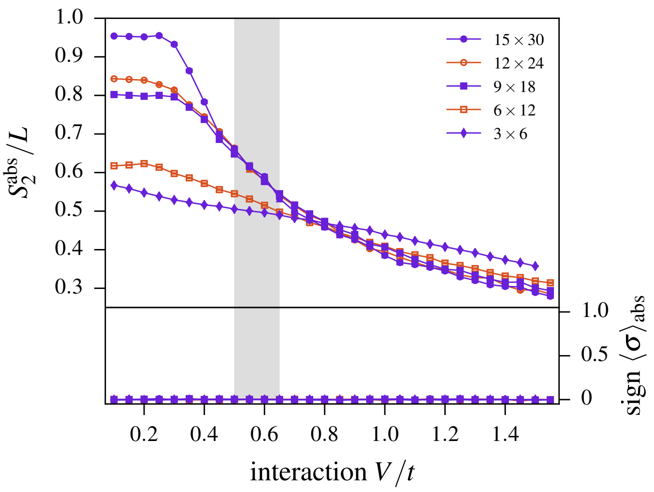


**Figure 5.25.:** Upper panel: Sign-ignoring entanglement entropy $S_2^{\mathrm{abs}}(L)/L$ for spinless fermions on the honeycomb lattice at one-third filling subject to a nearest-neighbor repulsive interaction $V$. The numerical data is obtained from projector auxiliary-field QMC calculations using a complex-valued Hubbard-Stratonovich transformation at $\theta = 20$, for which there is a severe sign problem for all coupling parameters and system sizes as indicated by the vanishing average sign $\langle\sigma\rangle_{\mathrm{abs}}$ shown in the lower panel. The grey bar indicates the best estimate for the region in which the logarithmic contribution vanishes and the system starts to exhibit pure area law behavior. Figure first appeared in [55].

scaling behavior for the nodal liquid at small $V$ to a clean boundary law for larger $V$ (with all curves collapsing onto each other) indicative of an ordinary charge-ordered state. This allows estimating the location of this fermionic quantum phase transition to be roughly located at $V_c/t \approx 0.58 \pm 0.04$ as indicated by the vertical bar in Fig. 5.25.

### 5.8.3. Discussion

This section showed that within the framework of DQMC, it is possible to extract some meaningful physical properties from some models that suffer from severe sign problems. The most likely reason for this stunning result is that the ignorance of negative signs in the interpretation of statistical weights manifests itself differently in various Monte Carlo flavors: In auxiliary-field QMC techniques the weight of a configuration is given by the determinant of the free fermion problem in the Hubbard-Stratonovich field. At this level, the propagation of the fermions in imaginary time and all fermionic exchange terms, have been fully taken into account and condensed into the determinant which only arises because of the fermionic exchange statistics. If one were to sample a bosonic system with a similar approach to DQMC, the resulting weight would be given by a permanent instead of a determinant. This stands in stark contrast to other Monte Carlo flavors, such as worldline QMC techniques where the ignorance of negative statistical weights truly implies that one considers bosonic instead of fermionic exchange statistics and / or couplings are actually changed on the Hamiltonian level from antiferromagnetic to ferromagnetic – there is no difference in the algorithm. These modifications have to be undone explicitly by counting the number of worldline permutations and the number of off diagonal operators from which one calculates the sign of a given configuration that can later be used in data analysis. In a similar spirit, one cannot expect that variational Monte Carlo approaches to be robust when ignoring negative statistical weights because one fully ignores the intricate sign structure of the underlying wavefunction which would usually be an important part of the sampling process. This distinct property of the DQMC method is the most likely source of the huge difference in the splitting of entanglement contributions observed for nodal line wave functions reported here and in earlier work on gapless spin liquids [141].

Despite these promising results, considerably more work needs to be done to further understand whether the results presented here are valid for a restricted class of wavefunctions only or might apply in more generality for the DQMC algorithm and if other Monte Carlo flavors can be modified to reach the same conclusion. From the point of view of algorithmic complexity theory, it would be interesting to have some guidance into whether the calculation of global ground state properties, such as the entanglement entropy, might generally be considered to be in a different complexity class than the calculation of ordinary observables.

## 5.9. Conclusion and outlook

This chapter introduced a numerically stable way to calculate entanglement entropies for many-fermion systems which was benchmarked first on the Hubbard chain and then applied to the Hubbard model on a bilayer square lattice. Finally, a novel perspective on the sign problem was explored by studying entanglement properties of a model of spinless fermions on the honeycomb lattice. With the help of the work presented here, in particular the implementation of the replica trick and the subsequent stabilization, boundary laws and their corrections can now be reliably studied for strongly interacting many-fermion systems.

There are three closely related topics to explore in the future: One is the role of the sign problem. The results presented in this work are very promising but need to be supported by similar studies on other models whose properties are known and can thus serve to verify the ideas. A second important task is to find a model without sign problem but that nevertheless exhibits interesting boundary laws to which the algorithm presented in this chapter can be applied to gain actual physical insight. Promising candidates are for example models with designer Hamiltonians that were already mentioned in the chapter about the sign problem. Finally, although the entanglement algorithm can be stabilized using the techniques described in this chapter, it comes at the price of higher computational cost compared to the conventional correlator based DQMC algorithm. This is where machine learning techniques could potentially make a huge difference by eliminating or severely reducing the need for local updates and thus many of the stabilization steps. The availability of an efficient global update scheme would open the door to studying much larger system sizes which is highly desirable for the study of the scaling behavior of the boundary laws in order to extract information about a potential topological entanglement entropy in a suitable model or, more concretely, the presence of a metallic phase in the Hubbard bilayer model.

# 6. Machine Learning

Machine learning defines a new paradigm of algorithm development. Machine learning algorithms implement a plastic model whose functionality depends on a number of parameters, often referred to as hyperparameters, that are determined during a process called learning at runtime. The values of these hyperparameters define in what way input data is turned into output data. In contrast, traditional algorithms are defined by a static set of hard-coded instructions that transform a given input and into a well-defined output. Note that even algorithms that rely heavily on the use of random numbers, such as the Monte Carlo simulations used throughout this thesis, are ultimately deterministic because the instructions how these random numbers are processed are statically implemented in the program.

Machine learning algorithms can be broadly classified by how the learning process of the hyperparameters is performed: In supervised learning, the hyperparameters are optimized with a pair of input and output data, i.e. with the knowledge which input should generate which output. A prototypical example is the classification of images, in which a set of images, the input data, comes with a set of labels, the output data, that describe what is shown on the respective image. The model should learn to assign labels according to the input images' content so that it can be used to classify unknown pictures that were not used in the learning procedure. In contrast, unsupervised learning algorithms are trained with input data only, i.e. there is no a priori defined output to each input and no restriction on the possible output classes. This is arguably the more powerful approach because allowing the algorithm to come up with its own output classes removes some of the human bias if done properly. The only thing the user has to provide is the metric according to which the algorithm attempts to classify the input images.

There are many different approaches to implement these algorithms, among them clustering [142], decision trees [143], random forests [144] or artificial neural networks (ANNs) [145]. Judging by the number of applications in physics, ANNs stand out among the aforementioned techniques. There are three core topics that are currently of interest to the community: The representation of wavefunctions [146] as neural networks, learning and sampling from distributions for improving the Monte Carlo sampling procedure using neural networks [147] or the elimination of the autocorrelation for significantly faster updating schemes [148, 149] and finally the discrimination and characterization of quantum phases [150, 151, 152, 153].

The results of this thesis pertain exclusively to the application last mentioned, the study of the properties of models for condensed matter systems using ANNs. Following a brief introduction to the inner workings of neural networks, a supervised learning method to extract information about (quantum) phases from Monte Carlo data of any flavor is presented. This method is then extended to an unsupervised method – one that is capable of

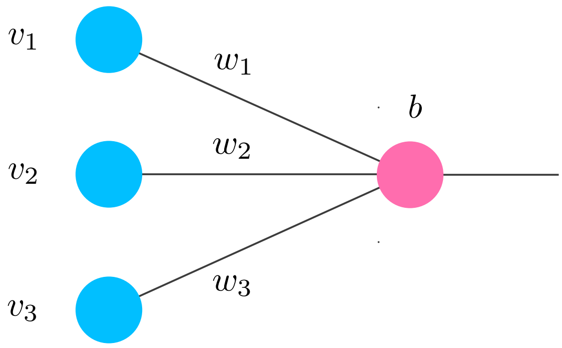


**Figure 6.1.:** A perceptron takes an input vector $\mathbf{v}$ of size $N$ and maps it onto a binary output using a set of $N$-dimensional weight vector $\mathbf{w}$ and a bias $b$.

mapping out entire phase diagrams autonomously from raw Monte Carlo data. In addition, with the help of the unsupervised method, it is also shown that the machine learning approach is capable of seamlessly distinguishing even topologically ordered phases from ordinary, symmetry broken phases without any additional modification or fine-tuning of the algorithm. The chapter ends with an extensive discussion on how this novel approach allows dealing with the fermion sign problem and includes some promising preliminary results on the classification of phases despite the presence of a sign problem. The algorithms and data presented in this chapter were first published in [152, 154] from which this chapter is compiled.

## 6.1. Artificial Neural networks

Artificial neural networks are constructed from interconnected, elementary units called artificial neurons that are designed to mimic the human brain. The simplest type of neurons are called perceptrons [155], see Fig. 6.1. A perceptron processes a given input vector $\mathbf{x}$ by computing the scalar product of the input vector with a weight vector $\mathbf{w}$, subtracting a bias $b$ and then feeding the result into a $\Theta$-function:

$$o(z) = \Theta(\mathbf{x} \cdot \mathbf{w} - b) = \Theta(z)\,, \tag{6.1}$$

where $z$ is called the weighted input and $o(z)$ is the activation function, sometimes simply referred to as the activation. A perceptron, therefore, turns a given input into a binary output. As a side note, perceptrons are, in fact, capable of representing functionally complete logical gates and allow for universal computation [156]. Unlike logical gates, however, their power does not stem exclusively from their connectedness but to a large extent also from the choice of hyperparameters and activation functions.

For actual applications, choosing a $\Theta$-function for calculating the activation is not the optimal choice, since it is not differentiable at zero and has a trivial derivative everywhere else. This will turn out to be a problem for the learning procedure. Instead, one chooses one of many differentiable activation functions, such as sigmoids, ReLUs or most recently *swish* [145, 157].

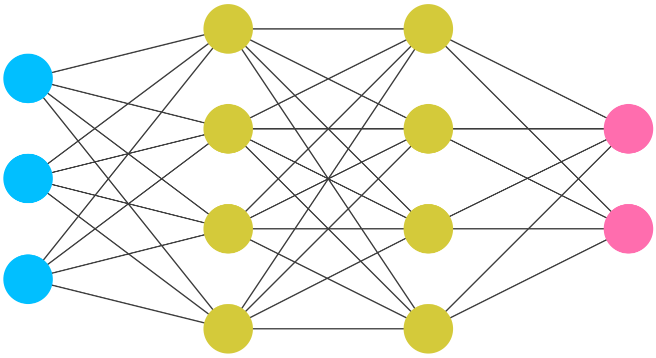


**Figure 6.2.:** A neural network is typically organized in layers of neurons where all neurons are connected to all neurons of the preceding and the succeeding layer.

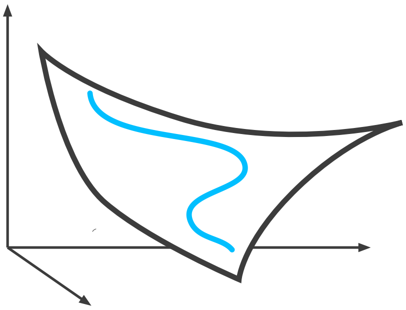


**Figure 6.3.:** The gradient descent method finds minima of functions defining fitness landscapes by starting from an initial point somewhere in the landscape and moving opposite to the gradient at that point.

## 6.1.1. Training neural networks

There are numerous ways of arranging and connecting a large number of neurons to form neural networks with potentially different computational capabilities. In practice, it has proven to be advantageous for training to organize the neurons in layers where neurons within one layer are connected only to neurons from other layers. One example for such a network is shown in Fig. 6.2.

The information enters a neural network via the input layer (blue) and is subsequently propagated through the hidden layers (yellow) until it finally reaches the output layer (red). Along the way, each neuron processes the input vector in combination with its own weights and biases and feeds the output to the next layer. It is thus obvious that finding the right values for the weights and biases is of crucial importance for the network to work as desired. Due to the sheer size of the space spanned by the hyperparameters, finding the correct set analytically is impossible for all but a few select cases, which is why the optimization is done numerically using a method called gradient descent[158]. The basic idea is to find minima of functions by starting at a random point and then moving along the opposite direction of the gradient, see Fig. 6.3. There are numerous technical improvements that facilitate the convergence towards the global minimum that are included in actual algorithms. For a more detailed discussion, the interested reader is referred to [159].

To apply the gradient descent method, a function defining a fitness landscape over the hyperparameters has to be introduced. In the context of neural networks, this function is called a cost function or loss function, usually denoted by $C(y,y')$, which quantifies how much the actual output $y$ differs from the desired output $y'$.

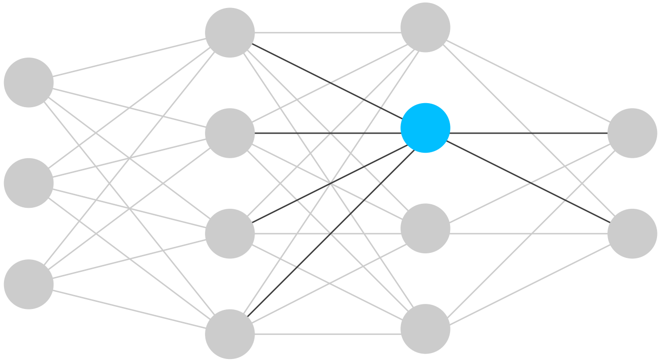


**Figure 6.4.:** In backpropagation, the weights and biases are not optimized all at the same time but iteratively layer by layer. A central step is therefore to derive a formula for the dependence of the cost function on a single neuron and its associated weights and biases which should only depend on values from the preceding and the succeeding layer.

## Implementing gradient descent

As the name suggests, the idea of gradient descent is to slowly descend to the minimum of the cost function along the direction of the gradient. To calculate the gradient, the set of partial derivatives with respect to all hyperparameters is needed which is easy for the output layer but becomes much more difficult for hidden and input layers. The problem is that the activation function of a given layer $l$ takes as inputs the results of the activation functions of layer $l-1$ which in turn uses as input the activation functions of layer $l-2$ and so forth. To avoid this problem, an algorithm called backpropagation was conceived[160, 161, 162, 163]. The basic idea of this algorithm is to start from the last layer, calculate the gradient and update the weights for this layer only and then proceed to do the same for each hidden and finally the input layer. The error that the network made by mapping the input to the output vector $y$ instead of $y'$ is thus propagated backwards through the network. Here, this procedure is illustrated with the concrete example of the quadratic cost function and arbitrary but differentiable activation functions $o(z)$. Figure 6.4 illustrates the idea of backpropagation where the hyperparameters of a neuron in a given layer are optimized based on data from the preceding and the succeeding layer.

*Optimizing the output layer*

The derivatives for the output are the easiest to compute because the weights and biases directly affect the cost function $C$. Output neurons are index by $k$ and the neurons of the previous hidden layer are indexed by $j$. The derivative of the cost function with respect to a weight $w_{jk}$ gives

$$\begin{aligned}
\frac{\partial C}{\partial w_{jk}} &= (a_k - y'_k)\frac{\partial}{\partial w_{jk}} a_k \\
&= (a_k - y'_k)\frac{\partial}{\partial w_{jk}} o(z_k) \\
&= (a_k - y'_k)\frac{do}{dz_k}\frac{\partial}{\partial w_{jk}} z_k \\
&= (a_k - y'_k)\frac{do}{dz_k} a_j \,. && (6.2) \\
& && (6.3)
\end{aligned}$$

The derivative is thus made up of the difference between the output neurons output and the actual output, the derivative of its activation function and the output $a_j$ of the neuron connected by the weight $w_{jk}$. Introducing the shorthand notation $\delta_k = (a_k - y'_k)\frac{do}{dz_k}$ allows rewriting this in the following concise form

$$\frac{\partial C}{\partial w_{jk}} = \delta_k a_j \,. \tag{6.4}$$

This notation is chosen to emphasize that $\delta_k$ is supposed to be the error of the signal. The closer the output signal of the network is to the actual value, the smaller $\delta_k$ becomes. From this equation, the output weights are updated as

$$w_{jk} \leftarrow w_{jk} - \eta \frac{\partial C}{\partial w_{jk}} \,, \tag{6.5}$$

where $\eta$ is called the learning rate which controls how large the step along the gradient is.

The biases are updated in a very similar fashion, with the exception that they are not proportional to the input signal, so they become

$$\frac{\partial C}{\partial b_k} = (a_k - y'_k)\frac{do}{dz_k} = \delta_k \tag{6.6}$$

and are updated just like the weights.

*Optimizing hidden layers*

Deriving the equations for hidden layers is a little more involved because they affect the cost function $C$ only indirectly. Nevertheless, the derivation starts out similarly:

$$\begin{aligned} \frac{\partial C}{\partial w_{ij}} &= \sum_k (a_k - y'_k)\frac{\partial}{\partial w_{ij}} a_k \\ &= \sum_k (a_k - y'_k)\frac{do}{dz_k}\frac{\partial}{\partial w_{ij}} z_k \,. \end{aligned} \tag{6.7}$$

Assuming that the hidden layer considered here is the last before the output layer, the derivative of $z_k$ is

$$\begin{aligned} \frac{\partial z_k}{\partial w_{ij}} &= \frac{\partial z_k}{\partial a_j}\frac{\partial a_j}{\partial w_{ij}} \\ &= w_{jk}\frac{\partial a_j}{\partial w_{ij}} \\ &= w_{jk}\frac{do}{dz_j}\frac{\partial z_j}{\partial w_{ij}} \\ &= w_{jk}\frac{do}{dz_j} a_i \,. \end{aligned} \tag{6.8}$$

Thus, the derivative of the cost function becomes

$$\begin{aligned} \frac{\partial C}{\partial w_{ij}} &= \sum_k (a_k - y'_k)\frac{do}{dz_k} w_{jk}\frac{do}{dz_j} a_i \\ &= \frac{do}{dz_j} a_i \sum_k \delta_k w_{jk} \\ &= \delta_j a_i \,. \end{aligned} \tag{6.9}$$

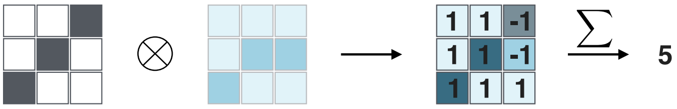


**Figure 6.5.:** A convolutional filter operates on a small section of the image that is of the same size as the filter itself. By a pixel-wise comparison of the filter and the image section and subsequent summation, a single value is obtained.

The procedure is identical for the layers further away from the output layer. Put into words, the gradient for a hidden layer is calculated from the backpropagated error signal and weigh it with the corresponding feed-forward signal.

Again, the biases are updated using the backpropagation error signal only:

$$\frac{\partial C}{\partial b_i} = \delta_j \, . \tag{6.10}$$

*Summary of backpropagation*

To make backpropagation work, one needs to first calculate all feed-forward signals from input to output. These are used to determine the value of the cost function. The difference between the output of the neural network and the desired output is used to calculate the backpropagation error which in turn is used to update the hyperparameters of a given layer. This procedure is repeated from output to input layer until all hyperparameters have been updated once.

In practice, the input vectors are not fed into the network one by one but in so-called mini batches which are small subsets of the entire input dataset. The gradient is calculated is averaged over the entire mini batch before the hyperparameters are adjusted. This has the advantage of preventing too frequent updates to the weights which is computationally expensive while at the same time preventing premature convergence of the algorithm which can happen when large portions of or even the entire input dataset is used for averaging.

Another problem one has to worry about is overfitting, which means that the neural network learns to handle the training data extremely well but performs poorly on newly presented data. One has to experiment with the various tuning options of the neural network, such as its size, the training rate, or the mini batch size to obtain results that are robust over a certain range of parameter values. More approaches to tackle this problem are described below.

### 6.1.2. Convolutional Layers

The neural networks introduced earlier are powerful tools that can realize an abundance of possible functions based on their connectedness and choice of hyperparameters. However, it is sometimes advantageous to preprocess the input data which can drastically reduce the number of layers, neurons, and thus the number of hyperparameters necessary to achieve a desired task. One such preprocessing layer is a convolutional layer. The fundamental

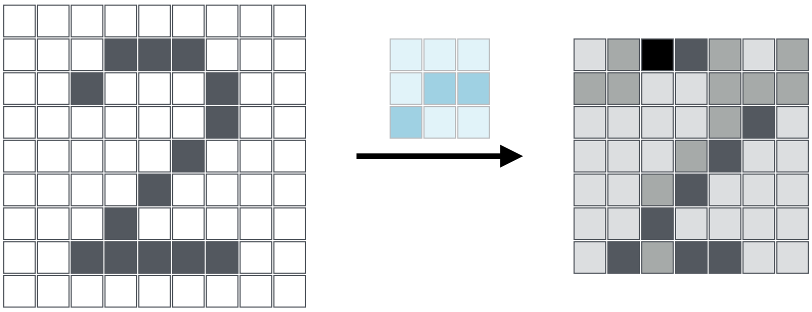


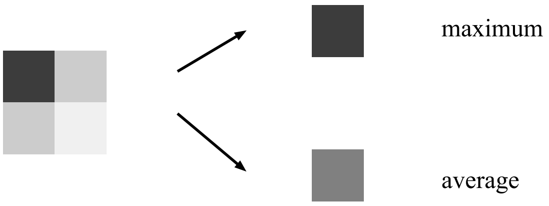
**Figure 6.6.:** Illustration of the action of a convolutional filter on a sample input image. By applying the convolutional filter to all possible sections of the image, a new image is generated.

advantage of using a convolutional layer is that it can encode some degree of locality present in the input vector, which in practice may translate to words being adjacent in text analysis or pixels of an image being direct or next-nearest neighbors. It does so by providing a number of trainable filters that are moved across the input data and optimized in such a way that they find distinct patterns that occur frequently. The filter outputs a number for each segment of the input data it is applied to, leading to as many new input vectors as the convolutional layer has filters, see Fig. 6.5. A full example is shown in Fig. 6.6 where a filter that was optimized to detect certain parts of how the number '2' is encoded in pixel values is shown. The filter is applied first in the top left corner of the image, resulting in a single number calculated from the sum of the element-wise product of the filters' values and the section of the input vector. It is then moved one pixel to the right and again evaluated to give a second data point. This procedure is repeated until the entire input vector is processed. At first sight, it might seem counterintuitive that creating an even larger input vector (due to the number of convolutional filters) would help in training the neural network. But experience has shown that only with the help of convolutional layers neural networks have become capable of successfully classifying images, sound and text bites. A fully connected neural network would need a much larger number of neurons to realize the same functionality which would in turn be very difficult to train as well as to physically store on a computer and thus the use of convolutional filters is the preferred approach.

### 6.1.3. Pooling layers

Another type of layer that can be inserted before the input is flattened and processed via a fully connected network is a pooling layer. Pooling layers downsample data with the goal of reducing the amount of data and therefore computational cost, as well as pronouncing the most important features in the data which aims to prevent overfitting. This is done by sliding yet another window over the input data which, in contrast to the convolutional layer, should be done with a stride size greater than one, in order to actually produce a smaller output. The most popular options for processing the data within a particular window are maximum (max) pooling and average pooling, with which either the maximum value or the average value of all values within a given window are used as the output, respectively. This is illustrated in Fig. 6.7 for a max pooling operation with a filter size of 2x2.

**Figure 6.7.:** A pooling layer is used to reduce the image size by reducing a small section of the image.



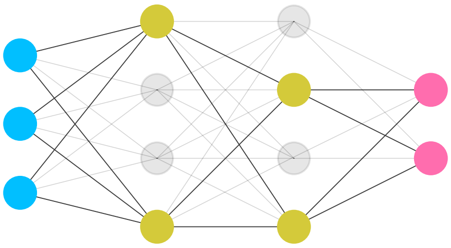


**Figure 6.8.:** A dropout filter deactivates a certain percentage of the neuron connection at random to avoid excessive specialization of the neurons on one particular input vector.

### 6.1.4. Dropout filters and regularization of hyperparameters

Overfitting can be a serious problem in practical applications. Two techniques that are specifically aimed at preventing overfitting are dropout filters and regularization of hyperparameters. A dropout filter, see Fig. 6.8 can be thought of as a gated connection between two layers where at each training step, only parts of the gates are open, typically 50%. Which connections are active is decided randomly for each mini batch with the goal of preventing specific neurons to learn one particular feature of the training image. By constantly changing the ensemble of active neurons, the gradient descent procedure is less likely to fall into local minima. Directly aimed at keeping neurons plastic, regularization penalizes large values of the hyperparameters which could potentially push neurons into saturation and prevent further training. It can be implemented in various ways, one of the most popular being $L_2$ regularization, which simply adds the weighted $L_2$ norm of the weight vector for a given layer to the loss function, aiming to keep it small.

## 6.2. Supervised approach to the discrimination of phases of matter

A Hamiltonian consisting of competing terms each is potentially capable of realizing a variety of different physical phases. Each of the terms is weighted with a tunable parameter and the particular choice of parameters decides which phase is realized. To discover which phase belongs to which parameter region, one typically chooses to measure order parameters, to calculate entanglement entropies [82] or resorts to measure other non-local features such as Wilson loops [164] or the modular S matrix[165] if possible.

However, which is of these methods is best suited to map out the phase diagram is not obvious because they are tailored towards identifying specific types of phases. In addition, the actual numerical implementation of the measurement process can be very costly or even impossible depending on the method at hand. This is where neural networks come in as a possible general purpose tool that can learn from the raw Monte Carlo data generated by the simulation. The motivation for such an approach is that neural networks excel at identifying patterns and structure in data and it will be shown that (quantum) phase transitions are accompanied by very distinct changes in precisely those patterns of the Monte Carlo data which can then be exploited for their identification and characterization.

### 6.2.1. Learning the characteristics of a phase

The Ising model [166] serves as a prototypical example to illustrate this idea: The presence of (anti-)ferromagnetic order is detected by an order parameter given by the (alternating) sum of all spin directions on the lattice. In a machine learning approach the entire spin configuration is used as the input. In the magnetized phases, a typical configuration looks like either a uniformly colored rectangle or a checkerboard pattern while the paramagnetic phase is characterized by images that look completely random. The observation that different types of orders are associated with certain patterns in fundamental quantities is not unique to the Ising model but carries over to other more complicated models such as Hubbard models of bosons and fermions which are the focus of this chapter.

In the general case, the procedure works as follows: One considers a (quantum) many-body Hamiltonian that, as a function of some parameter $\lambda$, exhibits a phase transition between two phases – such as the aforementioned thermal phase transition in the classical Ising model [150] or the zero-temperature quantum phase transition as a function of some coupling parameter [152]. In such a setting, where one has prior knowledge about the existence of two distinct phases, one can train a convolutional neural network in a supervised setting with labeled representations of the respective phases acquired by Monte Carlo sampling deep inside the two phases. One is in principle free to choose which specific representation to use for the training process, but often the employed flavor of Monte Carlo offers at least one straightforward choice. For the Ising model, this obvious choice is images of the spin configurations [150]. After successfully training the CNN to distinguish these two phases (which typically requires a few thousand training instances), one can then feed unlabeled instances, sampled for arbitrary intermediate parameter values of $\lambda$, to the CNN in order to locate the phase transition between the two phases, see also the schematic illustration of Fig. 6.9. This approach has been demonstrated to produce relatively good quantitative estimates for the location of the phase transition [150, 152, 167, 168, 169] and might even be finessed to be amenable to a finite-size scaling analysis for second-order phase transitions [150].

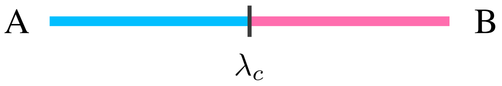


**Figure 6.9.:** Setup for supervised discrimination of two phases $A$ and $B$ (left and right) separated by a critical coupling strength $\lambda_c$ using neural networks.

### 6.2.2. Network architecture

The network architecture is chosen to be similar to the ones that have already been used successfully in other contexts of pattern recognition, in particular image recognition. One

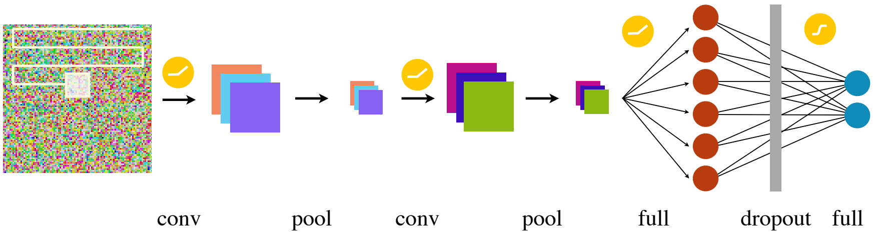


**Figure 6.10.:** Schematic illustration of the neural network used in this work. A combination of convolutional (conv) and max-pooling layers (pool) is first used to study the image, before the data is further analyzed by two fully connected neural networks separated by a dropout layer. The convolutional and the first fully connected layer are activated using rectified linear functions, while the final layer is activated by a softmax function. Figure first appeared in [152].

such architecture is shown in Fig. 6.10, which has been employed throughout this thesis, sometimes with minor modifications. It consists of two main components – a convolutional and a fully connected part. Because the exact values for the number of hyperparameters may vary slightly from project to project, only typical choices are reported here. The convolutional part processes the data by a combination of two convolutional and max-pooling units. Both of these units are activated by a rectified linear function (relu) and have filters typically of size $3 \times 3$ with about 32 filters for the first and 64 filters for the second layer. The data is then fed into a fully connected, relu activated layer of usually 512 neurons. To avoid overfitting, a dropout regularization at a rate of 0.5 is applied to this layer until the result can be read out from a softmax layer. The optimization of the neural network is performed using the cross-entropy as a cost function and ADAM [170] as a particularly efficient variant of the stochastic gradient at a learning rate of typically $\gamma = 0.0001$.

The number of epochs is adjusted dynamically based on the timeseries of the loss function. If desired, errorbars can be generated by recording data for a number of epochs and considering the mean and variance because they weights continue to be adjusted slightly as different mini batches are used in training. The networks used in this work were implemented using the TensorFlow library [171].

### 6.2.3. Finding the correct input

The neural network employed in this thesis was designed with image recognition in mind and therefore whatever type of input is chosen should resemble an image. In DQMC simulations, the two principal objects that lend themselves naturally to be used as input to the neural network are the auxiliary field and the Green's function. A priori, one would expect both of them to work equally well as they were both shown to play a fundamental role in the algorithm (see chapter 2). To verify this assumption, the by now familiar spinful Hubbard model on a honeycomb lattice is used:

$$H = -t \sum_{\langle i,j \rangle,\sigma} c^{\dagger}_{i,\sigma} c_{j,\sigma} + U \sum_i n n_{\downarrow,i} \,. \tag{6.11}$$

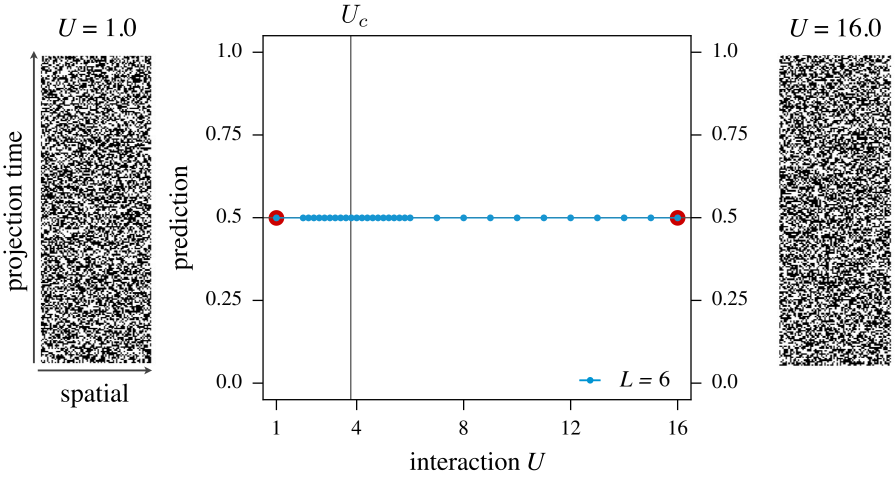


**Figure 6.11.:** Results from training the neural network on Hubbard-Stratonovich field configurations of a spinful Hubbard model on a $2 \cdot 6 \times 6$ lattice with on-site interaction $U$. Reference points for training were $U = 1.0$ and $U = 16.0$, marked by red dots in the figure. Despite intensive training, the network depicted in Fig. 6.10 is unable to distinguish the auxiliary field configurations of the two reference points and as a consequence cannot be used to discriminate between the two phases. Figure first appeared in [152].

For assessing the suitability of the auxiliary fields, the interaction is decoupled in both the charge and the magnetic channel, see (2.34). and (2.34), respectively. For each of these decouplings, 8192 auxiliary field configurations and Green's functions are sampled along the Markov chain for many different values of the on-site interaction $U$. Further parameters are a discretization step of $\Delta\tau = 0.1$ and a projection time $\theta = 10$. Thus, the auxiliary field for the spinful Hubbard model is of size $2 \cdot L^2 \times 200$ and the Green's functions are of size $2 \cdot L^2 \times 2 \cdot L^2$.

## Learning auxiliary field configurations

Regardless of the type of transformation employed, the training of the CNN from Fig. 6.10 is performed with auxiliary field configurations from two extremal values of the interaction strength $U$, namely $U = 1.0$ and $U = 16.0$. After 32 epochs, training is stopped and the network is asked to predict to which of the phases configurations from intermediate couplings strengths belong. The results of this training and prediction procedure for auxiliary fields from simulations with decoupling in the charge channel are shown in Fig. 6.11.

The side panels show one sample of the auxiliary field configuration from the two respective extremal points. At first sight, there is no discernible difference although these configurations belong to two very different points in the phase diagram. As indicated by the flat line and the constant prediction of 0.5, the neural network is equally incapable of distinguishing these two phases based on the auxiliary field when decoupling the charge channel.

The situation is improved when decoupling in the magnetic channel, the results for which are shown in Fig. 6.12. In stark contrast to the previous case, the auxiliary field configurations are visibly different for the two reference values of the interaction $U$. For weak interactions, $U = 1.0$, the auxiliary spins are arranged in a random, noisy pattern while at strong interactions, $U = 16.0$, the spins form long lines in the direction of projection time.

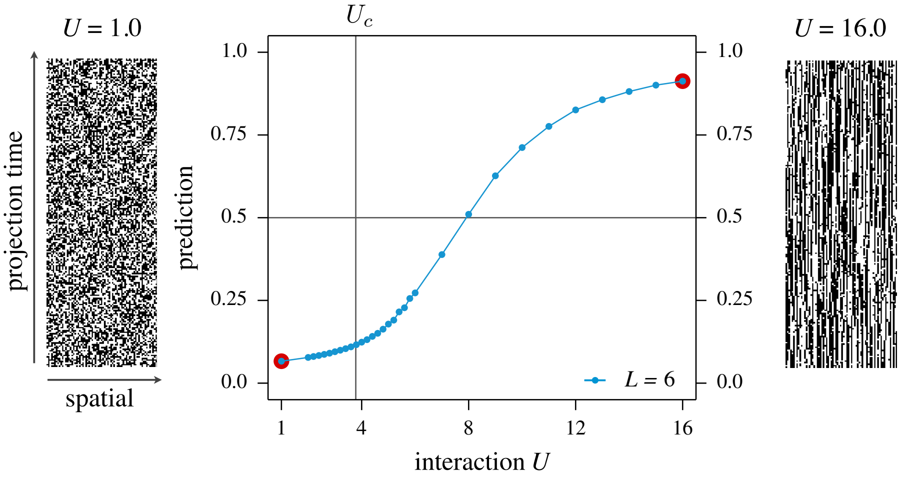


**Figure 6.12.:** Results from training the neural network on auxiliary field configurations of a spinful Hubbard model on a $2 \cdot 6 \times 6$ lattice with on-site interaction $U$ using a Hubbard-Stratonovich decoupling in the magnetic channel. Reference points for training were $U = 1.0$ and $U = 16.0$, marked by red dots in the figure. The neural network is able to distinguish the two phases, but the prediction of the critical point lacks accuracy. Figure first appeared in [152].

Judging purely from the difference in appearance, a neural network should be capable of distinguishing the two phases. This impression is confirmed by the prediction produced by the network which shows that the network is capable of finding enough unique features distinguishing the two respective phases. However, despite being generally capable of distinguishing the phases, the supposed critical point for this system size is located at $U \approx 8.0$, far away from the actual value $U \approx 3.86$ even when considering that $U \approx 3.86$ is the value for the infinite system.

## Learning Green's functions

The second option for constructing an input that naturally comes to mind is to use Green's functions, i.e. the two-point correlator $\langle c_i c_j^\dagger \rangle$. Its two index structure is already very suggestive of being representable as a square matrix, which in turn can be interpreted as an image, where the number of "color" channels is determined by whether the entries of the Green's function are real-valued or complex-valued. The Green's functions used as an input here are obtained from the same simulation and correspond each to one of the auxiliary field configurations used previously. Applying the same training procedure as for the auxiliary fields, one obtains the curves shown in Fig. 6.14. The side panels each show four different samples of Green's functions from deep inside the two phases. Because the entries of the Green's functions are actually complex numbers, they were transformed for the purpose of visualization as described in the figure's caption. Note, however, that the network was trained using the raw, complex-valued data. In contrast to the training with auxiliary field configurations, a relatively sharp signal is obtained that locates the value of the phase transition close to the actual value.

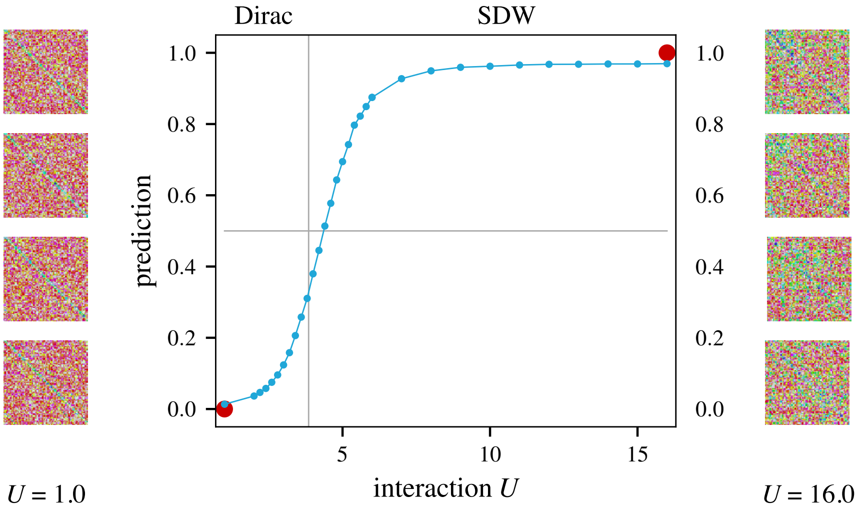


**Figure 6.13.:** The main panel shows results from training a network on Green's functions for the same parameter settings as in Fig. 6.11 and 6.12. For the visualization of the samples of Green's functions in the side panel, the complex entries of these matrices are color-converted by interpreting their absolute value as the hue of the color and their phase angle as the saturation (HSV coloring scheme [172]). The red dots indicate the location of the interaction values used for training. Figure first appeared in [152].

### Conclusion

In conclusion, the auxiliary field configurations can in principle be used as an input. However, their usability depends strongly on the type of Hubbard-Stratonovich transformation used and they show a strong finite size effect, at least for the system considered here. The results could possibly be improved by leaving the dimensions of the auxiliary field unflattened at the rather high cost of working with genuinely three-dimensional tensors as inputs to the network [173]. However, not every problem allows freely choosing the type of Hubbard-Stratonovich transformation due to its Hamiltonian that may only allow for one particular decoupling. Other reasons include the stability of the algorithm or the efficiency of the updates. In contrast to the auxiliary fields, the Green's functions perform much better and are always accessible regardless of the underlying type of decoupling. In particular, they are a general and quantity and accessible in other QMC methods that are not based on auxiliary fields such as Stochastic Series expansion. Thus, from now on, Green's functions will be used exclusively as the input for the neural network unless mentioned otherwise.

## 6.2.4. Machine learning a fermionic quantum phase transition

Having established that the Green's function is the object of choice, the transition of the spinful Hubbard model on the honeycomb lattice is studied more closely, see Fig. 6.14. The critical value of the interaction strength can now be located quite close to the actual value, right in between the value for the infinite system ($U \approx 3.85$) and the finite size value that was previously determined for the largest system studied here ($U \approx 4.2$ [174]).

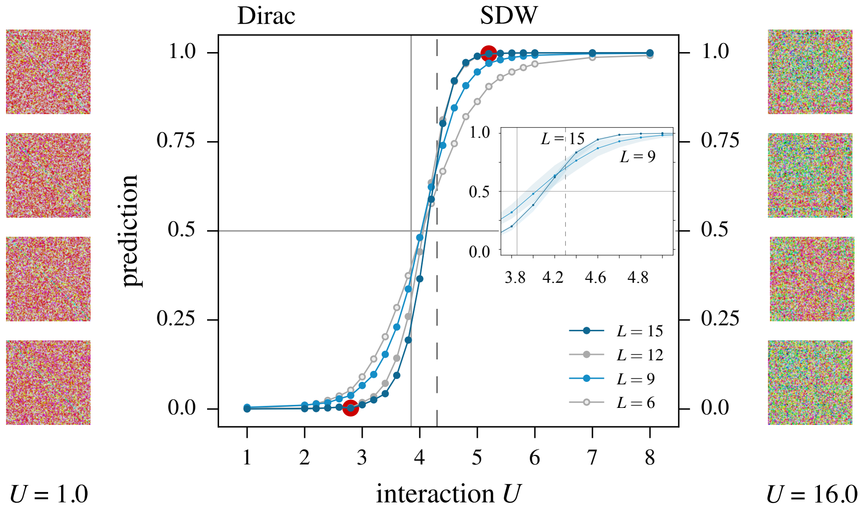


**Figure 6.14.:** Machine learning of the phase transition from a semi-metal to an antiferromagnetic insulator in the spinful Hubbard model (6.11) on a honeycomb lattice using the Green's function approach (see main text). Visualized in the side panels are representative samples of the Green's function (calculated from the auxiliary field) for a $2 \cdot 9 \times 9$ system in the two respective phases. The main panel shows the prediction obtained from a CNN trained for parameters in the two fermionic phases (indicated by the red dots). Data for different system sizes $2 \cdot L \times L$ are shown where the colors were selected to highlight an apparent even-odd effect in the linear system size. The vertical solid line indicates the position of the phase transition in the thermodynamic limit [175], while the dashed line marks the position at which the antiferromagnetic order breaks down [174] for the finite system sizes of the current study. Figure first appeared in [152].

## 6.3. Unsupervised approach to mapping out phase diagrams

One major shortcoming of the supervised scheme for distinguishing phases is that one has to have a rough idea of which range of parameters belongs to which phase. A priori, these ranges will be unknown and thus the supervised approach is not generally suitable. In particular, it might occur that in between two extremal values that realize different phases, one or more other phases are realized which would be missed by training on extremal values only. For those situations, it is desirable to have an unsupervised approach that is capable of autonomously discovering the relevant parameter ranges.

### 6.3.1. Turning supervised into unsupervised

The key idea of the unsupervised approach presented here is to slide a small window through the parameter space of the to-be-determined phase diagram and to test whether it is possible to positively discriminate two distinct phases for the two boundary parameters. The latter can be accomplished by employing the original supervised approach, i.e. by making an attempt to train a CNN on labeled instances for the two boundary parameters and to subsequently determine whether the training indeed led to a positive discrimination of the two sets of instances. For a parameter window that encloses a phase transition, the expectation is that a positive discrimination is indeed found, i.e. the CNN produces prediction values of $p = 0$ and $p = 1$ for the two sets of instances, respectively. A parameter window that for its full extent resides within one given phase, however, should not

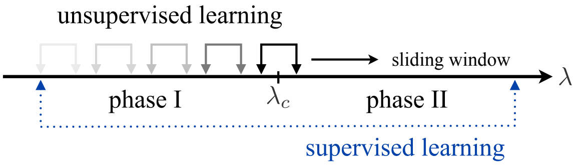


**Figure 6.15.:** Schematic illustration of the unsupervised machine learning approach. For a small parameter window, which is slid across parameter space, a discrimination of phases at its endpoints A and B is attempted via a supervised learning approach. A positive discrimination via the underlying convolutional neural network is expected only if the parameter window indeed encompasses a phase transition, while it should fail when points A and B reside in the same phase. Figure first appeared in [154].

result in a positive discrimination of the instances and should result in a prediction value of $p = 0.5$ for *all* instances (indicating a maximal confusion). This approach is indeed unsupervised in the sense that one can introduce a metric that quantifies how well the instances used in a given training procedure can be positively discriminated. Specifically, we consider the “label distance" $d(\lambda_1, \lambda_2)$ as the integral over the prediction values $p$ in $\lambda$-space between two points $\lambda_1$ and $\lambda_2$

$$d(\lambda_1,\lambda_2) = \Theta\left(\int_{\lambda_1}^{\lambda_2} \mathrm{d}\lambda\,(p(\lambda) - 0.5) - \varepsilon\right), \tag{6.12}$$

which indicates how close the assignment of labels for instances at these two points should be. If they belong to the same phase, the prediction will in theory always be 0.5 and the integral will evaluate to 0, while it will be 1 if the instances for the two values of $\lambda$ are distinguishable (and separated by a phase transition). In practice, deviations from the ideal values can be accounted for by introducing a threshold difference $\varepsilon$.

### 6.3.2. Application to hard-core bosons

The first test system is the Bose-Hubbard model[176], which captures the competition between kinetic and potential energies that stabilize superfluid and Mott insulating phases, respectively. Adding nearest-neighbor repulsion or ring exchange terms can enrich its phase diagram by supersolid [177, 178, 179] or $d$-wave-correlated Bose liquids [180]. For the purpose of evaluating the proposed method, a model of hard-core bosons on the square lattice [181] subject to a nearest-neighbor Coulomb repulsion $V$ and a chemical potential $\mu$ is chosen. Its Hamiltonian is

$$\mathcal{H} = -t\sum_{\langle i,j\rangle}\left(a_i^\dagger a_j + a_j^\dagger a_i\right) + V\sum_{\langle i,j\rangle} n_i n_j - \mu\sum_i n_i\,, \tag{6.13}$$

where $n_i = a_i^\dagger a_i$ are the usual boson operators in second quantization. Its ground-state phase diagram [182, 181] exhibits four different phasesas illustrated in Fig. 6.17a) – besides the trivial, fully filled or completely empty, ground states there is an extended superfluid phase along with a checkerboard solid. At finite temperatures, the model exhibits continuous phase transitions to a normal fluid both from the checkerboard solid (second order) and the superfluid (Kosterlitz-Thouless) as illustrated in Fig. 6.18a). This abundance of different phases and different types of phase transitions is ideally suited

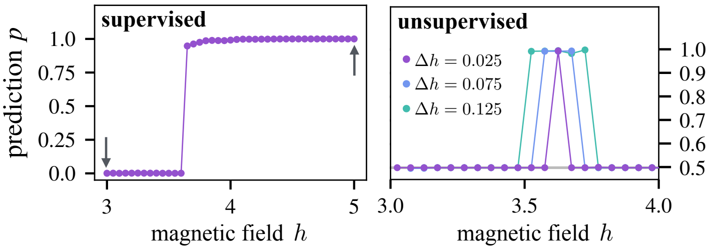


**Figure 6.16.:** Identification of the first-order quantum phase transition between the checkerboard solid and superfluid phase for $\Delta = 3$ employing (a) a supervised and (b) the unsupervised ML approach. Figure first appeared in [154].

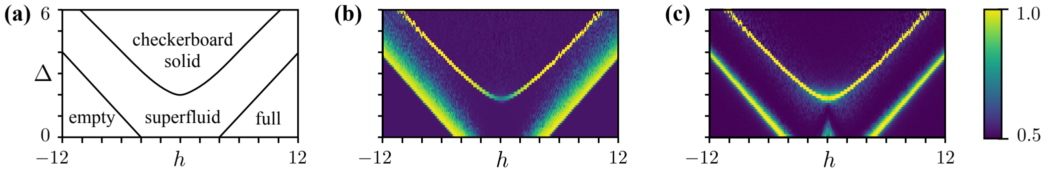


**Figure 6.17.:** Zero-temperature phase diagram of interacting hard-core bosons. Panel (a) shows the phase diagram extracted from quantum Monte Carlo (QMC) simulations [182, 181]. Panels (b) and (c) show the phase diagrams extracted from our unsupervised ML approach applied to correlation functions sampled in QMC simulations (for $L = 8$). Panel (b) is based on the diagonal correlation function $\langle S_i^z S_j^z \rangle$ and panel (c) on the off-diagonal correlation function $\langle S_i^+ S_j^- \rangle + \langle S_j^+ S_i^- \rangle$, respectively. Figure first appeared in [154].

to benchmark the proposed unsupervised approach against numerically exact results from large-scale Monte Carlo simulations [181]. The simulations are carried out using Stochastic Series Expansion [183] to study an anisotropic spin-1/2 model in a magnetic field to which the original Hubbard Hamiltonian (6.13) can be mapped [184] to give

$$\mathcal{H} = -\sum_{\langle i,j\rangle}\left(S_i^+ S_j^- + S_i^- S_j^+\right) + \Delta \sum_{\langle i,j\rangle} S_i^z S_j^z + h\sum_i S_i^z . \tag{6.14}$$

In this mapping, $\Delta = V$, $h = 2V - \mu$ for $t = 1$ The systems under study are of linear system sizes $L = 8, 16, 24, 32$ and were sampled at inverse temperatures of up to $\beta = 40$.

The first step is to map out the phase boundaries of the ground-state phase diagram in the $(\Delta, h)$ plane by shifting a training window (of width $\delta_\Delta = 0.2$) vertically across the parameter space. In the spirit of devising a rather general algorithm that uses no prior knowledge about the phase diagram and the specific nature of its phases, the CNN is fed with equal-time Green's functions sampled in the Monte Carlo simulation, as was done for the supervised scenario. Specifically, for this model the diagonal correlation function $\langle S_i^z S_j^z \rangle$ and the off-diagonal correlation function $\langle S_i^+ S_j^- \rangle + \langle S_j^+ S_i^- \rangle$ are used as input. Fig. 6.16 illustrates results for an example cut at $\Delta = 3$ as a function of $h$ comparing the supervised and the unsupervised learning approach.

Figure 6.16a) shows how the supervised method allows identifying the location of the phase transition via the change in the prediction function when trained deep in the two phases ($h_{1,2} = 3.0, 5.0$). Figure 6.16b) shows the same transcription from the perspective of the unsupervised scheme put forward in the preceding section where a training window of varying length is moved along the cut. A singular peak in the average prediction success clearly indicates the location of the phase transition, with the peak narrowing for shorter window width as expected. Results of such cuts are compiled into an entire phase diagram shown in Figs. 6.17b) and c) where the average prediction success is plotted. The diagrams reveal several sharp transitions and, in fact, traces out the phase diagram in superb quantitative agreement with the original MC analysis [181].

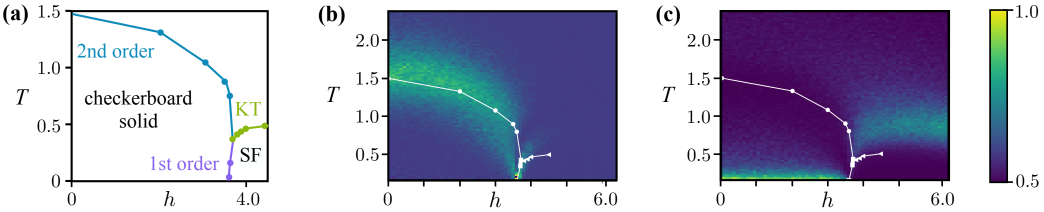


**Figure 6.18.:** Finite-temperature phase diagram of interacting hard-core bosons. Panel (a) shows the phase diagram found in quantum Monte Carlo simulations [181], panels (b) and (c) show results form our unsupervised ML approach (for $L = 8$). For Panel (b) the diagonal correlation function $\langle S_i^z S_j^z \rangle$ is fed into the CNN, for panel (c) the winding number per site. The white lines indicate the phase boundaries of panel (a). Figure first appeared in [154].

The broadening of the transition from one of the trivial states into the superfluid in the diagonal correlations reflects their slower decay in comparison with the rapid change of the off-diagonal correlations.

Tracing out the thermal transitions to construct the finite-temperature phase diagram of model (6.14) turns out to be somewhat more difficult. The results obtained via the unsupervised approach applied to the diagonal correlation function $\langle S_i^z S_j^z \rangle$ are shown in Fig. 6.18b). The second-order transition between checkerboard solid and normal fluid results in a relatively broad signature, which is mostly due to the moderate system size ($L = 8$) underlying this comprehensive sweep of the phase diagram. It is interesting to note that while the Kosterlitz-Thouless (KT) transition out of the superfluid leaves no visible trace in the analysis of the diagonal correlation function, it leaves a broad signal in the off-diagonal correlation function (not shown). This reflects the intrinsic inefficiency of local observables to capture the non-local nature of the vortex-antivortex unbinding at a KT transition. Alternatively, the CNN can be fed with explicit information about winding numbers for configurations sampled in the Monte Carlo simulation, e.g. the winding number per site in one of the spatial directions. Each sample of the winding number was normalized with respect to its largest absolute value to ensure values in the range of $(-1, 1)$. The idea is to emphasize the *pattern* of the distribution of windings rather than the absolute values, which would be used for a direct estimation of the order parameter. This results in a clear signal located slightly above the actual KT transition, see Fig. 6.18c). The finite-size trend of this peak is shown in Fig. 6.19. While the feature broadens with increasing system size, the peak systematically enhances for larger systems and clearly shifts towards the MC estimate.

### 6.3.3. Fermions and topological order

The second example to illustrate the effectiveness of the unsupervised learning approach is a model of Dirac fermions coupled to a fluctuating $Z_2$ gauge field,which exhibits a phase transition from a deconfined, topologically ordered phase to an antiferromagnet

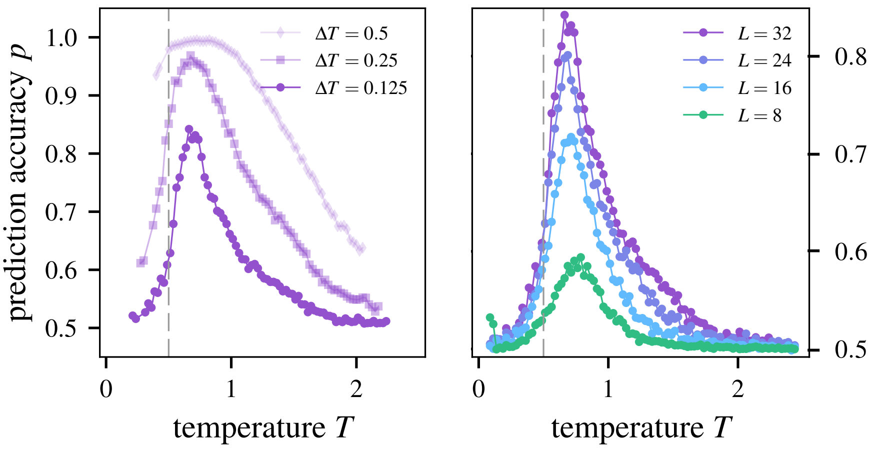


**Figure 6.19.:** Identification of the finite-temperature **Kosterlitz-Thouless transition** between superfluid and normal fluid for $h = 5.0$ when feeding the CNN with winding numbers of Monte Carlo configurations. The dashed lines indicate the known location of the transition. Figure first appeared in [154].

[185, 186]. Defined on a square lattice, its Hamiltonian reads

$$\begin{aligned}\mathcal{H} = \quad & \sum_{\langle i,j\rangle} Z_{\langle i,j\rangle} \left( \sum_{\alpha=1}^{N} c^{\dagger}_{i,\alpha} c_{j,\alpha} + h.c \right) - Nh \sum_{\langle i,j\rangle} X_{\langle i,j\rangle} \\ & + NF \sum_{\square} \prod_{\langle i,j\rangle \in \partial\square} Z_{\langle i,j\rangle} \,, \end{aligned} \tag{6.15}$$

where in this study there are $N = 2$ species of fermions with creation/annihilation operators $c^{\dagger}_{i,\alpha}/c_{i,\alpha}$ and bond spin operators $Z_{\langle i,j\rangle}$ and $X_{\langle i,j\rangle}$ that correspond to the usual Pauli spin-1/2 matrices. Since

$$Q_i = (-1)^{\sum_\alpha c^{\dagger}_{i,\alpha} c_{i,\alpha}} \prod_{\delta = \pm a_x, \pm a_y} X_{\langle i, i+\delta\rangle} \tag{6.16}$$

commutes with (6.15), the Gauss law, $Q_i = -1$, is imposed dynamically in the zero temperature limit and on any finite-sized lattice. The original model of Ref. [185] was supplemented with a flux term of magnitude $F = 1/2$ which shifts the critical value for the magnetic field toward larger values. The phase transition is then found to be $h_c \approx 0.40$ [187].

The data for the machine learning approach is generated with finite temperature auxiliary-field quantum Monte Carlo [188] as implemented in the ALF-package [189, 15] with the latter providing samples of the equal-time single-particle Green's function $\langle c^{\dagger}_i c_j \rangle$ for an inverse temperature $\beta = 40$. As Fig. 6.20a) clearly demonstrates, the highly non-trivial phase transition in model (6.15) can be readily located using the unsupervised approach – there is a sharp peak located right at the expected value of the transition for varying system sizes. This might be surprising at first sight as one might expect that the non-local nature of the topologically ordered phase might pose similar problems as the identification of the vortex-antivortex unbinding at a topological phase transition (as discussed above). Indeed, a recent machine learning based identification of topological order [168] succeeded only because of the addition of explicit non-local filters (akin to the convolutional filter of a CNN). In the context of model (6.15) such steps are not necessary as the topological nature of the deconfined Dirac phase can reveal itself already on relatively modest length

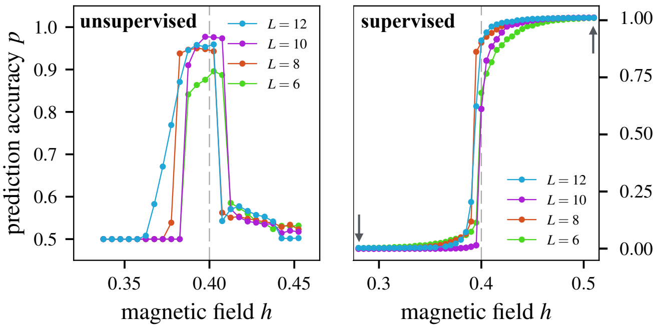


**Figure 6.20.: Detection of phase transition to topological order** in model (6.15) of fermions coupled to a fluctuating $Z_2$ gauge field employing (a) the unsupervised and (b) a supervised ML approach. Figure first appeared in [154].

scales – the proliferation of vison excitations at the transition are bound to plaquettes of the square lattice and as such easily detectable. Although visons are very local, the CNN is fed with snapshots of the Green's function $\langle c^\dagger_{i,\sigma} c_{j,\sigma}\rangle$ – a quantity that, taken at face value, contains very little information. Since the simulations are SU(2)-spin invariant, each snapshot also has no spin dependence. Furthermore, since $\{Q_i, c_{i,\sigma}\} = 0$, it follows that $\langle c^\dagger_{i,\sigma} c_{j,\sigma}\rangle = \delta_{i,j}/2$, reflecting the fact that the Green's function is a gauge-dependent quantity. Note that the latter equation holds only after averaging over snapshots. As such, it is certainly remarkable that the CNN can detect in such a precise manner the phase transition between a topologically ordered state and an antiferromagnet.

The supervised learning approach can be used as a consistency check, which is shown in Fig. 6.20b) where a CNN was trained deep inside the two respective phases (indicated by the arrows). As expected, the prediction changes from 0 to 1 right at the transition (dashed line). Note that both machine learning approaches like a standard analysis of the phase transition using RG-invariant quantities [185, 187] are relatively sensitive to finite-size effects, which makes it hard to infer the order of transition from the currently available data. Compared to the hard first-order transition in the boson model, the fermionic transition at hand certainly does not show a similarly sharp transition. On the other hand, the finite-size trends of Fig. 6.20 do not readily allow for a data collapse expected for a continuous transition [190].

## 6.4. Sign-problematic many-fermion systems

Similar in spirit to the work presented in chapter 5, this section explores the machine learning perspective on the sign problem. In contrast to how ordinary observables have to be treated in a model with sign problem (see chapter 4), a neural network only looks for patterns in characteristic quantities of the system such as the Green's functions, the auxiliary field (DQMC), operator string (SSE) or any other relevant quantity from the configuration space or one that is derived from those. As a disclaimer, it should be mentioned right away that the work presented in the following does not in any way solve the fermion sign problem in the sense that it becomes possible to calculate correlation functions and obtain any desired information about the model. What will be shown, however, are two examples of models with a very severe sign problem that realizes distinct patterns in their respective equal times Green's function that can be learnt by a neural network and used to detect the critical value of the interaction for the transition. With this information, it even becomes possible to establish a concept called transfer learning, where a neural network trained on one model can be used to detect the transition in another model that is structurally similar which potentially opens the way to a multi-purpose phase classification system. The issue of how representative these results are will be discussed at the end of the section.

The system of choice is again the spinless Hubbard model on the honeycomb lattice:

$$H = -t \sum_{\langle i,j \rangle} c_i^\dagger c_j + V \sum_{\langle i,j \rangle} n_i n_j. \tag{6.17}$$

As was done in the entanglement chapter, the model is studied in its sign-problematic complex fermion basis rather than the sign-problem free Majorana formulation.

### 6.4.1. Identifying phase boundaries

The network was set up with the same choice for the number of hyperparameters and trained with the same procedure as in the previous examples. The training set consisted of representative samples of the Green's function for interaction strengths deep within the two phases. For each of the two reference points $V = 0.1$ (semi-metal) and $V = 2.5$ (CDW), 8192 (4096 for $L = 15$) samples were generated from DQMC simulations using the modified statistical ensemble of absolute weights $|W_C|$. Thus optimized, unlabeled configurations from several different interaction values $0.1 < V < 2.5$ were fed to the CNN which was then asked to predict to which phase a particular configuration belongs.

Unlike for regular observables, there is no obvious way how to include the sign in the learning procedure if desired. One can envision many different ways such as including a separate pixel, a separate layer (i.e. image dimension) training multiple networks for different values of the sign or multiplying the entire sample by its sign. A separate pixel would likely be deemed irrelevant by the training procedure, considering that there are $(2 \cdot L^2)^2$ other fluctuating quantities. Training multiple networks is not possible for a model with a phase problem where the phases are numerically different at each step along the Markov chain and thus do not allow for any kind of grouping. Including a separate layer and multiplying each sample with its respective phase are similar in nature with the latter option being the less computationally intensive which is why it was chosen for this study. In addition, this method is also the closest in spirit to the measurement prescription for regular observables that include the average over the sign weighted observable $\langle \sigma O \rangle$.

Following the above considerations, two experiments were conducted. Once, the information about the sign was provided by multiplying each Green's matrix $G_s(i, j)$ with the sign associated with the underlying configuration. In a second run, the sign was ignored altogether and the "bare" Green's function was presented as is. It turns out that, as illustrated in the inset of Fig. 6.21, the prediction of the location of the phase transition is not notably improved by multiplying the Green's functions with its phase. While the prediction function moves slightly in parameter space, it also acquires a much broader spread as estimated from averaging over 12 epochs. In fact, multiplying the Green's matrices by a completely random phase is found to give a similar shift and broadening of the spread. Considering the data for different system sizes in Fig. 6.21 one can determine a quantitative estimate of the location of the fermionic phase transition, which is in very good

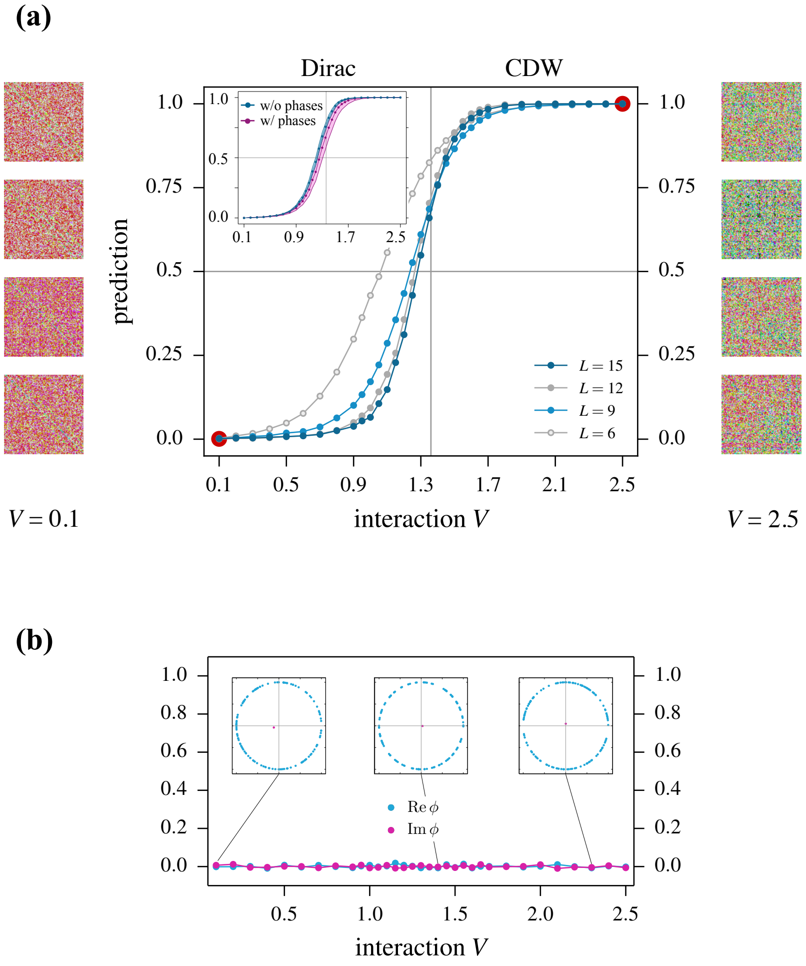
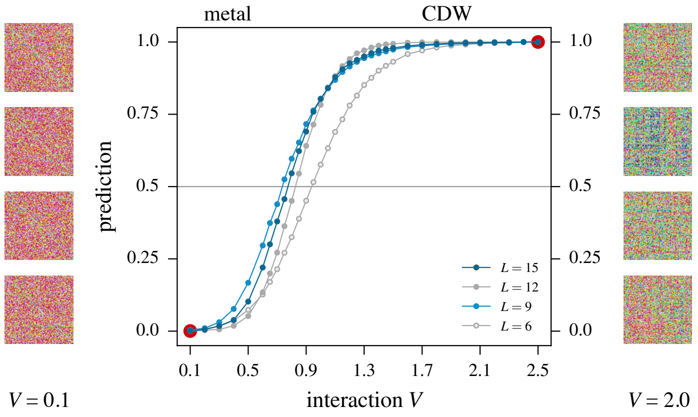


**Figure 6.21.:** (a) Prediction of a CNN for the phase transition from a Dirac semi-metal to a charge density wave (CDW) ordered state in the half-filled spinless fermion Hubbard model (6.17) on the honeycomb lattice of size $2 \cdot L \times L$. The CNN has been trained on 8192 representative samples of the bare Green's function deep inside the two phases (indicated by the red dots). The images in the left and right columns are color-converted instances of the Green's function used in the training. The inset shows a comparison of the prediction for the $L = 9$ system when feeding the CNN with the bare Green's function or the Green's function multiplied by the relative sign / complex phase associated with each configuration (of a given Markov chain). (b) The averaged real and imaginary part of the weight's phase $\phi$, Re($\phi$) and Im($\phi$), respectively, is shown in the main part of the figure for $L = 6$. The three insets show the distribution of the phase for a sequence of 128 measurements, with their average depicted by the pink dot. Figure first appeared in [152].

**Figure 6.22.:** CNN-based identification of the phase transition in the one-third filled, spinless Hubbard model (6.17) on the honeycomb lattice with nearest-neighbor repulsion $V$. The side panels show representative samples of the Green's functions at the two reference points $V = 0.1$ and $V = 2.5$. The network finds two clearly separated phases of which the weak coupling phase is known to be metallic. Figure first appeared in [152].



agreement with the Monte Carlo results [191]. This convincingly demonstrates that the CNN is capable of providing a high-quality prediction discriminating the two fermionic phases, even when the sign content of the configurations is ignored. Importantly, using bare Green's matrices can provide a significant gain in computational efficiency over including information about the relative sign of individual configurations because multiple parallel Markov chains can be sampled. In light of the results of Fig. 6.21 (inset) which show no systematic improvement of the predictive capabilities of the neural network when given additional information about the sign structure, all data presented below was obtained by ignoring the sign in the simulation.

The next experiment was conducted using the same spinless fermion system of Eq. (6.17) but at one-third filling. Going below half-filling turns the itinerant phase for small coupling $V$ into a conventional metal with a nodal Fermi line, while for large $V$ some sort of CDW-ordered Mott insulating state is still expected. In contrast to half-filling, the one-third-filled system has no known sign-free (Majorana) basis for this parameter regime. Despite a very strong sign problem, the supervised machine learning approach is nevertheless capable of finding a prediction function that can properly distinguish two phases . This procedure indicates the existence of a phase transition around $V_c \approx 0.7 \pm 0.1$ as illustrated in Fig. 6.22, which matches a recent estimate from the earlier entanglement calculations [55]. The precise nature of the Mott insulating phase at large $V$ has so far remained elusive, which unfortunately is not altered by the supervised learning approach employed in the current study.

## 6.4.2. Transfer learning

Finally, the possibility of "transfer learning" is explored, which means that a neural network is trained on one model but then used to discriminate phases for an entirely different Hamiltonian without any further training. This approach was highly successful for neural networks trained with classical Ising configurations in Ref. [150]. In the context of this work, the CNN was trained to discriminate the fermionic phases of the sign-problem free, spinful fermion model (6.11) and then applied to predict phases for the sign-problematic, spinless fermion model (6.17). This procedure seems justified based on the fact that at half-filling the two models exhibit similar physics, with the potential energy driving a Gross-Neveu type phase transition from a Dirac semi-metal to a SDW/CDW charge-

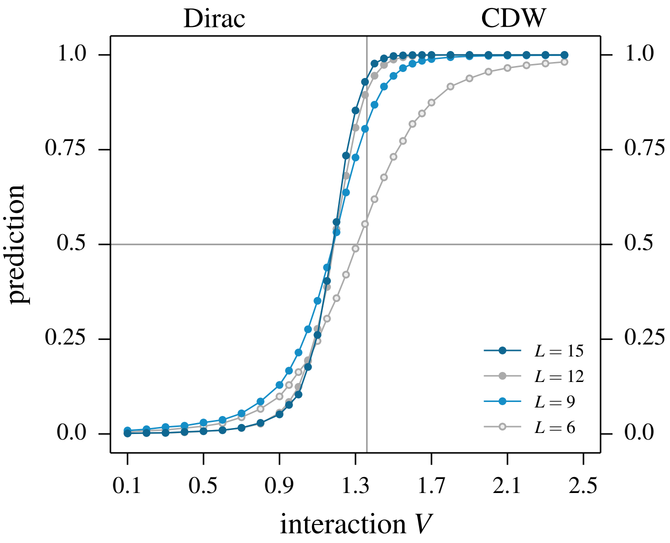


**Figure 6.23.:** An example of transfer learning in an artificial neural network. A CNN that was trained to discriminate the phases of the sign-problem free, *spinful* Hubbard model (6.11) and then applied to identify the phases and phase transition of the sign-problematic, *spinless* Hubbard model (6.17). The network is found to reliably distinguish the fermionic phases of the spinless model and provides a relatively accurate estimate for the location of the phase transition (the vertical line indicates the location of the transition in the thermodynamic limit of infinite system size). Figure first appeared in [152].

ordered phase, respectively. Results for the predictions of the averaged state function are illustrated in Fig. 6.23, which shows that the CNN is capable of reliably distinguishing the fermionic phases of the spinless model, even producing a rough estimate for the location of the phase transition. It is thus concluded that this approach indeed allows for a certain level of transfer learning between sign-problem free and sign-problematic Hamiltonians, suggesting a fruitful area of future study on the relationship between supervised machine learning and the sign problem.

### 6.4.3. On the validity of the machine learning approach to sign-problematic models

Following these promising results, the question remains how to interpret them and how representative they are. To reiterate, the motivation for conducting these experiments is that in contrast to the analysis of correlation functions, the sign does not enter explicitly and there is, in fact, not even an obvious way of including it at all. The only relevant information for the neural network is which patterns appear in the input and how these patterns are related to each other in the respective input images. If there are patterns that are unique to one phase, then these will be chosen as one of the distinguishing features which in turn can be used to detect the location of the phase transition.

The probably most pressing issue that can be raised about this procedure is the question of whether the phase transitions obtained from the modified ensemble are the same ones that occur in the real system. It seems that the answer to that question is again dependent on the particular Monte Carlo flavor, just like it was in the case of the entanglement entropies. The best example to discuss this is the model of spinless fermions at half filling. The SSE formulation for this problem is exactly the same as the one for a hardcore boson model at half filling. This bosonic version has a phase transition from a superfluid to an insulator at $V/t = 2.0$, so at much larger coupling strengths, that is accompanied by characteristic changes in the off-diagonal and diagonal correlation functions [192]. For this model, all of the information about the fermionic phase transition at around $V/t = 1.35$ is contained in the fermion sign, which is calculated from the windings of the hardcore bosons not around the simulation cell but around each other. One thus has to find a way of feeding this information to the network on top in a way that would expose some structure in the data so that a neural network would be capable of learning some characteristics. It is obvious that the unmodified data from the auxiliary ensemble in SSE is not sufficient to learn anything about the fermionic problem.

In stark contrast, the DQMC flavor is capable of coming up with the correct location of the phase transition within the margin of error set by the finite system sizes. Most likely this is due to the fact that the DQMC approach does not simply replace fermions by bosons (or antiferromagnetic by ferromagnetic interactions) when a sign problem appears. There is, in fact, a straightforward way of building up confidence that the DQMC is less affected by the sign problem: At least for Hubbard type models, it is possible to use many different Hubbard-Stratonovich transformations which lead to different sign problems and thus to different auxiliary ensembles. In addition, the Green's functions that are generated during the simulations are different (only expectation values after the averaging process have to be equal). Because each transformation creates its own unique auxiliary ensemble they can each exhibit phase transitions for certain coupling strengths or not. If each ensemble does, in fact, have the same critical interaction strength, then this shows that all of the auxiliary ensembles that can be simulated via the original model share a common feature from which one could conjecture that it is also present in their parent model.

Along with the attempt to come up with a way of encoding the sign structure in worldline methods, it would certainly be worthwhile to set up simulations for models whose physics is known but that suffer from a sign problem in DQMC using different Hubbard-Stratonovich transformation and checking whether they all realize the same phase diagrams in their respective auxiliary ensembles.

## 6.5. Conclusion and outlook

The work presented in this chapter played a key role in establishing that machine learning techniques are capable of learning certain distinct properties of (quantum) phases of matter from raw Monte Carlo data and that this information can be used to pinpoint the location of phase transitions. In some cases, it was shown that it is even possible to extract some information about critical properties as well. Applying this method to an unknown model offers an easy way of getting a first glimpse at what parameter regimes realize distinct phases of matter. This already is an advantage over traditional methods where one would have to test certain order parameters or apply other methods capable of finding phases that are topologically ordered and thus not identifiable with the help of a simple order parameter. In a second step, this information could be used to single out certain interesting values of the model's parameters for a more in-depth Monte Carlo analysis or for studies with equally sophisticated techniques such as exact diagonalization or tensor network algorithms that would be costly to apply to the entire parameter space.

There are two distinct types of models that this method could be useful for. One is the class of designer models, an example of which was actually studied in section 6.3.3. In this and other similar models, one considers a sign-problem free model of fermions which are subsequently coupled to bosonic degrees of freedom giving rise to many different interesting phases without introducing a sign problem. Because of the variety of possible couplings and the resulting variety phases that these models can realize, it is advantageous to have a technique at hand that allows getting a quick overview of what is happening. Applying the transfer learning approach that was pioneered in this work will not only allow to identify where different phases are realized but also of what type theses phases are. The other direction to pursue is to study whether the sign problem can be associated with some kind of structure that a neural network would be able to learn order to distinguish phases based on how the sign problem manifests itself.

Aside from the image recognition inspired approach to applying machine learning techniques in condensed matter, there are two more major research directions that show promising results. The representation of wave functions in terms of tensor network states and their variational optimization towards the physical target state has been an extremely successful approach to the many-body problem, in particular in one dimension where under certain conditions it is possible to find an exact representation of a quantum state as a tensor network state. Not all models are suited for such a procedure though because either the optimization takes too long or because it is even too difficult to setup an appropriate wave function to optimize in the first place. Machine learning techniques provide an alternative way worthy of further investigation to write down these variational wave functions that are potentially as powerful as their tensor network counterparts all while possibly less resource intensive and easier to optimize.

In connection with Monte Carlo, one very interesting application of image *generating* machine learning techniques is the construction of new update mechanisms that provide highly effective global updates, effectively eliminating any autocorrelation between samples and thus allowing for a more massive parallelization of the simulation that allows accessing significantly larger lattices. The idea relies on machine learning algorithms such as Restricted Boltzmann Machines (RBM) or Generative Adversarial Networks that are capable of generating new images, sometimes photo-realistic, after learning key properties of existing ones. Just like one find distinguishing features in Monte Carlo configurations for different values of a model's parameters that can be exploited to separate phases, one can also generate new Monte Carlo configurations with a sufficiently trained machine learning algorithm which can be proposed as updates in the Markov Chain. The advantages of having access to such an essentially autocorrelation free algorithm are manifold: When looking to determine critical properties of models, one often runs into strong finite-size effects due to the small accessible lattice sizes. Another type of analysis that would benefit from such an update is the calculation of entanglement entropies which one may use to find out whether a model realizes a topological phase or not. In chapter 5, it was shown that such simulations entail additional numerical instabilities that can be controlled but are nevertheless challenging and costly. With an efficient global update that eliminates the need for local updates, most of these difficulties would disappear and quantities like the topological entanglement entropy would be accessible without any problems. The possibility of efficiently proposing global updates has already been demonstrated conceptually, but what is still missing is the conception of a general approach for other models than the few selected studied thus far.

# 7. Conclusion and outlook

This thesis contributed two novel computational approaches to the toolbox of quantum Monte Carlo techniques to study the many-fermion problem and the related fermion sign problem. It was demonstrated how to calculate entanglement entropies of fermionic systems using DQMC in a stable way and how to harness the power of artificial neural networks to identify quantum phases and the transitions between them.

While very successful when it can be applied, the natural limit for quantum Monte Carlo approaches remains the fermion sign problem. It was, however, demonstrated that the sign problem is not created equal in all Monte Carlo flavors. DQMC in particular allowed to extract some exact results in the limit of finite-size effects. Much more work on the various aspects of the fermion sign problem is needed, i.e. which exact solutions exist, how certain interesting phenomena can be implemented in effective models, and how auxiliary ensembles without the sign taken into account relate to the actual physical ones. In a direct extension to this thesis, it would be interesting to see whether the apparent greater resiliency of the DQMC algorithm in both the entanglement and the machine learning approach is supported by models other than the simple spinless fermion model or whether this model is just a particularly easy example. One possible way to explore this further is to include a second interaction value on the next-nearest neighbor bonds. The ground state phase diagram of this model has recently been studied using state of the art exact diagonalization techniques thus providing a perfect testing ground for this question. A second well-suited model system are spinless fermions on a checkerboard lattice with complex hopping amplitudes which are believed to realize a fractional Chern insulator phase for certain parts of its parameter regime. For these systems as well, exact diagonalization results amenable for comparison exist for relatively large lattices.

Chronologically, the work on machine learning followed the implementations of the entanglement entropies. One of the most interesting applications of machine learning techniques would actually be to revisit the calculation of entanglement entropies in the context of novel global updates. If it was possible to implement such an update efficiently, it would immediately make many of the numerically costly stabilization procedures presented in this thesis unnecessary and at the same time allow reaching much larger system sizes. This is of particular importance because the central insights are gained from the scaling behavior of the entanglement entropy, the boundary law, which is best explored over a wide range of system sizes.

Exploring the relation of machine learning and numerical techniques is a very interesting endeavor, offering many exciting opportunities for future research. Certainly, the identification and characterization of phases is a very useful tool because it can be easily applied without much additional work on the Monte Carlo side. It will be able to show its strength when it comes to exploring phase diagrams of novel, sign problem free designer Hamiltonians that will then be easily compared to existing, known phases with the help of transfer learning. As was already mentioned earlier, a very promising idea with huge potential impact is to use machine learning techniques to devise global update schemes that could significantly push the existing limitations on system sizes. Outside of the realm of quantum Monte Carlo is the idea to represent and optimize wave function in neural networks complementing efforts from the matrix product states community. While first results have been promising, there much remains to be learnt about how general this approach really is and whether the problems that MPS based methods face are simply due to the underlying structure in which the problem is represented or whether the problem is intrinsically difficult to solve.

Without doubt, numerical techniques to study the many-fermion problem continue to evolve and to generate new insights into a decade old problem. Many of its mysteries still have not been solved and it continues providing new phenomena to explore. Both the entanglement approach and the machine learning approach provide fresh perspectives on the many-body problem that at least in the case of entanglement based measurements have already proven to be crucial to studying novel phenomena which would not have been possible with more traditional techniques. This is an exciting initial position for future studies that will hopefully help to illuminate new aspects of the beauty of the many-body problem.

# A. Determinant Quantum Monte Carlo

## A.1. Slater determinant calculus

An integral part of the DQMC algorithm are Slater determinants representing the fermionic states. This section discusses some elementary properties that are necessary to understand how the DQMC algorithm works.

### A.1.1. Representation as matrices

The starting point is a set of $N_s$ creation operators $\{\gamma_1, \dots \gamma_{N_s}\}$ which might for example derive from the diagonalization of a two particle Hamiltonian given in position space

$$H = \sum_{i,j} c_i^\dagger h_{ij} c_j \,.$$

These two sets of operators are related by

$$\gamma_\alpha = \sum_i U_{\alpha i}^\dagger c_i, \quad \gamma_\alpha^\dagger = \sum_i c_i^\dagger U_{i\alpha} \,.$$

Any $N_p$ particle state in the eigenbasis can be written as

$$\gamma_{s_1}^\dagger \gamma_{s_2}^\dagger \cdots \gamma_{s_{N_p}}^\dagger \left|0\right\rangle = \prod_{n=1}^{N_p} \left( c_i^\dagger U_{i,s_n} \right) \left|0\right\rangle = \prod_{n=1}^{N_p} \left( \mathbf{c}^\dagger \cdot \mathbf{P} \right)_{s_n} \left|0\right\rangle \,. \tag{A.1}$$

The transformations of the operators are contained in the matrix $\mathbf{P}$ of dimension $N_s \times N_p$, which completely characterizes the Slater determinant.

### A.1.2. Properties

#### How does an exponential of a quadratic operator act on a state vector?

The first important property is the action of the exponential of a Hamiltonian, bilinear in creation and annihilation operators, on the Slater determinant, i.e. what is the result of

$$e^{\mathbf{c}^\dagger \mathbf{H}' \mathbf{c}} \prod_{n=1}^{N_p} \left( \mathbf{c}^\dagger \cdot \mathbf{P} \right)_{s_n} \left|0\right\rangle \,. \tag{A.2}$$

By applying a unitary transformation $U$ on the matrix $\mathsf{H}'$ the Hamiltonian is diagonalized $\lambda' = \mathsf{U}^\dagger \mathsf{H}\mathsf{U}$. Using the same transformation on the physical states results in

$$\prod_{n=1}^{N_p}\left(\mathbf{c}^\dagger\cdot\mathbf{P}\right)_{s_n}|0\rangle \to \prod_{n=1}^{N_p}\left(\gamma^\dagger\cdot\mathbf{UP}\right)_{s_n}|0\rangle\,. \tag{A.3}$$

Thus, applying the exponential of the Hamiltonian to the state gives

$$\begin{aligned} e^{\mathbf{c}^\dagger\mathsf{H}'\mathbf{c}}\prod_{n=1}^{N_p}\left(\mathbf{c}^\dagger\cdot\mathbf{P}\right)_{s_n}|0\rangle &= e^{\gamma^\dagger\boldsymbol{\lambda}\gamma}\prod_{n=1}^{N_p}\left(\mathbf{c}^\dagger\cdot\mathbf{UP}\right)_{s_n}|0\rangle \\ &= \sum_{s_1,\dots,s_{N_p}} e^{\gamma^\dagger\boldsymbol{\lambda}\gamma}\gamma^\dagger_{s_1}\cdots\gamma^\dagger_{s_{N_p}}|0\rangle\,(UP)_{s_1,1}\cdots(UP)_{s_{N_p},N_p} \\ &= \sum_{s_1,\dots,s_{N_p}} \gamma^\dagger_{s_1}e^{\lambda_{s_1}}\cdots\gamma^\dagger_{s_{N_p}}e^{\gamma_{s_{N_p}}}|0\rangle\,(UP)_{s_1,1}\cdots(UP)_{s_{N_p},N_p} \\ &=\prod_n\left(\gamma^\dagger e^{\boldsymbol{\lambda}}UP\right)_{s_n}|0\rangle \\ &=\prod_n\left(\boldsymbol{c}^\dagger e^{\mathsf{H}'}P\right)_{s_n}|0\rangle\,. \end{aligned} \tag{A.4}$$

The creation operators in the the diagonalized basis can be commuted through, leaving only the exponential of the eigenvalue. If no particle is created in any of the eigenstates, the exponential acts on the vacuum and evaluates to unity. Ultimately, the vector of creation operators in the site basis is simply exchanged with the exponential of the Hamiltonian matrix. The same argument holds for any product of operators, in particular the exponentials making up the Trotter decomposed product. In conclusion, a Slater determinant remains a Slater determinant after the propagation by the exponentials of quadratic operators.

### What is the overlap of two state vectors?

Applying a product of exponential operators to a Slater determinant returns another Slater determinant. To actually calculate the statistical weight, the inner product of two Slater determinants has to be calculated. To find out how this is done, consider two state vectors with an equal number of particles $N_p$,

$$|\psi\rangle = \prod_{n=1}^{N_p}\left(\boldsymbol{c}^\dagger P\right)_n|0\rangle \tag{A.5}$$

$$|\psi'\rangle = \prod_{n=1}^{N_p}\left(\boldsymbol{c}^\dagger P'\right)_n|0\rangle \tag{A.6}$$

$$\tag{A.7}$$

and calculate their overlap $\langle\psi|\psi'\rangle$:

$$\begin{aligned}\langle\psi|\psi'\rangle &= \langle 0|\prod_{n=N_p}^{1}\prod_{n'=1}^{N_p}\left(P^\dagger\boldsymbol{c}\right)_n\left(\boldsymbol{c}^\dagger P'\right)_{n'}|0\rangle \\ &= \sum_{\substack{s_1,\dots,s_{N_p}\\ s'_1,\dots,s'_{N_p}}} P^\dagger_{N_p,s_p}\cdots P^\dagger_{1,s_1}P'_{1,s'_1}\cdots P'_{N_p,s'_{N_p}}\langle 0|c_{s_{N_p}}\cdots c_{s_{N_1}}c^\dagger_{s'_{N_1}}\cdots c^\dagger_{s'_{N_p}}|0\rangle \end{aligned} \tag{A.8}$$

The important part of the last expression is the expectation value of the creation and annihilation operators

$$\langle 0|\, c_{s_{N_p}} \cdots c_{s_{N_1}} c^{\dagger}_{s'_{N_1}} \cdots c^{\dagger}_{s'_{N_p}} \,|0\rangle \tag{A.9}$$

This expression is only non-zero if the two sets of indices are related by a permutation $\pi$,

$$\{s_1, \dots, s_{N_p}\} = \pi(\{s'_1, \dots, s'_{N_p}\})\,. \tag{A.10}$$

In that case, the expression evaluates to $(-1)^{\pi}$ and thus (A.8) reduces to

$$\begin{aligned}
&\sum_{\substack{s_1,\dots,s_{N_p}\\ \pi\in\mathcal{S}_{N_p}}} (-1)^{\pi} P^{\dagger}_{N_p,s_{N_p}} \cdots P^{\dagger}_{1,s_{N_1}}\; P'_{\pi(s_{N_1}),1} P'_{\pi(s_{N_p}),N_p} \\
&= \det\left(P^{\dagger}P'\right)
\end{aligned} \tag{A.11}$$

This establishes that the overlap of two Slater determinants is the determinant of the product of the matrices characterizing the states.

## Evaluating the grand-canonical trace

It was previously established that

$$e^{\mathbf{c}^{\dagger}\mathbf{H}'\mathbf{c}} \prod_{n=1}^{N_p} \left(\mathbf{c}^{\dagger}\cdot\mathbf{P}\right)_{s_n} |0\rangle = \prod_{n} \left(c^{\dagger} e^{\mathbf{H}'} P\right)_{s_n} |0\rangle$$

holds, even for an arbitrary number of exponentials of quadratic Hamiltonians,

$$\left(\prod_{l} e^{\mathbf{c}^{\dagger}\mathbf{H}^{(l)}\mathbf{c}}\right) \prod_{n=1}^{N_p} \left(\mathbf{c}^{\dagger}\cdot\mathbf{P}\right)_{s_n} |0\rangle = \prod_{n} \left[c^{\dagger}\left(\prod_{l} e^{\mathbf{H}^{(l)}}\right) P\right]_{s_n} |0\rangle\,. \tag{A.12}$$

The product of matrices on the right should now be viewed as originating from an effective, quadratic Hamiltonian, $H_{\text{eff}}$ written in the basis of the site operators. Associated with this effective Hamiltonian is an eigenbasis which can be used for taking a trace:

$$\begin{aligned}
\operatorname{Tr}\left(e^{\mathbf{c}^{\dagger}\mathbf{H}_{\text{eff}}\mathbf{c}}\right) &= \operatorname{Tr}\left(e^{\gamma^{\dagger}\boldsymbol{\lambda}\gamma}\right) \\
&= \operatorname{Tr}\left(\prod_{i} e^{\gamma_i^{\dagger}\lambda_i\gamma_i}\right) \\
&= \left(1+\prod_{i} e^{\lambda_i}\right) \\
&= \det\left(1+e^{\mathbf{H}_{\text{eff}}}\right) \qquad \text{(A.13)} \\
& \qquad \text{(A.14)}
\end{aligned}$$

### A.1.3. Deriving an expression for the Green's function

A central part of the algorithms presented in chapter 2 was a quantity that was referred to as the Green's function. This statement will now be justified by deriving the form of the Green's function and showing that it equals the quantities encountered earlier. A general single body observables is written as $O = \mathbf{c}^\dagger A\mathbf{c}$, where for the single particle Green's function the matrix $A$ is derived from

$$c_x c_y^\dagger = \delta_{x,y} - c_m^\dagger \delta_{m,y}\delta_{n,x} c_n = \delta_{x,y} - c_m^\dagger A_{mn}^{(xy)} c_n\,, \tag{A.15}$$

which is evaluated differently in the projective and the finite temperature formulation. Both cases start from the expression for the evaluation of an observable with the help of introducing the sought after operator as a source, which is later set to zero.

#### Zero temperature formulation

$$\begin{aligned}
\left\langle c_x c_y^\dagger \right\rangle &= \delta_{x,y} - \left\langle c_m^\dagger A_{mn}^{(xy)} c_n \right\rangle \\
&= \delta_{x,y} - \frac{\partial}{\partial\eta} \ln \langle \psi_T | \hat{B}(2\theta,\theta) e^{\eta c_m^\dagger A_{mn}^{(xy)} c_n} \hat{B}(\theta,0) | \psi_T \rangle \Big|_{\eta=0} \\
&= \delta_{x,y} - \frac{\partial}{\partial\eta} \ln\det \left( P^\dagger B(2\theta,\theta) e^{\eta A_{mn}^{(xy)}} B(\theta,0) P \right) \Big|_{\eta=0} \\
&= \delta_{x,y} - \frac{\partial}{\partial\eta} \mathrm{Tr}\ln \left( P^\dagger B(2\theta,\theta) e^{\eta A_{mn}^{(xy)}} B(\theta,0) P \right) \Big|_{\eta=0} \\
&= \delta_{x,y} - \frac{\partial}{\partial\eta} \mathrm{Tr}\ln \left( P^\dagger B(2\theta,\theta) e^{\eta A_{mn}^{(xy)}} B(\theta,0) P \right) \Big|_{\eta=0} \\
&= \delta_{x,y} - \mathrm{Tr}\left[ \left( P^\dagger B(2\theta,0) P \right)^{-1} P^\dagger B(2\theta,\theta) A_{mn}^{(xy)} B(\theta,0) P \right] \\
&= (1 - B(\theta,0) P \left( P^\dagger B(2\theta,0) P \right)^{-1} P^\dagger B(2\theta,\theta))_{x,y}
\end{aligned} \tag{A.16}$$

### Finite temperature formulation

$$\begin{aligned}
\left\langle c_x c_y^\dagger \right\rangle &= \delta_{x,y} - \left\langle c_m^\dagger A_{mn}^{(xy)} c_n \right\rangle \\
&= \delta_{x,y} - \left. \frac{\partial \ln \mathrm{Tr}\left[ B(\beta,\tau) e^{\eta c_m^\dagger A_{mn}^{(xy)} c_n} B(\tau,0) \right]}{\partial \eta} \right|_{\eta=0} \\
&= \delta_{x,y} - \left. \frac{\partial \ln \det\left[ 1 + B(\beta,\tau) e^{\eta A_{mn}^{(xy)}} B(\tau,0) \right]}{\partial \eta} \right|_{\eta=0} \\
&= \delta_{x,y} - \left. \frac{\partial \mathrm{Tr}\, \ln\left[ 1 + B(\beta,\tau) e^{\eta A_{mn}^{(xy)}} B(\tau,0) \right]}{\partial \eta} \right|_{\eta=0} \\
&= \delta_{x,y} - \mathrm{Tr}\left[ B(\tau,0) \left(1 + B(\beta,0)\right)^{-1} B(\beta,\tau) A_{mn}^{(xy)} \right] \\
&= \delta_{x,y} - \left[ \delta_{x,y} - \left(1 + B(\tau,0) B(\beta,\tau)\right)_{x,y} \right] \\
&= \left(1 + B(\tau,0) B(\beta,\tau)\right)_{x,y}
\end{aligned} \tag{A.17}$$

# Bibliography

[1] N. Metropolis, “The beginning of the Monte Carlo method,” *Los Alamos Science*, vol. 15, no. 584, pp. 125–130, 1987.

[2] W. Krauth, *Statistical Mechanics: Algorithms and Computations*. OUP Oxford, Sept. 2006.

[3] H. Fehske, R. Schneider, and A. Weiße, eds., *Computational Many-Particle Physics*, vol. 739 of *Lecture Notes in Physics*. Berlin, Heidelberg: Springer Berlin Heidelberg, 2008.

[4] M. Weigel and W. Janke, “Error estimation and reduction with cross correlations,” *Physical Review E*, vol. 81, June 2010.

[5] J. Gubernatis, N. Kawashima, and P. Werner, *Quantum Monte Carlo Methods: Algorithms for Lattice Models*. Cambridge: Cambridge University Press, 2016.

[6] N. Metropolis, A. W. Rosenbluth, M. N. Rosenbluth, A. H. Teller, and E. Teller, “Equation of State Calculations by Fast Computing Machines,” *The Journal of Chemical Physics*, vol. 21, no. 6, p. 1087, 1953.

[7] W. K. Hastings, “Monte Carlo Sampling Methods Using Markov Chains and Their Applications,” *Biometrika*, vol. 57, no. 1, pp. 97–109, 1970.

[8] U. Wolff, “Critical slowing down,” *Nuclear Physics B - Proceedings Supplements*, vol. 17, pp. 93–102, Sept. 1990.

[9] H. G. Evertz, “The loop algorithm,” *Advances in Physics*, vol. 52, no. 1, pp. 1–66, 2003.

[10] R. Blankenbecler, D. J. Scalapino, and R. L. Sugar, “Monte Carlo calculations of coupled boson-fermion systems. I,” *Physical Review D*, vol. 24, pp. 2278–2286, Oct. 1981.

[11] D. J. Scalapino and R. L. Sugar, “Monte Carlo calculations of coupled boson-fermion systems. II,” *Physical Review B*, vol. 24, no. 8, p. 4295, 1981.

[12] R. Blankenbecler and R. L. Sugar, “Projector Monte Carlo method,” *Physical Review D*, vol. 27, no. 6, p. 1304, 1983.

[13] D. Kung, D. Dahl, R. Blankenbecler, R. Deza, and J. R. Fulco, “New stochastic treatment of fermions with application to a double-chain polymer,” *Physical Review B*, vol. 32, no. 4, p. 2022, 1985.

[14] R. R. dos Santos, "Introduction to Quantum Monte Carlo simulations for fermionic systems," arXiv E-Print cond-mat/0303551, Mar. 2003. Braz. J. Phys. 33, 36 (2003).

[15] F. Assaad and H. Evertz, "World-line and Determinantal Quantum Monte Carlo Methods for Spins, Phonons and Electrons," in *Computational Many-Particle Physics* (H. Fehske, R. Schneider, and A. Weiße, eds.), vol. 739 of *Lecture Notes in Physics*, pp. 277–356, Berlin Heidelberg: Springer, 2008.

[16] J. Hubbard, "Electron correlations in narrow energy bands," *Proc. R. Soc. Lond. A*, vol. 276, pp. 238–257, Nov. 1963.

[17] C. J. Umrigar, J. Toulouse, C. Filippi, S. Sorella, and R. G. Hennig, "Alleviation of the Fermion-sign problem by optimization of many-body wave functions," *Physical Review Letters*, vol. 99, Oct. 2007.

[18] Y. Otsuka, S. Yunoki, and S. Sorella, "Universal Quantum Criticality in the Metal-Insulator Transition of Two-Dimensional Interacting Dirac Electrons," *Physical Review X*, vol. 6, Mar. 2016.

[19] R. L. Stratonovich, "On a Method of Calculating Quantum Distribution Functions," *Soviet Physics Doklady*, vol. 2, p. 416, July 1957.

[20] J. Hubbard, "Calculation of Partition Functions," *Phys. Rev. Lett.*, vol. 3, pp. 77–78, July 1959.

[21] J. E. Hirsch, "Two-dimensional Hubbard model: Numerical simulation study," *Phys. Rev. B*, vol. 31, pp. 4403–4419, Apr. 1985.

[22] F. F. Assaad, "SU(2)-spin Invariant Auxiliary Field Quantum Monte Carlo Algorithm for Hubbard models," arXiv E-Print cond-mat/9806307, June 1998.

[23] F. F. Assaad, M. Imada, and D. J. Scalapino, "Charge and spin structures of a dx2-y2 superconductor in the proximity of an antiferromagnetic Mott insulator," *Phys. Rev. B*, vol. 56, pp. 15001–15014, Dec. 1997.

[24] Z. Bai, C. Lee, R.-C. Li, and S. Xu, "Stable solutions of linear systems involving long chain of matrix multiplications," *Linear Algebra and its Applications*, vol. 435, no. 3, pp. 659 – 673, 2011. Special Issue: Dedication to Pete Stewart on the occasion of his 70th birthday.

[25] P. Broecker and S. Trebst, "Numerical stabilization of entanglement computation in auxiliary field quantum Monte Carlo simulations of interacting many-fermion systems," *Physical Review E*, vol. 94, Dec. 2016.

[26] A. W. Sandvik and J. Kurkijärvi, "Quantum Monte Carlo simulation method for spin systems," *Physical Review B*, vol. 43, no. 7, p. 5950, 1991.

[27] R. G. Melko, "Stochastic series expansion quantum Monte Carlo," in *Strongly Correlated Systems*, pp. 185–206, Springer, 2013.

[28] D. C. Handscomb, “The Monte Carlo method in quantum statistical mechanics,” *Mathematical Proceedings of the Cambridge Philosophical Society*, vol. 58, p. 594, Oct. 1962.

[29] A. W. Sandvik, “Finite-size scaling of the ground-state parameters of the two-dimensional Heisenberg model,” *Physical Review B*, vol. 56, no. 18, p. 11678, 1997.

[30] A. W. Sandvik, “Stochastic series expansion method for quantum Ising models with arbitrary interactions,” *Physical Review E*, vol. 68, Nov. 2003.

[31] S. Wessel, F. Alet, M. Troyer, and G. G. Batrouni, “Quantum Monte Carlo simulations of confined bosonic atoms in optical lattices,” *Physical Review A*, vol. 70, Nov. 2004.

[32] R. G. Melko and A. W. Sandvik, “Stochastic series expansion algorithm for the S=1/2 XY model with four-site ring exchange,” *Physical Review E*, vol. 72, Aug. 2005.

[33] S. Wessel, “Phase Diagram of Interacting Bosons on the Honeycomb Lattice,” *Physical Review B*, vol. 75, May 2007.

[34] R. G. Melko and A. W. Sandvik, “Stochastic series expansion algorithm for the S = 1 / 2 X Y model with four-site ring exchange,” *Physical Review E*, vol. 72, Aug. 2005.

[35] R. G. Melko, “Simulations of quantum XXZ models on two-dimensional frustrated lattices,” *Journal of Physics: Condensed Matter*, vol. 19, no. 14, p. 145203, 2007.

[36] O. F. Syljuåsen and A. W. Sandvik, “Quantum Monte Carlo with directed loops,” *Physical Review E*, vol. 66, Oct. 2002.

[37] O. F. Syljuåsen, “Directed loop updates for quantum lattice models,” *Physical Review E*, vol. 67, Apr. 2003.

[38] F. Alet, S. Wessel, and M. Troyer, “Generalized directed loop method for quantum Monte Carlo simulations,” *Physical Review E*, vol. 71, no. 3, p. 036706, 2005.

[39] A. Dorneich and M. Troyer, “Accessing the dynamics of large many-particle systems using the stochastic series expansion,” *Physical Review E*, vol. 64, no. 6, p. 066701, 2001.

[40] M. Troyer and U.-J. Wiese, “Computational Complexity and Fundamental Limitations to Fermionic Quantum Monte Carlo Simulations,” *Phys. Rev. Lett.*, vol. 94, p. 170201, May 2005.

[41] J. E. Hirsch, R. L. Sugar, D. J. Scalapino, and R. Blankenbecler, “Monte Carlo simulations of one-dimensional fermion systems,” *Phys. Rev. B*, vol. 26, pp. 5033–5055, Nov. 1982.

[42] E. Y. Loh, J. E. Gubernatis, R. T. Scalettar, S. R. White, D. J. Scalapino, and R. L.

Sugar, “Sign problem in the numerical simulation of many-electron systems,” *Phys. Rev. B*, vol. 41, pp. 9301–9307, May 1990.

[43] M. Iazzi, A. A. Soluyanov, and M. Troyer, “Topological Origin of the Fermion Sign Problem,” *Physical Review B*, vol. 93, Mar. 2016.

[44] S. Chandrasekharan and A. Li, “Fermion bag solutions to some sign problems in four-fermion field theories,” *Phys. Rev. D*, vol. 85, p. 091502, May 2012.

[45] S. Chandrasekharan and U.-J. Wiese, “Meron-cluster solution of fermion sign problems,” *Physical Review Letters*, vol. 83, no. 16, p. 3116, 1999.

[46] Z.-X. Li, Y.-F. Jiang, and H. Yao, “Solving fermion sign problem in quantum Monte Carlo by Majorana representation,” *arXiv:1408.2269*, Aug. 2014.

[47] Z.-X. Li, Y.-F. Jiang, and H. Yao, “Fermion-sign-free Majarana-quantum-Monte-Carlo studies of quantum critical phenomena of Dirac fermions in two dimensions,” *New Journal of Physics*, vol. 17, no. 8, p. 085003, 2015.

[48] Z.-C. Wei, “Semigroup Approach to the Sign Problem in Quantum Monte Carlo Simultions,” *arXiv preprint arXiv:1712.09412*, 2017.

[49] E. Berg, M. A. Metlitski, and S. Sachdev, “Sign-Problem–Free Quantum Monte Carlo of the Onset of Antiferromagnetism in Metals,” *Science*, vol. 338, pp. 1606–1609, Dec. 2012.

[50] Y. Tang and A. W. Sandvik, “Confinement and Deconfinement of Spinons in Two Dimensions,” *Physical Review Letters*, vol. 110, May 2013.

[51] F. F. Assaad and T. Grover, “Simple Fermionic Model of Deconfined Phases and Phase Transitions,” *Physical Review X*, vol. 6, Dec. 2016.

[52] X. Y. Xu, K. Sun, Y. Schattner, E. Berg, and Z. Y. Meng, “Non-Fermi Liquid at ( 2 + 1 ) D Ferromagnetic Quantum Critical Point,” *Physical Review X*, vol. 7, Sept. 2017.

[53] A. Einstein, B. Podolsky, and N. Rosen, “Can Quantum-Mechanical Description of Physical Reality Be Considered Complete?,” *Phys. Rev.*, vol. 47, p. 777, May 1935.

[54] P. Broecker and S. Trebst, “Rényi entropies of interacting fermions from determinantal quantum Monte Carlo simulations,” *Journal of Statistical Mechanics: Theory and Experiment*, vol. 2014, p. P08015, Aug. 2014.

[55] P. Broecker and S. Trebst, “Entanglement and the fermion sign problem in auxiliary field quantum Monte Carlo simulations,” *Phys. Rev. B*, vol. 94, p. 075144, Aug. 2016.

[56] J. Bell, “On the Einstein-Podolsky-Rosen paradox,” *Physics*, vol. 1, pp. 195–200, 1964.

[57] J. F. Clauser, M. A. Horne, A. Shimony, and R. A. Holt, “Proposed Experiment to Test Local Hidden-Variable Theories,” *Phys. Rev. Lett.*, vol. 23, pp. 880–884, Oct. 1969.

[58] S. J. Freedman and J. F. Clauser, “Experimental Test of Local Hidden-Variable Theories,” *Phys. Rev. Lett.*, vol. 28, pp. 938–941, Apr. 1972.

[59] A. Aspect, P. Grangier, and G. Roger, “Experimental Realization of Einstein-Podolsky-Rosen-Bohm Gedankenexperiment: A New Violation of Bell’s Inequalities,” *Phys. Rev. Lett.*, vol. 49, pp. 91–94, July 1982.

[60] J.-W. Pan, D. Bouwmeester, M. Daniell, H. Weinfurter, and A. Zeilinger, “Experimental test of quantum nonlocality in three-photon Greenberger–Horne–Zeilinger entanglement,” *Nature*, vol. 403, pp. 515–519, Feb. 2000.

[61] M. A. Rowe, D. Kielpinski, V. Meyer, C. A. Sackett, W. M. Itano, C. Monroe, and D. J. Wineland, “Experimental violation of a Bell’s inequality with efficient detection,” *Nature*, vol. 409, pp. 791–794, Feb. 2001.

[62] M. Ansmann, H. Wang, R. C. Bialczak, M. Hofheinz, E. Lucero, M. Neeley, A. D. O’Connell, D. Sank, M. Weides, J. Wenner, A. N. Cleland, and J. M. Martinis, “Violation of Bell’s inequality in Josephson phase qubits,” *Nature*, vol. 461, pp. 504–506, Sept. 2009.

[63] J.-A. Larsson, M. Giustina, J. Kofler, B. Wittmann, R. Ursin, and S. Ramelow, “Bell-inequality violation with entangled photons, free of the coincidence-time loophole,” *Phys. Rev. A*, vol. 90, p. 032107, Sept. 2014.

[64] M. Giustina, A. Mech, S. Ramelow, B. Wittmann, J. Kofler, J. Beyer, A. Lita, B. Calkins, T. Gerrits, S. W. Nam, R. Ursin, and A. Zeilinger, “Bell violation using entangled photons without the fair-sampling assumption,” *Nature*, vol. 497, p. 227, Apr. 2013.

[65] C. E. Shannon, “A Mathematical Theory of Communication,” *Bell System Technical Journal*, vol. 27, no. 3, pp. 379–423, 1948.

[66] J. von Neumann, “Mathematische Begründung der Quantenmechanik,” *Nachrichten von der Gesellschaft der Wissenschaften zu Göttingen, Mathematisch-Physikalische Klasse*, vol. 1927, pp. 1–57, 1927.

[67] A. Wehrl, “General properties of entropy,” *Rev. Mod. Phys.*, vol. 50, pp. 221–260, Apr. 1978.

[68] M. M. Wolf, F. Verstraete, M. B. Hastings, and J. I. Cirac, “Area Laws in Quantum Systems: Mutual Information and Correlations,” *Phys. Rev. Lett.*, vol. 100, p. 070502, Feb. 2008.

[69] X. G. Wen, “Topological orders in rigid states,” *Int. J. Mod. Phys. B*, vol. 04, pp. 239–271, Feb. 1990.

[70] A. Kitaev and J. Preskill, “Topological Entanglement Entropy,” *Phys. Rev. Lett.*,

vol. 96, p. 110404, Mar. 2006.

[71] M. Levin and X.-G. Wen, “Detecting Topological Order in a Ground State Wave Function,” *Phys. Rev. Lett.*, vol. 96, p. 110405, Mar. 2006.

[72] U. Schollwoeck, “The density-matrix renormalization group,” *Reviews of Modern Physics*, vol. 77, pp. 259–315, Apr. 2005.

[73] G. Vidal, “Entanglement Renormalization: An introduction,” *arXiv:0912.1651 [cond-mat]*, Dec. 2009.

[74] U. Schollwoeck, “The density-matrix renormalization group in the age of matrix product states,” *Annals of Physics*, vol. 326, pp. 96–192, Jan. 2011.

[75] J. C. Bridgeman and C. T. Chubb, “Hand-waving and Interpretive Dance: An Introductory Course on Tensor Networks,” *Journal of Physics A: Mathematical and Theoretical*, vol. 50, p. 223001, June 2017.

[76] H. Li and F. D. M. Haldane, “Entanglement Spectrum as a Generalization of Entanglement Entropy: Identification of Topological Order in Non-Abelian Fractional Quantum Hall Effect States,” *Phys. Rev. Lett.*, vol. 101, p. 010504, July 2008.

[77] H. F. Song, S. Rachel, C. Flindt, I. Klich, N. Laflorencie, and K. L. Hur, “Bipartite Fluctuations as a Probe of Many-Body Entanglement,” *Physical Review B*, vol. 85, Jan. 2012.

[78] C.-M. Chung, L. Bonnes, P. Chen, and A. M. Läuchli, “Entanglement Spectroscopy using Quantum Monte Carlo,” arXiv E-Print 1305.6536, May 2013.

[79] F. F. Assaad, T. C. Lang, and F. P. Toldin, “Entanglement Spectra of Interacting Fermions in Quantum Monte Carlo Simulations,” *Physical Review B*, vol. 89, Mar. 2014.

[80] F. F. Assaad, “Stable quantum Monte Carlo simulations for entanglement spectra of interacting fermions,” *Phys. Rev. B*, vol. 91, p. 125146, Mar. 2015.

[81] A. Rényi, “On measures of entropy and information,” tech. rep., HUNGARIAN ACADEMY OF SCIENCES Budapest Hungary, 1961.

[82] J. Eisert, M. Cramer, and M. B. Plenio, “Area laws for the entanglement entropy,” *Rev. Mod. Phys.*, vol. 82, pp. 277–306, Feb. 2010.

[83] H.-C. Jiang, H. Yao, and L. Balents, “Spin liquid ground state of the spin-$\frac{1}{2}$ square ${J}_{1}$-${J}_{2}$ Heisenberg model,” *Phys. Rev. B*, vol. 86, p. 024424, July 2012.

[84] H.-C. Jiang, Z. Wang, and L. Balents, “Identifying topological order by entanglement entropy,” *Nature Physics*, vol. 8, p. 902, Dec. 2012.

[85] S. Depenbrock, I. P. McCulloch, and U. Schollwöck, “Nature of the Spin-Liquid Ground State of the $S=1/2$ Heisenberg Model on the Kagome Lattice,” *Phys.*

*Rev. Lett.*, vol. 109, p. 067201, Aug. 2012.

[86] M. A. Metlitski and T. Grover, "Entanglement Entropy of Systems with Spontaneously Broken Continuous Symmetry," *arXiv:1112.5166 [cond-mat, physics:hep-th, physics:quant-ph]*, Dec. 2011.

[87] E. M. Stoudenmire, P. Gustainis, R. Johal, S. Wessel, and R. G. Melko, "Corner contribution to the entanglement entropy of strongly-interacting O(2) quantum critical systems in 2+1 dimensions," *Physical Review B*, vol. 90, Dec. 2014.

[88] B. Swingle, J. McMinis, and N. M. Tubman, "Oscillating terms in the Renyi entropy of Fermi liquids," arXiv E-Print 1211.0006, Oct. 2012.

[89] B. Swingle, J. McMinis, and N. M. Tubman, "Oscillating terms in the Renyi entropy of Fermi gases and liquids," *Phys. Rev. B*, vol. 87, p. 235112, June 2013.

[90] P. Calabrese and J. Cardy, "Entanglement entropy and conformal field theory," *J. Phys. A: Math. Theor.*, vol. 42, p. 504005, Dec. 2009.

[91] M. B. Hastings, I. González, A. B. Kallin, and R. G. Melko, "Measuring Renyi Entanglement Entropy in Quantum Monte Carlo Simulations," *Phys. Rev. Lett.*, vol. 104, p. 157201, Apr. 2010.

[92] R. G. Melko, A. B. Kallin, and M. B. Hastings, "Finite Size Scaling of Mutual Information: A Scalable Simulation," arXiv E-Print 1007.2182, July 2010. Phys. Rev. B 82, 100409(R) (2010).

[93] J. McMinis and N. M. Tubman, "Renyi Entropy of the Interacting Fermi Liquid," *Physical Review B*, vol. 87, Feb. 2013.

[94] M.-C. Chung and I. Peschel, "Density-matrix spectra for two-dimensional quantum systems," *Physical Review B*, vol. 62, no. 7, p. 4191, 2000.

[95] M.-C. Chung and I. Peschel, "Density-matrix spectra of solvable fermionic systems," *Physical Review B*, vol. 64, July 2001.

[96] T. Grover, "Quantum Entanglement of Interacting Fermions in Monte Carlo," arXiv E-Print 1307.1486, July 2013. Phys. Rev. Lett. 111, 130402 (2013).

[97] S. Humeniuk and T. Roscilde, "Quantum Monte Carlo calculation of entanglement Renyi entropies for generic quantum systems," *Physical Review B*, vol. 86, Dec. 2012.

[98] C. Holzhey, F. Larsen, and F. Wilczek, "Geometric and renormalized entropy in conformal field theory," *Nucl. Phys.*, vol. B424, pp. 443–467, 1994.

[99] P. Calabrese and J. Cardy, "Entanglement entropy and quantum field theory," *J. Stat. Mech.*, vol. 2004, p. P06002, June 2004.

[100] T. Grover, "Entanglement of Interacting Fermions in Quantum Monte Carlo Calculations," *Phys. Rev. Lett.*, vol. 111, p. 130402, Sept. 2013.

[101] L. N. Trefethen and D. Bau, *Numerical Linear Algebra*. SIAM, 1997.

[102] R. M. Fye and J. E. Hirsch, “Monte Carlo study of the symmetric Anderson-impurity model,” *Phys. Rev. B*, vol. 38, pp. 433–441, July 1988.

[103] J. W. Demmel, *Applied Numerical Linear Algebra*. Philadelphia, PA, USA: Society for Industrial and Applied Mathematics, 1997.

[104] F. Anfuso and A. Rosch, “String order and adiabatic continuity of Haldane chains and band insulators,” *Phys. Rev. B*, vol. 75, p. 144420, Apr. 2007.

[105] F. H. L. Essler and A. M. Tsvelik, “Weakly coupled one-dimensional Mott insulators,” *Phys. Rev. B*, vol. 65, p. 115117, Mar. 2002.

[106] F. H. L. Essler and A. M. Tsvelik, “Theory of hybrid state in a metal with a small Fermi surface and strong collective excitations,” *Phys. Rev. B*, vol. 71, p. 195116, May 2005.

[107] A. Fuhrmann, D. Heilmann, and H. Monien, “From Mott insulator to band insulator: A dynamical mean-field theory study,” *Phys. Rev. B*, vol. 73, p. 245118, June 2006.

[108] R. M. Konik, T. M. Rice, and A. M. Tsvelik, “Doped Spin Liquid: Luttinger Sum Rule and Low Temperature Order,” *Phys. Rev. Lett.*, vol. 96, p. 086407, Mar. 2006.

[109] C. Berthod, T. Giamarchi, S. Biermann, and A. Georges, “Breakup of the Fermi Surface Near the Mott Transition in Low-Dimensional Systems,” *Phys. Rev. Lett.*, vol. 97, p. 136401, Sept. 2006.

[110] A. Garg, H. R. Krishnamurthy, and M. Randeria, “Can Correlations Drive a Band Insulator Metallic?,” *Phys. Rev. Lett.*, vol. 97, p. 046403, July 2006.

[111] T. D. Stanescu, P. Phillips, and T.-P. Choy, “Theory of the Luttinger surface in doped Mott insulators,” *Phys. Rev. B*, vol. 75, p. 104503, Mar. 2007.

[112] H. v. Löhneysen, A. Rosch, M. Vojta, and P. Wölfle, “Fermi-liquid instabilities at magnetic quantum phase transitions,” *Rev. Mod. Phys.*, vol. 79, pp. 1015–1075, Aug. 2007.

[113] S. S. Kancharla and S. Okamoto, “Band insulator to Mott insulator transition in a bilayer Hubbard model,” *Phys. Rev. B*, vol. 75, p. 193103, May 2007.

[114] N. Paris, K. Bouadim, F. Hebert, G. G. Batrouni, and R. T. Scalettar, “Quantum Monte Carlo Study of an Interaction-Driven Band-Insulator to Metal Transition,” *Phys. Rev. Lett.*, vol. 98, p. 046403, Jan. 2007.

[115] S. S. Kancharla and E. Dagotto, “Correlated Insulated Phase Suggests Bond Order between Band and Mott Insulators in Two Dimensions,” *Phys. Rev. Lett.*, vol. 98, p. 016402, Jan. 2007.

[116] M. Z. Hasan and C. L. Kane, “*Colloquium* : Topological insulators,” *Rev. Mod.*

*Phys.*, vol. 82, pp. 3045–3067, Nov. 2010.

[117] X.-L. Qi and S.-C. Zhang, "Topological insulators and superconductors," *Rev. Mod. Phys.*, vol. 83, pp. 1057–1110, Oct. 2011.

[118] C. L. Kane and E. J. Mele, "Quantum Spin Hall Effect in Graphene," *Phys. Rev. Lett.*, vol. 95, p. 226801, Nov. 2005.

[119] C. L. Kane and E. J. Mele, "Z_2 Topological Order and the Quantum Spin Hall Effect," *Phys. Rev. Lett.*, vol. 95, p. 146802, Sept. 2005.

[120] C. Wang, A. C. Potter, and T. Senthil, "Classification of Interacting Electronic Topological Insulators in Three Dimensions," *Science*, vol. 343, no. 6171, pp. 629–631, 2014.

[121] C.-C. Chang, R. R. P. Singh, and R. T. Scalettar, "Entanglement properties of the antiferromagnetic-singlet transition in the Hubbard model on bilayer square lattices," *Phys. Rev. B*, vol. 90, p. 155113, Oct. 2014.

[122] M. Golor, T. Reckling, L. Classen, M. M. Scherer, and S. Wessel, "Ground-state phase diagram of the half-filled bilayer Hubbard model," *Phys. Rev. B*, vol. 90, p. 195131, Nov. 2014.

[123] Z. Weihong, "Various series expansions for the bilayer S= Heisenberg antiferromagnet," *Phys. Rev. B*, vol. 55, pp. 12267–12275, May 1997.

[124] L. Wang, K. S. D. Beach, and A. W. Sandvik, "High-precision finite-size scaling analysis of the quantum-critical point of S=$1/2$ Heisenberg antiferromagnetic bilayers," *Phys. Rev. B*, vol. 73, p. 014431, Jan. 2006.

[125] S. S. Kancharla and S. Okamoto, "Band insulator to Mott insulator transition in a bilayer Hubbard model," *Phys. Rev. B*, vol. 75, p. 193103, May 2007.

[126] K. Bouadim, G. G. Batrouni, F. Hébert, and R. T. Scalettar, "Magnetic and transport properties of a coupled Hubbard bilayer with electron and hole doping," *Phys. Rev. B*, vol. 77, p. 144527, Apr. 2008.

[127] C. J. Hamer, J. Oitmaa, and Z. Weihong, "Restoration of symmetry in the spectrum of the bilayer Heisenberg antiferromagnet," *Phys. Rev. B*, vol. 85, p. 014432, Jan. 2012.

[128] J. Helmes and S. Wessel, "Entanglement entropy scaling in the bilayer Heisenberg spin system," *Phys. Rev. B*, vol. 89, p. 245120, June 2014.

[129] R. Rüger, L. F. Tocchio, R. Valenti, and C. Gros, "The phase diagram of the square lattice bilayer Hubbard model: A variational Monte Carlo study," *New Journal of Physics*, vol. 16, no. 3, p. 033010, 2014.

[130] J. Eisert, M. Cramer, and M. B. Plenio, "Area laws for the entanglement entropy," *Rev. Mod. Phys.*, vol. 82, pp. 277–306, Feb. 2010.

[131] M. M. Wolf, "Violation of the Entropic Area Law for Fermions," *Phys. Rev. Lett.*, vol. 96, p. 010404, Jan. 2006.

[132] D. Gioev and I. Klich, "Entanglement Entropy of Fermions in Any Dimension and the Widom Conjecture," *Phys. Rev. Lett.*, vol. 96, p. 100503, Mar. 2006.

[133] Y. Zhang, T. Grover, and A. Vishwanath, "Entanglement Entropy of Critical Spin Liquids," *Phys. Rev. Lett.*, vol. 107, p. 067202, Aug. 2011.

[134] D. J. Gross and A. Neveu, "Dynamical symmetry breaking in asymptotically free field theories," *Phys. Rev. D*, vol. 10, pp. 3235–3253, Nov. 1974.

[135] N. Zerf, L. N. Mihaila, P. Marquard, I. F. Herbut, and M. M. Scherer, "Four-loop critical exponents for the Gross-Neveu-Yukawa models," *Physical Review D*, vol. 96, Nov. 2017.

[136] L. Wang, P. Corboz, and M. Troyer, "Fermionic Quantum Critical Point of Spinless Fermions on a Honeycomb Lattice," *arXiv:1407.0029*, June 2014.

[137] J. Motruk, A. G. Grushin, F. de Juan, and F. Pollmann, "Interaction driven phases in the half-filled honeycomb lattice: An infinite density matrix renormalization group study," *arXiv preprint arXiv:1505.01674*, 2015.

[138] S. Capponi and A. M. Läuchli, "Phase diagram of interacting spinless fermions on the honeycomb lattice: A comprehensive exact diagonalization study," *Physical Review B*, vol. 92, p. 085146, Aug. 2015.

[139] D. Gioev and I. Klich, "Entanglement Entropy of Fermions in Any Dimension and the Widom Conjecture," *Physical Review Letters*, vol. 96, p. 100503, Mar. 2006.

[140] M. M. Wolf, "Violation of the Entropic Area Law for Fermions," *Physical Review Letters*, vol. 96, p. 010404, Jan. 2006.

[141] Y. Zhang, T. Grover, and A. Vishwanath, "Entanglement Entropy of Critical Spin Liquids," *Physical Review Letters*, vol. 107, p. 067202, Aug. 2011.

[142] V. Estivill-Castro, "Why So Many Clustering Algorithms: A Position Paper," *SIGKDD Explor. Newsl.*, vol. 4, pp. 65–75, June 2002.

[143] M. Bramer, *Principles of Data Mining*. Springer London, 2016.

[144] T. K. Ho, "Random Decision Forests," in *Proceedings of the Third International Conference on Document Analysis and Recognition (Volume 1) - Volume 1*, ICDAR '95, (Washington, DC, USA), pp. 278–, IEEE Computer Society, 1995.

[145] M. Nielsen, *Neural Networks and Deep Learning*. Online only, 2018.

[146] G. Carleo and M. Troyer, "Solving the quantum many-body problem with artificial neural networks," *Science*, vol. 355, pp. 602–606, Feb. 2017.

[147] G. Torlai and R. G. Melko, "Learning thermodynamics with Boltzmann machines,"

*Physical Review B*, vol. 94, Oct. 2016.

[148] L. Huang and L. Wang, “Accelerated Monte Carlo simulations with restricted Boltzmann machines,” *Physical Review B*, vol. 95, Jan. 2017.

[149] J. Liu, Y. Qi, Z. Y. Meng, and L. Fu, “Self-learning Monte Carlo method,” *Physical Review B*, vol. 95, Jan. 2017.

[150] J. Carrasquilla and R. G. Melko, “Machine learning phases of matter,” *Nature Physics*, vol. 13, pp. 431–434, Feb. 2017.

[151] K. Ch’ng, J. Carrasquilla, R. G. Melko, and E. Khatami, “Machine Learning Phases of Strongly Correlated Fermions,” *arXiv preprint arXiv:1609.02552*, 2016.

[152] P. Broecker, J. Carrasquilla, R. G. Melko, and S. Trebst, “Machine learning quantum phases of matter beyond the fermion sign problem,” *Scientific Reports*, vol. 7, p. 8823, Aug. 2017.

[153] E. P. L. van Nieuwenburg, Y.-H. Liu, and S. D. Huber, “Learning phase transitions by confusion,” *Nature Physics*, vol. 13, p. 435, May 2017.

[154] P. Broecker, F. F. Assaad, and S. Trebst, “Quantum phase recognition via unsupervised machine learning,” *arXiv:1707.00663 [cond-mat]*, July 2017.

[155] F. Rosenblatt, “The perceptron: A probabilistic model for information storage and organization in the brain.,” *Psychological Review*, vol. 65, pp. 386–408, Nov. 1958.

[156] H. T. Siegelmann and E. D. Sontag, “On the Computational Power of Neural Nets,” *Journal of Computer and System Sciences*, vol. 50, pp. 132–150, Feb. 1995.

[157] P. Ramachandran, B. Zoph, and Q. V. Le, “Searching for Activation Functions,” *arXiv:1710.05941 [cs]*, Oct. 2017.

[158] A. Cauchy, “Méthode générale pour la résolution des systèmes d’équations simultanées,” *Comptes Rendus Hebd. Seances Acad. Sci.*, vol. 25, pp. 536–538, 1847.

[159] Y. N. Dauphin, R. Pascanu, C. Gulcehre, K. Cho, S. Ganguli, and Y. Bengio, “Identifying and attacking the saddle point problem in high-dimensional non-convex optimization,” in *Advances in Neural Information Processing Systems*, pp. 2933–2941, 2014.

[160] H. J. Kelley, “Gradient Theory of Optimal Flight Paths,” *ARS Journal*, vol. 30, pp. 947–954, Oct. 1960.

[161] A. E. Bryson, “A gradient method for optimizing multi-stage allocation processes,” in *Proc. Harvard Univ. Symposium on Digital Computers and Their Applications*, 1961.

[162] S. Linnainmaa, “Taylor expansion of the accumulated rounding error,” *BIT Numerical Mathematics*, vol. 16, pp. 146–160, June 1976.

[163] D. E. Rumelhart, G. E. Hinton, and R. J. Williams, "Parallel Distributed Processing: Explorations in the Microstructure of Cognition, Vol. 1," pp. 318–362, Cambridge, MA, USA: MIT Press, 1986.

[164] K. G. Wilson, "Confinement of quarks," *Phys. Rev. D*, vol. 10, pp. 2445–2459, Oct. 1974.

[165] E. Fradkin, "Field Theories of Condensed Matter Physics," Feb. 2013.

[166] E. Ising, "Beitrag zur Theorie des Ferromagnetismus," *Zeitschrift für Physik*, vol. 31, pp. 253–258, Feb. 1925.

[167] K. Ch'ng, J. Carrasquilla, R. G. Melko, and E. Khatami, "Machine Learning Phases of Strongly Correlated Fermions,"

[168] Y. Zhang and E.-A. Kim, "Quantum Loop Topography for Machine Learning," *Phys. Rev. Lett.*, vol. 118, p. 216401, May 2017.

[169] Y. Zhang, R. G. Melko, and E.-A. Kim, "Machine Learning $\mathbb{Z}_2$ Quantum Spin Liquids with Quasi-particle Statistics,"

[170] D. Kingma and J. Ba, "Adam: A Method for Stochastic Optimization,"

[171] M. Abadi itet al., "TensorFlow: Large-Scale Machine Learning on Heterogeneous Distributed Systems,"

[172] J. D. Foley, A. van Dam, S. K. Feiner, and J. F. Hughes, *Computer Graphics: Principles and Practice (2nd Ed.).* Boston, MA, USA: Addison-Wesley Longman Publishing Co., Inc., 1990.

[173] K. Ch'ng, J. Carrasquilla, R. G. Melko, and E. Khatami, "Machine Learning Phases of Strongly Correlated Fermions," *Phys. Rev. X*, vol. 7, p. 031038, Aug. 2017.

[174] Z. Y. Meng, T. C. Lang, S. Wessel, F. F. Assaad, and A. Muramatsu, "Quantum spin liquid emerging in two-dimensional correlated Dirac fermions," *nat*, vol. 464, pp. 847–851, Apr. 2010.

[175] Y. Otsuka, S. Yunoki, and S. Sorella, "Universal Quantum Criticality in the Metal-Insulator Transition of Two-Dimensional Interacting Dirac Electrons," *Phys. Rev. X*, vol. 6, p. 011029, Mar. 2016.

[176] M. P. A. Fisher, P. B. Weichman, G. Grinstein, and D. S. Fisher, "Boson localization and the superfluid-insulator transition," *Phys. Rev. B*, vol. 40, pp. 546–570, July 1989.

[177] S. Wessel and M. Troyer, "Supersolid Hard-Core Bosons on the Triangular Lattice," *Phys. Rev. Lett.*, vol. 95, p. 127205, Sept. 2005.

[178] D. Heidarian and K. Damle, "Persistent Supersolid Phase of Hard-Core Bosons on the Triangular Lattice," *Phys. Rev. Lett.*, vol. 95, p. 127206, Sept. 2005.

[179] R. G. Melko, A. Paramekanti, A. A. Burkov, A. Vishwanath, D. N. Sheng, and L. Balents, "Supersolid Order from Disorder: Hard-Core Bosons on the Triangular Lattice," *Phys. Rev. Lett.*, vol. 95, p. 127207, Sept. 2005.

[180] D. N. Sheng, O. I. Motrunich, S. Trebst, E. Gull, and M. P. A. Fisher, "Strong-coupling phases of frustrated bosons on a two-leg ladder with ring exchange," *Phys. Rev. B*, vol. 78, p. 054520, Aug. 2008.

[181] G. Schmid, S. Todo, M. Troyer, and A. Dorneich, "Finite-Temperature Phase Diagram of Hard-Core Bosons in Two Dimensions," *Phys. Rev. Lett.*, vol. 88, p. 167208, Apr. 2002.

[182] G. G. Batrouni and R. T. Scalettar, "Phase Separation in Supersolids," *Phys. Rev. Lett.*, vol. 84, pp. 1599–1602, Feb. 2000.

[183] A. W. Sandvik, "Stochastic series expansion method with operator-loop update," *Phys. Rev. B*, vol. 59, pp. R14157–R14160, June 1999.

[184] T. Matsubara and H. Matsuda, "A Lattice Model of Liquid Helium, I," *Progress of Theoretical Physics*, vol. 16, no. 6, p. 569, 1956.

[185] F. F. Assaad and T. Grover, "Simple Fermionic Model of Deconfined Phases and Phase Transitions," *Phys. Rev. X*, vol. 6, p. 041049, Dec. 2016.

[186] S. Gazit, M. Randeria, and A. Vishwanath, "Emergent Dirac fermions and broken symmetries in confined and deconfined phases of Z2 gauge theories," *Nat Phys*, vol. 13, no. 5, pp. 484–490, 2017.

[187] F. F. Assaad emet al. *in preparation*.

[188] J. Gubernatis, N. Kawashima, and P. Werner, *Quantum Monte Carlo Methods: Algorithms for Lattice Models*. Cambridge, UK: Cambridge University Press, 2016.

[189] M. Bercx, F. Goth, J. S. Hofmann, and F. F. Assaad, "The ALF (Algorithms for Lattice Fermions) project release 1.0. Documentation for the auxiliary field quantum Monte Carlo code," *ArXiv:1704.00131*, Apr. 2017.

[190] J. Carrasquilla and R. G. Melko, "Machine learning phases of matter," *Nat Phys*, vol. 13, pp. 431–434, May 2017.

[191] Z.-X. Li, Y.-F. Jiang, and H. Yao, "Fermion-sign-free Majarana-quantum-Monte-Carlo studies of quantum critical phenomena of Dirac fermions in two dimensions," *New Journal of Physics*, vol. 17, no. 8, p. 085003, 2015.

[192] S. Wessel, "Phase diagram of interacting bosons on the honeycomb lattice," *Physical Review B*, vol. 75, May 2007.

# Erklärung zur Dissertation

Ich versichere, dass ich die von mir vorgelegte Dissertation selbständig angefertigt, die benutzten Quellen und Hilfsmittel vollständig angegeben und die Stellen der Arbeit - einschließlich Tabellen, Karten und Abbildungen -, die anderen Werken im Wortlaut oder dem Sinn nach entnommen sind, in jedem Einzelfall als Entlehnung kenntlich gemacht habe; dass diese Dissertation noch keiner anderen Fakultät oder Universität zur Prüfung vorgelegen hat; dass sie - abgesehen von unten angegebenen Teilpublikationen - noch nicht veröffentlicht worden ist sowie, dass ich eine solche Veröffentlichung vor Abschluss des Promotionsverfahrens nicht vornehmen werde. Die Bestimmungen der Promotionsordnung sind mir bekannt. Die von mir vorgelegte Dissertation ist von Prof. Dr. Simon Trebst betreut worden.

# Publikationen

- P. Broecker, F. F. Assaad, and S. Trebst, *Quantum phase recognition via unsupervised machine learning* arXiv:1707.00663 [cond-mat], Jul. 2017.

- P. Broecker, J. Carrasquilla, R. G. Melko, and S. Trebst, *Machine learning quantum phases of matter beyond the fermion sign problem* Scientific Reports, vol. 7, no. 1, p. 8823, Aug. 2017.

- P. Broecker and S. Trebst, *Rényi entropies of interacting fermions from determinantal quantum Monte Carlo simulations*, Journal of Statistical Mechanics: Theory and Experiment, vol. 2014, no. 8, p. P08015, Aug. 2014.

- P. Broecker and S. Trebst, *Entanglement and the fermion sign problem in auxiliary field quantum Monte Carlo simulations*, Phys. Rev. B, vol. 94, no. 7, p. 075144, Aug. 2016.

- P. Broecker and S. Trebst, *Numerical stabilization of entanglement computation in auxiliary field quantum Monte Carlo simulations of interacting many-fermion systems* Physical Review E, vol. 94, no. 6, Dec. 2016.

Gefördert durch

Deutsche
Telekom
**Stiftung**

bcgs

CRC183
ENTANGLED STATES OF MATTER