

Simulating fermionic systems on classical and quantum computing devices

Dissertation
zur Erlangung des Grades
des Doktors der Naturwissenschaften
der Naturwissenschaftlich-Technischen Fakultät
der Universität des Saarlandes

von

Michael Peter Kaicher

Saarbrücken

2020

Tag des Kolloquiums: 17. September 2020

Dekan: Prof. Dr. Guido Kickelbick

Berichterstatter: Prof. Dr. Frank Wilhelm-Mauch
Prof. Dr. Giovanna Morigi

Akad. Mitglied: Dr. Adam Wysocki

Vorsitz: Prof. Dr. Jürgen Eschner

ABSTRACT

This thesis presents a theoretical study of topics related to the simulation of quantum mechanical systems on classical and quantum computers. A large part of this work focuses on strongly interacting fermionic systems, more precisely, the behavior of electrons in presence of strong magnetic fields. We show how the energy spectrum of a Hamiltonian describing the fractional quantum Hall effect can be computed on a quantum computer and derive a closed form for the Hamiltonian coefficients in second quantization. We then discuss a mean-field method and a multi-reference state approach that allow for an efficient classical computation and an efficient initial state preparation on a quantum computer. The second part of the thesis presents a detailed description on how long-range interacting fermionic systems can be simulated on classical computers using a variational method, introduce an Ansatz which could potentially simplify numerical simulations and give an explicit quantum circuit that shows how the variational state can be used as an initial state and how it can be implemented on a quantum computer. In the last part, two novel protocols are presented that generate a variety of prominent many-body operators from two-body interactions and show how these protocols improve over previous construction schemes for a number of important examples.

ZUSAMMENFASSUNG

Diese Arbeit behandelt verschiedene zentrale Probleme theoretischer Natur, welche im Rahmen der Simulation quantenmechanischer Systeme auf klassischen und Quantencomputern auftreten. Ein Großteil dieser Arbeit beschäftigt sich mit stark wechselwirkenden fermionischen Systemen, genauer gesagt, dem Verhalten von Elektronen innerhalb eines starken Magnetfelds. Es wird dargelegt, wie das Energiespektrum des Quanten-Hall-Effekt-Hamiltonoperators auf einem Quantencomputer berechnet werden kann, und es werden geschlossene Ausdrücke für dessen Hamilton-Koeffizienten in zweiter Quantisierung hergeleitet. Anschließend werden sowohl ein Molekularfeld- als auch ein Multi-Referenz-Ansatz diskutiert, welche eine effiziente Berechnung auf klassischen Rechnern zulassen sowie eine effiziente Implementierung auf Quantencomputern ermöglichen. Der zweite Teil dieser Arbeit erläutert, wie man langreichweitige, wechselwirkende fermionische Systeme mit Hilfe einer neuen Variationsmethode, welche über die Molekularfeldnäherung hinaus geht, auf einem klassischen Computer simulieren kann. Es wird darüber hinaus ein alternativer Ansatz vorgestellt, der Teile dieser Variationsmethode vereinfachen könnte, und gezeigt, wie sich der Ansatz auf einem Quantencomputer realisieren lässt. Im letzten Teil werden zwei neue Methoden vorgestellt, welche es ermöglichen, eine Reihe wichtiger Vielteilchen-Operatoren aus Zweiteilchen-Wechselwirkungen zu erzeugen. Beide Methoden werden durch eine Vielzahl an wichtigen Beispielen veranschaulicht.

CONTENTS

1	Introduction	1
	Introduction	1
1.1	Preface	2
1.2	Many-body fermionic systems in second quantization	6
1.3	Fermionic Gaussian states	10
1.3.1	Fermionic linear optics	10
1.3.2	Grassmann numbers	11
1.3.3	Fermionic Gaussian states and their Grassmann representation	16
1.4	Imaginary time evolution of a quantum system	18
1.5	Quantum computation	20
1.5.1	Quantum circuits	22
1.5.2	A fault-tolerant quantum computer	24
1.5.3	Approximating unitary operators	26
1.6	Quantum simulation	28
1.6.1	Quantum Fourier transformation	29
1.6.2	Quantum phase estimation	30
1.6.3	Hamiltonian simulation through linear combination of unitaries	32
1.7	The quantum Hall effect	38
1.7.1	The classical model	38
1.7.2	The discovery of the quantum Hall effect	40
2	Roadmap for quantum simulation of the fractional quantum Hall effect	43
2.1	Introduction and overview	44
2.2	System Hamiltonian	48
2.2.1	The Hamiltonian in first quantization	48
2.2.2	The Hamiltonian in second quantization	49
2.2.3	System Hamiltonian in the LLL	53
2.2.4	Mapping the second-quantized fermionic Hamiltonian to the Pauli basis	54
2.3	Hamiltonian simulation through linear combination of unitaries	57
2.4	Finding an initial state	60
2.4.1	Single-reference state	60
2.4.2	Multi-reference state	63
2.5	Numerical results	66

2.6	Discussion	70
2.6.1	Finite size studies	70
2.6.2	Augmenting the model	70
2.6.3	Using the Laughlin wave function as a sanity check for the variational Ansatz	71
2.6.4	Computing correlation functions	71
2.7	Conclusion and outlook	73
Appendices		75
2.A	Derivation of analytical result for the Coulomb matrix elements	76
2.B	Hamiltonian simulation through linear combination of unitaries using the self-inverse matrix decomposition strategy	82
2.C	Derivations for the equations of motion of the CM	85
2.D	Rotating the system Hamiltonian into the eigenbasis of the CM	86
2.E	Imaginary time evolution - formal integration	87
2.F	Computing correlation functions for single-reference and multi-reference states	89
2.G	Computing the overlaps	91
2.G.1	Computing the diagonal terms	91
2.G.2	Computing the off-diagonal terms	92
2.G.3	Identity map	92
2.G.4	One-body excitations	92
2.G.5	Two-body excitations	93
2.H	Convergence of the ASCI energy	94
3	Efficiently computable approximate ground states for strongly interacting fermionic systems beyond Gaussian states	95
3.1	Motivation and main result	96
3.2	Notations	99
3.3	Emerging expectation values	102
3.3.1	Unitary transformations of fermionic and Majorana operators . . .	103
3.3.2	Deriving the energy functional	104
3.4	Analytical formulas for emerging expectation values	107
3.4.1	Derivation of implicit formula	107
3.4.2	Derivation of explicit expressions	115
3.4.3	General explicit formula	116
3.5	Energy expectation value	118
3.6	Minimizing the energy through imaginary-time evolution	119
3.6.1	Computing the Gram matrix	120
3.6.2	Mean-field matrix of rotated Hamiltonian	122
3.6.3	The role of the chemical potential	123
3.6.4	Mean-field matrix of NGS Ansatz	124
3.6.5	Time evolution of the energy	124
3.7	Minimizing the energy through a hybrid imaginary-time- gradient-descent algorithm	126
3.7.1	Gradient descent	126
3.7.2	The HITGD Ansatz	126
3.7.3	Computing the energy gradient w.r.t. the NGS variational parameters	128

3.8	Efficient implementation of the non-Gaussian state Ansatz on a quantum computer	129
3.9	Summary and outlook	132
Appendices		133
3.A	System Hamiltonian and chemical potential	133
3.B	Wick's theorem	136
3.C	Canonical transformation through flux attachment operator	138
3.D	Deriving the rotated Hamiltonian	139
3.E	Calculation of expectation values	140
3.E.1	Matrix elements of \mathcal{M}	141
3.E.2	Zeroth order	141
3.E.3	Second order	142
3.E.4	Fourth order	144
3.F	Symmetry properties of the fermionic expectation values	146
3.F.1	Expectation values belonging to one-body terms	146
3.F.2	Expectation values belonging to quartic terms	147
3.G	Derivatives of \mathcal{A} and \mathcal{G} w.r.t. the covariance matrix Γ_m and variational parameters ω	148
3.G.1	Computing the gradient of a Pfaffian	148
3.G.2	Derivative of \mathcal{A} with respect to the covariance matrix Γ_m	148
3.G.3	Derivative of \mathcal{G} with respect to the covariance matrix Γ_m	149
3.G.4	Derivative of \mathcal{A} with respect to the NGS variational parameters ω	151
3.G.5	Derivative of \mathcal{G} with respect to the NGS variational parameters ω_{ij}	152
3.H	Normal-order expansion of a product of Majorana operators	154
3.I	Computations for equations of motion under imaginary-time evolution	155
3.I.1	Moving the non-Gaussian operator to the left-hand side	155
3.I.2	Compute the Gaussian transformation of u_L and O	156
3.J	Derivation of the mean-field NGS matrix	158
3.J.1	Second line simplification	159
3.J.2	First and third line simplifications	159
3.J.3	Resulting expression	161
3.K	Ensuring a monotonic decrease of the energy during the HITGD evolution	162
3.L	Mean-field expression of rotated Hamiltonian	168
3.M	Computing the commutator of the number-number operator w.r.t. the rotated Hamiltonian	170
3.M.1	Contributions from rotated one-body Hamiltonian terms	170
3.M.2	Contributions from rotated two-body Hamiltonian terms	171
3.N	Derivations of the equations of motion of the variational parameters through imaginary-time evolution	174
3.N.1	Left-hand side of the equation of motion	174
3.N.2	Right-hand side of the equation of motion	175
3.N.3	Getting the equations of motion for the covariance matrix	175
3.O	Behavior of the energy in time	177
3.O.1	Differential geometry perspective on the imaginary-time evolution	177
3.O.2	Time-dependence of the energy as a function of the CM and NGS parameters	178
3.P	Miscellaneous computations	179

3.P.1	179
3.P.2	179
3.P.3	180
3.P.4	180
3.P.5	180
4 Linear and logarithmic time compositions of quantum many-body operators	181
4.1 Introduction	183
4.2 Notation	185
4.3 The decoupling protocol	186
4.4 The selection protocol	188
4.5 Examples	191
4.5.1 Strings of Pauli operators	191
4.5.2 Number- and parity-conserving strings	191
4.5.3 Multi-control NOT gates	192
4.5.4 Unitary coupled cluster	193
4.6 Architectural considerations	195
4.7 Conclusion	196
Appendices	197
4.A Action of the staircase circuit	198
4.B Derivations for the number and parity conserving terms via iSWAP gates	199
4.C Derivations for multi-CNOT gates	200
Bibliography	203

CHAPTER 1

INTRODUCTION

Section 1.1 motivates the work carried out by the author and collaborators. It then introduces the basic language and working principles of second quantization in Section 1.2, fermionic Gaussian states in Section 1.3, imaginary time evolution in Section 1.4, quantum computation in Section 1.5, and quantum simulation in Section 1.6. Since Chapters 2-4 each contain individual introductions to the employed methods, this chapter is not an exhaustive introduction. We conclude by giving a short introduction to the quantum Hall effect in Section 1.7.

Part of the preface is based on Ref. [1]. The introduction to second quantization is based on Ref. [2] and to fermionic linear optics and Grassmann numbers on Refs. [3, 4]. Some basic concepts of imaginary time evolution are taken from Ref. [5]. Most of the introduction to quantum computing is based on Ref. [6], while the theorems, lemmas, corollaries and definitions about Hamiltonian simulation are taken from Ref. [7]. The introduction of the fractional quantum Hall effect is mainly based on Refs. [8, 9]

Section 1.1

PREFACE

Depending on the scientific background of the reader, the title *Simulating fermionic systems on classical and quantum computing devices* may lead to preconceived expectations of what the following chapters might entail. Such expectations can differ considerably depending on whether the reader’s scientific background lies in condensed matter, quantum computing, or for instance theoretical chemistry. The purpose of these introductory remarks is to put in context the contents of the following chapters in view of the rather lofty title.

The simulation of quantum mechanical systems consisting of interacting particles is considered to be a hard problem for classical computers and is just one example of a variety of problems quantum computers are promised to solve more *efficiently* than any classical processor. A quantum computer is said to solve a problem more *efficiently* than a classical computer, if the *asymptotic scaling* of the quantum algorithm’s¹ complexity measure (such as the runtime or space usage) with problem size n scales better than that of any known classical algorithm². Computing and understanding the properties of strongly interacting quantum systems is not only of pivotal interest to fundamental physics, but could also have a tremendous impact on humanity, as it could lead to huge leaps in the design of new drugs [10] and the discovery of new materials [11], to name but a few. Other problems—such as those in classical optimization or machine learning—might be completely unrelated to quantum mechanics, yet quantum algorithms could provide substantial improvements over all known classical algorithms and give rise to a variety of possible applications of quantum computing in diverse fields [12–16]. On the other hand, there are a number of examples of problems that can be solved efficiently on a classical computer, even though they are quantum mechanical in nature. Examples include Boson sampling under certain conditions [17] and fermionic linear optical systems [18], which we discuss in more detail in Section 1.3.1. In this thesis, we will tackle the problem of simulating interacting and strongly correlated fermionic systems, more specifically, the task of finding an approximate ground state for such systems. Even the most sophisticated classical methods³ (such as the quantum Monte Carlo algorithm) are

¹A classical (quantum) *algorithm* is a step-by-step instruction on how to solve a given problem with operations that can run on a classical (quantum) computer.

²The *runtime* is measured by the number of elementary operations used by the respective quantum or classical algorithm. For the former, this can be measured in terms of the quantum circuit model, which is just a specific sequence of elementary quantum operations applied to a number of qubits (a *qubit* is the quantum analogue to a classical *bit*). All of this and more is detailed in Sections 1.5 and 1.6.

³Some of the most prominent classical methods include density functional theory (DFT), which exploits the electron density distribution rather than the many electron wave function using a variety of approximations [19], but fails at describing strongly interacting systems. Another approach based on the wave function representation is the quantum Monte Carlo (QMC) method, but its efficient implementation suffers from the infamous fermionic sign problem, that leads to an exponential increase in the error ϵ of the simulation with system size [20]. Another classical algorithm used to find approximate ground states to the many-body problem is density matrix renormalization group (DMRG) [21], which very successfully describes one-dimensional systems, but has trouble building up enough entanglement to describe most strongly correlated two- and three-dimensional systems. More on the chemistry side, full configuration quantum Monte Carlo (FCIQMC) is an approach based on QMC, that deals with the

incapable of efficiently computing the ground state of highly correlated fermionic systems to desired precision (i.a. due to the fermionic sign problem [25]). It is exactly for these types of problems—those where classical algorithms break down—where the use of a quantum computer is warranted.

How hard it is to actually solve a given problem is studied in *complexity theory* and the question which problems are efficiently solvable (on a quantum computer) in the physical world—if quantum physics is true—falls into the complexity class of Bounded-Error Quantum Polynomial-Time (**BQP**). This describes the class of decision problems solvable by a quantum computer in polynomial time, with an error probability of at most $1/3$ for all instances [6]. To answer whether a given problem falls into both categories, namely it can be solved on a quantum computer more efficiently than on a classical one, while also lying in the **BQP**-class, is tricky. Problems that can be solved efficiently—that is, in polynomial complexity measure—on probabilistic classical computers, lie in the complexity class Bounded-Error Probabilistic Polynomial-Time (**BPP**). This describes the class of decision problems solvable by a probabilistic Turing machine in polynomial time with an error probability of at most $1/3$ for all instances. A probabilistic Turing machine models the abilities of a classical computer and has the additional ability to choose available transitions at each point of a computation according to some probability distribution. It is rather simple to show that $\mathbf{BPP} \subseteq \mathbf{BQP}$, but whether $\mathbf{BPP} \neq \mathbf{BQP}$, in other words, whether quantum computers are more powerful than probabilistic classical computers, has actually not been proven yet. What *has* been proven is that given access to a certain *oracle*⁴, quantum algorithms exist which have a provable better asymptotic scaling over *any* classical algorithm in an oracle-relative sense [1]. The first such algorithm was discovered by Bernstein and Vazirani [26] in 1993, which provided a quasipolynomial speedup over classical algorithms. Grover’s algorithm [27]—designed to search a classical database—provides a quadratic speedup over any classical algorithm [28]. The first quantum algorithm with a provable exponential improvement over any classical algorithm was found by Simon [29], which motivated the algorithm for prime factorization of an integer by Shor [30]. In Shor’s algorithm, quantum phase estimation (which we will discuss in Section 1.6.2) is used as a period-finding subroutine (oracle). To summarize, there is yet no mathematical proof that quantum computers are more powerful than classical computers, unless both are given access to certain oracles, in which case quantum computers may provide an exponential improvement in complexity (depending on the problem).

The Google Quantum AI team’s recent experiment [31] demonstrated that we are now entering the post-quantum-supremacy era [32], meaning that we have reached a stage where a certain computational task can actually be executed exponentially faster on a quantum processor than on any classical processor. This breakthrough in quantum com-

many-body wave function of an interacting fermionic system [22]. It is a projector method in imaginary time which expands the electronic wave function as a linear combination of Slater determinants based on stochastic sampling of amplitudes that describe the weights of the respective Slater determinants in the expansion. The recently introduced adaptive sampling configuration interaction (ASCI) improves upon FCIQMC by making it deterministic [23]. In this work, we will implement the ASCI algorean-field method based on generalized Hartree-Fock theory [24].

⁴An oracle is a black-box operation designed to realize a specific task. One distinguishes between classical- and quantum-oracles, which take a classical (quantum) input string, perform the desired operation, and output a string (quantum state). The operation the classical (quantum) oracle performs on the given input is known as a (quantum-) subroutine. Two of the most prominent quantum oracles are the quantum Fourier transformation, or the time evolution operator $U = e^{-iHt}$ we come across in Sections 1.6.1 and 1.6.3, respectively.

puting does not contradict the previous paragraph, since currently no claim can be made that there does *not* exist a classical algorithm that could do the same sampling task as Google’s quantum processor with similar or even better complexity scaling. We are, however, still quite far away from being able to execute an algorithm of disruptive size with provable exponential speedup—like Shor’s algorithm—in an actual experiment. These algorithms require quantum circuits that possess running times which are several orders of magnitude larger than the coherence time of current quantum processors, where *coherence time* refers to the time for which there exists a definite phase relation between different quantum states of the quantum processor. Such phase relations are easily destroyed by e.g. spurious magnetic fields, making quantum systems much more susceptible to noise than classical ones. While classical computing allows to protect the computational state by simply copying it to several bit registers, a similar procedure to avert decoherence is not possible on a quantum computer due to the *no-cloning theorem* [33], which forbids the copying of an arbitrary unknown quantum state. Instead, one can spread the information of the encoded quantum state among many parties to make it more robust against local errors. While current quantum computing platforms strive to implement this sort of protection through an underlying quantum algorithm—namely, the realization of the *toric code* as proposed by Kitaev [34], whose two dimensional spin-lattice analogue is known as the *surface code* [35] — other, more exotic types of systems are believed to possess this sort of topological protection already on a hardware level. One prominent type of physical system which is believed to display this behavior on a hardware level are fractional quantum Hall systems, which will be introduced in Section 1.7 and whose quantum simulation will be discussed in Chapter 2. The concepts of quantum error-correction have to be extended also to faulty gate operations, faulty quantum state preparation and faulty measurements. Only then is it possible to realize a fault-tolerant quantum computer, which can execute quantum operations indefinitely. Luckily, in order to implement e.g. the surface code, it is acceptable to use faulty quantum resources—more on this in Section 1.5.2— and with recent progress in the design of quantum processors, the realization of a quantum error-corrected processor might no longer be so far out of reach.

Where does the content of this thesis fit into all of this? As mentioned before, the simulation of strongly interacting fermionic systems is a task which quickly becomes infeasible on a classical computer, requiring the use of a universal quantum simulator, such as a quantum computer. At sufficiently low temperatures⁵ the quantum system will be in its ground state and its computation (as well as the computation of its first couple of excitations) will allow us to compute its quantum mechanical properties. In Chapter 2, we provide a roadmap for simulating the fractional quantum Hall effect on a quantum computer. Its understanding is not only of interest to fundamental physics, but could also lead to potential leaps in material designs for new platforms of quantum computing with intrinsically protected quantum states and operations [36]. While this chapter also puts a lot of emphasize on the derivation of the Hamiltonian in second quantization and the complexity cost for its simulation on an error-corrected quantum computer, it also lays emphasis on one subroutine that is routinely required for many quantum algorithms⁶—

⁵The rule of thumb here is that the energy gap, i.e. the energy difference $\Delta E = E_1 - E_0$, between the first excited state and the ground state should be much larger than $k_B T$, where k_B is the Boltzmann constant and T is the temperature.

⁶In adiabatic quantum algorithms [37], one does not require the computation of such an initial/reference state, as adiabatic state generation algorithms follow a completely different strategy, starting in a well-known initial state and adiabatically evolving the state to the ground state of the problem Hamiltonian [38]. Adiabatic quantum computing has been shown to be equivalent to standard quantum

such as quantum phase estimation— namely, finding an initial/reference state which has a non-vanishing overlap with the (unknown) ground state. As we will learn in Section 1.6.2, the number of times and thus the duration of a quantum computation depends significantly on our ability to initialize the quantum computer in an initial state that possesses a large overlap with the desired. While the overlap of a trial wave function with the desired eigenstate will decrease exponentially with system size [19], a quantum simulation on a quantum computer will most likely only involve modest system sizes for which one has to try to "postpone" the exponential decay of the overlap as long as possible by using elaborate classical simulation methods [40]. In this context, we study algorithms which can be efficiently computed on classical computers and efficiently be implemented on noisy, or error-corrected quantum computers in Chapters 2 and 3. We show how the recently introduced variational method by Shi et al. [41], which is designed to efficiently study bosonic and fermionic systems beyond mean-field, can be applied to fermionic systems and implemented on a quantum computer in Chapter 3. The simplicity of the quantum circuit for the variational Ansatz allows for a straightforward implementation even on noisy quantum computers. Another prerequisite for the quantum simulation of fermionic systems is the ability to construct many-body operators from two-body interactions native to the underlying quantum computing architecture. In this context, we introduce two new algorithms designed for this task and improve upon long-standing construction schemes for a variety of prominent many-body operators in Chapter 4. In order to understand the ensuing chapters, we will introduce the most important physical concepts used in this thesis in the remainder of this chapter, with a main focus on quantum computing and simulation.

computing, i.e. the gate model [39], but also requires error-correction and will not be considered in this work.

Section 1.2

MANY-BODY FERMIONIC SYSTEMS IN SECOND QUANTIZATION

We will be working in second quantization, which differs from the standard formulation of quantum mechanics in that also the wave functions are represented in terms of operators, namely the bosonic or fermionic creation and annihilation operators applied to the vacuum state. Unlike an approach based on first quantization, in second-quantized representation one no longer has to worry about whether a state is properly symmetrized or antisymmetrized, since this follows automatically from the commutation and anti-commutation relations of the bosonic and fermionic operators. Since we will be dealing with electrons in this work, we will give an introduction to second quantization for fermionic systems.

A one-electron function is known as a *spin-orbital*. Let $\{\phi_p(\mathbf{x})\}$ be a basis of N_{so} orthonormal spin-orbitals, where \mathbf{x} represents the three spatial coordinates of an electron. A *Slater-determinant* is an antisymmetrized product of one or more spin-orbitals,

$$|\phi_{p_1} \phi_{p_2} \cdots \phi_{p_{N_{\text{el}}}}\rangle = \frac{1}{\sqrt{N_{\text{el}}!}} \begin{vmatrix} \phi_{p_1}(\mathbf{x}_1) & \phi_{p_2}(\mathbf{x}_1) & \cdots & \phi_{p_{N_{\text{el}}}}(\mathbf{x}_1) \\ \phi_{p_1}(\mathbf{x}_2) & \phi_{p_2}(\mathbf{x}_2) & \cdots & \phi_{p_{N_{\text{el}}}}(\mathbf{x}_2) \\ \vdots & \vdots & \ddots & \vdots \\ \phi_{p_1}(\mathbf{x}_{N_{\text{el}}}) & \phi_{p_2}(\mathbf{x}_{N_{\text{el}}}) & \cdots & \phi_{p_{N_{\text{el}}}}(\mathbf{x}_{N_{\text{el}}}) \end{vmatrix}. \quad (1.2.1)$$

Slater determinants are anti-symmetric wave functions used for a first-quantization description of fermionic systems. Each determinant can be represented by an *occupation number vector* (sometimes also called a *Fock basis vector*)

$$|\mathbf{k}\rangle = |k_1, k_2, \dots, k_{N_{\text{so}}}\rangle, \quad (1.2.2)$$

where $k_p = 1$ if ϕ_p is present in the determinant and $k_p = 0$ if it is not. The inner product for two occupation vectors is defined as⁷

$$\langle \mathbf{k} | \mathbf{m} \rangle = \prod_{p=1}^{N_{\text{so}}} \delta_{k_p m_p}, \quad (1.2.3)$$

where $\delta_{ij} = 1$ if $i = j$ and $\delta_{ij} = 0$ if $i \neq j$. The occupation vectors in Eq. (1.2.2) constitute an orthonormal basis of the *Fock space* $F(N_{\text{so}})$, which may be written as a direct sum of subspaces $F(N_{\text{so}}, N_{\text{el}})$ containing all occupation number vectors obtained by distributing N_{el} electrons among N_{so} orbitals,

$$F(N_{\text{so}}) = F(N_{\text{so}}, 0) \oplus F(N_{\text{so}}, 1) \oplus \cdots \oplus F(N_{\text{so}}, N_{\text{so}}), \quad (1.2.4)$$

where \oplus denotes the direct sum. The subspace $F(N_{\text{so}}, 0)$ contains a single vector, the *vacuum state*

$$|\text{vac}\rangle = |0_1, 0_2, \dots, 0_{N_{\text{so}}}\rangle. \quad (1.2.5)$$

⁷We use the convention of omitting the comma in subscripts whenever it can be done without causing confusion. For instance, we will write the comma in expressions such as $M_{a+b,c}$, since M_{a+bc} is misleading.

The *fermionic creation operator* c_p^\dagger acts on an occupation number vector as follows

$$c_p^\dagger |k_1, k_2, \dots, 0_p, \dots, k_{N_{\text{so}}}\rangle = \prod_{q=1}^{p-1} (-1)^{k_q} |k_1, k_2, \dots, 1_p, \dots, k_{N_{\text{so}}}\rangle \quad (1.2.6)$$

$$c_p^\dagger |k_1, k_2, \dots, 1_p, \dots, k_{N_{\text{so}}}\rangle = 0. \quad (1.2.7)$$

The phase factor in Eq. (1.2.6) is necessary in order for the second-quantized representation to be consistent with first quantization, while Eq. (1.2.7) is a manifestation of the Pauli principle, i.e. that a spin-orbital can only be occupied at most once. Any occupation vector can be written as a string of creation operators acting on the vacuum state. The adjoint operator to c_p^\dagger is known as the fermionic annihilation operator which acts on an occupation number vector as

$$c_p |k_1, k_2, \dots, 1_p, \dots, k_{N_{\text{so}}}\rangle = \prod_{q=1}^{p-1} (-1)^{k_q} |k_1, k_2, \dots, 0_p, \dots, k_{N_{\text{so}}}\rangle \quad (1.2.8)$$

$$c_p |k_1, k_2, \dots, 0_p, \dots, k_{N_{\text{so}}}\rangle = 0. \quad (1.2.9)$$

Creation (annihilation) operators anti-commute among themselves. The anti-commutator between creation and annihilation on the other hand is given by

$$\{c_p, c_q^\dagger\} = c_p c_q^\dagger + c_q^\dagger c_p = \delta_{p,q}. \quad (1.2.10)$$

An important operator we will later encounter checks whether a spin-orbital is occupied by an electron or not, $n_p = c_p^\dagger c_p$. The total number of electrons in an occupation number vector is then given by $\sum_{p=1}^{N_{\text{so}}} n_p$. If a product of annihilation and creation operator is in a form where all creation operators are on the left-hand side of the expression, this is known as *normal order*, denoted by two colons. For instance, $:a_p a_q^\dagger := -a_q^\dagger a_p$ ⁸. The correspondence between Slater determinants and occupation number vectors is given by

$$\langle \mathbf{x}_1, \mathbf{x}_2, \dots, \mathbf{x}_{N_{\text{el}}} | c_{p_1}^\dagger c_{p_2}^\dagger \cdots c_{p_{N_{\text{el}}}}^\dagger | 0 \rangle = \frac{1}{\sqrt{N_{\text{el}}!}} \begin{vmatrix} \phi_{p_1}(\mathbf{x}_1) & \phi_{p_2}(\mathbf{x}_1) & \cdots & \phi_{p_{N_{\text{el}}}}(\mathbf{x}_1) \\ \phi_{p_1}(\mathbf{x}_2) & \phi_{p_2}(\mathbf{x}_2) & \cdots & \phi_{p_{N_{\text{el}}}}(\mathbf{x}_2) \\ \vdots & \vdots & \ddots & \vdots \\ \phi_{p_1}(\mathbf{x}_{N_{\text{el}}}) & \phi_{p_2}(\mathbf{x}_{N_{\text{el}}}) & \cdots & \phi_{p_{N_{\text{el}}}}(\mathbf{x}_{N_{\text{el}}}) \end{vmatrix}. \quad (1.2.11)$$

Any physical Hamiltonian that is described by electromagnetic interactions can be expressed in terms of one- and two-body operators. The former may describe the kinetic energy of the fundamental particles or local fields, while the latter describes the (e.g. Coulomb-) interaction between the various particles. Expectation values correspond to observables and should therefore be independent on whether one chooses a first- or second-quantized representation (i.e. Slater determinants or occupation number vectors). Therefore, an operator in the Fock space can be constructed from the condition that its matrix elements should be identical to the corresponding matrix elements in first quantization. While matrix elements between Slater determinants depend on the spatial form of the spin-orbitals, the occupation number operators are independent on the spatial form of

⁸Normal ordering has to be understood as a rule which gives a minus whenever a fermionic operator— independent of its spin-orbital index—is moved past another fermionic operator to obtain the desired ordering. So we have $:a_p a_q^\dagger := -a_q^\dagger a_p$, but $a_p a_q^\dagger = -a_q^\dagger a_p + \delta_{pq}$.

the spin-orbitals, however, the second-quantized operators themselves depend on the spatial form of the spin-orbitals. We will see that this spatial dependence is hidden in the one- and two-body integrals which are central to the description of a many-body system's Hamiltonian in second quantization.

The first quantization one-body operators have the general form

$$T^c = \sum_{i=1}^{N_{\text{el}}} T^c(\mathbf{x}_i), \quad (1.2.12)$$

where the superscript indicates that one is working in first quantization (c as in coordinate representation). Since states of different particle numbers do not overlap, see Eq. (1.2.3), and since only a single electron appears in each term of the above expression, a general one-body operator in second quantization will have the following form

$$T = \sum_{p,q=1}^{N_{\text{so}}} f_{pq} c_p^\dagger c_q. \quad (1.2.13)$$

The form of f_{pq} is given by demanding that the matrix elements should coincide with its counterpart in first quantization, where the matrix elements between Slater determinants can be obtained from the Slater-Condon rules [42–44]. The matrix elements in second quantization are identical to their first quantization versions by choosing

$$f_{pq} = \int d\mathbf{x} \phi_p^*(\mathbf{x}) T^c(\mathbf{x}) \phi_q(\mathbf{x}). \quad (1.2.14)$$

Similarly, two-body operators in first quantization are given by

$$V^c = \frac{1}{2} \sum_{i \neq j}^{N_{\text{el}}} V^c(\mathbf{x}_i, \mathbf{x}_j), \quad (1.2.15)$$

where typically $V^c(\mathbf{x}_i, \mathbf{x}_j) = 1/|\mathbf{x}_i - \mathbf{x}_j|$ (Coulomb interaction) and their second-quantized version reads

$$V = \frac{1}{2} \sum_{p,q,r,s=1}^{N_{\text{so}}} h_{pqrs} c_p^\dagger c_r^\dagger c_s c_q. \quad (1.2.16)$$

The matrix elements of both representations are identical when choosing the two-body coefficients as⁹

$$h_{pqrs} = \iint d\mathbf{x}_1 d\mathbf{x}_2 \phi_p^*(\mathbf{x}_1) \phi_r^*(\mathbf{x}_2) V^c(\mathbf{x}_1, \mathbf{x}_2) \phi_q(\mathbf{x}_1) \phi_s(\mathbf{x}_2). \quad (1.2.17)$$

One can combine Eqs. (1.2.13) and (1.2.16) and obtain the electronic structure Hamiltonian $H = T + V$. In doing so, we have quietly neglected the fact that matter consists not only of electrons, but also of nuclei. The nuclei masses are however three orders of magnitudes larger than the masses of the electrons and one can assume the electrons to move within a field of fixed nuclei within good approximation, which is known as the Born-Oppenheimer approximation [45, 46]. A precise non-relativistic treatment of matter would include electron-nucleon, nucleon-nucleon, electron-electron interaction as well as single-body electron and nucleon terms.

⁹Note that this definition slightly deviates from the definition we use in Chapters 2 and 3.

Within the context of quantum computation, approaches based on a second-quantized representation of the Hamiltonian clearly outnumber first quantized approaches. This is owed to the fact that even though the number of Hamiltonian terms in first quantization has a more favorable scaling than second-quantized operators, the former requires the initialization in an anti-symmetric state (which is potentially costly), as well as large number of qubits in order to discretize space [47–50]. Second-quantized approaches on the other hand avert the anti-symmetrization dilemma, since the anti-symmetric algebraic properties are shifted to the fermionic operators, while the geometry of the problem is contained in the one- and two-body integrals. We will work in second quantization throughout this thesis.

So far we have used Slater determinants or occupation number vectors as a basis for constructing the Fock space. In the following we will introduce fermionic coherent states, which are the eigenstates of the fermionic annihilation operator, as an alternative basis. We will see that this is a natural basis when working with fermionic Gaussian states and that the appearing eigenvalues must necessarily be anti-commuting numbers, which are known as Grassmann numbers.

Section 1.3

FERMIONIC GAUSSIAN STATES

1.3.1 Fermionic linear optics

Much of the work in this thesis is based on quantum computation with fermionic linear optics (FLO), which has been studied for example in Refs. [3, 51, 52]. Performing quantum computation (which will be introduced in more detail in Section 1.5) with FLO requires a sequence of linear 'optical elements'. These can be either unitary operators describing an evolution under a Hamiltonian quadratic in the number of creation and annihilation operators, or non-unitary operations which are just single-mode projective measurements of $n_j = c_j^\dagger c_j$, applied to a Fock state. In a FLO quantum computation one can alternate between linear optical operations and measurements, so that the choice of a unitary operation in a computation step can be made dependent on the measurement outcome in a previous step. Its bosonic analogue (which consists of passive linear optical elements, squeezers, single photon sources and photodetectors) was shown to provide a universal set of quantum gates¹⁰ for quantum computing, meaning that bosonic linear optics is a potential platform for realizing a quantum computer [53]. One could now argue that since bosonic linear optics is universal for quantum computing, FLO should also be universal for quantum computing, since bosons can be represented by paired fermions (see e.g. Ref. [54]). However, this is not the case because the transformation from bosons to paired fermionic leads to an evolution under a quartic Hamiltonian expression and not a quadratic one [52] and thus cannot be described by FLO. This is hardly surprising, since FLO can efficiently be simulated by classical means [3], a property we will make extensive use of throughout the thesis. A large part of this thesis is based on *fermionic Gaussian states* (FGS), which are the workhorse of FLO quantum computation. A FGS is a state which can be constructed from a Fock state undergoing evolution under a *quadratic Hamiltonian*

$$H = \sum_{j=1}^{N_{\text{so}}} \varepsilon_j c_j^\dagger c_j + \sum_{1 \leq j < k \leq N_{\text{so}}} \left(t_{jk} c_j^\dagger c_k + t_{jk}^* c_k^\dagger c_j \right) + \sum_{1 \leq j < k \leq N_{\text{so}}} \left(s_{jk} c_j^\dagger c_k^\dagger + s_{jk}^* c_k c_j \right), \quad (1.3.1)$$

where $\varepsilon_j \in \mathbb{R}$, $t_{jk}, s_{jk} \in \mathbb{C}$ are parameters describing the mode energies, tunnelling processes and non-number conserving terms [55]. Quadratic Hamiltonians such as the one displayed above can also be seen as Hamiltonians describing *non-interacting fermions*. This is owed to the fact that by performing a generalized transformation (a change of basis known as a Bogoliubov-Valatin transformation) the Hamiltonian can be brought into standard form, i.e. a sum of terms each acting only on a single fermionic mode [56, 57].

Instead of using the fermionic creation and annihilation operators defined by Eqs. (1.2.6)-(1.2.10), it will be convenient to introduce $2N_{\text{so}}$ Hermitian operators as in Ref. [3], the so-called *Majorana operators* which are defined as

$$a_{1,p} = c_p^\dagger + c_p \quad (1.3.2)$$

$$a_{2,p} = i(c_p^\dagger - c_p). \quad (1.3.3)$$

¹⁰We will discuss what a universal set of quantum gate means in more detail in Section 1.5.3.

Their respective canonical anti-commutation relations are given by

$$\{a_{\alpha,p}, a_{\beta,q}\} = 2\delta_{\alpha,\beta}\delta_{p,q}, \quad (1.3.4)$$

making them Hermitian operators by definition. An algebra generated by the operators A_j , where A_j can be either $a_{1,p}$ or $a_{2,p}$, is called the *Clifford algebra* $\mathcal{C}_{2N_{\text{so}}}$. Arbitrary operators $X \in \mathcal{C}_{2N_{\text{so}}}$ can be represented as polynomials of Majorana operators,

$$X = \frac{1}{2^n} \text{tr}(X) \mathbb{1}_{2N_{\text{so}}} + \sum_{p=1}^{2N_{\text{so}}} \sum_{1 \leq a_1 < \dots < a_p \leq 2N_{\text{so}}} \alpha_{a_1, \dots, a_{2N_{\text{so}}}} A_{a_1} \dots A_{a_{2N_{\text{so}}}}, \quad (1.3.5)$$

where $\mathbb{1}_{2N_{\text{so}}}$ denotes the $(2^{N_{\text{so}}} \times 2^{N_{\text{so}}})$ identity matrix [3]. Since it is usually clear which dimension the identity matrix should have we often omit the subscript and just write $\mathbb{1}$. An operator X is called even (odd) if involves only even (odd) powers of the generators A_j . The Hamiltonian in Eq. (1.3.1) has the same form as Eq. (1.3.5) since it can be written as

$$H = \frac{i}{4} \sum_{p,q=1}^{2N_{\text{so}}} H_{pq} A_p A_q, \quad (1.3.6)$$

where H_{pq} are elements of a real and anti-symmetric matrix. We will see in Chapter 3, that the unitary elements of FLO constitute a group of canonical transformation. If $U = e^{iH}$ with H as given by Eq. (1.3.6), then the adjoint action of U on A_p is just a rotation $U_m \in \text{SO}(2N_{\text{so}})$,

$$U^\dagger A_p U = \sum_{q=1}^{2N_{\text{so}}} (U_m)_{pq} A_q, \quad (1.3.7)$$

where $U_m U_m^T = \mathbb{1}_{2N_{\text{so}}}$ and $\det(U_m) = 1$.

1.3.2 Grassmann numbers

Much of quantum optics is described in terms of bosonic coherent states, which are the eigenstates of the bosonic annihilation operators and allow for a treatment of all states of the electromagnetic field. Let us assume that a fermionic analogue to the bosonic coherent state exists and let us denote it by $|\gamma\rangle$, where $\gamma = \gamma_1, \gamma_2, \dots$ are some unspecified numbers. Since the fermionic coherent state is an eigenstate of the fermionic annihilation operator, we expect that

$$c_p |\gamma\rangle = \gamma_p |\gamma\rangle. \quad (1.3.8)$$

Using the anti-commutation relations, one observes that [58]

$$c_p c_q |\gamma\rangle = \gamma_p \gamma_q |\gamma\rangle \quad (1.3.9)$$

$$c_p c_q |\gamma\rangle = -c_q c_p |\gamma\rangle = -\gamma_q \gamma_p |\gamma\rangle. \quad (1.3.10)$$

By comparing the right-hand sides of Eqs. (1.3.9) and (1.3.10), one realizes that equality requires γ to be composed of *anti-commuting variables* to obtain non-trivial solutions.

These variables are known as *Grassmann variables* [59] and obey the following set of anti-commutation relations

$$\{\gamma_p, \gamma_q\} = 0 \quad (1.3.11)$$

$$\{\gamma_p^*, \gamma_q\} = 0 \quad (1.3.12)$$

$$\{\gamma_p^*, \gamma_q^*\} = 0, \quad (1.3.13)$$

where γ_p^* denotes the complex conjugate of γ_p and $(\gamma_p \gamma_q)^* = \gamma_q^* \gamma_p^*$. An important property of Grassmann numbers is that they square to zero,

$$\gamma_p^2 = 0, \quad (1.3.14)$$

which follows from Eq. (1.3.11). When dealing with fermionic systems it turns out that many quantities of interest (e.g. expectation values) can be solved *exactly* by the following natural linear map [3].

Definition 1.3.1 (Grassmann representation of the operator X). *One can assign a polynomial of Grassmann variables*

$$\omega(A_p A_q \cdots A_r, \boldsymbol{\gamma}) = \gamma_p \gamma_q \cdots \gamma_r \quad (1.3.15)$$

$$\omega(\mathbf{1}, \boldsymbol{\gamma}) = 1 \quad (1.3.16)$$

to any set of Majorana operator A_p, A_q, \dots, A_r , which extends by linearity to any $X \in \mathcal{C}_{2N_{\text{so}}}$. The map $\omega(X, \boldsymbol{\gamma})$ is called the *Grassmann representation of the operator X* .

The map ω is just an isomorphism of linear spaces. Similar to Eq. (1.3.7), a canonical transformation by a unitary U of the FLO elements is equivalent to an orthogonal change of basis in the space of the Grassmann variables, i.e. for $X \in \mathcal{C}_{2N_{\text{so}}}$, $R \in \text{SO}(2N_{\text{so}})$,

$$\omega(U^\dagger X U, \boldsymbol{\theta}) = \omega(X, \boldsymbol{\eta}) \quad (1.3.17)$$

$$\eta_p = \sum_{q=1}^{2N_{\text{so}}} R_{pq} \theta_q. \quad (1.3.18)$$

We will mainly use Greek letters to denote Grassmann variables, with some minor exceptions in later sections owed to the fact that we wanted to be consistent with notation employed in literature.

We will now consider a set $\{\gamma_i\}$ of Grassmann variables and state all major properties which we will make use of in Section 3.4. Many of the definitions that follow can also be found in Chapter 3 and are included here solely for the sake of completeness.

Fermionic coherent states

For fermionic fields, the only physically realizable eigenstate of the fermionic annihilation operator is the vacuum state. Other eigenstates can be constructed only in a formal way and are merely introduced as a means to do analytical computations. One can show that the unitary displacement operator

$$D(\boldsymbol{\gamma}) = e^{\sum_i (c_i^\dagger \gamma_i - \gamma_i^* c_i)} \quad (1.3.19)$$

acts as follows on fermionic creation and annihilation operators

$$D(\boldsymbol{\gamma})^\dagger c_i D(\boldsymbol{\gamma}) = c_i + \gamma_i \quad (1.3.20)$$

$$D(\boldsymbol{\gamma})^\dagger c_i^\dagger D(\boldsymbol{\gamma}) = c_i^\dagger + \gamma_i^*. \quad (1.3.21)$$

Deriving typical identities for Grassmann variables as in Eqs. (1.3.20)-(1.3.21) often involves the trick of writing out the matrix exponential in its series expansion and using Eq. (1.3.14). One can use the above identities to show that a *fermionic coherent state* can be written with respect to the displacement operator acting on the fermionic vacuum state,

$$|\boldsymbol{\gamma}\rangle = D(\boldsymbol{\gamma}) |\text{vac}\rangle, \quad (1.3.22)$$

since

$$c_i |\boldsymbol{\gamma}\rangle = \gamma_i |\boldsymbol{\gamma}\rangle. \quad (1.3.23)$$

The adjoint of the coherent state obeys

$$\langle \boldsymbol{\gamma} | c_i^\dagger = \langle \boldsymbol{\gamma} | \gamma_i^*. \quad (1.3.24)$$

The inner product of two coherent states is given by

$$\langle \boldsymbol{\alpha} | \boldsymbol{\beta} \rangle = e^{\sum_i (\alpha_i^* \beta_i - \frac{1}{2} (\alpha_i^* \alpha_i + \beta_i^* \beta_i))}. \quad (1.3.25)$$

Coherent states are commonly used to describe fermionic path integrals [59].

Grassmann calculus

As a result of Eq. (1.3.14), the most general function of a single Grassmann variable γ is linear in γ ,

$$f(\gamma) = u + t\gamma, \quad (1.3.26)$$

where t, u can be either complex or Grassmann numbers. One can define the *left derivative* of such a function $f(\gamma)$ as

$$\frac{df(\gamma)}{d\gamma} = t \quad (1.3.27)$$

The word 'left' emphasizes that the variable one wants to differentiate needs to be moved all the way to the left (i.e. γ has to be moved past t *before* taking the derivative. If t is also a Grassmann number, say τ , this will lead to a minus sign due to the anti-commutation relations, more specifically

$$\frac{d}{d\gamma}(u + \tau\gamma) = -\frac{d}{d\gamma}\gamma\tau = -\tau. \quad (1.3.28)$$

Functions of Grassmann variables $f(\boldsymbol{\gamma})$ which do commute with a Grassmann number are called *even*, those that anti-commute are called *odd* functions.

One can also take the derivative w.r.t. a Grassmann variable γ_i of a product of two functions of Grassmann variables $f(\boldsymbol{\gamma})g(\boldsymbol{\gamma})$. The Grassmann product-rule reads

$$\frac{d}{d\gamma_i}(f(\boldsymbol{\gamma})g(\boldsymbol{\gamma})) = \begin{cases} \frac{df(\boldsymbol{\gamma})}{d\gamma_i}g(\boldsymbol{\gamma}) + f(\boldsymbol{\gamma})\frac{dg(\boldsymbol{\gamma})}{d\gamma_i}, & \text{if } f \text{ is even} \\ \frac{df(\boldsymbol{\gamma})}{d\gamma_i}g(\boldsymbol{\gamma}) - f(\boldsymbol{\gamma})\frac{dg(\boldsymbol{\gamma})}{d\gamma_i}, & \text{if } f \text{ is odd.} \end{cases} \quad (1.3.29)$$

In the Grassmann representation, left integration is equivalent to differentiation and is defined as

$$\int d\gamma_p = \int d\gamma_p^* = 0 \quad (1.3.30)$$

$$\int d\gamma_p d\gamma_q = \delta_{pq} \quad (1.3.31)$$

$$\int d\gamma_p^* d\gamma_q^* = \delta_{pq}. \quad (1.3.32)$$

We will often encounter integrals of pairs of Grassmann variables, which will be denoted

$$\int d^2\gamma_i = \int d\gamma_i^* d\gamma_i \quad (1.3.33)$$

$$\int d^2\boldsymbol{\gamma} = \int \prod_i d^2\gamma_i. \quad (1.3.34)$$

Using the properties of Grassmann calculus one can show that the coherent states $|\boldsymbol{\gamma}\rangle$ are (over-) complete¹¹.

Operators and their expansions

As we have seen in Section 1.2, operators in quantum mechanics appear as sums of products of annihilation and creation operators. In quantum mechanics, due to the law (or rather the conjecture) that parity is conserved, these sums contain either odd or even products of such operators, but never both. Since Grassmann variables do not appear in the operators of quantum mechanics, even (odd) operators commute with Grassmann variables.

Since coherent states form a complete set of states, the identity operator and the trace operation of an arbitrary operator B are given by

$$\mathbb{1} = \int d^2\boldsymbol{\alpha} |\boldsymbol{\alpha}\rangle \langle \boldsymbol{\alpha}| \quad (1.3.35)$$

$$\text{tr}(B) = \int d^2\boldsymbol{\alpha} \langle \boldsymbol{\alpha}|B|-\boldsymbol{\alpha}\rangle = \int d^2\boldsymbol{\alpha} \langle -\boldsymbol{\alpha}|B|\boldsymbol{\alpha}\rangle \quad (1.3.36)$$

$$\text{tr}(B|\boldsymbol{\beta}\rangle \langle \boldsymbol{\gamma}|) = \langle -\boldsymbol{\beta}|B|\boldsymbol{\gamma}\rangle = \langle \boldsymbol{\beta}|B|-\boldsymbol{\gamma}\rangle \quad (1.3.37)$$

and the coherent state overlap can be computed through Eq. (1.3.25). One can define a Grassmann-type delta function which plays the role of the Dirac delta function for Grassmann variables,

$$\delta(\boldsymbol{\beta} - \boldsymbol{\gamma}) = \prod_i (\beta_i - \gamma_i) (\beta_i^* - \gamma_i^*). \quad (1.3.38)$$

¹¹An arbitrary single-mode quantum state $|f\rangle = (a + bc^\dagger)$, where $a, b \in \mathbb{C}$, its coherent state expansion is given by $\int d^2\gamma \langle \boldsymbol{\gamma}|f\rangle |\boldsymbol{\gamma}\rangle = \int d^2\gamma (a + b\gamma^*)(1 + \gamma\gamma^* - \gamma c^\dagger) |0\rangle = |f\rangle$. This can be extended to the multi-mode case. Similarly one can show that the displacement operators form an overcomplete set of operators for an arbitrary multi-mode fermionic operator [4].

Another representation of the (single-mode) delta function is

$$\delta(\beta - \gamma) = \text{tr}(D_n(\gamma)E_A(-\beta)), \quad (1.3.39)$$

where

$$D_n(\gamma) = D(\gamma)e^{\frac{1}{2}\sum_i \gamma_i^* \gamma_i} \quad (1.3.40)$$

is the normal-ordered displacement operator and

$$E_A(\beta) = \int d^2\alpha e^{\beta\alpha^* - \alpha\beta^*} |\alpha\rangle \langle -\alpha| \quad (1.3.41)$$

is the Fourier transform of the coherent state dyadic $|\alpha\rangle \langle -\alpha|$. Therefore, an arbitrary operator B can for instance be expanded in terms of the normally ordered displacement operator through

$$B = \int d^2\gamma \text{tr}(BE_A(\gamma))D_N(-\gamma). \quad (1.3.42)$$

Mean values of operators

In the following we will introduce additional expressions in order to derive a simple equation for computing expectation values of products of fermionic creation and annihilation operators.

For a density operator ρ ¹² we define the *characteristic function*

$$\chi(\gamma) = \text{tr} \left(\rho e^{\sum_i (\gamma_i c_i^\dagger - c_i \gamma_i^*)} \right). \quad (1.3.43)$$

At this point we introduce a generalized ordering which allows one to switch between normal ordering ($s = -1$) and anti-normal ordering ($s = 1$),

$$\{c_p^\dagger c_p\}_s = c_p^\dagger c_p + \frac{1}{2}(s - 1). \quad (1.3.44)$$

Using Eqs. (1.3.43)-(1.3.44), one can define the s -ordered characteristic function and the s -ordered displacement operator

$$\chi(\gamma, s) = \chi(\gamma) e^{\frac{s}{2}\sum_i \gamma_i^* \gamma_i} \quad (1.3.45)$$

$$D(\gamma, s) = D(\gamma) e^{\frac{s}{2}\sum_i \gamma_i^* \gamma_i}, \quad (1.3.46)$$

which shows that the s -ordered characteristic function is an even function of even parity, $\chi(-\gamma, s) = \chi(\gamma, s)$, that can be written as a trace including the s -ordered displacement operator through

$$\chi(\gamma, s) = \text{tr}(\rho D(\gamma, s)). \quad (1.3.47)$$

¹²We will consider a physical density operator ρ which is a positive Hermitian operator of unit trace, i.o.w. ρ must fulfill

$$\begin{aligned} \langle \psi | \rho | \psi \rangle &\geq 0 \quad \forall |\psi\rangle \\ \rho^\dagger &= \rho \\ \text{tr}(\rho) &= 1. \end{aligned}$$

The *expectation value* of a fermionic operator X w.r.t. a normalized quantum state ρ is given by $\langle X \rangle_\rho = \text{tr}(\rho X)$.

With the definition of Eq. (1.3.41), one defines

$$E(\boldsymbol{\gamma}, s) = E_A(\boldsymbol{\gamma}) e^{\frac{s+1}{2} \sum_i \gamma_i^* \gamma_i}. \quad (1.3.48)$$

Using the above definition, one can do a general expansion of an arbitrary operator (here, the density operator ρ) through

$$\rho = \int d^2 \boldsymbol{\gamma} \chi(\boldsymbol{\gamma}, s) E(-\boldsymbol{\gamma}, -s). \quad (1.3.49)$$

The s -ordered quasiprobability distribution $W(\boldsymbol{\gamma}, s)$ is defined as the Fourier transform of the s -ordered characteristic function,

$$W(\boldsymbol{\gamma}, s) = \int d^2 \boldsymbol{\beta} e^{\sum_i (\gamma_i \beta_i^* - \beta_i \gamma_i^*)} \chi(\boldsymbol{\beta}, s). \quad (1.3.50)$$

All the above definitions allow us to write an equation for computing mean values of s -ordered monomials of fermionic creation and annihilation operators which are of the form

$$\prod_i \{c_i^{\dagger n_i} c_i^{m_i}\}_s, \quad (1.3.51)$$

where $n_i, m_i \in \{0, 1\}$. Such mean values of fermionic operators may be computed by performing the Grassmann integral

$$\text{tr} \left(\rho \prod_i \{c_i^{\dagger n_i} c_i^{m_i}\}_s \right) = \int d^2 \boldsymbol{\gamma} \prod_i \gamma_i^{* n_i} \gamma_i^{m_i} W(\boldsymbol{\gamma}, s). \quad (1.3.52)$$

One of the most used representations for the density operator ρ is the P -representation (due to its resemblance with its bosonic counterpart) where $s = 1$ and one defines $P(\boldsymbol{\alpha}) = W(\boldsymbol{\alpha}, 1)$. The P -representation allows one to compute expectation values as in Eq. (1.3.52) in a straight forward manner,

$$\text{tr} \left(\rho c_p^{\dagger n} c_q^m \right) = \int d^2 \boldsymbol{\gamma} P(\boldsymbol{\gamma}) \langle \boldsymbol{\gamma} | c_p^{\dagger n} c_q^m | \boldsymbol{\gamma} \rangle = \int d^2 \boldsymbol{\gamma} P(\boldsymbol{\gamma}) \gamma_p^{* n} \gamma_q^m. \quad (1.3.53)$$

1.3.3 Fermionic Gaussian states and their Grassmann representation

A Gaussian state is a state whose density matrix can be written as the matrix exponential of a quadratic Hamiltonian. A more general definition, which avoids certain issues when ρ can not be written as an exponential of an operator as it may not have full rank is given below [3].

Definition 1.3.2 (Fermionic Gaussian state). *A quantum state of N_{so} fermionic modes is Gaussian iff its density operator $\rho \in \mathcal{C}_{2N_{so}}$ has a Gaussian Grassmann representation, that is*

$$\omega(\rho, \boldsymbol{\theta}) = \frac{1}{2^{N_{so}}} e^{\frac{i}{2} \sum_{p,q=1}^{2N_{so}} \theta_p (\Gamma_m)_{pq} \theta_q} \quad (1.3.54)$$

for some real and anti-symmetric (i.e. skew-symmetric) $(2N_{so} \times 2N_{so})$ -matrix Γ_m , which is called the correlation matrix.

If $\rho \in \mathcal{C}_{2N_{\text{so}}}$ is a Gaussian state, its *correlation matrix* is given by

$$(\Gamma_m)_{pq} = \frac{i}{2} \text{tr}(\rho[A_p, A_q]), \quad (1.3.55)$$

where A_p can be either c_p or c_p^\dagger . A Gaussian state is said to be *pure* iff $\Gamma_m^T \Gamma_m = \mathbf{1}$. Under FLO, a Gaussian state remains Gaussian. This property is implicitly assumed in Chapters 2 and 3 and has been rigorously proven in Ref. [3].

The following formula, known as the *Wick formula*¹³, allows one to compute all higher order correlators through

$$i^p \text{tr}(\rho A_{a_1} A_{a_2} \cdots A_{a_{2p}}) = \text{Pf}(\Gamma_m|_{a_1, a_2, \dots, a_{2p}}), \quad (1.3.56)$$

where $1 \leq a_1 < a_2 < \cdots < a_{2p} \leq 2N_{\text{so}}$, $\Gamma_m|_{a_1, a_2, \dots, a_{2p}}$ is the $(2p \times 2p)$ -submatrix of Γ_m and Pf denotes the Pfaffian for a $(2n \times 2n)$ skew-symmetric matrix A , which is defined as

$$\text{Pf}(A) = \frac{1}{2^n n!} \sum_{\sigma \in S_{2n}} \text{sgn}(\sigma) \prod_{i=1}^n A_{\sigma_{2i-1}, \sigma_{2i}}, \quad (1.3.57)$$

where S_{2n} is the symmetric group containing $(2n)!$ elements and $\text{sgn}(\sigma)$ is the signature of σ , i.e. it gives an overall minus (plus) sign if the number of permutations is odd (even)¹⁴.

To summarize, in this section we have seen how FGS are defined in terms of fermionic operators as well as their Grassmann representation. Furthermore, FGS remain Gaussian under FLO operations and mean values of a product of fermionic annihilation and creation operators w.r.t. a physical state can be computed through Grassmann variables using e.g. Eqs. (1.3.52) or (1.3.53), or in case when ρ is a Gaussian state through Eq. (1.3.56).

¹³ *Wick's theorem* will be discussed in detail in Appendix 3.B.

¹⁴ A basic example for Eq. (1.3.56) and $p = 2$ is given by

$$i^2 \text{tr}(\rho A_1 A_2 A_3 A_4) = (\Gamma_m)_{12} (\Gamma_m)_{34} - (\Gamma_m)_{13} (\Gamma_m)_{24} + (\Gamma_m)_{14} (\Gamma_m)_{23}.$$

Section 1.4

IMAGINARY TIME EVOLUTION OF A QUANTUM SYSTEM

A quantum mechanical system is fully described by its Hamiltonian H , which is an Hermitian matrix $H^\dagger = H$ and thus possesses only real-valued eigenvalues $E_0 \leq E_1 \leq E_2 \leq \dots$. Exactly solving the time-dependent Schrödinger equation (setting here $\hbar = 1$)

$$i \frac{d}{dt} \Psi(x, t) = H \Psi(x, t) \quad (1.4.1)$$

for an initial state $\Psi(x, t = 0)$ is only possible for a handful of simple systems¹⁵. We will be interested in approximating the ground state of H , since an exact solution is generally too difficult to obtain. One powerful technique developed for evolving to the ground state is the *imaginary time propagator method*, which we will simply call *imaginary time evolution* in this thesis. Here, a Wick rotation ($t \rightarrow -i\tau$) turns Eq. (1.4.1) into [61]

$$-\frac{d}{d\tau} \Psi(x, \tau) = H \Psi(x, \tau), \quad (1.4.2)$$

which possesses the formal solution

$$\Psi(x, \tau) = e^{-H\tau} \Psi(x, \tau = 0). \quad (1.4.3)$$

The initial state $\Psi(x, \tau = 0)$ can be expanded in the basis of eigenfunctions $\phi_k(x)$ through

$$\Psi(x, \tau = 0) = \sum_k \langle \phi_k(x) | \Psi(x, \tau = 0) \rangle \phi_k(x) = \sum_k c_k \phi_k(x). \quad (1.4.4)$$

We will assume in the following, that the coefficient describing the overlap of the initial state with the ground state is non-zero. The time evolution of the diffusion equation Eq. (1.4.2) can be formally written as

$$\Psi(x, \tau) = \sum_k e^{-E_k \tau} c_k \phi_k(x). \quad (1.4.5)$$

The above state has yet to be normalized. Due to the ordering $E_0 \leq E_1 \leq E_2 \leq \dots$, for sufficiently long times, we see from Eq. (1.4.5) that we will eventually be lead to the ground state of the system Hamiltonian since all the amplitudes of higher energy states will decay more rapidly. In cases where the ground state is degenerate, e.g. $E_0 = E_1$, the solution of Eq. (1.4.5) will be a linear combination of eigenfunctions and one has to repeat the imaginary time evolution for various distinct initial state configurations multiple times to extract the vectors which span the degenerate subspace. The term *sufficiently long* which characterizes the simulation time mainly depends on the gap between the ground state and the first excited state, similar to adiabatic quantum computing [62]. One can also get trapped in local minima, which can be detected by repeating over various initial state instances and various step sizes and simulation times [63].

¹⁵See Ref. [60] for an example list of quantum systems which possess analytical solutions.

If the initial state is a FGS described by the density matrix ρ_{GS}^{16} , the imaginary time evolution of the density matrix is given by [63]

$$\rho_{\text{GS}}(\tau) = \frac{e^{-H\tau} \rho_{\text{GS}}(0) e^{-H\tau}}{\text{tr}(e^{-2H\tau} \rho_{\text{GS}}(0))}, \quad (1.4.6)$$

where the denominator takes care of the normalization. Note that the above equation will in general take us out of the setting of FGS if H includes more than just quadratic polynomials of fermionic creation and annihilation operators, which is the case for the Hamiltonians we will consider in this work, i.e. those described by $H = T + V$ with T and V given by Eqs. (1.2.13) and (1.2.16), respectively.

Recent work proposed how imaginary time evolution could be realized on a quantum computer [64, 65]. We emphasize that this is not the subject of this work, as we will perform imaginary time evolution of quantum systems on classical computers. We will give a brief introduction to the field of quantum computing in the following section.

¹⁶The time evolution of a density operator ρ is described by the *von-Neumann equation*

$$i\hbar \frac{d}{dt} \rho(t) = [H, \rho(t)].$$

For pure states, the von-Neumann equation is equivalent to the Schrödinger equation.

Section 1.5

QUANTUM COMPUTATION

In the early 1980s it was suggested that a quantum computer could enable the execution of new kinds of algorithms, so-called quantum algorithms, which are entirely different from their classical counterparts as they could exploit properties that are inherently quantum mechanical [66, 67]. The promise is that the realization of a quantum computer would provide a means to render certain problems feasible, for which (to this day) no classical algorithm is known that could solve them efficiently. While most problems are generically hard to solve even with a quantum computer [68], the problems we will be considering fall all into the small, but nevertheless extremely important class of efficiently solvable problems, which could have significant impact on the simulation of chemicals [69] and materials [70], drug design [10], breaking existing public-key cryptography [71] and machine learning [72], to name but a few. The study of capabilities and limitations of an idealized quantum computer, which Scott Aaronson eloquently described as *the study of what we can't do even with computers we don't have* [73] will however not be discussed here.

First of all, we have to clearly state what is meant by saying that a *problem is solved efficiently*. One considers a problem to be 'solved', if the solution to a problem can be obtained within some error $\epsilon > 0$, where the accuracy itself depends on the problem at hand. For example, in computational chemistry one is typically interested in numerically solving the time-independent Schrödinger equation for a number of interacting electrons and nuclei. The goal of a quantum chemistry simulation is to match or exceed the accuracy $1 - \epsilon$ that can be obtained in experiment, which for quantum chemistry calculations is around $1 \frac{\text{kcal}}{\text{mol}}$ ¹⁷ for *ab-initio*¹⁸ methods, which is also known as (thermo-)chemical accuracy [74]. It turns out that even though we know the equations that predict the behavior of atoms, molecules and materials, solving these equations within error ϵ requires an amount of classical resources (i.e. number of bits and basic logic operations for running a classical algorithm on a classical computer) that scales exponentially with problem size. Analogous to a classical algorithm, a quantum algorithm also requires a certain amount of *quantum resources* (the quantum analogues to bits and logic operations) and for certain types of problems - here the solution of the Schrödinger equation -, the quantum resources required scale much more favorable in comparison to their classical counterparts, where in some cases even an exponential reduction in resource requirements is observed [71]. If the required amount of quantum resources needed to solve a problem via a quantum algorithm running on a quantum computer scales polynomially with the size of the problem, the corresponding algorithm is considered to be *efficient* and the problem to be *efficiently solvable*.

The fact that a certain quantum algorithm shows a better scaling in terms of required resources than a classical algorithm is sometimes referred to as a *quantum speedup*. Loosely speaking, the reason for a speedup using a quantum computer over a classical machine is that in the latter case one tries to solve a quantum mechanical problem on a classical machine, which lacks the ability to make use of the key ingredients that distinguish

¹⁷In physicists units $1 \frac{\text{kcal}}{\text{mol}} \approx 0.043\text{eV}$.

¹⁸In this context, *ab-initio* methods are based entirely on quantum mechanics and physical constants, with no additional empirical data.

quantum from classical systems, namely quantum *superposition* and *entanglement*. While one could in principle easily construct a uniform superposition of an astronomically large number of states on a quantum computer - of which one will be the desired answer to the considered problem - one will have to perform a measurement of an observable at the end. Since there is an astronomically large number of possible outcomes which are all equally likely, the output is random and the quantum computer would not be any better than a random guess at a solution. There would really be no need to spend millions on designing a device whose guess would have as good of a shot at the actual solution as the infamous monkey trying his luck to reproduce Shakespeare [75]. An additional tool is required, which allows one to single out the 'right' solution from all other 'wrong' solutions. This tool is known as a *quantum algorithm*, which is a finite sequence of instructions on how to apply the available quantum resources in order to obtain the 'right' solution to a given problem within error ϵ in as few measurements and runs as possible (where a *run* refers to a single execution of the respective quantum algorithm). In order to better understand what makes a quantum computer so special, one has to understand the concept of superposition. A quantum state $|\Psi\rangle$ is in general a superposition of various states, each state $|M_i\rangle$ corresponds to a possible measurement outcome and carries a complex amplitude $m_i \in \mathbb{C}$,

$$|\Psi\rangle = \sum_i m_i |M_i\rangle, \quad (1.5.1)$$

where $\sum_i |m_i|^2 = 1$. When one is 'not looking' (so while the quantum computer is running and left to itself), the amplitudes m_i can interfere constructively or destructively with one another. The working principle of a quantum algorithm and thus of a quantum computer is to choreograph things in a way that the amplitudes of paths leading to 'wrong' solutions can destructively interfere with other amplitudes of paths which also lead to 'wrong' solutions, while the paths leading to 'right' solutions constructively interfere with one another (the *path* is here just the evolution of the respective quantum state through the quantum circuit) [1]. This explanation, while lacking any mathematical rigor, demonstrates that quantum computing is a fundamentally different way of solving certain problems by harnessing the quantum mechanical properties of nature.

While a lot of progress has been made within the last 40 years since the suggestion of building a quantum computer [76], a fault-tolerant quantum computer, i.e. a quantum computer that can run in principle arbitrarily long without accumulating any considerable error (for instance by realizing the surface code [35]), has still to be realized. Nevertheless, a recent experiment led by the Google group of John Martinis succeeded in demonstrating for the first time a proof-of-principle calculation of a very specific computational task on a 53 qubits (a *qubit* is the quantum-analog to a classical bit) superconducting quantum processor, which would take the worlds largest supercomputer a significant longer time¹⁹ to compute [31]. In addition, substantial progress has been made in finding useful applications for noisy quantum computers operating with 50-100 qubits [78]. However, most of the algorithms of the so-called noisy intermediate scale quantum (NISQ) era, such as the variational quantum eigensolver [79, 80], are heuristically motivated and do not possess provable speedup over classical methods. The claim that quantum computers could give a provable quadratic [81], or even exponential

¹⁹There is some ambiguity over how much longer a supercomputer would actually need. However, this can be considered nitpicking, since simply increasing the simulation by only a handful of qubits would render the classical simulation essentially non-simulatable [77].

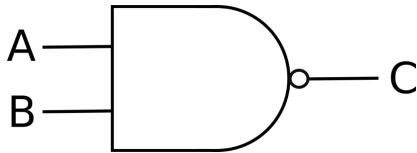


Figure 1.1: NAND gate symbol

Table 1.1: Truth table of NAND gate

A	B	C
0	0	1
0	1	1
1	0	1
1	1	0

[71] speedup over classical algorithms is still only promised for fault-tolerant quantum processors.

In the following sections, we will try to elucidate the basic working principles of a quantum computer and motivate why certain quantum algorithms require a good "starting point" (which we will later call a *reference-* or *initial state*), since improving the latter with methods that can both be *efficiently computed* on classical computers and *efficiently implemented* on quantum computers is one of the main focuses of this dissertation.

1.5.1 Quantum circuits

A classical computer is essentially an electronic circuit which realizes logical gates that implement Boolean functions, meaning that a gate performs a logical operation on one or several input bits (a bit can only be either in the state 0 or in the state 1) and outputs a single bit output, realizing the respective logical operation. There are various logical operations, e.g. an AND or OR operation, however, only a single logical (so-called universal-) gate is sufficient to realize all other possible logical gates [82]. Universal gates for Boolean logic are the NOR and NAND gates, the latter is displayed in Fig. 1.1 and its corresponding truth table is given in Table 1.1.

Similar to logical binary bits, a quantum bit (*qubit*) is one of the most simple systems to display non-trivial quantum behavior. While a classical bit can only either be in 0 or 1, a qubit - if not disturbed by any measurement - can be in the state

$$|\psi\rangle = \alpha |0\rangle + \beta |1\rangle, \quad (1.5.2)$$

where in the single qubit subspace, the canonical basis vectors are defined as $|0\rangle = (1, 0)^T$, $|1\rangle = (0, 1)^T$ and $\alpha, \beta \in \mathbb{C}$ are complex quantum amplitudes with $|\alpha|^2 + |\beta|^2 = 1$. In order to experimentally determine the quantum state $|\psi\rangle$ in Eq. (1.5.2), one would have to perform a series of measurements in the computational basis, which is defined as $\{|0\rangle, |1\rangle\}$. Each measurement will output the value 0 or 1 with a probability $|\alpha|^2$ or $|\beta|^2$, respectively. The larger the number of measurements, the more accurate will be the determination of the quantum amplitudes and thus of the single qubit state $|\psi\rangle$. Similarly, for two qubits, a general state will be of the form²⁰

$$|\psi\rangle = \alpha |00\rangle + \beta |01\rangle + \gamma |10\rangle + \delta |11\rangle, \quad (1.5.3)$$

²⁰In quantum computing one uses the convention to write tensor products of quantum states as regular products, i.e.

$$|i\rangle \otimes |j\rangle = |i\rangle |j\rangle = |ij\rangle,$$

where i, j are single qubit states of the form described in Eq. (1.5.2) and the left state belongs to the first qubit and the right state belongs to the second qubit. A similar convention will later be used for the tensor product of quantum operators.

where $\alpha, \beta, \gamma, \delta \in \mathbb{C}$ are complex quantum amplitudes with $|\alpha|^2 + |\beta|^2 + |\gamma|^2 + |\delta|^2 = 1$, thus containing four computational basis states. This can be extended to $n \in \mathbb{N}$ qubits, for which the number of basis states is given by 2^n . The span of that basis describes the Hilbert space of an n qubit quantum computer. Similar to logical operations such as the NAND gate depicted in Fig. 1.1, one can perform operations on the qubit states, which are required to be norm-preserving and thus unitary. The most prominent examples of single qubit gates are given by the Pauli-matrices²¹, which are defined as [83]

$$\sigma^x = \begin{pmatrix} 0 & 1 \\ 1 & 0 \end{pmatrix}; \quad \sigma^y = \begin{pmatrix} 0 & -i \\ i & 0 \end{pmatrix}; \quad \sigma^z = \begin{pmatrix} 1 & 0 \\ 0 & -1 \end{pmatrix}, \quad (1.5.4)$$

where the last matrix (the matrix is the operator representation of the desired operation in the computational basis) is diagonal in the computational basis. Other important single qubit gates include the Hadamard gate H , phase gate S and T gate,

$$H = \frac{1}{\sqrt{2}} \begin{pmatrix} 1 & 1 \\ 1 & -1 \end{pmatrix}; \quad S = \begin{pmatrix} 1 & 0 \\ 0 & i \end{pmatrix}; \quad T = e^{i\frac{\pi}{8}} \begin{pmatrix} e^{-i\frac{\pi}{8}} & 0 \\ 0 & e^{i\frac{\pi}{8}} \end{pmatrix}. \quad (1.5.5)$$

As a simple example, the action of each operator on a qubit state can be computed through simple linear algebra, e.g. for the single qubit state $|\psi\rangle$ in Eq. (1.5.2), we have $\sigma^x |\psi\rangle = \beta |0\rangle + \alpha |1\rangle$.

There also exist operations which act simultaneously on several qubits, which are known multi-qubit gates or operators. A simple example of such an operator is the controlled-NOT operation (CNOT), which applies the σ^x operator on the *target* qubit if the *control* qubit is in the state $|1\rangle$ and acts as the identity if the *control* is in the state $|0\rangle$. For instance, if the target qubit is in the state $|\psi\rangle$, applying the CNOT gate will act as $\text{CNOT} |\psi\rangle |0\rangle = |\psi\rangle |0\rangle$, or $\text{CNOT} |\psi\rangle |1\rangle = (\beta |0\rangle + \alpha |1\rangle) |0\rangle$. An important extension of the CNOT-gate to gates with multiple control qubits will be discussed in Chapter 4. The extension of the CNOT to two control qubits and one target qubit is known as a Toffoli gate, which applies σ^x on the target qubit only if both control qubits are in the state $|1\rangle$ and else acts as the identity²².

Similar to the way a classical computer is build from an electrical circuit of wires and logical gates as depicted in Fig. 1.1, a quantum computer is build from a *quantum circuit*, where wires represent qubits, single- and multi-qubit gates are indicated by square boxes labeled with an appropriate symbol to indicate which operator it represents (e.g. a Hadamard gate will carry a box with H written on it) and there also exist special symbols for frequently appearing operations, such as the CNOT gate. The size of a quantum circuit is characterized mainly by two measures, namely the circuit *length* and *depth*. The former describes the number of qubits appearing in a given quantum circuit (essentially the number of horizontal wires) and the latter is the number of non-commuting gates which appear in a given circuit. For instance, a circuit containing only two qubits that applies a H gate on qubit 1 and a S gate on qubit 2 has depth 1, since the single qubit gates operate on distinct qubits we have $[H_1, S_2] = [H \otimes \mathbb{1}_2, \mathbb{1}_2 \otimes S] = 0$, where we defined

$$\mathbb{1}_2 = \begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix},$$

²¹Different conventions exist for denoting Pauli gates. We will use either σ_j^x or X_j for a Pauli-X gate (Pauli-Y and -Z gates are similarly defined), where j is the qubit index.

²²A truth-table for the Toffoli gate, which is a CNOT gate with two control qubits, is given in Table 4.4.1

and therefore the single qubit gates can be executed simultaneously. A simple example of a quantum circuit of length 3 and depth 13 which realizes the Toffoli gate from the gate set CNOT, H , S , T is given in Fig. 1.2.

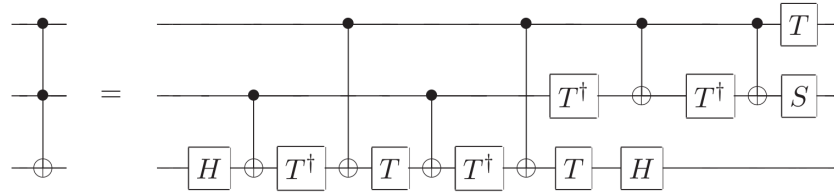


Figure 1.2: Quantum circuit realizing a Toffoli gate from CNOT, H , S , T gates, where the CNOT gate is represented by a filled and empty circle (indicating the control and target qubit) connected by a horizontal line. The \dagger symbol indicates the adjoint of the respective operator. The circuit depicted here has a *circuit depth* 13 and *circuit length* 3.

In order to extract information about the output of a quantum circuit, i.e. the quantum computation, one has to measure the state of (not necessarily all) the qubits after going through the circuit. There are various types of ways to measure a quantum state and the experimental realization strongly differs for the various species of quantum computers [84]. For our purposes, it is sufficient to note that measurements which project onto the computational basis are denoted in quantum circuits with a 'meter' symbol (see Fig. 1.4) and the outcome will tell us whether the state was $|0\rangle$ or $|1\rangle$.

Time runs from left to right in a quantum circuit. Regarding the example circuit in Fig. 1.2, this means that the first operator acting on a three qubit state is a Hadamard gate acting on the third qubit. The way of reading a quantum circuit is thus opposite to the way one would write the operators acting on a quantum state, e.g. in a circuit realizing $HS|\psi\rangle$, the rightmost gate (we use the expressions 'gate' and 'operator' interchangeably) S would be all the way on the left in the corresponding quantum circuit. This closes the discussion of quantum circuits. A quantum computer is a machine that can execute quantum circuits with low error. While small quantum circuits can be executed on current quantum hardware, the more the circuits grow, the more demanding it is to execute them with low error and one quickly reaches a circuit size that can no longer be executed reliably on today's error-prone quantum processors. Somewhat counter intuitively, the next section will show that in order to run large scale quantum circuits, it is not necessary to produce perfect qubits, it suffices to have many imperfect qubits instead.

1.5.2 A fault-tolerant quantum computer

This thesis considers an *idealized quantum computer*, arguably the most frustrating assumption for anyone who is trying to run an actual experiment. An idealized quantum computer performs state initialization, gates and measurements without any errors or losses and is perfectly isolated from the environment. While this seems to be the somewhat most unrealistic assumption one can make, it turns out that using *quantum error-correction* one can not only protect stored and transmitted quantum states, but even protect quantum states which dynamically undergo a quantum computation. This is the content of the following theorem, which we state due to its significance for quantum computing.

Theorem 1.5.1 (Threshold theorem for quantum computing). *A quantum circuit consisting of a polynomial number $P(n)$ of quantum gates may be simulated within error ϵ using $O(\text{poly}(\log(P(n)/\epsilon))P(n))$ gates on quantum hardware whose components fail with probability at most p , provided that p stays below a certain constant threshold $p < p_{\text{th}}$ and given reasonable assumptions about the noise in the hardware.*

This theorem states that if you stay below the threshold p_{th} you're correcting errors faster than they're created. That's the whole point, and the whole non-trivial thing that the theorem shows [85]. It is extremely difficult to boil down what the value of p_{th} would have to be exactly, as it depends on the quantum error-correction code one employs, hardware-specific noise models, etc. Having reached the goal of *quantum supremacy*, Google and other leading groups in the field of quantum computing are now aiming to realize a quantum error-corrected qubit by implementing the *surface code*, where one estimates $p_{\text{th}} \approx 0.01$ and one would need anywhere between $10^3 - 10^4$ physical qubits to implement one logical (error-corrected) qubit [35]. One thus distinguishes between *logical* and *physical* qubits, where the latter is the total number of required qubits (most of which are consumed for quantum error-correction) and the latter is the number of 'perfect' qubits available for the actual computation. So even though our main assumption - the existence of an idealized quantum computer - is not realizable, we can approximate an idealized quantum computer in an efficient way by a fault-tolerant (i.e. error-corrected) quantum computer which is comprised of many 'good' but nevertheless error-prone qubits. DiVincenzo introduced the following five criteria that an experimental setup would have to satisfy in order to be able to serve as such 'good' qubits and be able to implement a quantum algorithm [86].

1. Well characterized qubits, i.o.w. qubits that are described by two eigenstates $|0\rangle, |1\rangle$, separated by a non-vanishing energy gap and almost always stay within their two-level subspace (no leakage to other levels). These qubits need to be implemented in a scalable architecture.
2. The qubits can reliably initialized to a fiducial- (or reference-) state. In our work, we assume that one is able to produce the vacuum state where all qubits are initialized to $|0\rangle$, which can be realized for instance by quantum non-destructive projective measurements or quantum annealing (which we will not discuss in this work).
3. Often times theorists treat a quantum computer as an isolated system. This is of course not a valid assumption and interactions with the environment and leakage out of the computational subspace destroy the definite phase relations between the different states describing the wave function of the quantum computer. These phase relations are essential for quantum computing as they allow for superposition and entanglement. The loss of a definite phase relation is known as quantum *decoherence* and an experimental setup should produce qubits where the phase relations survive as long as possible, in order for quantum error-correction schemes to be applicable.
4. A quantum computer has to be able to approximate any unitary operator within its Hilbert space. In order to do so, it needs to be capable of implementing a (finite) *universal set of quantum gates*, which is a set of gates that can approximate any unitary of the $(2^n \times 2^n)$ -dimensional Hilbert space, see Section 1.5.3.
5. One has to be able to read out the state of a qubit (in e.g. the computational basis) at the end or even in between the computation.

The five points above is a check list that a working quantum computer has to complete in order to implement a quantum circuit and thus perform a quantum computation, which is in general a unitary operation on the Hilbert space of the system (ignoring measurement operations for the sake of simplicity). The following section introduces a key result for quantum computation, namely that any unitary operation²³ can be approximated to arbitrary precision on a quantum computer that satisfies DiVincenzo's criteria, however realizing the circuit that approximates the unitary operator may be exponentially costly in terms of quantum resources (e.g. the number of gates).

1.5.3 Approximating unitary operators

Similarly to the classical NAND or NOR gate being universal, there exist sets of gates which are termed *universal for quantum computing*, meaning that any unitary operator may be approximated to arbitrary accuracy by a quantum circuit composed only of gates from that (finite) gate set. What does it mean that a unitary operator is *approximated* by another unitary? Following the convention of Ref. [6], if U is the target unitary we wish to implement and let V be the unitary that one is able to implement on a quantum computer which is supposed to approximate U , we define the error that is caused by the approximation as

$$E(U, V) = \max_{|\psi\rangle} \sqrt{\langle\psi| (U - V)^\dagger (U - V) |\psi\rangle}, \quad (1.5.6)$$

where the maximum is over all normalized quantum states $|\psi\rangle$. If P_U (P_V) is the probability of a measurement if U (V) were applied on a state $|\psi\rangle$, one can show that

$$|P_U - P_V| \leq 2E(U, V). \quad (1.5.7)$$

This means that if E is small, the measurement statistics of $U|\psi\rangle$ and $V|\psi\rangle$ for any $|\psi\rangle$ is approximately the same. This can also be extended to a product of m unitaries $U_m U_{m-1} \cdots U_1$ (remember that the right-most operator is applied first), where each U_j is approximated by V_j , the errors add at most linearly

$$E(U_m U_{m-1} \cdots U_1, V_m V_{m-1} \cdots V_1) \leq \sum_{j=1}^m E(U_j, V_j). \quad (1.5.8)$$

From Eqs. (1.5.7)-(1.5.8) it immediately follows that for a circuit of m gates U_j , where each gate U_j is approximated by V_j , in order to be approximated within error $\Delta > 0$ of the correct measurement statistics, it suffices that $E(U_j, V_j) \leq \Delta/(2m)$. Now that we have introduced the notion of approximating unitaries, we can put it in the context of quantum computing.

We will see that in general, the number of required gates from the universal gate set to approximate a unitary operator to accuracy ϵ will scale exponentially with the number of qubits involved. One of the goals of quantum computing is to find useful families of unitary operators which can be approximated to arbitrary accuracy using only polynomial resources and thus *efficiently*. It is evident that an exact decomposition of a unitary operator in terms of the universal gate set is in general impossible, since the universal gate set contains only a finite number of gates, but unitary operations are continuous.

²³By *any unitary operation* we restrict ourselves to those operations which lie in the $(2^n \times 2^n)$ -Hilbert subspace of a quantum computer with n logical qubits.

- As a first property of unitary matrices, we note that an arbitrary unitary matrix U acting on n qubits can be written as a product of at most $2^{n-1}(2^{n-1} - 1)$ two-level unitary matrices, where a two-level unitary matrix is defined as a matrix that acts non-trivially (i.e. not as the identity) only on two-or-fewer vector components.
- An arbitrary two-level unitary operation on the state space of n qubits can be implemented using $O(n^2)$ single qubit and CNOT operations²⁴.

Combining the two properties above shows that any unitary operation U can be implemented exactly using $O(n^2 4^n)$ single qubit and CNOT operations. This exponential scaling is a first hint as to why finding efficient quantum algorithms (i.e. special-purpose quantum circuits) is such a difficult endeavor and why until this day, there are only a handful of efficient quantum algorithms that could in principle outperform all known classical algorithms designed for that very same specific task. The following property shows that two gates are sufficient to approximate any single-qubit gate.

- Any single qubit gate can be approximated to arbitrary accuracy using the set $\{H, T\}$.
- The set $\{H, T, \text{CNOT}\}$ is sufficient to approximate any unitary U to accuracy ϵ . The addition of the phase gate S makes the construction fault-tolerant, which is why one considers $\{H, T, S, \text{CNOT}\}$ as a *universal gate set* for quantum computing.

One of the most central theorems of quantum computing, the *Solovay-Kitaev theorem*, states that an arbitrary single qubit gate may be approximated within an error ϵ using $O(\log^c(1/\epsilon))$ gates from a discrete gate set, where c is a constant close to 2 [87]. In addition, the approximation of a quantum circuit containing m CNOT and m single qubit gates to an accuracy ϵ can be achieved using $O(m \log^c(m/\epsilon))$ gates. Unfortunately, as we have seen in the previous paragraph, the scaling of the number m with system size is almost always exponential, since combining the Solovay-Kitaev theorem and the universality construction to approximate an arbitrary unitary operator U on n qubits within error ϵ will require $O(n^2 4^n \log^c(n^2 4^n / \epsilon))$ gates from the universal gate set.

The fact that a quantum computer can approximate any unitary operator makes it in some sense superior to most other *quantum simulators*. The latter are quantum systems whose Hamiltonian can be controlled within the experiment and one tries to shape it in a way that simulates (most of the times only specific parts of) the physics of the quantum system of interest. Quantum simulators do not have to be universal. They have been realized in various platforms, including trapped ions [88], ultracold quantum gases [89], photonic systems [90], quantum dots [91] and superconducting qubits [92]. Quantum simulation of (non-classical) physical systems presents one of the most useful applications of the quantum circuit model and will be introduced more thoroughly in the following sections.

²⁴We use the notation $O(f(x))$ to denote the limiting behavior of a function $f(x)$ for large values of x

Section 1.6

 QUANTUM SIMULATION

The following sections will give a brief glimpse into one of the predicted "killer applications" of quantum computing, the digital quantum simulation of strongly interacting quantum systems²⁵, whose properties can no longer be described adequately by simulations on classical computers²⁶.

The postulates of quantum mechanics state that every quantum system is fully characterized by its wave function $|\Psi\rangle$ whose time evolution is described by the Schrödinger equation²⁷

$$i\frac{d}{dt}|\Psi\rangle = H|\Psi\rangle. \quad (1.6.1)$$

It is a linear equation and—in the non-relativistic limit—completely describes all atomic and molecular systems [2]. For strongly interacting systems, one is facing an exponential number of coupled differential equations where all classical approximations to reduce the number of equations so far have failed. As an example of the importance of the above equation, we consider chemistry. A large part of theoretical chemistry's effort is directed to finding approximate methods to solve Eq. (1.6.1) [2, 44]. More precisely, in chemistry one is interested in finding the lowest energy configuration of electrons in presence of a given nuclear configuration, i.o.w. the *ground state* of the so-called electronic structure Hamiltonian. A high precision estimate of the ground state is needed in order to be able to predict chemical reaction rates, which govern the mechanism of chemical reactions. For instance, the impact of solving the above equation for certain parts of an enzyme could potentially lead to an understanding of how to produce fertilizers under ambient condition, which is currently achieved under high temperatures and pressures using an extremely energy consuming industrial process. While classical methods resoundingly fail at this task, a fault-tolerant quantum computer with around 150-200 logical qubits is expected to solve this task in reasonable time [95, 96].

The formal solution of Eq. (1.6.1) is extremely simple, in fact it is just given by

$$|\Psi(t)\rangle = e^{-iHt}|\Psi(t=0)\rangle, \quad (1.6.2)$$

but actually computing the resulting wave function is in general impossible as it involves the matrix exponentiation of an exponentially large matrix. We note that the operator $U = e^{-iHt}$ on the right-hand side of Eq. (1.6.2) is unitary. Just as we have seen in Section 1.5.3 that not all unitaries can be approximated by a quantum computer, not all Hamiltonians H appearing in the argument of the matrix exponential can be simulated

²⁵A *digital* quantum simulation is a quantum simulation on a device with a universal set of quantum gates, where a proposed quantum algorithm is executed by applying a finite sequence of one- and two-qubit gates in order to approximate the desired unitary operation. Most realizations of quantum simulation are however analog, see [93]. Proposals exist to combine digital with analog quantum computing [94].

²⁶As we will later see, there exist quantum systems which can be simulated efficiently using classical computers, for instance systems whose ground state is described by a Gaussian state.

²⁷Throughout this thesis, we will only consider the *time-independent* Schrödinger equation, where the system Hamiltonian $H(t) = H$ does not depend on time and we will often set \hbar to 1.

efficiently. We will expand a bit more on the topic of which kind of Hamiltonians can actually be simulated efficiently in Section 1.6.3. One of the biggest discoveries within the field of quantum algorithms was that a quantum algorithm exists, which can efficiently solve Eq. (1.6.1), i.e. it can compute the eigenenergies of a certain class of Hamiltonians H and how they evolve in time using the *phase estimation algorithm*, providing an *exponential speedup* over all known classical algorithms.

In order to explain the underlying algorithm, we will introduce the *quantum Fourier transform* in Section 1.6.1 before getting to the actual algorithm in Section 1.6.2 and introducing the current state-of-the-art method to realize the required time evolution operator $U = e^{-iHt}$ through the *linear combination of unitaries method* in Section 1.6.3.

1.6.1 Quantum Fourier transformation

The quantum Fourier transformation, denoted by the symbol FT, is a unitary operation whose action on a state $|j\rangle = |j_1 j_2 \dots j_n\rangle$ from the orthonormal basis set comprised of the Fock states $\{|0\rangle, |1\rangle, \dots, |N-1\rangle\}$, where $N = 2^n$ is a fixed positive integer²⁸, is given by [71, 97, 98]

$$\text{FT} |j\rangle = \frac{1}{\sqrt{N}} \sum_{k=0}^{N-1} e^{2\pi i j k / N} |k\rangle \tag{1.6.3}$$

$$= \frac{1}{2^{n/2}} \left(|0\rangle + e^{2\pi i 0 \cdot j_n} |1\rangle \right) \left(|0\rangle + e^{2\pi i 0 \cdot j_{n-1} j_n} |1\rangle \right) \dots \left(|0\rangle + e^{2\pi i 0 \cdot j_1 j_2 \dots j_n} |1\rangle \right), \tag{1.6.4}$$

where we used binary representation in the exponent. Eq. (1.6.3) underscores the close resemblance to the common Fourier transformation, while the identical expression in Eq. (1.6.4) makes it easier to derive the quantum circuit that realizes FT using only unitary gates, from which it immediately follows that FT itself must be unitary. One of the central gates to the FT is the controlled operation of the following operator,

$$R_k = \begin{pmatrix} 1 & 0 \\ 0 & e^{2\pi i / 2^k} \end{pmatrix}. \tag{1.6.5}$$

In Fig. 1.3 we give an explicit quantum circuit realizing FT using $\Omega(n^2)$ elementary gates²⁹ and follows from Eq. (1.6.4).

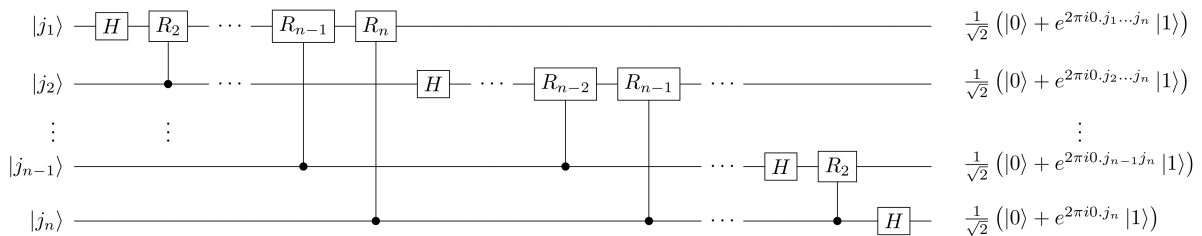


Figure 1.3: Efficient quantum circuit depicting the quantum Fourier transformation. This circuit follows from the representation of $\text{FT} |j\rangle$ in Eq. (1.6.4). The controlled-operation is given by Eq. (1.6.5). Not shown in the above circuits are a series of SWAP gates which reverse the order of the qubits at the end of the computation.

²⁸ N must not necessarily be a power of 2, however it simplifies notation.

²⁹A scaling $\Omega(g(n))$ means that the respective function is both upper and lower bounded by $g(n)$.

1.6.2 Quantum phase estimation

The FT is utilized as a subroutine in Quantum Phase Estimation (QPE), a quantum algorithm designed to find the phase φ (w.l.o.g. we let $\varphi = 0.\varphi_1\varphi_2\dots\varphi_n$ in binary representation) of the eigenvalue $e^{2\pi i\varphi}$ of a unitary operator U , that has a *provable* exponential speedup over all known classical algorithms. A *subroutine* (often times also called a *black box* or *oracle*) is a task-specific operation for which a quantum circuit either already exists, or is assumed to exist, and its respective explicit quantum circuit is often times just replaced by a box to simplify the circuit representation. QPE is itself a subroutine, requiring the existence of two oracles, one that efficiently prepares the eigenstate $|u\rangle$ of U and one that implements controlled- U^j operations, where U is applied j -times (j is a positive integer) if the control qubit is in the state $|1\rangle$.

The QPE algorithm is depicted in Fig. 1.4. QPE requires two *registers*, where a register can be thought of as a group of qubits dedicated to accomplish the same task. The first register contains t qubits and sets the accuracy (the number of digits) for estimating the phase φ . The second register contains the eigenstate $|u\rangle$ of U . We will not go into too many details of the algorithm, i.e. performing a step-by-step analysis of the evolution of the quantum state of the combined register state, but only briefly describe the main steps.

1. Create the state $|u\rangle$ by applying an appropriate state generation circuit U_{init} on the vacuum state,

$$|\Psi_{\text{init}}\rangle = U_{\text{init}}|0\rangle = |u\rangle. \quad (1.6.6)$$

2. Now create the state

$$\frac{1}{2^{\frac{n}{2}}} \left(|0\rangle + e^{2\pi i 0.\varphi_t} |1\rangle \right) \left(|0\rangle + e^{2\pi i 0.\varphi_{t-1}\varphi_t} |1\rangle \right) \dots \left(|0\rangle + e^{2\pi i 0.\varphi_1\varphi_2\dots\varphi_t} |1\rangle \right) |u\rangle \quad (1.6.7)$$

by applying a series of Hadamard gates followed by controlled- U^j operations. Note that since $|u\rangle$ is an eigenstate of U , it remains invariant throughout the whole computation. This operation is known as a *phase-kickback*, since information about the second register is somehow transferred to the first register, without altering the state of the second register.

3. Apply the FT[†] to encode the phase information into the state. This will result in

$$|\varphi_1\varphi_2\dots\varphi_t\rangle. \quad (1.6.8)$$

4. Measure the first register in the computational basis to read out the phase information of the state in Eq. (1.6.8).

If φ may be expressed exactly with t bits (assuming that $|u\rangle$ is an eigenstate of U), then measuring the final state in Eq. (1.6.8) will give us the phase φ exactly. In general, φ may not be exactly expressed with t gates. One can show that if one wants to *successfully obtain φ accurate to n bits of precision with probability of success at least $1 - \epsilon$* , one has to choose [6]

$$t = n + \lfloor \log(2 + 1/(2\epsilon)) \rfloor, \quad (1.6.9)$$

where $\lfloor \cdot \rfloor$ denotes the floor function.

We will now loosen the restriction that we have to be able to prepare the exact eigenstate $|u\rangle$ of U and assume that we can efficiently prepare some other state $|\Psi_{\text{init}}\rangle$ which can be expanded in the basis of eigenstates of U ,

$$|\Psi_{\text{init}}\rangle = \sum_u c_u |u\rangle, \quad (1.6.10)$$

where $\sum_u |c_u|^2 = 1$.

The key observation here is that the QPE algorithm can still be applied to the initial state described by Eq. (1.6.10), but will now give us information about the phases of all eigenstates that have a non-vanishing amplitude $c_u \neq 0$. Running the QPE algorithm with an input state $|0^t\rangle |\Psi_{\text{init}}\rangle$ will output a state close to $\sum_u c_u |\tilde{\varphi}_u\rangle |u\rangle$, where $\tilde{\varphi}_u$ is a good approximation to the phase φ_u . Reading out the first register will result in a good estimate of φ_u with probability $|c_u|^2$, since we have not prepared an exact eigenstate, but rather the weighted superposition given by Eq. (1.6.10). This leads to the following corollary, which is the main motivation for finding 'good' initial/reference states $|\Psi_{\text{init}}\rangle$ which are both *efficiently computable* on classical computers and *efficiently preparable* on either NISQ or fault-tolerant quantum computers.

Corollary 1.6.0.1 (Preparing 'good' initial states). *Given an input state $|\Psi_{\text{init}}\rangle$ as given by Eq. (1.6.10), if the number of qubits t in the first register is chosen according to Eq. (1.6.9), then the probability p_{φ_u} of measuring the phase φ_u accurate to n bits of precision at the conclusion of the algorithm is at least*

$$p_{\varphi_u} = |c_u|^2(1 - \epsilon). \quad (1.6.11)$$

We have assumed a couple of subroutines in the QPE algorithm, most notably an oracle which efficiently implements the (controlled) unitary U and one which prepares a good initial state $|\Psi_{\text{init}}\rangle$. In context of digital quantum simulation of physical systems, the former oracle will have to generate the time evolution operator $U = e^{-iHt}$, while the latter has to find an initial state with sufficient support on e.g. the ground state of the system. In the following section we will describe the currently most efficient quantum algorithm to accomplish the implementation of $U = e^{-iHt}$ which we employ in Chapter 2, whereas the issue of finding $|\Psi_{\text{init}}\rangle$ with an algorithm that can be *executed* efficiently on *classical computers* and whose result can efficiently be *implemented* on a *quantum computer* will be discussed in Chapters 2 and 3.

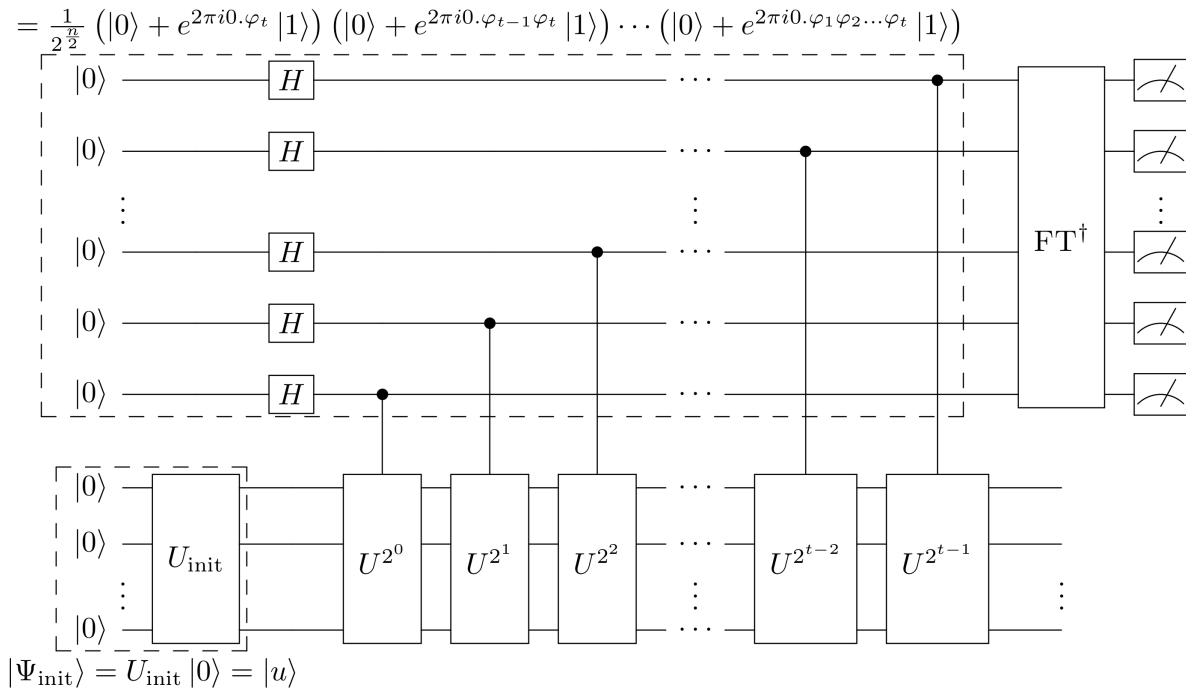


Figure 1.4: Quantum circuit depicting the QPE algorithm. The upper block contains the first register of t qubits, while the lower block is the second register containing as many qubits as necessary to store the state $|\Psi_{\text{init}}\rangle = |u\rangle = U_{\text{init}}|0\rangle$, or more generally the state described by Eq. (1.6.10). The inverse quantum Fourier transformation FT^\dagger is represented by a blackbox, whose specific form is given by the adjoint circuit of Fig. (1.3). We have highlighted two parts of the QPE circuit, the upper dashed box represents the part which realizes the phase-kickback, i.e. it encodes the phase information about $|u\rangle$ in the ancillary system (the first register). The lower dashed box highlights the operation which generates the initial state $|u\rangle$ out of the vacuum.

1.6.3 Hamiltonian simulation through linear combination of unitaries

In this section, we will give an introduction to the field of Hamiltonian simulation, more specifically, to the Linear Combination of Unitaries (LCU) method, which currently possesses the lowest T gate complexity³⁰ for simulating the time evolution (i.o.w. the black box unitary that realizes $U = e^{-iHt}$ in the QPE of Fig. 1.4) of electronic structure-type Hamiltonians [100]. The content of this subsection is mainly based on the work presented in Ref. [101]³¹. For a list of references of the field see e.g. Ref. [102], which also includes explicit quantum circuits for the various types of oracles—such as the PREPARE subroutine—which we assume in Section 2.3. An in-depth treatment of Hamiltonian sim-

³⁰A quantum algorithm's complexity for fault-tolerant quantum computing is typically stated in terms of T gates, which are single qubit gate of the form given by the last expression in Eq. (1.5.5). The reason for this is that the realization of a single T gate consumes hundreds of thousands of physical qubits and takes significantly longer than any other operation [99]. So even though the number of T gates might have a lower scaling than other quantum gates, they still usually are the most expensive ones to execute.

³¹All definitions, theorems, corollaries and lemmas within Section 1.6.3 are taken from Ref. [7] with kind approval of Robin Kothari.

ulation would go way beyond the scope of this thesis. In this section we will use the notation of Ref. [7], which will differ slightly from the notation used in Chapter 2, but the connection between them will become obvious from context. In Section 1.6.3, we shortly introduce the qubitization algorithm [100, 103, 104] for estimating the ground state energy of a Hamiltonian on an error-corrected quantum computer.

One of the central problems studied in quantum information science is to find quantum algorithms and their respective complexities that efficiently implement the unitary evolution of a quantum state³². The number of known quantum algorithms that give an improvement in complexity over known classical algorithms is very small. This can be traced back to the fact that it is generally much harder to have intuition about algorithms based on quantum mechanics than over algorithms based on classical mechanics. Furthermore, once a quantum algorithm has been found that theoretically allows to carry out a certain task, it has to compete against all known classical algorithms, proving that it outcompetes every single one of them (and even combinations thereof). Manin and Feynman were among the first to argue that a quantum computer would allow one to overcome the exponential growth of the Hilbert space and thus provide us with a means to simulate certain classes of Hamiltonians [66, 67], a conjecture that was proven to be correct by Lloyd [105], who provided an efficient algorithm to simulate Hamiltonians of the form $H = \sum_{j=1}^{\eta} H_j$ through application of the Lie-product formula $e^{-iHt} \approx \left(e^{-iH_1 t/r} \dots e^{-iH_{\eta} t/r} \right)^r$ for large³³ positive integers r [106]. More precisely, Lloyd considered only Hamiltonians which can be written as a sum of tensor products that act non-trivially—i.o.w. not as the identity $\mathbb{1}$ —only on a small number k of subsystems, known as k -local Hamiltonians. If a Hamiltonian has at most d nonzero entries in any row or column, it is said to be d -sparse. k -local Hamiltonians are sparse. In the sparse Hamiltonian simulation problem, one is given access to the d -sparse Hamiltonian H that acts on n qubits. A black box (oracle) takes as input the row index i and a number $j = 1, \dots, d$ and returns the position and the value of the j -th nonzero entry of H in row i .

The objective of Hamiltonian simulation is to construct a circuit which implements the unitary time evolution e^{-iHt} with error at most $\epsilon > 0$ using as few queries to H as possible (see Section 1.5.3 for the definition of error in context of approximating a unitary operator). In addition, the number of additional two-qubit gates should be upper-bounded, mainly for practical purposes regarding experimental realization. The total number of queries to the oracle and additional two-qubit gates gives the time complexity of the Hamiltonian simulation. While there are subtle differences in the various approaches to Hamiltonian simulation, all state-of-the-art algorithms employ the linear combination of unitaries (LCU) which strictly improve all previous approaches based on product formulas. We will give an introduction to LCU in the following. We will state the most relevant definitions, theorems and lemmas required for the simulation of the time evolution operator e^{-iHt} of a sparse Hamiltonian operator H using the truncated Taylor expansion LCU method as introduced in Ref. [101] that forms the basis of Sections 2.3 and 2.B.

³²Within the context of quantum information science, *complexity* often refers to the quantum query complexity, where the respective algorithm only obtains information about the input by querying an oracle and evolving the quantum state by successively querying said oracle. Quantum query complexity is then defined as the smallest number of times the oracle has to be used in order to compute the desired function to a given precision $\epsilon > 0$ and it provides a lower bound on the overall time complexity required to calculate that function.

³³The Lie-product formula becomes exact in the limit $r \rightarrow \infty$.

Preliminaries

The LCU algorithm's objective is to implement a unitary V which can be written as a linear combination of efficiently³⁴ implementable unitary gates U_i , $V = \sum_i a_i U_i$. We require the set of unitaries U_i to be efficiently implemented in superposition, i.e. the map $U = \sum_i |i\rangle \langle i| \otimes U_i$ is efficiently implementable. It maps the state $|i\rangle |\psi\rangle$ to $|i\rangle U_i |\psi\rangle$. The coefficients a_i are assumed to be real and positive, any phases can be absorbed in U_i . The LCU method also requires a second type of operation, which maps the state $|0^m\rangle$ (all m qubits in the register are in the state $|0\rangle$) to $1/\sqrt{a} \sum_i \sqrt{a_i} |i\rangle$, where $a = \sum_i |a_i|$. These two maps will later be referred to as the SELECT and PREPARE operations. Efficient quantum circuits for the implementation of SELECT and PREPARE are given in Ref. [102] and will not further be discussed. In the following we will give a short motivation of the working principles behind the LCU method. First, we define what is meant by a p -implementation of an operator.

Definition 1.6.1 (p -implementation). *Let V be an operator acting on a total number of n qubits and let $p \geq 0$. The unitary W acting on $m + n$ qubits p -implements V , if for all n -qubit states $|\psi\rangle$, we have*

$$W |0^m\rangle |\psi\rangle = \sqrt{p} |0^m\rangle V |\psi\rangle + |\Phi^\perp\rangle, \quad (1.6.12)$$

where $|\Phi^\perp\rangle$ is an unnormalized state satisfying $(|0^m\rangle \langle 0^m| \otimes \mathbb{1}_{2^n}) |\Phi^\perp\rangle = 0$.

By measuring the first register, one can then simply check whether one has exactly performed V , in which case all qubits will have to be in the state $|0\rangle$.³⁵ The following lemma shows that any operator V that can be expressed as a LCU can be p -implemented by a simple quantum circuit.

Lemma 1.6.1 (p -implementation of V). *Let $V = \sum_i a_i U_i$ be a LCU with $a_i > 0 \forall i$ and A be the unitary matrix that maps $|0^m\rangle$ to $1/\sqrt{a} \sum_i \sqrt{a_i} |i\rangle$, where $a = \sum_i |a_i|$ and let $U \equiv \sum_i |i\rangle \langle i| \otimes U_i$ and $p = 1/a^2$. Then $W \equiv A^\dagger U A$ satisfies Eq. (1.6.12) for all states $|\psi\rangle$ and the unnormalized state $|\Phi^\perp\rangle$ satisfies $(|0^m\rangle \langle 0^m| \otimes \mathbb{1}_{2^n}) |\Phi^\perp\rangle = 0$.*

The first term on the right-hand side of Eq. (1.6.12) can be considered as the 'good' state, i.e. the one that one wishes to implement. Berry et al. [101] introduced a method that was able to boost the amplitude of the 'good' state by means of *oblivious amplitude amplification*. Essentially, given a unitary W that p -implements a unitary V following Eq. (1.6.12), oblivious amplitude amplification provides a $p' > p$ implementation of V . What distinguishes oblivious amplitude amplification from regular amplitude amplification (see e.g. [107]) is that one has to be able to reflect about the state $|\psi\rangle$ in the latter, which cannot be done since $|\psi\rangle$ is unknown, whereas oblivious amplitude amplification requires no prior knowledge about $|\psi\rangle$.

Lemma 1.6.2 (Oblivious amplitude amplification). *Let W (V) be a unitary matrix which acts on $n + m$ (n) qubits and let $\theta \in (0, \pi/2)$. For any $|\psi\rangle$, we let*

$$W |0^m\rangle |\psi\rangle = \sin(\theta) |0^m\rangle V |\psi\rangle + \cos(\theta) |\Phi^\perp\rangle, \quad (1.6.13)$$

³⁴Where *efficiently implementable* here again refers to its respective quantum circuit scaling at most polynomially in circuit size and depth (time) with system size.

³⁵When V is unitary, p can be interpreted as a probability, however, if V is not unitary, p can be larger than 1.

where $|\Phi^\perp\rangle$ satisfies $\Pi|\Phi^\perp\rangle = 0$ with $\Pi \equiv (|0^m\rangle\langle 0^m| \otimes \mathbf{1}_{2^n})$. Define $R \equiv 2\Pi - \mathbf{1}_{2^{n+m}}$ and $S \equiv -WRW^\dagger R$. Then, for any $t \in \mathbb{Z}$, we have

$$S^t W |0^m\rangle |\psi\rangle = \sin((2t+1)\theta) |0^m\rangle V |\psi\rangle + \cos((2t+1)\theta) |\Phi^\perp\rangle. \quad (1.6.14)$$

From Lemma 1.6.2, it follows that for a specific choice of θ , one can get an exact LCU algorithm, i.e. where $p = 1$. This is the content of the following theorem.

Theorem 1.6.3 (Exact LCU algorithm). *Let V be unitary, such that $V = \sum_i a_i U_i$ is a linear combination of unitary matrices U_i with $a_i > 0 \forall i$. Let A be the unitary matrix that maps $|0^m\rangle$ to $1/\sqrt{a} \sum_i a_i |i\rangle$. Then, a quantum algorithm exists that performs the map V exactly with $O(a)$ uses of the oracles A , $U = \sum_i |i\rangle\langle i| \otimes U_i$ and their respective inverses.*

This result can be also extended to the important case, where \tilde{V} is the desired matrix one wants to implement, but it is no longer unitary, but rather satisfies $\|V - \tilde{V}\| \leq \delta$ (we call this \tilde{V} being δ -close to V) for some unitary V , $\delta > 0$ and where

$$\|A\| = \max(\sqrt{\text{eig}(A^\dagger A)})$$

denotes the spectral norm. In that case, an analogous version to Theorem 1.6.3 exists.

Theorem 1.6.4 (Approximate LCU algorithm). *Let \tilde{V} be a matrix that is δ -close to some unitary in spectral norm, such that $\tilde{V} = \sum_i a_i U_i$ is a linear combination of unitary matrices U_i with $a_i > 0 \forall i$. Let A be the unitary matrix that maps $|0^m\rangle$ to $1/\sqrt{a} \sum_i \sqrt{a_i} |i\rangle$. Then there exists a quantum algorithm that performs the map \tilde{V} with error $O(a\sqrt{\delta})$ and makes $O(a)$ uses of A , $U \equiv \sum_i |i\rangle\langle i| \otimes U_i$ and their inverses.*

If a is a constant (which it will be for the Coulomb matrix elements we discuss in Section 2.2— a is called λ in Chapter 2), the following corollary follows from Theorem 1.6.4.

Corollary 1.6.4.1. *Let \tilde{V} be a matrix that is δ -close to some unitary in spectral norm, such that $\tilde{V} = \sum_i a_i U_i$ is a linear combination of unitary matrices U_i , where a is constant, i.e. $a = O(1)$ and any unitary U_i requires at most q queries to perform. Then the map \tilde{V} can be performed with error $O(\sqrt{\delta})$ using $O(q)$ queries.*

These theorems, lemmas and corollaries lay the foundation for the simulation of sparse Hamiltonians.

Hamiltonian simulation

We consider a Hamiltonian H which is the sum of m unitaries U_j and its evolution for a time $t = 1/m$. Note that U_m is not required to be Hermitian, but it will be whenever H is a physical Hamiltonian of interacting fermions in the spin-basis (i.e. when H is represented as a sum of products of Pauli operators through the Jordan-Wigner transformation, which is discussed in Section 2.2.4). The desired operator we wish to implement is

$$V = e^{-iHt/m} = e^{-i \sum_j U_j/m}. \quad (1.6.15)$$

By using the definition of the matrix exponential, one can write V as a linear combination of unitary matrices. Truncating the series after k terms gives

$$\tilde{V} = \sum_{l=0}^k \frac{1}{l!m^l} \left(-i \sum_j U_j \right)^l. \quad (1.6.16)$$

The error caused by this truncation of the Taylor series is at most $\delta = \|V - \tilde{V}\| < 1/(k!)$. One can show that the following lemma follows from Corollary 1.6.4.1.

Lemma 1.6.5. *Let $H = \sum_{j=1}^m U_j$ be a Hamiltonian, where each U_j is unitary and costs $O(1)$ queries to implement. Then the unitary $e^{-iH/m}$ can be implemented up to error ϵ with query complexity $O\left(\frac{\log(1/\epsilon)}{\log(\log(1/\epsilon))}\right)$. Thus for $t \geq 1/m$, the unitary e^{-iHt} can be implemented up to error ϵ with query complexity $O\left(mt \frac{\log(mt/\epsilon)}{\log(\log(mt/\epsilon))}\right)$.*

Lemma 1.6.5 can in principal be used to simulate any Hamiltonian, since they can always be written as a LCU. The problem however is, that these unitaries U_j have to be efficiently implementable and that $a = \sum_i a_i$ should be a small number. In order to handle arbitrary sparse Hamiltonians, Ref. [101] introduces a method that decomposes the Hamiltonian into a in general much larger sum of 1-sparse Hamiltonians and then decompose the latter into a linear combination of unitaries with eigenvalues ± 1 . We will encounter this strategy again in Appendix 2.B.

Lemma 1.6.6. *If H is a d -sparse Hamiltonian, there exists a decomposition $H = \sum_{j=1}^{d^2} H_j$ where each H_j is 1-sparse and a query to any H_j can be simulated with $O(1)$ queries to H .*

Using the following lemma, one can decompose a 1-sparse Hamiltonian G into a sum of $O(\|G\|_{\max}/\gamma)$ unitary Hamiltonians G_j up to error $O(\gamma)$.

Lemma 1.6.7. *For any 1-sparse Hamiltonian G and precision $\gamma > 0$, there exist $O(\|G\|_{\max}/\gamma)$ unitary sparse Hamiltonians G_j with eigenvalues ± 1 such that $\|G - \gamma \sum_j G_j\| \leq 3\gamma$.*

Lemma 1.6.7 leads to the following important theorem for the simulation of sparse Hamiltonians.

Theorem 1.6.8 (Sparse Hamiltonian simulation). *A d -sparse Hamiltonian H can be simulated for time t with error at most ϵ using $O\left(\tau \frac{\log(\tau/\epsilon)}{\log(\log(\tau/\epsilon))}\right)$ queries, where $\tau = d^2 \|H\|_{\max} t \geq 1$.*

The dominant source of error for the sparse Hamiltonian simulation is here due to the truncation of the Taylor series expansion of the matrix exponential. Importantly, it was shown that the dependence of the query complexity in Theorem 1.6.8 on the error ϵ is tight up to constant factors to the lower bound, which was shown to require $\Omega\left(\frac{\log(1/\epsilon)}{\log(\log(1/\epsilon))}\right)$ discrete queries to obtain error at most ϵ . Theorem 1.6.8 with Lemma 1.6.7 give the principal idea behind the Hamiltonian simulation method used in Section 2.3, see also Chapter 4.4. of Ref. [50].

Qubitization algorithm

One of the drawbacks of the algorithm presented here is that it requires the use of oblivious amplitude amplification. Such modern Hamiltonian simulation methods aiming at simulating e^{-iHt} , including signal processing [108] and the qubitization algorithm [109], have achieved the provable optimal scaling in the number of queries to a certain primitive. There are however encodings other than e^{-iHt} which can be used if one only is interested in sampling the spectrum of a Hamiltonian H . The motivating principle here is that in order to simulate a quantum system, it is not necessary to imitate nature, a quantum computer can sometimes employ routes that are physically inaccessible [110]. One such encoding is described in Ref. [100] which is based on the work of Refs. [103, 104] where it was suggested to perform phase estimation on a quantum walk operator $\mathcal{W}(H) = e^{\pm i \arccos(H/\lambda)}$ instead. The steps of the quantum walk can be performed exactly (it avoids i.a. the error due to the truncation of the Taylor expansion of the time evolution operator in Eq. (1.6.16)) and lead to eigenvalues $\pm e^{\pm i \arcsin(E_k/\lambda)}$ which are isomorphic to E_k , where E_k is the k -th eigenvalue of H . We will use the quantum walk algorithm in Sections 2.3 and 2.B. For our purposes, it is sufficient to know that the quantum walk algorithm is also based on the use of the SELECT and PREPARE oracles. The interested reader should consider Ref. [100] for a more detailed explanation of the employed algorithm.

Section 1.7

THE QUANTUM HALL EFFECT

The classical Hall effect was discovered in 1879 by Edwin Hall [111]. It describes how a voltage builds up which is transversal both to the direction of an electrical current in a conductor and to a magnetic field that is perpendicular to the current. This voltage-buildup is a result of the charge carriers being under the influence of the Lorentz force. In the following, we will motivate the expected *classical* behavior of the resistivities when a electric field \mathbf{E} , which is the longitudinal resistivity ρ_{xx} and the behavior of the ρ_{xy} which is transversal to ρ_{xx} in the plane. We will make use of the Drude model [112], which was designed to predict the charge transport of electrons in materials following a model. Drude himself was aware that his theory would most likely not be able to explain all observed phenomena in experiments [112], and we will see how the QHE experiments' findings differ from the classically expected behavior below. The derivations for the conductivity and resistivity below are adapted from Ref. [8].

1.7.1 The classical model

The Drude model assumes that the metal is made up of positively charged ions, where a number of free electrons are detached from. These electrons are called free since one assumes that the Valence levels of a certain atom has a non-negligible overlap with the other atoms' potentials, making it possible for the electrons to move around the material. There are a number of strong assumptions made, such as the negligence of long-range electron-electron, or electron-ion interactions, and the only interaction considered is that of electrons with the ions due to instantaneous collisions. The nature of the collisions is not of importance. The forces of the electron are due to the Lorentz force [113] and the material properties which hinder the electrons' motion are summarized in a linear friction term. For a given electric (magnetic) field \mathbf{E} (\mathbf{B}), electron charge e and electron mass m , the equations of motions within the Drude model are given by

$$m \frac{d^2 \mathbf{x}}{dt^2} = -e\mathbf{E} - e \frac{d\mathbf{x}}{dt} \times \mathbf{B} - \frac{m}{\tau} \frac{d\mathbf{x}}{dt}, \quad (1.7.1)$$

where τ is a friction term known as the scattering time—which can be thought of as the average time between collisions—and the current density \mathbf{J} is linearly proportional to the velocity of the electron,

$$\mathbf{J} = -ne \frac{d\mathbf{x}}{dt}, \quad (1.7.2)$$

where n is the density of charge carriers. We search for solutions of Eq. (1.7.1) where $\frac{d^2 \mathbf{x}}{dt^2} = 0$, which leads to

$$-\frac{1}{ne} \mathbf{J} - \frac{\tau}{mn} \mathbf{J} \times \mathbf{B} = -\frac{e\tau}{m} \mathbf{E}. \quad (1.7.3)$$

We let the movement of the electrons be restricted to the xy -plane, the magnetic field of magnitude $B = |\mathbf{B}|$ pointing in the z direction and define the cyclotron frequency

$\omega_B = \frac{eB}{m}$. Then, the equilibrium condition can be written in matrix notation as

$$\mathbf{J} = \sigma \mathbf{E}, \quad (1.7.4)$$

which is known as *Ohm's law* [114], where

$$\sigma = \begin{pmatrix} \sigma_{xx} & \sigma_{xy} \\ -\sigma_{xy} & \sigma_{xx} \end{pmatrix} = \frac{ne^2\tau}{m} \begin{pmatrix} 1 & \omega_B\tau \\ -\omega_B\tau & 1 \end{pmatrix}^{-1} = \frac{\sigma_{\text{DC}}}{1 + \omega_B^2\tau^2} \begin{pmatrix} 1 & -\omega_B\tau \\ \omega_B\tau & 1 \end{pmatrix}, \quad (1.7.5)$$

is known as the *conductivity* tensor and $\sigma_{\text{DC}} = \frac{ne^2\tau}{m}$ is the conductivity in absence of the magnetic field. The off-diagonal elements in Eq. (1.7.5) are responsible for the classical Hall effect. The *resistivity* ρ is defined as the inverse of the conductivity,

$$\rho = \sigma^{-1} = \begin{pmatrix} \rho_{xx} & \rho_{xy} \\ -\rho_{xy} & \rho_{xx} \end{pmatrix} = \frac{1}{\sigma_{\text{DC}}} \begin{pmatrix} 1 & \omega_B\tau \\ -\omega_B\tau & 1 \end{pmatrix}. \quad (1.7.6)$$

Most notably, the off-diagonal elements

$$\rho_{xy} = \frac{\omega_B\tau}{\sigma_{\text{DC}}} = \frac{B}{ne} \quad (1.7.7)$$

are independent of the scattering effects taking place in the probe, hinting that it captures a fundamental property of the material rather than of the impurities responsible for the scattering. Another peculiar property of the off-diagonal part is that in two dimensions, the transversal resistance R_{xy} (which is the quantity measured in experiments—it depends on the geometry of the probe) and the transversal resistivity coincide, $R_{xy} = \rho_{xy}$. The longitudinal resistivity is given by

$$\rho_{xx} = \frac{m}{ne^2\tau}. \quad (1.7.8)$$

In the Drude model, a current flowing in the x -direction within a magnetic field in the z -directions, charges accumulate at the edges of the probe, building up an electric field E_y in the y -direction. The associated resistance,

$$R_H = -\frac{E_y}{J_x B} = \frac{1}{ne}, \quad (1.7.9)$$

is called the *Hall coefficient* and it only depends on the charge and the density of the conducting particles, i.e. the microscopic properties of the material and not on the "dirt" (i.e. the friction term) located in the sample.

From Eqs. (1.7.7)-(1.7.8), it follows that, within the classical picture of the Drude model, one expects the longitudinal resistivity ρ_{xx} to be constant and the transversal resistivity ρ_{xy} to be increasing linearly w.r.t. the magnetic field strength B , as sketched in Fig. 1.5.

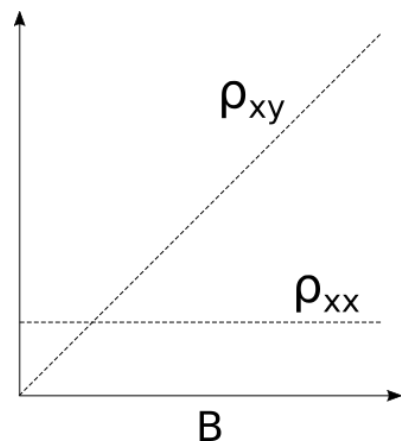


Figure 1.5: Classical expectation of the behavior of the transversal (longitudinal) resistivities ρ_{xy} (ρ_{xx}) w.r.t. an applied transversal magnetic field B .

1.7.2 The discovery of the quantum Hall effect

In 1980, about a century after the discovery of the Hall effect, as experiments were able to go to lower temperatures and thus entering the quantum regime, it was reported by von Klitzing et al. [115] that instead of the behavior of the resistivities displayed in Fig. 1.5, the transversal resistivity ρ_{xy} was observed to be constant over finite ranges of the applied magnetic field before jumping to another plateau,

$$\rho_{xy} = \frac{h}{e^2} \frac{1}{\nu}, \quad (1.7.10)$$

where ν was found to be an integer to extraordinarily high precision, namely to one part in a billion. By comparison, the density required to get the measured value of the resistivity at the ν -th plateau is $n = \frac{B\nu}{\Phi_0}$, where $\Phi_0 = \frac{h}{2e}$ is the flux quantum. The constant h/e^2 is also known as the von Klitzing constant and is recognized as the resistance quantum. The observation of integer-valued plateaus of the transversal resistivity is known as the Integer Quantum Hall effect (IQHE). On the respective plateaus, the longitudinal resistivity ρ_{xx} vanishes, while displaying sharp peaks whenever ρ_{xy} jumps to another plateau. This behavior can be clearly seen in Fig. 1.6, where two plateaus of ρ_{xy} and two dips in ρ_{xx} are visible. The fact that ρ_{xx} is not constant in these plots is due to the fact that a voltage has to be applied in order to keep the current flowing in the longitudinal direction. Quantization is well known to appear in quantum systems on a microscopic level, however the quantization observed here is that of an emergent, macroscopic property of a "dirty" sample involving billions of fundamental particles, constituting a so-called quantum liquid. The theoretical explanation of the IQHE is well understood, it requires the concept of *Landau levels*³⁶, which are discussed in Section 2.2.1, in order to explain the values at which the plateaus appear and Anderson localization due to local random potentials ("dirt") for explaining the persistence of these plateaus over a finite range of the magnetic field. Its explanation does, however, not require to take into account the Coulomb interactions between the electrons.

The notion that the behavior of Eq. (1.7.10) would only be observed for integer valued ν was quickly refuted by the discovery of a plateau at fractional filling $\nu = 1/3$ by Tsui et al. [117] in 1982 in probes with reduced disorder. They observed plateaus at $\nu = 1$ and $\nu = 1/3$ (see Fig. 1.6, to be compared with the behavior as predicted by a classical theory as sketched in Fig. 1.5), which lead to a plethora of additional discoveries of other plateaus at various filling factors [118]. This is known as the Fractional Quantum Hall Effect (FQHE). The appearance of the FQHE requires an hierarchy of energy scales which is roughly given by [8, 9]

$$\hbar\omega_B \gg E_{\text{Coulomb}} \gg V_{\text{disorder}}, \quad (1.7.11)$$

where E_{Coulomb} and V_{disorder} is the strength of the Coulomb interaction and the disorder, respectively³⁷. Therefore, the effect of Coulomb interactions between the electrons plays

³⁶It is well known that (here, non-interacting) particles moving in two-dimensions in presence of a magnetic field can only occupy discrete energy levels, known as Landau levels, which are macroscopically degenerate. An excellent explanation of Landau levels and the QHE is given in Ref. [116].

³⁷Roughly speaking, the reason why the IQHE does not require to take into account disorder is due to the fact that they can be observed in "dirtier" samples, where $V_{\text{disorder}} \gtrsim E_{\text{Coulomb}}$. A perfectly clean sample, i.e. one with no disorder is not believed to display any plateaus, since it is the disorder which is responsible for the persistence of the plateaus over a finite range of B .

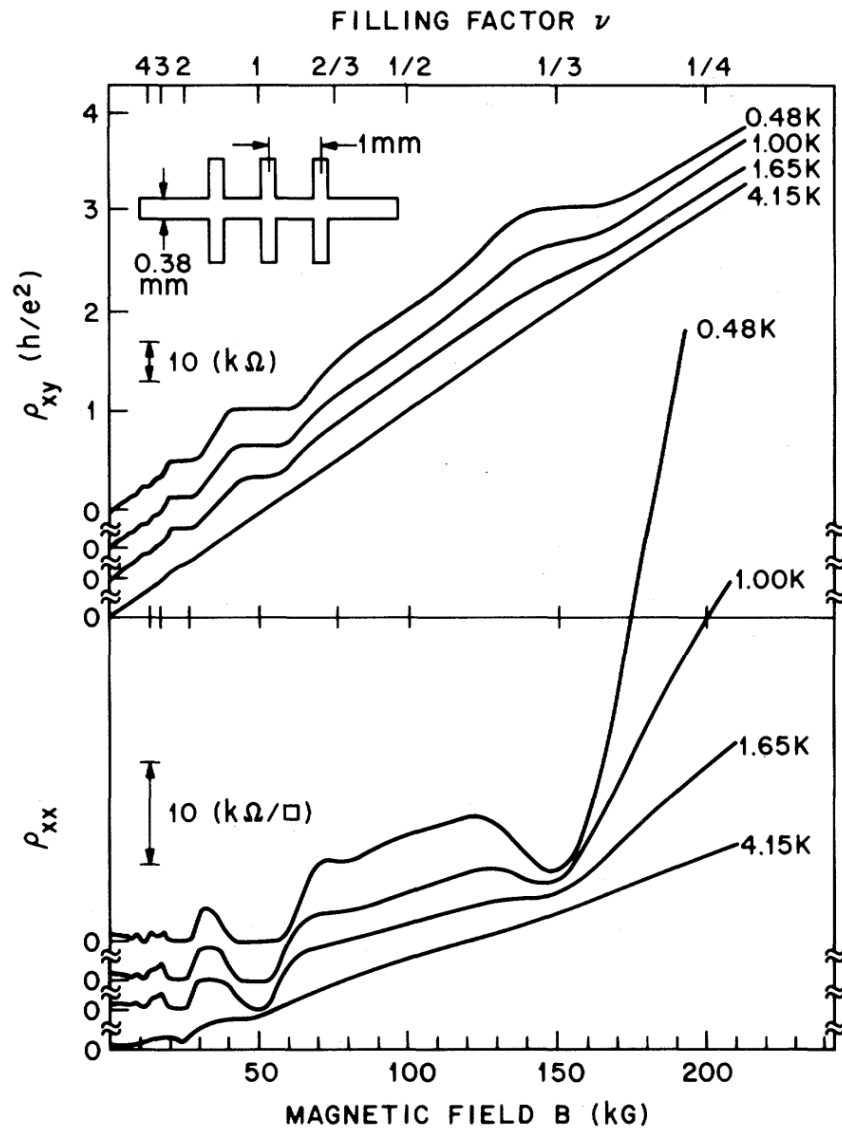


Figure 1.6: Experimental data of Ref. [117] showing the transversal and longitudinal resistivities ρ_{xy} and ρ_{xx} for a *GaAs-AlGaAs* sample. For an electron density of $n = 1.23 \times 10^{11}/(\text{cm})^2$, a plateau at $\nu = nh/(eB) = 1/3$ becomes visible at temperatures $T < 0.5\text{K}$, the first reported observation of the FQHE. Also clearly visible is an IQHE plateau at $\nu = 1$. In the lower part of the figure, one can see the dips in the longitudinal resistivity whenever a jump to another plateau happens. The fact that ρ_{xx} is not constant but grows linearly is due to the fact that a voltage is applied in longitudinal direction to keep the current alive in the probe. Reprinted with kind permission of the American Physical Society under license number RNP/20/MAR/023933.

a crucial role in its explanation³⁸ which makes it well-suited to be studied on a quantum computer, as argued in Section 1.1.

Almost all discovered plateaus were observed at odd denominator filling fractions, while the mechanism behind the even denominator plateau at $\nu = 5/2$ is to this day still not understood [119]. Robert Laughlin provided a trial wave function for the ground state of a FQH system at filling $\nu = 1/m$, with m being an odd integer [120], while Moore and Read proposed a wave function describing the $\nu = 5/2$ FQH state [121]. These wave functions are not ground states of the QHE Hamiltonian, but rather ground states of different, so-called parent Hamiltonians [122–124]. One of the main impacts the FQHE had on physics, was that it provided an example of a correlated phase that violates *both* of Landau’s paradigms and did not fit into the then established paradigms of condensed matter theory [125]. More precisely, the FQHE breaks with the two main concepts of Landau’s Fermi Liquid theory [126] which describes the behavior of electrons in a metal at sufficiently low temperatures. The first concept of Landau’s theory is that the electron can be viewed as a quasiparticle excitation above the quantum ground state of the many particle system. In the FQHE, the electron does not retain its integrity as a quasiparticle excitation, since excitations out of a FQHE ground state possess fractional charge [120, 127, 128]. The second paradigm of Landau’s Fermi Liquid theory is that of the local order parameter to classify and distinguish phases of matter. The FQHE also breaks with this second paradigm, as its phases display a certain kind of order that is not captured by a local Landau order parameter, but the order has to be rather seen as a global property of the many-electron ground state wave function [129, 130]. The QHE was the first system that displayed non-trivial topological quantum states in an experiment [125, 131], which are believed to be a platform for topological quantum computing [34, 55], an architecture which would implement quantum error-correction on a hardware level.

The microscopic origin of the FQHE is a major research area of modern condensed matter theory, whose goal is to identify simple emergent principles that provide a theory which can unify, predict, explain and compute the structures that emerge when many particles interact in the system at hand [132]. While it is not clear whether a quantum computer will be helpful for *all* these tasks, it will provide a means to test theories beyond what would be possible on a classical computer (for reasons outlined in Section 1.1) and will be discussed in more detail in Chapter 2.

³⁸It turns out that the FQHE gives rise to new types of quasiparticles which are composed of electrons binding magnetic flux lines in order to lower their energy. One of the most prominent theories which describes and predicted many of the observed FQHE patterns is the *composite fermion theory* [9]. This microscopic theory explains the fractional states of the electrons in the FQHE as the IQHE of the composite quasiparticles. The composite fermion theory recovers the Laughlin wave function as a special case, in contrast, composite fermions are not part of, nor can they be derived from the Laughlin wave function [9], which makes the composite fermion theory much more general. The states these composite particles condense into possess excitations that display the bizarre property of being described by fractional quantum numbers. Excitations and holes of the underlying ground state possess fractional charges and fractional statistics.

CHAPTER 2

ROADMAP FOR QUANTUM SIMULATION
OF THE
FRACTIONAL QUANTUM HALL EFFECT

A major motivation for building a quantum computer is that it provides a tool to efficiently simulate strongly correlated quantum systems. In this work, we present a detailed roadmap on how to simulate a two-dimensional electron gas—cooled to absolute zero and pierced by a strong transversal magnetic field—on a quantum computer. This system describes the setting of the Fractional Quantum Hall Effect (FQHE), one of the pillars of modern condensed matter theory. We give analytical expressions for the two-body integrals that allow for mixing between N Landau levels at a cutoff M in angular momentum and give gate count estimates for the efficient simulation of the energy spectrum of the Hamiltonian on an error-corrected quantum computer. We then focus on studying efficiently preparable initial states and their overlap with the exact ground state for noisy as well as error-corrected quantum computers. By performing an imaginary time evolution of the covariance matrix we find the generalized Hartree-Fock solution to the many-body problem and study how a multi-reference state expansion affects the state overlap. We perform small-system numerical simulations to study the quality of the two initial state Ansätze in the Lowest Landau Level (LLL) approximation.

Section 2.1

INTRODUCTION AND OVERVIEW

Feynman's conjecture that quantum computers could provide a means for efficiently simulating other quantum systems was proven by Lloyd in 1996 [105], where a simulation is considered to be *efficient*, if the computational cost scales at most polynomially with the system size. The following year, Abrams and Lloyd [133] showed how a fermionic quantum system could be simulated on such a device in either first or second quantization. 25 years after the proposal of quantum computing [67, 134], Aspuru-Guzik et al. [69] demonstrated that the calculation time for the energy of atoms and molecules scales polynomially using quantum algorithms given an initial state with sufficient support on the desired eigenstate. This provided the initial spark to ignite a plethora of studies on molecular electronic systems using quantum computers (see e.g. [135] for a recent summary). Until then, quantum computing was more famously known for being able to break RSA-encryption [71] but with the proposed simulation of quantum mechanical systems, quantum computing gained a lot of interest across various fields.

While the study of strongly correlated fermionic systems has been advocated as a strong suit for quantum computers, one of its most prominent phenomena, the FQHE, has so far been rather sparsely covered¹. This effect occurs when electrons are confined to two dimensions², cooled to near absolute zero and are subject to a strong transversal magnetic field. The FQHE manifests itself by a quantization of the Hall conductance over a finite range of the applied magnetic field for certain electron densities and led to various theories and proposed new quasiparticles, such as composite fermions, aimed at describing the observed patterns [9]. The plateaus appear at integer or fractional values of e^2/h (where e is the electron charge and h is Planck's constant) and while the integer value plateaus can be well explained by Landau quantization and the effect of disorder (without having to take into account interactions), the Coulomb interaction between electrons plays a key role for the understanding of the observation of plateaus at fractional values of e^2/h . Deriving a microscopic theory to explain the fractional plateaus is an active field of research in condensed matter physics. It is believed that quasi-hole and -particle excitations of the ground state of Fractional Quantum Hall (FQH) systems display anyonic statistics, which form the building blocks of a topological quantum computer [36].

It is not known whether a quantum computer will help us find underlying universal principles that enable us to explain the phenomena of the simulated correlated quantum system. However, a quantum computer does provide a tool to test such theories against exact and approximate solutions for system sizes far beyond what any classical computer will be able to simulate. Our aim is to give an *ab-initio* roadmap that paves the way towards a digital quantum simulation of FQH systems.

We will consider two different types of quantum computers, on the one hand those

¹With the exception of Ref. [136], where a quantum algorithm to compute the entanglement spectrum of a quantum state such as the Laughlin state on a quantum computer is presented, but a detailed state creation analysis is not included.

²Only the movement of the electrons is restricted to be (approximately) two-dimensional, we are not referring to the electrons living in a universe with two spatial dimensions, where the form of the Coulomb potential would be quite different from the three dimensional version that we are studying.

Table 2.1: List containing all abbreviations used in this chapter.

Abbreviation	
FQH(E)	Fractional Quantum Hall (Effect)
(L)LL	(Lowest) Landau Level
NISQ	Near Intermediate-Scale Quantum
VQE	Variational Quantum Eigensolver
LCU	Linear Combination of Unitaries
FGS	Fermionic Gaussian State
CM	Covariance Matrix
ASCI	Adaptive Sampling Configuration Interaction
FCI	Full Configuration Interaction

which are error-corrected and potentially able to perform millions of gate operations and on the other hand those available today, i.e. error-prone quantum processors, which are limited to execute quantum operations well within their coherence times.

Within the context of error-corrected quantum computers, we study the scaling of current state-of-the-art quantum algorithms based on the Linear Combination of Unitaries (LCU) method, which is designed to compute the energy spectrum of a given Hamiltonian H to desired precision ΔE [137]. These quantum algorithms realize a unitary alternative to the usual time evolution operator [138] of the quantum phase estimation algorithm [139] and allow one to efficiently extract information about the Hamiltonian’s spectrum.

While the quantum phase estimation algorithm has a theoretically proven exponential speedup in sampling a Hamiltonian or eigenvalue sampling of a unitary matrix generated by the exponential of a sparse matrix, current and near-term quantum computers are not fault-tolerant and applying the quantum phase estimation algorithm is impossible due to the tremendous amount of gate operations that need to be applied coherently. On the other hand, algorithms which are applicable to Noisy Intermediate-Scale Quantum (NISQ) [78] devices, i.e. non-error-corrected quantum computers—such as the Variational Quantum Eigensolver (VQE) [79, 80]—are restricted to coherence time limited circuit depths and are of heuristic nature. Such heuristic algorithms are intuitively compelling and capable of systematic refinement, but lack rigorous bounds on their performance³.

A large part of our work will focus on finding an initial state $|\Psi_{\text{init}}\rangle$ (sometimes also called a trial-, or reference state) which approximates the ground state $|\Psi_0\rangle$ of H . We restrict ourselves to initial states which are *efficiently computable on a classical-* and *efficiently preparable on a quantum computer* and need to possess a non-vanishing overlap with the desired eigenstate of the Hamiltonian. We engage in the task of finding an initial state which would serve as the starting point of a given quantum algorithm to approximate the ground state of the Hamiltonian describing the FQH system and how one could then extract physically meaningful properties from it, e.g. by means of computing the one- and two-particle correlation functions. The problem of finding an initial state $|\Psi_{\text{init}}\rangle$ with above mentioned prerequisites has largely been ignored in literature and has only recently been studied thoroughly for a variety of electronic systems [40], with the exception of FQH systems. Such initial states are not only of interest for NISQ algorithms, but also for quantum-error-corrected algorithms such as in Refs. [100, 142].

³It is a topic of current discussion which type of shallow circuit Ansatz might provide an advantage over classical algorithms [140] and the study of VQE-type algorithms revealed other challenges, such as exponentially vanishing gradients [141].

Table 2.2: This table lists and explains the most important symbols appearing in the main text and refers to their respective definitions and appearances in the last column.

Symbol	Explanation	Equation
$O(f(x))$	Limiting behaviour of a function $f(x)$ for large values of x	
$\hat{O}(f(x))$	Limiting behaviour of a function $f(x)$ for large values of x suppressing polylogarithmic factors	
H	System Hamiltonian (both in first and second-quantized representation) $H = H_1 + H_2$, where H_1 contains all single particle terms and H_2 contains all interaction terms between particles	(2.2.1),(2.2.9), (2.3.1),(2.4.4)
$ \Psi_0\rangle$	Exact ground state energy of the system Hamiltonian H	
$ \Psi_{\text{init}}\rangle$	Initial state / reference state that approximates $ \Psi_0\rangle$	
N	Denotes largest considered Landau level (LL). Individual LLs are indexed by $P_1 = 0, 1, \dots, N$	
M	Denotes cutoff in angular momentum, individual angular momenta are indexed by $P_2 = 0, 1, \dots, M$	
$N_{\text{so}} \approx NM$	Number of spin-orbitals—in numerical simulations one chooses $N \ll M$	
N_{el}	Number of electrons	
$\mathbf{P}, \mathbf{Q}, \mathbf{R}, \mathbf{S}$	Quantum number tuples, $\mathbf{P} = (P_1, P_2), \dots, \mathbf{S} = (S_1, S_2)$, we use the notation $P_\Sigma = P_1 + P_2$	
$\psi_{\mathbf{P}}(\mathbf{r})$	Single particle wave function, depending on \mathbf{P} and particle coordinate \mathbf{r} , eigenfunctions of H_1	(2.2.5)
$f_{\mathbf{PQ}}$	One-body Hamiltonian coefficients of H_1 in second-quantized representation	(2.2.10),(2.2.8)
$h_{\mathbf{PQRS}}$	Coulomb-matrix elements of H_2 in second-quantized representation	(2.2.11)
$h_{\mathbf{PQRS}}^{(i)}$	Coulomb-matrix elements for case (i) where $P_2 - S_2 \geq 0$ —case (ii) $P_2 - S_2 < 0$ follows from symmetry	(2.2.19)
$c_{\mathbf{P}}, c_{\mathbf{P}}^\dagger, c_p, c_p^\dagger$	Fermionic annihilation and creation operators satisfying the anticommutation relations	(2.2.9),(2.4.4)
$F_A^{(4)}[\dots]$	Lauricella function (here, a finite hypergeometric series)	(2.2.20), (2.2.22)
$(\lambda)_n$	Pochhammer symbol (also known as rising factorial), result of division of two Gamma functions	(2.2.21)
R_d	Radius of simulated 2D disk. Describes the disk boundary due to the cutoff in angular momenta at M	(2.2.27)
ν	Filling factor, defined as the number of electrons per flux quantum penetrating the disk	(2.2.29),(2.2.30)
M_{mn}^l	Coulomb matrix elements in the LLL approx., identical to $h_{\mathbf{PQRS}}$ for $P_1 = Q_1 = R_1 = S_1 = 0$	(2.2.31)
l, m, n	Coefficients of M_{mn}^l , corresponding to $l = P_2 - S_2, m = S_2$ and $n = Q_2$	
Σ	Number of non-zero terms of the system Hamiltonian H in Jordan-Wigner representation, $\Sigma \propto L$	(2.2.37)
L	Two meanings: Either number of terms in LCU expansion, or number of determinants in ASCI state	(2.3.1),(2.4.17)
U_ℓ	Unitary matrices from the linear combination of unitaries (LCU) method	(2.3.1)
ω_ℓ	Coefficients of the LCU method, related to Hamiltonian coefficients, $H = \sum_\ell \omega_\ell U_\ell$	(2.3.1)
λ	Sum of absolute values of all ω_ℓ values. Important for determining complexity of LCU method	(2.3.1)
SELECT	Oracle of LCU method efficiently implementing the U_ℓ in superposition	(2.3.2)
PREPARE	Oracle of LCU method generating a linear combination of states indexed by ℓ and weighted by ω_ℓ/λ	(2.3.3)
C_S, C_P	Gate complexity of SELECT and PREPARE oracles	(2.3.4)
ΔE	Target precision of the energy in phase estimation	(2.3.5)
Γ	Covariance matrix (CM) characterizing the FGS	(2.4.3)
f_{pq}, h_{pqrs}	One- and two-body coefficients of H (the latter being chosen to fulfill Eq. (2.4.5)) in LLL	(2.4.4)
$h_m(\Gamma), E_m$	Mean-field matrix of system Hamiltonian H and corresponding mean-field energy	(2.4.10),(2.4.12)
$cdtes, tdtes$	Number of core- and target-space determinants of ASCI algorithm	
C_i	Expansion coefficients of the ASCI algorithm	(2.4.17)
$ D_i\rangle$	Expansion determinants of the ASCI algorithm	(2.4.17)
A_i	Perturbed wave functions amplitudes over all single and double excitations in ASCI	(2.4.18)

This work is structured as follows. In Section 2.2 we present the Hamiltonian of interacting electrons in a disk geometry pierced by a strong magnetic field. We provide efficiently computable analytical expressions for the two-body coefficients of the Hamiltonian in second quantization and describe how this Hamiltonian can be mapped from the fermionic to the spin basis using the Jordan-Wigner transformation. In Section 2.3, we present an efficient strategy for simulating the FQHE on an error-corrected quantum computer using a quantum algorithm proposed in Ref. [143] based on the LCU method [137]. In Section 2.4 we discuss the classically efficient computation of initial states from the family of Fermionic Gaussian States (FGS), which can be implemented on NISQ devices. We extend our discussion by including a multi-reference state approach suited for error-corrected quantum computers, which is based on linear combination of Slater determinants using a state-of-the-art quantum chemistry algorithm [40]. The results of the numerical simulations are presented in Section 2.5, where we compare fidelities of the respective initial state and the actual ground state $|\Psi_0\rangle$ for small system sizes (which corresponds to the typical size of current cloud-based quantum computing hardware). In Section 2.6 we discuss possible avenues one could explore in order to improve the FQH Hamiltonian model. We sum up our findings in Section 2.7. The Appendix provides further details mainly on the derivations of the Coulomb matrix elements, an alternative Hamiltonian simulation strategy based on the self-inverse matrix decomposition strategy of [144], the equations of motion for the imaginary time evolution of the Covariance Matrix (CM), and helpful relations for the implementation of the multi-reference state approach. We provide a list of the main abbreviations in Table 2.1 and of symbols used throughout the main text in Table 2.2.

The first draft on the topic of Hamiltonian simulation in Section 2.3 and Appendix 2.B was written by Ryan Babbush, who also proposed the idea of exploiting the fact that we derived a closed form for the coefficients using an alternative approach [50, 144], while the author contributed the numerical simulation results for the complexity analysis. The current versions of Section 2.3 and Appendix 2.B have been written by the author based on the joint work on Hamiltonian simulation with Ryan Babbush.

Section 2.2

SYSTEM HAMILTONIAN

This Section presents the considered Hamiltonian of electrons under Coulomb repulsion in a strong magnetic field. We present analytical solutions in symmetric gauge disk geometry for the one- and two-body matrix elements of the second-quantized Hamiltonian which allows for Landau Level (LL) mixing. A similar result has been reported for a spherical geometry in [145].

We introduce Landau levels and the single-particle basis states in Section 2.2.1. Section 2.2.3 presents the system Hamiltonian in the LLL which is the setting of our numerical simulations. We conclude this section by showing how to map the fermionic Hamiltonian through the Jordan-Wigner transformation to a spin Hamiltonian in Section 2.2.4.

2.2.1 The Hamiltonian in first quantization

We analyze a two-dimensional electron gas in the x - y plane with no disorder and in a strong magnetic field $\mathbf{B} = (0, 0, B)^T$ allowing for discarding the spin degrees of freedom. It is described by the Hamiltonian [9]

$$H = H_1 + H_2, \tag{2.2.1}$$

that is the sum of the single-particle terms

$$H_1 = \sum_j \frac{1}{2m_b} \left(-i\hbar\nabla_j + \frac{e}{c}\mathbf{A}(\mathbf{r}_j) \right)^2, \tag{2.2.2}$$

and the two-particle interactions described by

$$H_2 = \frac{e^2}{\epsilon} \sum_{j < k} \frac{1}{|\mathbf{r}_j - \mathbf{r}_k|}. \tag{2.2.3}$$

Eq. (2.2.2) describes the energy of the electrons with effective band mass m_b in absence of interactions and in a constant magnetic field $\mathbf{B} = \nabla \times \mathbf{A}$. We use the vector potential in symmetric gauge [9]

$$\mathbf{A} = \frac{\mathbf{B} \times \mathbf{r}}{2}, \tag{2.2.4}$$

which breaks translational symmetry in x - and y -direction, but preserves the rotational symmetry about the origin, which makes the angular momentum a good quantum number. Here, $\mathbf{r}_j = (x_j, y_j, 0)$ is the position of the electron j in the x - y plane and e , c the electron charge and speed of light, respectively. The Hamiltonian in Eq. (2.2.3) describes the Coulomb interactions between the atoms where $\epsilon = 4\pi\epsilon_0$ and ϵ_0 is the dielectric constant.

Eigenfunctions of single-particle Hamiltonian

The eigenfunctions and energies of H_1 (Eq. (2.2.2)) are known analytically and will be used later to describe the full Hamiltonian H (Eq. (2.2.1)) in second quantization. The corresponding single-particle states are the basis of choice for the second-quantized Hamiltonian and are described by a set of two quantum numbers $\mathbf{P} = (P_1, P_2)$, where P_1 denotes the LL and the second quantum number P_2 denotes the angular momentum.

For a given LL $P_1 = 0, 1, \dots$, the angular momentum can take the values $P_2 = -P_1, -P_1 + 1, \dots$. The single-particle wave function are given by

$$\psi_{\mathbf{P}}(\mathbf{r}) = \frac{(-1)^{P_1}}{\sqrt{2\pi}} \sqrt{\frac{P_1!}{2^{P_2}(P_1 + P_2)!}} L_{P_1}^{(P_2)}\left(\frac{r^2}{2}\right) z^{P_2} e^{-\frac{1}{4}r^2}, \quad (2.2.5)$$

with $z_j = x_j - iy_j = r_j e^{-i\theta_j}$ and $r_j = \sqrt{z_j z_j^*}$ being the complex particle coordinates and where we defined the associated Laguerre polynomials of degree n and order α

$$L_n^{(\alpha)}(x) = \sum_{i=0}^n (-1)^i \binom{n+\alpha}{n-i} \frac{x^i}{i!}. \quad (2.2.6)$$

The functions $\psi_{\mathbf{P}}(\mathbf{r})$ fulfill

$$H_1 \psi_{\mathbf{P}}(\mathbf{r}) = E_{P_1} \psi_{\mathbf{P}}(\mathbf{r}), \quad (2.2.7)$$

with eigenenergy

$$E_{P_1} = \hbar\omega_c \left(P_1 + \frac{1}{2} \right), \quad (2.2.8)$$

where $\omega_c = eB/(\hbar m_b c)$ is called the cyclotron frequency. The discrete energy levels of the kinetic terms—the LLs—are the workhorse of the quantum Hall problem. The formation of Landau levels provide the key insight for the understanding of the integer quantum Hall effect and the fractional quantum Hall effect can be explained by a splitting of a Landau level into “Landau-like” energy levels in presence of interactions [9]. We note that other basis choices might provide a more compact representation of the system Hamiltonian (even though it is unclear how simple restrictions to single LLs would be possible in such representations), however the Landau level basis is a reasonable representation of the FQH problem.

2.2.2 The Hamiltonian in second quantization

For the purpose of simulating the quantum mechanical system on a quantum computer we derive the Hamiltonian in second quantization. The second-quantized form of H (as in Eq. (2.2.1)) in the single-particle basis of Eq. (2.2.5) is given by [2]

$$H = \sum_{\mathbf{P}, \mathbf{Q}} f_{\mathbf{P}\mathbf{Q}} c_{\mathbf{P}}^\dagger c_{\mathbf{Q}} + \frac{1}{2} \sum_{\mathbf{P}, \mathbf{Q}, \mathbf{R}, \mathbf{S}} h_{\mathbf{P}\mathbf{Q}\mathbf{R}\mathbf{S}} c_{\mathbf{P}}^\dagger c_{\mathbf{Q}}^\dagger c_{\mathbf{R}} c_{\mathbf{S}}, \quad (2.2.9)$$

where the one- and two-body coefficients are given by

$$f_{\mathbf{P}\mathbf{Q}} = \int d\mathbf{r} \psi_{\mathbf{P}}(\mathbf{r})^* H_1 \psi_{\mathbf{Q}}(\mathbf{r}), \quad (2.2.10)$$

$$h_{\mathbf{P}\mathbf{Q}\mathbf{R}\mathbf{S}} = \iint d\mathbf{r}_1 d\mathbf{r}_2 \psi_{\mathbf{P}}(\mathbf{r}_1)^* \psi_{\mathbf{Q}}(\mathbf{r}_2)^* H_2 \psi_{\mathbf{S}}(\mathbf{r}_1) \psi_{\mathbf{R}}(\mathbf{r}_2), \quad (2.2.11)$$

using Eq. (2.2.2) and Eq. (2.2.3), respectively and $\int d\mathbf{r}_j = \int_{-\infty}^{\infty} dx_j \int_{-\infty}^{\infty} dy_j$ for $j = 1, 2$. The operators $c_{\mathbf{P}}$ and $c_{\mathbf{P}}^{\dagger}$ are the fermionic annihilation and creation operators fulfilling the anticommutator relations $\{c_{\mathbf{P}}, c_{\mathbf{Q}}\} = 0$ and $\{c_{\mathbf{P}}, c_{\mathbf{Q}}^{\dagger}\} = \delta_{\mathbf{P}\mathbf{Q}}$ with the Kronecker delta $\delta_{\mathbf{P}\mathbf{Q}}$. The total number of terms in Eq. (2.2.9) scales as $O(N_{\text{so}}^4)$, where the number of spin orbitals is approximately given by $N_{\text{so}} \approx NM$, with N and M denoting the cutoff in the number of LLs and angular momentum and $N \ll M$.

Kinetic term

Since we used an eigenbasis of H_1 for the representation of the Hamiltonian in second quantization the one-particle coefficients are diagonal. They are given by

$$f_{\mathbf{P}\mathbf{Q}} = E_{P_1} \delta_{\mathbf{P}\mathbf{Q}}, \quad (2.2.12)$$

where the eigenenergies are given by Eq. (2.2.8).

Coulomb term

In order to derive the second-quantized representation of the Coulomb operator, we will give an analytical solution of Eq. (2.2.11) which is valid for all possible values of $\mathbf{P}, \mathbf{Q}, \mathbf{R}, \mathbf{S}$. For the evaluation of Eq. (2.2.11), we use the Fourier representation [146],

$$\frac{1}{|\mathbf{r}_1 - \mathbf{r}_2|} = \frac{1}{2\pi} \int d\mathbf{q} \frac{1}{q} e^{i\mathbf{q}(\mathbf{r}_1 - \mathbf{r}_2)}, \quad (2.2.13)$$

where $\int d\mathbf{q} = \int_{-\infty}^{\infty} dq_x \int_{-\infty}^{\infty} dq_y$. We insert Eq. (2.2.13) into Eq. (2.2.11), using polar coordinates we obtain

$$h_{\mathbf{P}\mathbf{Q}\mathbf{R}\mathbf{S}} = \frac{e^2 \mathcal{C}}{\epsilon} \int_0^{\infty} dq K_{\mathbf{P},\mathbf{S}}(q) K_{\mathbf{R},\mathbf{Q}}(q)^* \delta_{P_2-S_2, R_2-Q_2}, \quad (2.2.14)$$

where the delta-function on the right-hand side reflects the conservation of total angular momentum and we defined the coefficient

$$\mathcal{C} = \frac{(-1)^{P_1+Q_1+S_1+R_1}}{\pi^2 2^{(P_2+Q_2+S_2+R_2+4)/2}} \sqrt{\frac{P_1! Q_1! S_1! R_1!}{P_{\Sigma}! Q_{\Sigma}! S_{\Sigma}! R_{\Sigma}!}} \quad (2.2.15)$$

and the integral

$$K_{\mathbf{P},\mathbf{S}}(q) = \frac{2\pi}{i^{S_2-P_2}} \int_0^{\infty} dr_1 r_1^{P_2+S_2+1} e^{-\frac{1}{2}r_1^2} L_{P_1}^{(P_2)}\left(\frac{r_1^2}{2}\right) L_{S_1}^{(S_2)}\left(\frac{r_1^2}{2}\right) J_{P_2-S_2}(qr_1). \quad (2.2.16)$$

In the above derivation we made use of the integral representation of the Bessel function [147]

$$J_n(x) = \frac{i^n}{2\pi} \int_0^{2\pi} d\theta e^{i(n\theta - x \cos(\theta))}. \quad (2.2.17)$$

Integrating the right-hand side of Eq. (2.2.16) leads to

$$K_{\mathbf{P},\mathbf{S}}(q) = 2^{S_2+1} \pi i^{P_2-S_2} \frac{S_{\Sigma}!}{S_1!} (-1)^{P_1+S_1} q^{P_2-S_2} e^{-\frac{1}{2}q^2} L_{P_1}^{(S_1-P_1)}\left(\frac{q^2}{2}\right) L_{S_{\Sigma}}^{(P_{\Sigma}-S_{\Sigma})}\left(\frac{q^2}{2}\right). \quad (2.2.18)$$

By substituting $x_j = q_j^2/2$ for $j = 1, 2$ and using Eq. (2.2.18), the solution for Eq. (2.2.14) is given by

$$h_{\mathbf{PQRS}}^{(i)} = \frac{e^2 \mathcal{C}^{(i)} \Gamma(p)}{\epsilon} \frac{1}{2^p} \left(\prod_{k=1}^4 \binom{n_k + \alpha_k}{n_k} \right) F_A^{(4)} \left[\begin{matrix} p, -n_1, -n_2, -n_3, -n_4; & \frac{1}{2}, \frac{1}{2}, \frac{1}{2}, \frac{1}{2} \\ \alpha_1 + 1, \alpha_2 + 1, \alpha_3 + 1, \alpha_4 + 1; & \frac{1}{2}, \frac{1}{2}, \frac{1}{2}, \frac{1}{2} \end{matrix} \right] \times \delta_{P_2 - S_2, R_2 - Q_2}, \quad (2.2.19)$$

where $\Gamma(x)$ denotes the Gamma function, $\mathcal{C}^{(i)}$, p , and n_j and α_j with $j = 1, 2, 3, 4$, are given in Table (2.3) for all possible values of the quantum numbers $P_1, P_2, Q_1, \dots, S_2$. The superscript (i) indicates that we consider the case where $P_2 - S_2 \geq 0$, the remaining case (ii) , where $P_2 - S_2 < 0$ can be obtained from symmetry, as indicated in the last row of Table 2.3. The function

$$F_A^{(4)} \left[\begin{matrix} p, -n_1, -n_2, -n_3, -n_4; & \frac{1}{2}, \frac{1}{2}, \frac{1}{2}, \frac{1}{2} \\ \alpha_1 + 1, \alpha_2 + 1, \alpha_3 + 1, \alpha_4 + 1; & \frac{1}{2}, \frac{1}{2}, \frac{1}{2}, \frac{1}{2} \end{matrix} \right] = \sum_{k_1, k_2, k_3, k_4=0}^{\infty} \frac{(p)_{k_1+k_2+k_3+k_4} (-n_1)_{k_1} (-n_2)_{k_2} (-n_3)_{k_3} (-n_4)_{k_4}}{(\alpha_1 + 1)_{k_1} (\alpha_2 + 1)_{k_2} (\alpha_3 + 1)_{k_3} (\alpha_4 + 1)_{k_4} 2^{k_1+k_2+k_3+k_4} k_1! k_2! k_3! k_4!}, \quad (2.2.20)$$

is known as the Lauricella function, where

$$(\lambda)_n = \frac{\Gamma(\lambda + n)}{\Gamma(\lambda)} \quad (2.2.21)$$

is the rising factorial (Pochhammer symbol). Since $-n_1, -n_2, -n_3, -n_4$ in Eq. (2.2.20) are non-positive integers, the series terminates after a finite number of terms. One can represent the Lauricella function as an integral of a product of lower-order hypergeometric functions [148], which results in

$$F_A^{(4)} \left[\begin{matrix} p, -n_1, -n_2, -n_3, -n_4; & \frac{1}{2}, \frac{1}{2}, \frac{1}{2}, \frac{1}{2} \\ \alpha_1 + 1, \alpha_2 + 1, \alpha_3 + 1, \alpha_4 + 1; & \frac{1}{2}, \frac{1}{2}, \frac{1}{2}, \frac{1}{2} \end{matrix} \right] = \boldsymbol{\xi} \cdot (\mathbf{p}), \quad (2.2.22)$$

where we defined the two column vectors

$$(\mathbf{p}) = ((p)_0, (p)_1, \dots, (p)_{n_1+n_2+n_3+n_4})^T \quad (2.2.23)$$

$$\boldsymbol{\xi} = (\xi_0, \xi_1, \dots, \xi_{n_1+n_2+n_3+n_4})^T, \quad (2.2.24)$$

with convolution coefficients

$$\xi_k = \sum_{p=0}^k \sum_{q=0}^p \sum_{r=0}^q \frac{(-n_1)_r (-n_2)_{q-r}}{(\alpha_1 + 1)_r (\alpha_2 + 1)_{q-r} (\alpha_3 + 1)_{p-q}} \frac{(-n_3)_{p-q} (-n_4)_{k-p}}{(\alpha_4 + 1)_{k-p} r! (q-r)! (p-q)! (k-p)! 2^k}. \quad (2.2.25)$$

A detailed derivation of the results of this section can be found in Appendix 2.A. While recent work provided analytic expressions for the two-body matrix elements in finite spherical quantum Hall systems [145], we are not aware of prior analytic expressions for the two-body matrix elements that include general LL mixing for a two-dimensional disk geometry setting.

Table 2.3: This table defines the coefficients $\mathcal{C}^{(i)}$, p , n_j and α_j for $j = 1, 2, 3, 4$ in Eq. (2.2.19) and defines the explicit integral form of the Coulomb matrix elements of Eq. (2.A.13). The various sub-cases (i.i)-(i.ix) are defined in Table 2.A.1 in the appendix. Note that the values for α_j follow from the definition of $L_{[n_1, n_2, n_3, n_4]}$ in Eq. (2.A.2) and we defined the compact notation $P_\Sigma = P_1 + P_2$. The expressions for case (ii) do not need to be calculated, as they follow from $h_{\mathbf{PQRS}}^{(ii)} = h_{\mathbf{SRQP}}^{(i)}$ as indicated by the last row of this table.

$h_{\mathbf{PQRS}}^{(i)} = \mathcal{C}^{(i)} \int_0^\infty dx x^{p-1} e^{-2x} L_{[n_1, n_2, n_3, n_4]}$							
case	$\mathcal{C}^{(i)}$	p	$[n_1, n_2, n_3, n_4]$	α_1	α_2	α_3	α_4
(i.i)	$(-1)^{S_1 - P_1 + Q_1 - R_1} \sqrt{\frac{P_1! P_\Sigma! R_\Sigma! R_1!}{2 S_1! S_\Sigma! Q_\Sigma! Q_1!}}$	$S_1 - P_1 + Q_\Sigma - R_\Sigma + 1/2$	$[P_1, P_\Sigma, R_1, R_\Sigma]$	$S_1 - P_1$	$S_\Sigma - P_\Sigma$	$Q_1 - R_1$	$Q_\Sigma - R_\Sigma$
(i.ii)	$(-1)^{S_\Sigma - P_\Sigma} \sqrt{\frac{P_\Sigma! P_1! Q_\Sigma! R_1!}{2 Q_1! S_1! S_\Sigma! R_\Sigma!}}$	$S_1 - P_1 + 1/2$	$[P_1, P_\Sigma, R_1, Q_\Sigma]$	$S_1 - P_1$	$S_\Sigma - P_\Sigma$	$Q_1 - R_1$	$R_\Sigma - Q_\Sigma$
(i.iii)	$(-1)^{S_\Sigma - P_\Sigma + R_1 - Q_1} \sqrt{\frac{P_1! P_\Sigma! Q_1! Q_\Sigma!}{2 S_1! S_\Sigma! R_1! R_\Sigma!}}$	$S_1 - P_1 + R_1 - Q_1 + 1/2$	$[P_1, P_\Sigma, Q_1, Q_\Sigma]$	$S_1 - P_1$	$S_\Sigma - P_\Sigma$	$R_1 - Q_1$	$R_\Sigma - Q_\Sigma$
(i.iv)	$(-1)^{Q_\Sigma - R_\Sigma} \sqrt{\frac{P_1! S_\Sigma! R_1! (R_\Sigma!)}{2 P_\Sigma! Q_1! Q_\Sigma! S_1!}}$	$Q_1 - R_1 + 1/2$	$[P_1, S_\Sigma, R_1, R_\Sigma]$	$S_1 - P_1$	$P_\Sigma - S_\Sigma$	$Q_1 - R_1$	$Q_\Sigma - R_\Sigma$
(i.v)	$\mathcal{B} = \sqrt{\frac{P_1! Q_\Sigma! S_\Sigma! R_1!}{2 P_\Sigma! Q_1! S_1! R_\Sigma!}}$	$P_2 - S_2 + 1/2$	$[P_1, S_\Sigma, R_1, Q_\Sigma]$	$S_1 - P_1$	$P_\Sigma - S_\Sigma$	$Q_1 - R_1$	$R_\Sigma - Q_\Sigma$
(i.vi)	$(-1)^{R_1 - Q_1} \sqrt{\frac{P_1! Q_\Sigma! Q_1! S_\Sigma!}{2 P_\Sigma! S_1! R_\Sigma! R_1!}}$	$R_1 - Q_1 + P_2 - S_2 + 1/2$	$[P_1, S_\Sigma, Q_1, Q_\Sigma]$	$S_1 - P_1$	$P_\Sigma - S_\Sigma$	$R_1 - Q_1$	$R_\Sigma - Q_\Sigma$
(i.vii)	$(-1)^{P_1 - S_1 + Q_\Sigma - R_\Sigma} \sqrt{\frac{S_1! S_\Sigma! R_1! R_\Sigma!}{2 P_\Sigma! P_1! Q_\Sigma! Q_1!}}$	$P_1 - S_1 + Q_1 - R_1 + 1/2$	$[S_1, S_\Sigma, R_1, R_\Sigma]$	$P_1 - S_1$	$P_\Sigma - S_\Sigma$	$Q_1 - R_1$	$Q_\Sigma - R_\Sigma$
(i.viii)	$(-1)^{P_1 - S_1} \sqrt{\frac{Q_\Sigma! S_1! S_\Sigma! R_1!}{2 P_\Sigma! P_1! Q_1! R_\Sigma!}}$	$P_\Sigma - S_\Sigma + 1/2$	$[S_1, S_\Sigma, R_1, Q_\Sigma]$	$P_1 - S_1$	$P_\Sigma - S_\Sigma$	$Q_1 - R_1$	$R_\Sigma - Q_\Sigma$
(i.ix)	$(-1)^{P_1 - S_1 + R_1 - Q_1} \sqrt{\frac{Q_1! Q_\Sigma! S_1! S_\Sigma!}{2 P_\Sigma! P_1! R_1! R_\Sigma!}}$	$(P_\Sigma - S_\Sigma + R_1 - Q_1 + 1/2)$	$[S_1, S_\Sigma, Q_1, Q_\Sigma]$	$P_1 - S_1$	$P_\Sigma - S_\Sigma$	$R_1 - Q_1$	$R_\Sigma - Q_\Sigma$
$h_{\mathbf{PQRS}}^{(ii)} = h_{\mathbf{SRQP}}^{(i)}$							

Due to the conservation of angular momentum in Eq. (2.2.19), the number of terms in the Hamiltonian scales at most as $O(N^4 M^3)$, reducing the order of the polynomial by one in M (which is the most costly parameter, since $N \ll M$).

2.2.3 System Hamiltonian in the LLL

As indicated by the absence of any P_2 dependence in Eq. (2.2.8), each LL is degenerate in the absence of interactions. For the LLL, the single-particle wave functions in symmetric gauge in Eq. (2.2.5) simplify to

$$\psi_{(0,P_2)}(\mathbf{r}) = \frac{1}{\sqrt{2\pi 2^{P_2} P_2!}} z^{P_2} e^{-\frac{1}{4}r^2}. \quad (2.2.26)$$

The above wave functions are peaked on concentric rings, whose distance from the origin is proportional to the square root of the the angular coordinate,

$$R_d = l_B \sqrt{2(M+1)}, \quad (2.2.27)$$

where $l_B = \sqrt{\hbar c / (eB)}$. Not only for computational purposes is it of interest to introduce a cut-off for the angular momentum M that fulfills

$$P_2 \leq M. \quad (2.2.28)$$

Physically, this can be interpreted as a confinement of the electrons on a disk with radius R_d . Coulomb repulsion will typically force the electrons to fly away from each other while a confinement counteracts this repulsion by forcing them to stay within a confined region of space. Note that by fixing the Radius R_d , the cut-off M —and therefore the degeneracy of the LLs—can be tuned by changing the magnetic field B .

The cyclotron energy $\hbar\omega_c$ is proportional to the transversal magnetic field B and sets the spacing between the LLs. From the form of the wave functions one can deduce that the degeneracy within each LL is approximately given by $N_{\text{deg}} = AB/\phi_0$, where A is the area spanned by the confinement and B is the transversal magnetic field [9]. We will restrict our simulations to systems where the number of particles N_{el} is smaller than the degeneracy N_{deg} within each LL. For such a configuration, in the limit of sufficiently large magnetic field, only states in the LLL will be occupied and we can neglect coupling to states to higher LLs.

The filling factor ν is the number of electrons per flux quantum penetrating the sample and defined as [9]

$$\nu = \frac{\rho\phi_0}{B}, \quad (2.2.29)$$

where ρ is the 2D electron density and the flux quantum $\phi_0 = hc/e$. Assuming a homogeneous density of the N_{el} electrons on a disk with radius R_d (Eq. (2.2.27)) and restricting to the LLL, we get the density $\rho = N_{\text{el}}/(\pi R_d^2)$. With this we can derive

$$\nu = \frac{N_{\text{el}}}{M+1}. \quad (2.2.30)$$

Fixing the filling factor ν at constant magnetic field thus also results in a constant electron density ρ . The factor e^2/ϵ which appears in front of the Coulomb term in Eq. (2.2.14)

merely sets the overall energy scale when working in the LLL, which is why we set it equal to one in our numerical simulations and the integral expressions. For simulations that incorporate the effect of LL mixing, one would have to include the factor e^2/ϵ in front of the Coulomb terms again, as well as the cyclotron energy $\hbar\omega_c$, since it sets the energy spacing between LLs and depends on the strength of the transversal magnetic field.

For the disk geometry in the LLL approximation, we use the compact result of Ref. [146], where the matrix elements $h_{\mathbf{PQRS}} = M_{mn}^l$ are expressed as finite sums of fractions of factorials

$$M_{mn}^l = C_{lmn} (A_{mn}^l B_{nm}^l + B_{mn}^l A_{nm}^l) \quad (2.2.31)$$

$$C_{lmn} = \sqrt{\frac{(m+l)!(n+l)! \Gamma(l+m+n+3/2)}{m!n! \pi 2^{l+m+n+2}}} \quad (2.2.32)$$

$$A_{mn}^l = \sum_{i=0}^m \binom{m}{i} \frac{\Gamma(i+1/2)\Gamma(l+i+1/2)}{(l+i)!\Gamma(l+n+i+3/2)} \quad (2.2.33)$$

$$B_{mn}^l = \sum_{i=0}^m \binom{m}{i} \frac{\Gamma(i+1/2)\Gamma(l+i+1/2)}{(l+i)!\Gamma(l+n+i+3/2)} \times (l+2i+1/2), \quad (2.2.34)$$

where all indices P_1, Q_1, R_1, S_1 are equal to zero, $l = P_2 - S_2$, $m = S_2$, $n = Q_2$. The fact that instead of the four angular momentum quantum numbers P_2, Q_2, R_2, S_2 , only three such numbers, l, m, n , appear in Eq. (2.2.31) is a manifestation of the conservation of angular momentum. Due to the appearance of fractions of large integers in the coefficients in Eqs. (2.2.32)-(2.2.34), numerical implementation of Eq. (2.2.31) for large system sizes has to be performed with great caution. A rash implementation will lead to numerical instabilities already below one hundred spin-orbitals. We compared the coefficients in Eq. (2.2.31) with Eqs. (2.2.19) and (2.2.22) for various system sizes and fillings within the LLL and they were in exact agreement up to numerical precision errors.

2.2.4 Mapping the second-quantized fermionic Hamiltonian to the Pauli basis

If one wants to simulate a fermionic system on a quantum computer, one needs to map the fermionic creation and annihilation operators onto qubit operators. Various such encodings have been studied, each with its own benefits and drawbacks [18, 149–157]. Let $\sigma_j^{x,y,z}$ denote the Pauli- X, Y, Z matrix. We choose the Jordan-Wigner transformation [149] where a single fermionic raising or lowering operator is mapped to a simple qubit raising or lowering operator $\sigma_j^\pm = (\sigma_j^x \mp i\sigma_j^y)/2$, at the cost of up to $N_{\text{so}} - 1$ additional Pauli- Z operators,

$$c_j^\dagger = \sigma_j^+ \prod_{k=j+1}^{N_{\text{so}}} \sigma_k^z \quad (2.2.35)$$

$$c_j = \sigma_j^- \prod_{k=j+1}^{N_{\text{so}}} \sigma_k^z. \quad (2.2.36)$$

The Pauli- Z operator's role is to produce the sign factor that appears when acting with a fermionic operator on a Fock state [2], leading to the canonical fermionic anti-commutation

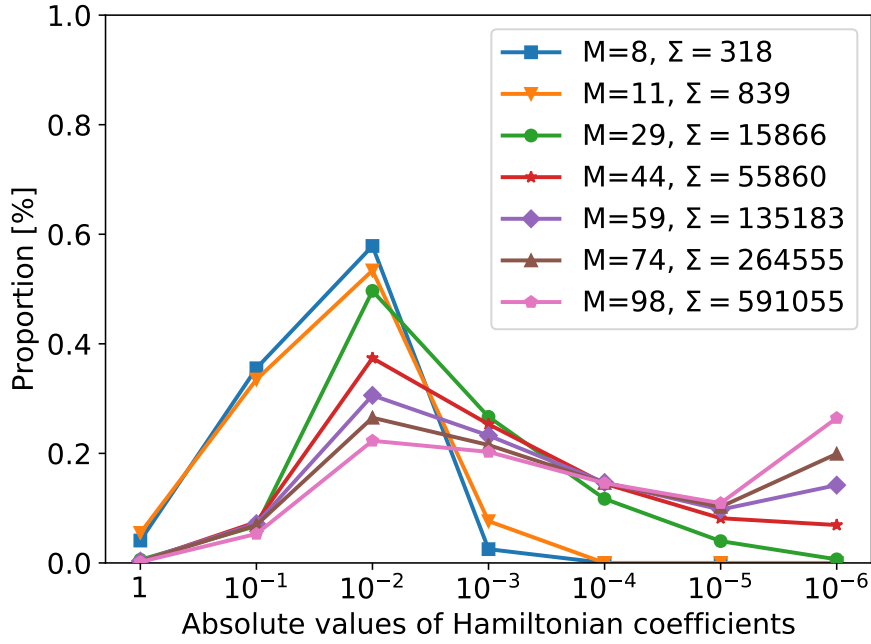


Figure 2.1: Scatter plot of the distribution of coefficient magnitudes of the real-valued coefficients of Eq. (2.2.37) for various numbers of spin orbitals at a filling $\nu = 1/3$ in the LLL $N = 0$. Each discrete point on the x -axis describes the value range of $|\omega_j|^2$, for instance 10^{-2} contains all values x within the range $x \in [10^{-3}, 10^{-2})$. Note that the largest coefficients are on the left-hand side, while the smallest coefficients are on the right-hand side of the graph. The y -axis displays the ratio of the number of terms within the range of a given x w.r.t. the total number Σ of non-zero Hamiltonian coefficients in Eq. (2.2.37).

relations. Inserting Eqs. (2.2.35) and (2.2.36) into Eq. (2.2.9) results in the qubit Hamiltonian which is equivalent to the original system Hamiltonian and that can be written as a sum of positive real-valued coefficients ω_j (not to be confused with the cyclotron frequency ω_c) times a phase factor $e^{i\theta_j}$ (whose sole purpose is to absorb the minus sign of negative Hamiltonian coefficients f_{pq} and h_{pqrs}) times a tensor product of Pauli operators $P_j \in \{\mathbb{1}_2, \sigma^x, \sigma^y, \sigma^z\}^{\otimes N_{so}}$,

$$H = \omega_0 e^{i\theta_0} \mathbb{1}_{2^{N_f}} + \sum_{j=1}^{\Sigma} \omega_j e^{i\theta_j} P_j. \quad (2.2.37)$$

In Fig. 2.1, we study the distribution of the range of values for the sum of squared coefficients for various system sizes. A general shift of the coefficients towards much smaller coefficient magnitudes with growing system size becomes apparent. Scaling analysis like these are important for determining upper bounds on the number of required measurements to estimate the ground state energy within a given precision and for various variational Ansatzes $U(\theta)$ of the VQE, such as the Hamiltonian variational Ansatz [158], as it depends on both the number and the relative weight of the non-zero terms appearing in the Hamiltonian of Eq. (2.2.37).

Now that we have derived the system Hamiltonian of the FQH system in second quantization, the following section will give an estimate for the gate complexity to estimate

its ground state energy using a state-of-the-art Hamiltonian simulation algorithm designed for an error-corrected universal quantum computer.

Section 2.3

HAMILTONIAN SIMULATION THROUGH LINEAR COMBINATION OF UNITARIES

While Trotter based methods are likely the most efficient technique for implementing quantum simulations of the fractional quantum Hall effect on near-term quantum computers, other methods might be more competitive within cost models appropriate for error-corrected quantum computing. Within fault-tolerance the key cost model of interest is often the number of non-Clifford gates (usually T gates) required for the simulation because within error-correcting codes, T gates require orders of magnitude more resource to realize than Clifford gates and thus limit the calculation size [159].

When studying quantum simulations of electronic structure within the context of error-correction we usually focus on state preparation using phase estimation. The quantum phase estimation algorithm [139] allows one to measure the phase accumulated on a quantum register under the action of a unitary operator. To estimate this phase to within error ϵ one must apply the unitary a number of times scaling as $O(1/\epsilon)$. Furthermore, some varieties of phase estimation allow one to perform this measurement projectively, which enables sampling in the eigenbasis of the unitary. In the context of quantum simulation, this unitary usually corresponds to time evolution under the system Hamiltonian H for time t with eigenvalues e^{-iHt} [138]. However, some recent papers [100, 102] have advocated instead that one perform phase estimation on a quantum walk with eigenvalues $e^{\pm i \arccos(H)}$ which is often possible to realize with lower overhead. Performing phase estimation on either operator will give the same information [102]. For either strategy, performing projective phase estimation on this operator will collapse the system register $|\psi\rangle$ to an eigenstate of the Hamiltonian with a probability that depends on the initial overlap between $|\psi\rangle$ and the eigenstate of interest. Thus, if $H|n\rangle = E_n|n\rangle$ then performing phase estimation will project the system register to the eigenstate $|n\rangle$, and readout the associated eigenvalue E_n with probability $p_n = \langle\psi|n\rangle\langle n|\psi\rangle$. Therefore, the number of times that one must repeat phase estimation to prepare eigenstate $|n\rangle$ with high probability scales as $O(1/p_n)$. Here, we focus on the implementation of circuits that realize a quantum walk with eigenvalues $e^{\pm i \arccos(H)}$. The same strategies can be used to synthesize time evolution with additional logarithmic overheads, by using quantum signal processing [108].

The FQHE Hamiltonian described in Section 2.2 is a special case of the electronic structure Hamiltonian studied in quantum chemistry. Currently, the lowest T complexity quantum algorithms for simulating chemistry are all based on LCU methods [137]. LCU methods include Taylor series methods [160], qubitization [161], and Hamiltonian simulation in the interaction picture [162]. These methods were applied to realize quantum algorithms for electronic structure in Refs. [48, 50, 102, 143, 163, 164] and elsewhere. All LCU methods involve simulating the Hamiltonian as a linear combination of unitaries,

$$H = \sum_{\ell=1}^L \omega_{\ell} U_{\ell}, \quad \lambda = \sum_{\ell=1}^L |\omega_{\ell}|, \quad (2.3.1)$$

where U_{ℓ} are unitary operators, ω_{ℓ} are scalars, and λ is a parameter that determines the

complexity of these methods. The Hamiltonians in this paper satisfy this requirement once mapped to qubits (see Section 2.2.4) since strings of Pauli operators are unitary.

LCU methods perform quantum simulation in terms of queries to two oracle circuits defined as

$$\text{SELECT } |\ell\rangle |\psi\rangle \mapsto |\ell\rangle U_\ell |\psi\rangle, \quad (2.3.2)$$

$$\text{PREPARE } |0\rangle^{\otimes \log(L)} \mapsto \sum_{\ell=1}^L \sqrt{\frac{\omega_\ell}{\lambda}} |\ell\rangle, \quad (2.3.3)$$

where $|\psi\rangle$ is the system register and $|\ell\rangle$ is an ancilla register which usually indexes the terms in the linear combinations of unitaries in binary and thus contains $\log(L)$ ancillae. LCU methods can perform time evolution with gate complexity scaling as

$$\tilde{O}((C_S + C_P) \lambda t), \quad (2.3.4)$$

where \tilde{O} indicates that polylogarithmic factors in the scaling are suppressed, C_S and C_P are the gate complexities of SELECT and PREPARE respectively, and t is time. Specifically, if the goal is to implement quantum phase estimation to estimate energies or project into an eigenstate of the Hamiltonian then the T cost (with constant factors) scales as

$$\frac{\sqrt{2}\pi\lambda(C_S + C_P)}{\Delta E}, \quad (2.3.5)$$

where ΔE is the target precision in phase estimation (in the same units as λ) [102].

In order to simplify scaling arguments, we will only consider scaling in terms of the cutoff in angular momentum M and neglect the contribution due to the N LLs in the following. In numerical studies of the FQHE, one typically only considers a handful of LLs (most of the times only a single one), while trying to push the state space describing each LL (described by M) as high as possible, thus $N \ll M$, which leads to $O(N_{\text{so}}) \approx O(NM) \approx O(M)$. We also neglect the cost of performing FT † , which is a negligible additive cost to the complexity of phase estimation [165].

To implement the LCU oracles one must be able to coherently (i.e., using a quantum circuit) translate the index ℓ into the associated U_ℓ and ω_ℓ . U_ℓ are related to the second-quantized fermion operators (e.g., the $c_{\mathbf{P}\mathbf{Q}}^\dagger c_{\mathbf{R}\mathbf{S}}$) and the ω_ℓ are related to the coefficients (e.g., the $h_{\mathbf{P}\mathbf{Q}\mathbf{R}\mathbf{S}}$) described in Section 2.2.2. The U_ℓ have a structure that is straightforward to unpack in a quantum circuit using techniques described in Refs. [102, 143]. In particular, those papers show that one can implement the SELECT oracle with a complexity of $O(M)$ T gates and low constant factors in the scaling. In the context of quantum chemistry the ω_ℓ are typically challenging to compute directly from this index. However, as described in the prior section, for the Hamiltonians of interest in this paper we are able to compute the ω_ℓ efficiently from ℓ (which is essentially equivalent to computing the $h_{\mathbf{P}\mathbf{Q}\mathbf{R}\mathbf{S}}$ from the indices $\mathbf{P}, \mathbf{Q}, \mathbf{R}$ and \mathbf{S}). Still, the primary bottleneck for this implementation will be the realization of PREPARE rather than SELECT.

The spectrum of the fractional quantum Hall effect Hamiltonian derived in Section 2.2 can be simulated on a quantum computer using the low rank factorization strategy described in Ref. [143]. There, it is shown that one can perform phase estimation on an arbitrary basis electronic structure system with T complexity scaling as $O(N_{\text{so}}^{3/2}\lambda/\Delta E)$ where this λ is the true 1-norm of the Hamiltonian as defined in Eq. (2.3.1).

In Fig 2.2, we plot the scaling of this quantity for various system sizes in the LLL, where $O(N_{\text{so}}) = O(M)$ and M again denoting the cutoff in angular-momentum. Empirically

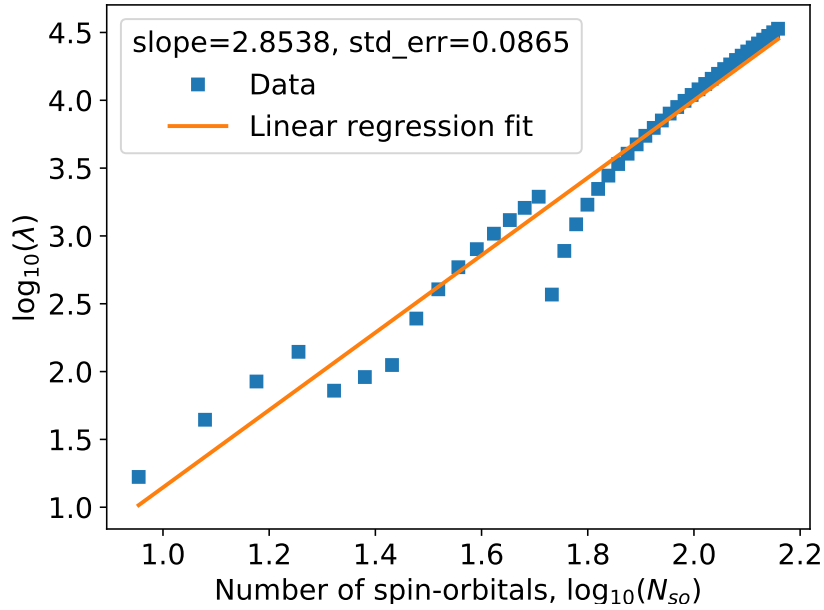


Figure 2.2: Linear regression fit of the scaling of the parameter $\lambda = \sum_{\ell=1}^L |\omega_{\ell}|$ for various system sizes in the LLL ranging from $M = 8, 11, 14, \dots, 144$, each blue square representing a system instance. Both, the x - and y -axes are on a (base-10) logarithmic scale.

we find that in this context $\lambda = O(M^{2.85})$ which leads to an overall T complexity of $O(M^{4.35}/\Delta E)$. Since the approach described in [143] is currently the lowest scaling approach to electronic structure simulations, the low rank factorization method with T complexity $O(M^{4.35}/\Delta E)$ is at present the most effective strategy in the current literature for simulating FQHE Hamiltonians restricted to the LLL. It should be noted that having a closed form for the one- and two-body Hamiltonian coefficients did not lead to a better scaling when we used an alternative simulation strategy, see Appendix 2.B for more details.

Section 2.4

FINDING AN INITIAL STATE

In this section we focus on the preparation of initial states on a gate-model based quantum computer. Our aim is to find an initial state $|\Psi_{\text{init}}\rangle$ which approximates the true ground state $|\Psi_0\rangle$ of the system Hamiltonian and possesses a non vanishing overlap

$$|\langle \Psi_{\text{init}} | \Psi_0 \rangle|^2 > 0, \quad (2.4.1)$$

where the left-hand side of the above equation defines the state fidelity. Moreover, we require these initial states to be both *efficiently computable on classical computers* and *efficiently preparable on a gate-based quantum computer*. Note that the initial state can serve as the starting point of quantum algorithms such as in Ref. [142], which is of course in general no longer efficiently simulatable on classical computers. The efficient construction of $|\Psi_{\text{init}}\rangle$ and the realization of e^{-iHt} , or in our case $e^{\pm i \arccos H}$ (neglecting the inverse quantum Fourier transform) are the main black box operations needed for Hamiltonian simulation. Even though one is in general not able to construct the accurate eigenstate, one can show that the success probability of measuring the desired energy using quantum phase estimation improves quadratically with the overlap of an initial state that is not the eigenstate of H [6].

Quantum algorithms designed to perform a digital quantum simulation of large system sizes often ignore the problem of finding an initial state fulfilling the above prerequisites with reasonable support on the ground state [40], even though it is well-known that overlaps of approximate states will decrease exponentially with system size due to the Van Vleck catastrophe [19]. While it is unclear whether this orthogonality catastrophe can ever be overcome, it is possible to delay the vanishing of the overlap by using more elaborate initial states.

We consider two algorithms to find a suitable initial state for our FQH system. The first algorithm, described in Section 2.4.1, makes use of generalized Hartree-Fock theory to find an initial state within the family of FGS following an imaginary time evolution [63]. The second algorithm, introduced in Section 2.4.2, uses a deterministic algorithm which samples from a large set of Slater determinants (which are contained in the family of FGS), to find a subset of determinants that are likely to have a large support on the exact ground state [40]. This state can be efficiently constructed using the PREPARE oracle defined in Eq. (2.3.3). While only the former algorithm is well-suited for NISQ era quantum computers, both algorithms may be used for state initialization of quantum phase estimation algorithms on error-corrected quantum computers.

2.4.1 Single-reference state

The goal of this section is to find an initial state within the family of pure FGS, since they can be prepared efficiently on a linearly connected qubit architecture [166–169]. A FGS is defined as [3, 41]

$$|\Psi_{\text{GS}}\rangle = U_{\text{GS}} |0\rangle, \quad (2.4.2)$$

where $|0\rangle$ is the fermionic vacuum and U_{GS} is a unitary operator that can be written as an exponential of a quadratic Hamiltonian times an imaginary prefactor. FGS are the ground states of non-interacting fermionic systems and are uniquely described by the one-particle reduced density matrix, which in case of particle number conservation is identical to the reduced covariance matrix (CM)

$$\Gamma_{ij} = \langle \Psi_{\text{GS}} | c_j^\dagger c_i | \Psi_{\text{GS}} \rangle, \quad (2.4.3)$$

where we want to highlight the (in the following derivation) convenient but unusual index ordering in the above definition. Since the CM is of dimension $(N_{\text{so}} \times N_{\text{so}})$, it can be efficiently computed on a classical computer, even though the state vector in Eq. (2.4.2) grows exponentially with system size. Since we consider number-conserving Hamiltonians, studying number-conserving FGS, for which the terms $\langle \Psi_{\text{GS}} | c_j^\dagger c_i^\dagger | \Psi_{\text{GS}} \rangle$ and $\langle \Psi_{\text{GS}} | c_j c_i | \Psi_{\text{GS}} \rangle$ vanish [170] is sufficient. It is for this reason that we choose the CM definition as in Eq. (2.4.3), which omits such correlators. Following Ref. [63], we describe in the remainder of this section how to find $|\Psi_{\text{GS}}\rangle$ as the lowest energy state which results from an imaginary time evolution of the CM.

Since our simulations are restricted to the LLL, we will neglect the quantum numbers indicating the LLs. The number-preserving system Hamiltonian can then be written as

$$H = \sum_{p,q=0}^{N_{\text{so}}} f_{pq} c_p^\dagger c_q + \frac{1}{2} \sum_{p,q,r,s=0}^{N_{\text{so}}} h_{pqrs} c_p^\dagger c_q^\dagger c_r c_s. \quad (2.4.4)$$

We will use a short-hand notation for the above Hamiltonian that summarizes the quadratic and quartic terms to $H = T + V$. Due to the anti-commuting properties of fermionic raising and lowering operators, the above Hamiltonian can always be recast in a form where the two-body matrix elements h_{pqrs} possess the following symmetries

$$h_{pqrs} = -h_{qprs} = -h_{pqsr} = h_{qpsr}. \quad (2.4.5)$$

Following [63], the imaginary time evolution of the density matrix $\rho(\tau)$ of a Hamiltonian is given by

$$\rho(\tau) = \frac{e^{-H\tau} \rho(0) e^{-H\tau}}{\text{tr}[e^{-2H\tau} \rho(0)]}, \quad (2.4.6)$$

and guides us to the ground state in the limit of τ going to infinity (τ denotes the imaginary time), provided the overlap of $\rho(0)$ with the ground state is non-zero [171]. Since the exponential contains quartic terms due to the interaction terms in Eq. (2.4.4), the imaginary time evolution will in general take us out of the family of FGS. By imposing that Wick's theorem holds, we restrict the evolution of Eq. (2.4.6) to a state-dependent quadratic Hamiltonian. Therefore, the solution of the imaginary time evolution will be the lowest energy state of the state-dependent quadratic Hamiltonian.

To derive an equation of motion for the CM, we first note that the time derivative of the density matrix is given by

$$d_\tau \rho = -\{H, \rho\} + 2\rho \text{tr}[H\rho], \quad (2.4.7)$$

—where $\{A, B\} = AB + BA$ is the anti-commutator—by simply taking the time derivative $d_\tau = \frac{d}{d\tau}$ on both sides of Eq. (2.4.6). Since the time evolution of the expectation value

of an (not explicitly time-dependent) operator A is given by $d_\tau \langle A \rangle = \text{tr}[A\dot{\rho}(\tau)]$, where $\langle A \rangle = \text{tr}[A\rho]$, we arrive at the following expression for the time evolution of the CM,

$$d_\tau \Gamma_{ji} = -\text{tr}[\{H, c_i^\dagger c_j\}\rho] + 2\Gamma_{ji}\text{tr}[H\rho]. \quad (2.4.8)$$

By inserting the Hamiltonian of Eq. (2.4.4) into Eq. (2.4.8) and restricting the density matrix to be drawn from the family of number conserving FGS, we can express the time evolution of the CM in terms of a state-dependent mean-field term,

$$d_\tau \Gamma_{ji} = -\{\Gamma, h_m(\Gamma)\}_{ji} + 2[\Gamma h_m(\Gamma)\Gamma]_{ji}, \quad (2.4.9)$$

and where

$$h_m(\Gamma) = f + 2\text{tr}_{1,4}[h\Gamma] \quad (2.4.10)$$

is the mean-field term describing the quadratic, but state-dependent Hamiltonian, where f is a two dimensional matrix with entries f_{pq} , h is a four-dimensional tensor with elements h_{pqrs} and

$$\text{tr}_{1,4}[h\Gamma] = \sum_{p,s=0}^{N_{\text{so}}} h_{pqrs}\Gamma_{sp} \quad (2.4.11)$$

is a partial trace operation. We present an explicit derivation of Eq. (2.4.9) in Appendix 2.C and note that our result is identical to the results in Refs. [41, 63]. We solve Eq.(2.4.9) numerically through a formal integration method as outlined in Appendix 2.E. The energy of the mean field state is given by

$$E_m = \text{tr}[f\Gamma] + \text{tr}[\text{tr}_{1,4}[h\Gamma]\Gamma]. \quad (2.4.12)$$

Since the matrix $[h_m, \Gamma]$ is anti-symmetric, $[h_m, \Gamma]^2$ is negative definite and leads to a monotonic decrease of the energy in time,

$$d_\tau E_m = 2\text{tr} \left[([h_m, \Gamma])^2 \right] \leq 0, \quad (2.4.13)$$

which is also observed in the numerical simulations, see Fig. 2.E.1 in Appendix 2.E. The imaginary time evolution will thus lead us to a (local) minimum in the energy landscape of a quadratic, but state-dependent Hamiltonian described by h_m in Eq. (2.4.10). If we denote with O_Γ the $(N_{\text{so}} \times N_{\text{so}})$ orthogonal matrix which diagonalizes the CM through

$$\Gamma = O_\Gamma \begin{pmatrix} 0 & & & & & & \\ & \ddots & & & & & \\ & & 0 & & & & \\ & & & 1 & & & \\ & & & & \ddots & & \\ & & & & & \ddots & \\ & & & & & & 1 \end{pmatrix} O_\Gamma^T, \quad (2.4.14)$$

where the number of 1s on the diagonal corresponds to the number of electrons N_{el} in the system, we can write the result of the imaginary time evolution in the basis where the FGS is a single Slater determinant of the form

$$|\Psi_{\text{init}}\rangle = \tilde{c}_1^\dagger \cdots \tilde{c}_{N_{\text{el}}}^\dagger |0\rangle, \quad (2.4.15)$$

where we defined a new set of fermionic creation and annihilation operators in the rotated spin-orbital basis

$$\tilde{c}_j = \sum_i (O_\Gamma)_{ij} c_i. \quad (2.4.16)$$

Using the generalized Hartree-Fock method of Ref. [63] as summarized in this section, one can readily apply the constructions scheme of e.g. Ref. [169] to implement a single Slater determinant as in Eq. (2.4.15) on a quantum computer in $N_{\text{so}}/2$ circuit depth using $\binom{N_{\text{so}}}{2}$ Givens rotations.

2.4.2 Multi-reference state

A single Slater determinant (as introduced in Section 2.4.1) is a state of independent particles and from the particle's perspective, it is unentangled [172]. Since the ground state of the FQH system is expected to be a highly entangled state, eventually, a single Slater determinant will have a poor overlap with the exact ground state. In order to simulate larger system sizes, one faces the challenge of improving the state overlap using a method complementary to the generalized Hartree-Fock approach, which is both, efficiently computable on a classical computer and efficiently implementable on a quantum computer. One way of improving the initial state overlap is by generating a multi-reference state, i.e. a linear combination of Slater determinants similar to Eq. (2.3.3),

$$|\Psi_{\text{init}}\rangle = \sum_{i=1}^L C_i |D_i\rangle, \quad (2.4.17)$$

where the sum runs over $L \ll 2^{N_{\text{so}}}$ values, C_i are real-valued coefficients with $\sum_i |C_i|^2 = 1$ and $|D_i\rangle$ are the "most important" Slater determinants according to a physically motivated ranking criterion (the symbol L used here should not be confused with the identical symbol we used to denote the number of terms of the LCU Hamiltonian in Eq. (2.3.1)). We will study the performance of the Adaptive Sampling Configuration Interaction (ASCI) algorithm [23, 173–175] in the FQH setting, which is a state-of-the-art algorithm used in quantum chemistry calculations to obtain highly accurate energy estimates for strongly correlated molecules, competitive with full configuration quantum Monte Carlo and density matrix renormalization group methods [23]. At the core of the algorithm lies a ranking criterion for the expansion coefficients C_i that determines which determinants $|D_i\rangle$ should be included in Eq. (2.4.17). We will give a brief overview of ASCI following Ref. [23] in Section 2.4.2, explain how we derive the fidelity of the resulting state in Section 2.4.2, and conclude with how a linear combination of Slater determinants could efficiently be implemented on a quantum computer in Section 2.4.2.

The ASCI algorithm

The ASCI algorithm is an iterative method to find the most important Slater determinants by sampling determinants based on a ranking criterion derived from conditions on a steady-state solution following an imaginary time evolution. Two determinant subspaces define the ASCI algorithm, namely, the core space and the target space, each containing $cdets$ - and $tdets$ -many determinants ($tdets \leq cdets$), respectively.

In the first iteration step the core space consists only of a single Slater determinant $|\Psi_{\text{GS}}\rangle$ obtained from the method outlined in Section 2.4.1, with corresponding energy

E_m as given by Eq. (2.4.12). The first step in each iteration consists of computing the space of all determinants which are connected with the core space through single- and double excitations, e.g. determinants generated by applying $c_p^\dagger c_q$ and $c_p^\dagger c_q^\dagger c_r c_s$. For all determinants generated in that manner one has to compute the coefficients

$$A_i = \sum_{\substack{j \neq i \\ j \in cdets}} \frac{H_{ij} C_j}{H_{ii} - E}. \quad (2.4.18)$$

Here, E describes the lowest energy eigenvalue from the previous diagonalization and $H_{ij} = \langle D_i | H | D_j \rangle$ are off-diagonal Hamiltonian matrix elements. In the first iteration we set $E = E_m$.

The computation of the amplitudes in Eq. (2.4.18) is motivated by the stationary state solution of an imaginary time propagation of a state Ansatz of the form defined by Eq. (2.4.17). One then chooses the largest *tdets* determinants from the sets $\{|C_i|\}$ and $\{|A_i|\}$ of core space and single- and double-excited core space determinants and diagonalizes the (*tdets* \times *tdets*)-dimensional reduced system Hamiltonian, keeping only the eigenvector belonging to the lowest eigenvalue E ⁴. This eigenvector will have entries $(C_1, C_2, \dots, C_{tdets})^T$, with each entry belonging to a unique Slater determinant of the target space. The *cdets* largest coefficients are kept and re-normalized and their respective determinants form the new core space in the next iteration step. One repeats these steps until the energy converges, which we generally observe after around four to five iterations for all system sizes studied (see Fig. 2.H.1 in Appendix 2.H).

One of the computationally more costly steps is the evaluation of the overlaps H_{ij} , which we discuss in more detail in Appendix 2.G.1 and 2.G.2. For all our ASCI simulations, we choose the core space to be identical to the target space of the previous iteration step, $L = tdets = cdets$. As outlined in Appendix (2.D), we transformed the Hamiltonian in Eq. (2.4.18) for the ASCI simulation into the eigenbasis of the CM using the transformation given by Eq. (2.4.16), where the Hartree-Fock state is a simple tensor product of N_{el} distinct fermionic creation operators acting on the fermionic vacuum state.

Overlap estimation

If the ASCI expansion in Eq. (2.4.17) includes all $\binom{N_{so}}{N_{el}}$ Slater determinants containing N_{el} electrons, the ASCI solution is identical to the Full Configuration Interaction (FCI) solution and will give the exact ground state of the system Hamiltonian⁵. We expand the exact solution as

$$|\Psi_0\rangle = \sum_{k=1}^{\text{FCI}} \tilde{C}_k |D_k\rangle \quad (2.4.19)$$

and compute the squared overlap w.r.t. the ASCI state in Eq. (2.4.17) containing $L \leq$ FCI determinants, which is identical to the support of the ASCI expansion on the exact solution, i.e. the state fidelity defined on the left-hand side of Eq. (2.4.1). Since the number of determinants in a FCI expansion grows exponential with system size, once

⁴Clearly, if you take a core determinant $|C_k\rangle$ and search all single- and double excitations of that determinant, chances are high that you will obtain determinants which are also elements of the core set. In that case, we keep the coefficient with the largest value (by magnitude) and discard the rest.

⁵FCI in our case refers to including all number-conserving determinants in the ASCI expansion—which grows exponentially with system size—and provides an exact solution, see e.g. Ref. [2].

we go beyond exactly solvable system sizes, we will no longer be able to talk about the support of a subset of determinants on the exact ground state of the system Hamiltonian, but rather on the ground state of the reduced system Hamiltonian which is spanned by the *tdets* determinants of the ASCI expansion.

Preparing a linear combination of Slater determinants on a quantum computer

Recent work showed that a linear combination of Slater determinants, required e.g. for realizing the mapping described by the PREPARE oracle in Section 2.3, could be implemented efficiently on a quantum computer through the use of a quantum read-only memory, whose purpose is to read classical data indexed by a quantum register [102]. The construction scheme was improved upon by reducing the number of ancillary qubits needed to 1, resulting in a state preparation protocol, where $|\Psi_{\text{init}}\rangle$ can be constructed using only $O(N_{\text{so}}L)$ gates [40], where L is here identical to the number of core and target space determinants in the ASCI expansion. As previously stated, while the single reference state method introduced in Section 2.4.1 is suitable for NISQ devices, the preparation of linear combination of Slater determinants outlined in Section 2.4.2 will require error-corrected quantum computers, as it demands the implementation of many layers of multi-qubit Toffoli-type gates, which are costly to implement [176].

Section 2.5

NUMERICAL RESULTS

In this section we present our numerical results for implementing a FGS state and a multi-reference state as proposed in Sections 2.4.1 and 2.4.2 for small instances.

We study the quality of the initial state Ansatz of a system containing N_{el} electrons in $N_{\text{so}} = 3N_{\text{el}}$ spin-orbitals, which corresponds to a filling of $\nu = 1/3$ in the LLL. This corresponds to a fixed electron density, which can be seen from Eqs. (2.2.29)-(2.2.30).

By performing a formal integration of the equations of motion of the CM given by Eq. (2.4.9), we obtain the mean-field solution $|\Psi_{\text{GS}}\rangle$ of the system Hamiltonian. The numerical method is detailed in Appendix 2.E and was performed using 10^5 time steps at step size $\Delta\tau = 0.01$ for all simulation results in Fig. 2.3 and Fig. 2.4, as well as in the simulations shown in Appendix 2.H. The mean-field energy converges for all cases well before the end of the imaginary time evolution and the number of particles is conserved throughout the simulation, as exemplified in Fig. 2.E.1 in the Appendix.

In Fig. 2.3, we study the support of the most important Slater determinants (i.o.w. those carrying the largest coefficients $|C_i|$) in the ASCI expansion of Eq. (2.4.17) for system sizes $N_{\text{so}} = 9, 12, 15, 18$. For each set of data points, we study how the support changes when enlarging the space of core determinants, keeping in mind that we set $cdets = tdets$. The horizontal axis displays the fraction of core determinants in the current ASCI expansion w.r.t. the FCI expansion. The very last data point in each of the plots compares the sum of the squared coefficients to the FCI expansion and the corresponding value is thus equivalent to the state fidelity F defined in Eq.(2.4.1) of the ASCI expansion. The single determinant expansion is equivalent to $|\Psi_{\text{GS}}\rangle$ and thus describes the mean-field behaviour. It drops from around $F \approx 0.4$ for the smallest system size in the upper-left corner to $F \approx 0.25$ for the largest simulated system size in the lower-right corner of Fig. 2.3. For all simulations, constructing an ASCI expansion of ten Slater determinants guarantees an initial state fidelity well above $F = 0.5$, where we assumed an error-free construction of the linear combination of Slater determinants.

In Fig. 2.4, we investigate the convergence of both, the fidelity F , as well as the energy E —which corresponds to the lowest energy eigenvalue obtained from diagonalizing the reduced system Hamiltonian in the ASCI algorithm—for system sizes $N_{\text{so}} = 9, 12, 15, 18$. The first (last) data point in each individual plot corresponds to the mean-field solution (FCI expansion / exact ground state). Each marker in Fig. 2.4 corresponds to an individual ASCI simulation. The convergence of the energy of the reduced Hamiltonian for each individual ASCI simulation is displayed in Appendix 2.H in Fig. 2.H.1 for a variety of core determinants, which shows that ASCI typically converges after about five iterations for the respective system sizes.

One can observe from Fig. 2.4, that the fidelity does not converge much faster than the energy, which makes ASCI an unsuitable candidate for estimating state overlap for intractable system sizes (given that this trend continuous) unlike the findings observed for the various physical systems studied in Ref. [40]. There, the argument is that if the fidelity were to converge much faster than the energy *and* the latter would start to converge already at reasonable system sizes, one would have a heuristic argument that supports the legitimacy of approximating the overlap of the initial state with the true ground state

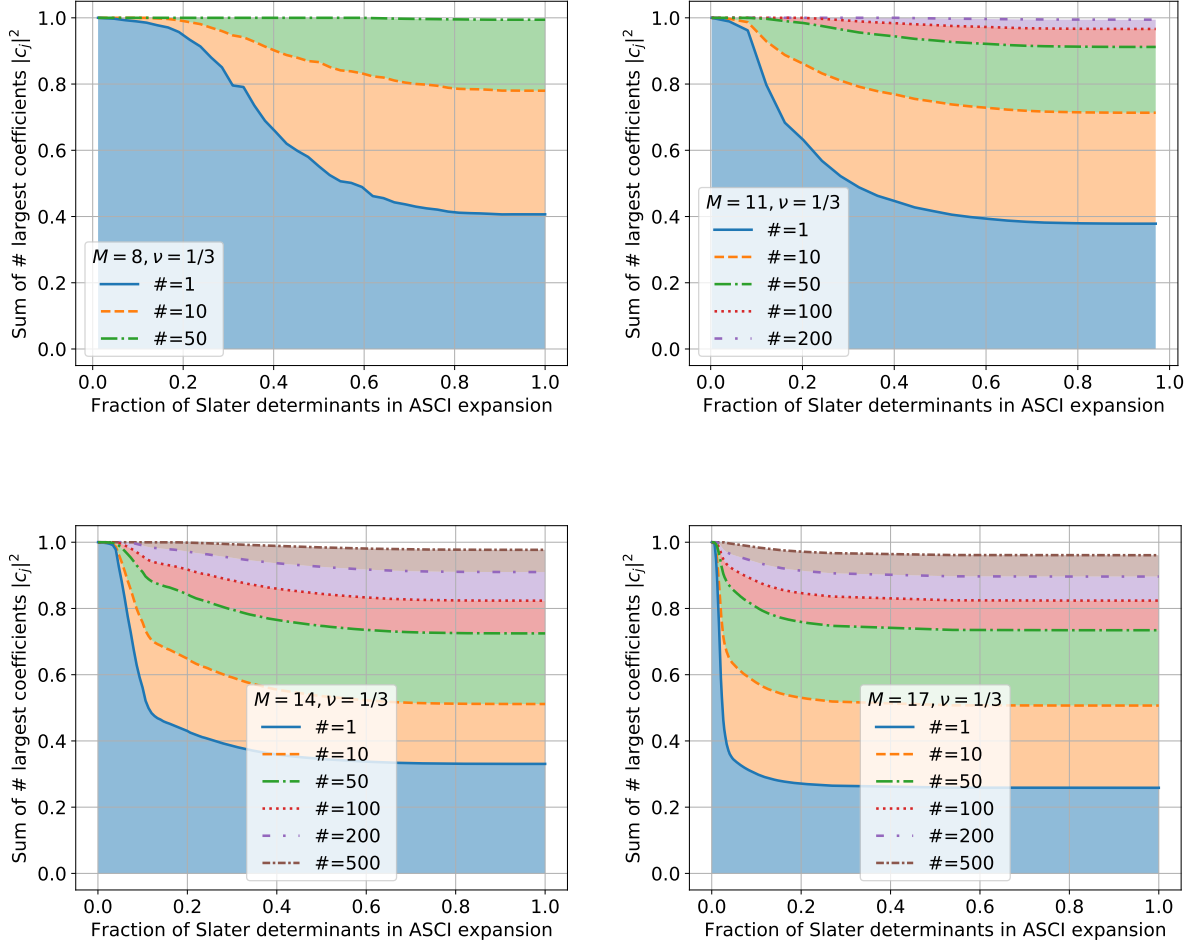


Figure 2.3: Scatter plots showing the sum of the squared coefficients in the ASCI expansion for various numbers of $tdets(=cdets)$ for system sizes of $N_{so} = 9, 12, 15, 18$ spin-orbitals at filling $\nu = 1/3$ in ascending order from the upper-left to the lower-right figure.. The blue curve (solid line) shows the behaviour of the mean-field solution $|\Psi_{GS}\rangle$. The rightmost points (where the fraction of Slater determinants in ASCI expansion is identical to 1) within each figure corresponds to the FCI expansion, i.e. all $\binom{N_{so}}{N_{el}}$ relevant Slater determinants are taken into account for those points and the sum of the $\#$ (where $\#$ is to be replaced with the number indicated in the grey box) is identical to the fidelity defined on the left-hand side of Eq. (2.4.1).

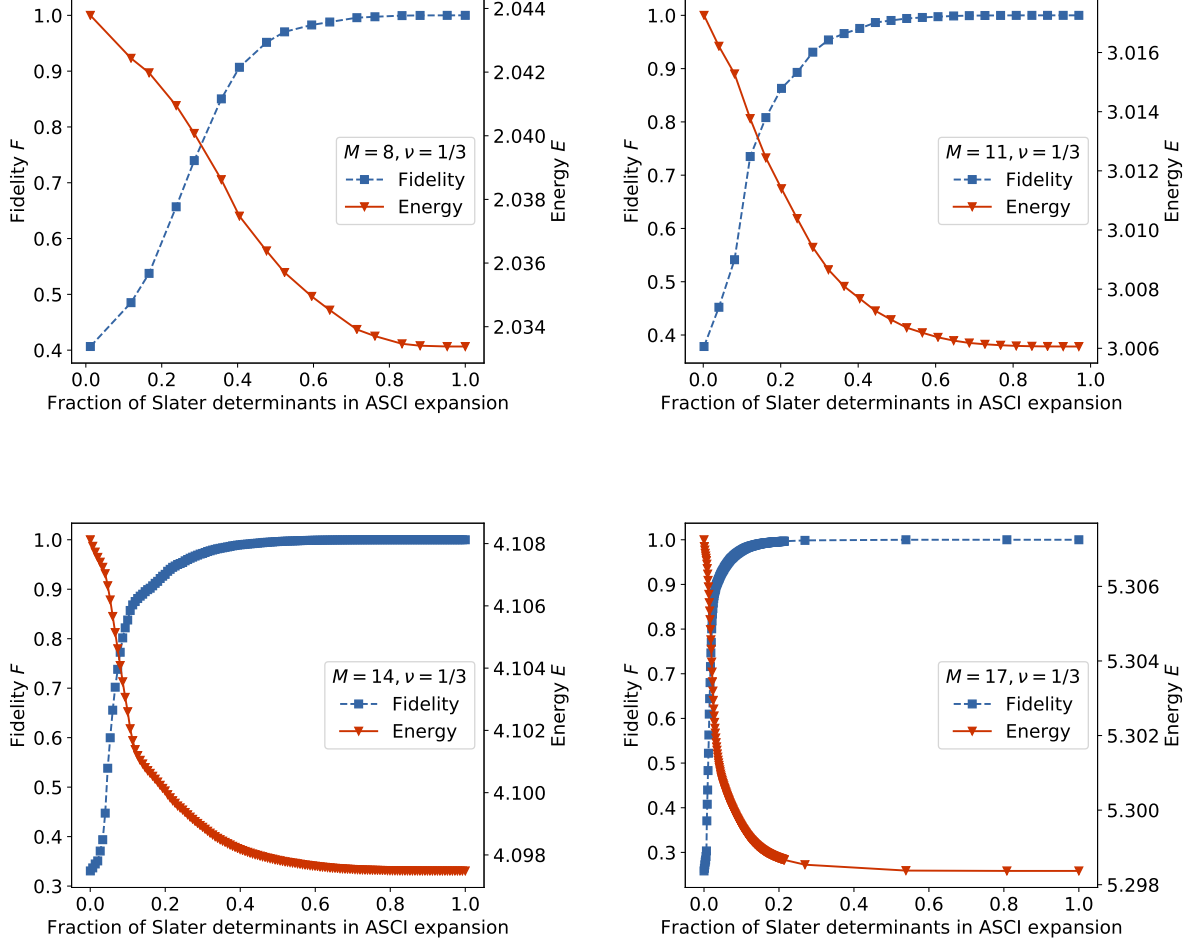


Figure 2.4: Scatter plots of the fidelity $F = |\langle \Psi_{\text{init}} | \Psi_0 \rangle|^2$ (blue colored squares) and the convergence of the energy E (red colored triangles) for various numbers of determinants in the ASCI expansion of Eq. (2.4.17). Instead of the total number of Slater determinants in the expansion, we plot the ratio w.r.t. the FCI expansion on the x-axis, with the first data point corresponding to the single reference state $|\Psi_{\text{GS}}\rangle$ and the last to the FCI expansion. The four plots show system sizes with $N_{\text{so}} = 9, 12, 15, 18$ spin orbitals at filling $\nu = 1/3$ in ascending order from the upper-left to the lower-right figure.

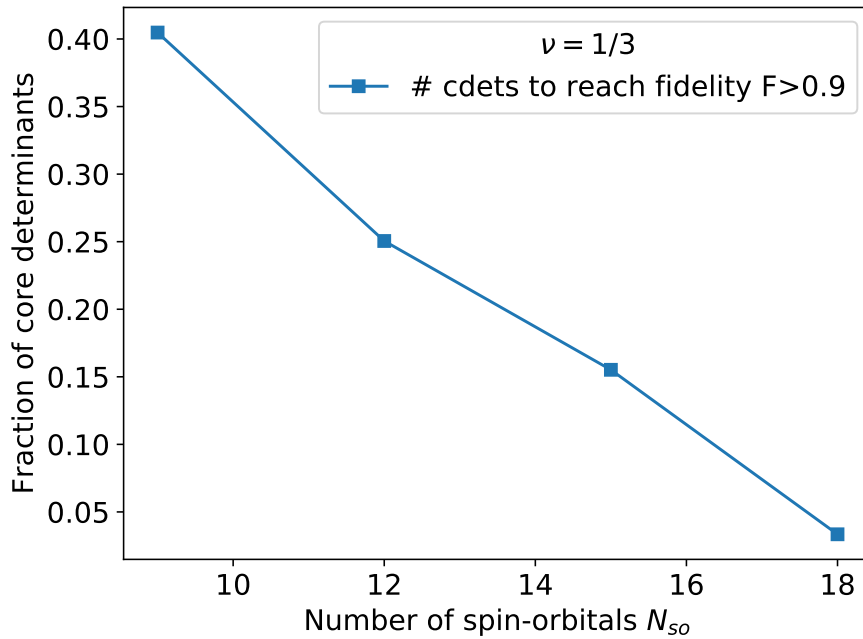


Figure 2.5: Scatter plot of the minimal number of ASCI core determinants (in terms of its ratio to the FCI expansion) needed to obtain state fidelities $F > 0.9$ for system sizes $N_{so} = 9, 12, 15, 18$. The values were obtained by linear extrapolation of the two data sets belonging to the largest (lowest) fidelity below (above) the threshold value $F = 0.9$. We note that the size of the determinant space corresponding to a FCI expansion displayed on the x-axis grows exponentially.

by using the largest possible ASCI expansion instead of $|\Psi_0\rangle$, since the latter is unknown. However, for the system sizes studied here, this behaviour was not observed.

In Fig. 2.5 we show how the minimal number of determinants needed to reach a fidelity of at least $F = 0.9$ scales with system size. The horizontal axis shows the number of spin-orbitals studied, where the number of FCI determinants grows exponentially, while the vertical axis displays the number of core determinant size w.r.t. the FCI to reach the desired fidelity, where the latter was obtained by linear extrapolation of the two simulated core sets displaying the largest (lowest) fidelity below (above) the threshold value $F = 0.9$. The close-to-linear behavior in Fig. 2.5 shows that for the system sizes studied here, only a sub-exponential increase in terms of the number $L = cdets = tdets$ of Slater determinants in the ASCI expansion is required to obtain an initial state $|\Psi_{init}\rangle$ with an overlap of at least $F = 0.9$ with the true ground state $|\Psi_0\rangle$. Larger-scale numerical simulation are needed to vindicate or disprove the observed trend for increasing values of N_{so} .

Section 2.6

DISCUSSION

In the following section, we discuss various avenues that could be explored in future studies, such as improving the model FQHE Hamiltonian, and choosing different geometries and basis sets for the system Hamiltonian, to using the Laughlin state as a proving ground to test heuristic Ansätze for NISQ algorithms. For completeness, we show how correlation functions (which contain all information about the respective physical system) can be computed for both the FGS and the multi-reference expansion.

2.6.1 Finite size studies

A natural question to ask, is whether it makes sense to perform a digital quantum simulation of a FQH system on a non error-corrected architecture, where one might be restricted to anywhere between tens up to a few hundreds of qubits. Current exact simulations of FQH systems are restricted to a handful of particles, but it turns out that the largest computer simulations today of around 50 spin-orbitals already exceeds the typical length scale (which is given by the magnetic length $l_B \approx 25\text{nm}/\sqrt{B[\text{T}]}$) of the problem considerably and it is therefore sensible to assume that one can make simulations that reflect properties which may extend to the thermodynamic limit even for relatively small numbers of particles. The goal of a digital quantum simulation of a quantum system can not be to try to simulate the actual system size, as the quantum resource requirements would be astronomical. As a small example taken from Ref. [9], a typical $1(\text{mm})^2$ sample contains roughly 10^9 electrons. A toy system of 100 electrons distributed among 250 spin-orbitals in the LLL (corresponding to $\nu = 0.4$) would lead to 10^{72} distinct ground state configurations, a number comparable to the number of particles in our universe. An error-corrected quantum computer would however only require 250 logical qubits (neglecting additional qubits required for the employed quantum algorithm) to represent this state.

2.6.2 Augmenting the model

In our discussion, we focused on the Coulomb interaction as it provides the key to the understanding of the FQHE. In order to make the system more realistic by taking into account effects that play a subdominant role in comparison to the electron-electron interaction described by H_2 , one can add additional terms to the system Hamiltonian of Eq. (2.2.1).

A two-dimensional electron gas is typically realized in experiments in dirty samples where random one-particle potentials of e.g. positive donor ions are scrambled across the probe (this is known as disorder). To account for their effect on the electrons, one therefore has to include one body potentials $\sum_j U(\mathbf{r}_j)$ as well, whose specific form depend on material properties. By computing the one-body coefficients due to the disorder terms, its effect could as well be included at a free cost in terms of qubit resources.

The role of the electron spin has been neglected in our derivations entirely, since we assumed that the magnetic field is large enough that all spin degrees of freedom are frozen. In order to account for the effect of the spin, one would have to add the Zeeman term

$g\mu\mathbf{B} \cdot \sum_{k=1}^{N_{\text{el}}} (\mathbf{S}_z)_j$, where $(\mathbf{S}_z)_j$ is the z -component of the spin of electron j , μ is the Bohr magneton and g the Landé g -factor. This would double the number of required qubits, since an additional register for each state would be required as a placeholder for the orbital spin component.

We have chosen a "soft" boundary (it is not a physical boundary) by introducing a cutoff in angular momentum. By using an harmonic trapping potential instead, one can simulate a physical boundary that allows one to exert pressure on the system by tuning the strength of the trapping potential.

We restrict ourselves to the disk geometry in symmetric gauge, but one could have also chosen a different gauge, such as the Landau gauge $\mathbf{A} = B(-y, 0, 0)^T$. Similarly, one can choose other geometries, for instance geometries which do not possess a boundary and are useful when studying bulk properties. Two prominent examples of such geometries are a two-dimensional sheet of electrons wrapped around the surface of a sphere, known as the Haldane sphere, or a two-dimensional sheet of electrons wrapped around a cylinder with periodic boundary conditions, which constitutes a torus geometry. See e.g. Ref. [177] for more details on the torus geometry and Ref. [145] for Hamiltonians that incorporate LL mixing within the Haldane sphere geometry.

2.6.3 Using the Laughlin wave function as a sanity check for the variational Ansatz

It is well known that for small system sizes the Laughlin wave function has a large overlap with the ground state of the FQH Hamiltonian in the LLL [120]. However, it is not the exact ground state of the FQH Hamiltonian, but rather the ground state of different, so-called parent Hamiltonian [122, 178–180]. To our knowledge, there has yet to appear a quantum circuit that efficiently constructs the Laughlin wave function for various filling factors ν , with the exception of integer filling factors [181]. Even though a Fock-space representation of the Laughlin state exists [182], it is not clear to us how this could be efficiently mapped onto a quantum circuit. An efficient quantum algorithm for generating the Laughlin state (or related states describing higher filling factors such as the Moore-Read state [121]) would most likely be of vital importance for digital quantum simulations of the FQHE Hamiltonian. Furthermore, a recent paper introduced a classically efficient variational method going beyond FGS to enable the study of FQHE systems in the spirit of composite fermions [41], but so far this method has not yet been applied to FQH systems and is not clear how well it will improve over a generalized Hartree-Fock Ansatz.

Even without an efficient algorithm for the implementation of the Laughlin state at hand, it could still play an important role for choosing appropriate variational Ansätze of the VQE algorithms. If a variational Ansatz would approximate the Laughlin wave function (by performing a VQE simulation with its corresponding parent Hamiltonian), it would be a strong indicator that the variational Ansatz can construct states that lie in the same universality class as the Laughlin wave function. Since the Laughlin wave function is an analytic expression, one can compare the results measured by a quantum computer with the theoretically predicted behavior even for large system sizes.

2.6.4 Computing correlation functions

In order to be able to extract ground state properties, such as the one-particle reduced density matrix, the pair correlation function and static structure factor, one has to com-

pute the expectation values of products of the fermionic field operators, which can be performed efficiently on a quantum computer [167, 169]. We define the fermionic field operators

$$\hat{\Psi}^\dagger(\mathbf{r}) = \sum_p \eta_p^*(\mathbf{r}) c_p^\dagger \quad (2.6.1)$$

$$\hat{\Psi}(\mathbf{r}) = \sum_p \eta_p(\mathbf{r}) c_p, \quad (2.6.2)$$

and the one-particle reduced density matrix and pair correlation function

$$G_1(\mathbf{r}, \mathbf{r}') = \langle \hat{\Psi}^\dagger(\mathbf{r}) \hat{\Psi}(\mathbf{r}') \rangle \quad (2.6.3)$$

$$G_2(\mathbf{r}, \mathbf{r}') = \langle \hat{\Psi}^\dagger(\mathbf{r}) \hat{\Psi}^\dagger(\mathbf{r}') \hat{\Psi}(\mathbf{r}') \hat{\Psi}(\mathbf{r}) \rangle. \quad (2.6.4)$$

The one-particle reduced density matrix $G_1(\mathbf{r}, \mathbf{r}')$ measures the values of the fermionic field operators at points \mathbf{r} and \mathbf{r}' and is identical to the electron density for $\mathbf{r} = \mathbf{r}'$. Thus, the number of electrons is given by $N_{\text{el}} = \text{tr}[G_1(\mathbf{r}, \mathbf{r})]$. The pair correlation function $G_2(\mathbf{r}, \mathbf{r}')$ is a measure of the density correlations and is proportional to the pair distribution function. By combining Eqs. (2.6.1)-(2.6.4), the measurement of correlation functions can be broken down into measurements of sums of quartic and quadratic fermionic operator expectation values.

For FQH states and more specifically for states describing a uniform density (at least inside the disk) isotropic liquid, one expects from extrapolation of finite system results that the one-particle reduced density matrix has an absence of off-diagonal long-range order [183],

$$\lim_{|\mathbf{r}-\mathbf{r}'| \rightarrow \infty} G_1(\mathbf{r}, \mathbf{r}') = 0 \quad (2.6.5)$$

and that the FQH state is a quantum liquid, which is characterized by [184]

$$\lim_{|\mathbf{r}-\mathbf{r}'| \rightarrow \infty} G_2(\mathbf{r}, \mathbf{r}') = \text{constant}, \quad (2.6.6)$$

(see e.g. chapters 8 and 12 in [9]) as opposed to the mean-field solution that produces a crystal and whose pair correlation function oscillates all the way to infinity. Any approximate ground state generated either through VQE approaches on NISQ devices or more elaborate methods such as ASCI (or the method introduced in [142]) should be able to reproduce the characteristic behavior as predicted by Eqs. (2.6.5)-(2.6.6). In Appendix 2.F, we give analytic expressions on how $G_2(\mathbf{r}, \mathbf{r}')$ may be efficiently computed on a classical computer for the FGS and show how multi-reference state approaches can be computed, given that the latter is kept to tractable system sizes. We also show the crystal-like patterns observed in the pair correlation function for a FGS Ansatz in Fig. 2.F.1 of Appendix 2.F.

Another physical quantity of interest regarding FQH states is the Rényi entropy, which contains information about whether the underlying entanglement obeys an area or volume law and whether the system is in a insulating or conducting phase. An explicit quantum circuit for measuring the Rényi entropy w.r.t. the Laughlin state on a quantum computer is given in Ref. [136].

Section 2.7

CONCLUSION AND OUTLOOK

We have presented an *ab-initio* roadmap to simulate the FQH Hamiltonian. We derived efficiently computable analytical expressions for the respective one- and two-body Hamiltonian coefficients which allow for LL mixing. Using the low-rank factorization method of Ref. [143] to extract the Hamiltonian eigenspectrum, we found a T gate complexity of $O(M^{4.35}/\Delta E)$ to estimate the energy to precision ΔE . This presents the current most efficient method to simulate the spectrum of the FQH Hamiltonian on an error-corrected quantum computer. We performed small-scale numerical simulations within the LLL to investigate the initial state fidelities of two efficiently computable and preparable Ansätze based on the generalized Hartree-Fock method and the ASCI algorithm, suitable for NISQ and error-corrected quantum processors, respectively. While the latter method shows a sub-exponential scaling in the required number of determinants to reach high fidelity initial states, larger scale numerical simulations are needed to better determine the large system-size behavior. In addition, scaling analysis for the parameter λ for systems including higher LLs are needed to discover the respective gate complexity for simulations beyond the LLL. To further improve the initial state Ansatz, an efficient implementation of Laughlin-type states should be a major focus of future work.

The authors thank Ryan Babbush for initiating the discussion on Hamiltonian simulation, suggesting to use a strategy that exploits the analytical form of the Hamiltonian coefficients and for critically reviewing parts of the draft.

At the time of this writing, a preprint of the work presented in this chapter can be found on <https://arxiv.org/abs/2003.02517> and is currently under review at a scientific journal.

APPENDIX

Section 2.A

DERIVATION OF ANALYTICAL RESULT FOR THE COULOMB MATRIX ELEMENTS

We will evaluate Eq. (2.2.14), which displays the trivial symmetry $h_{\mathbf{P}\mathbf{Q}\mathbf{R}\mathbf{S}} = h_{\mathbf{Q}\mathbf{P}\mathbf{S}\mathbf{R}}$ due to the indistinguishability of electrons. For the evaluation of Eq. (2.2.14), we use the Fourier representation of the Coulomb operator [146], more specifically,

$$\frac{1}{|\mathbf{r}_1 - \mathbf{r}_2|} = \frac{1}{2\pi} \int d\mathbf{q} \frac{1}{q} e^{i\mathbf{q}(\mathbf{r}_1 - \mathbf{r}_2)}. \quad (2.A.1)$$

For simplicity, we introduce a shorthand notation for a product of Laguerre polynomials of two quantum number tuples belonging to the same particle ⁶,

$$L_{[S_j, R_k, \dots]}(x) = L_{S_j}^{(P_j - S_j)}(x) L_{R_k}^{(Q_k - R_k)}(x) \dots \quad (2.A.2)$$

In the following, one has to distinguish between two cases: case (i), where $P_2 - S_2 \geq 0$ and case (ii), where $P_2 - S_2 < 0$. We will however only have to consider case (i), since case (ii) follows from the integral symmetry $h_{\mathbf{P}\mathbf{Q}\mathbf{R}\mathbf{S}}^{(ii)} = h_{\mathbf{S}\mathbf{R}\mathbf{Q}\mathbf{P}}^{(i)*}$. Moving to complex plane by substituting \mathbf{r}_j with $r_j e^{-i\theta_j}$ and \mathbf{q} with $q e^{-i\alpha}$, we insert the Fourier transformation defined in Eq. (2.A.1) into Eq. (2.2.14) and write $\mathbf{q} \cdot \mathbf{r}_i = q r_i \cos(\alpha - \theta_i)$, which results in

$$h_{\mathbf{P}\mathbf{Q}\mathbf{R}\mathbf{S}} = \frac{e^2}{2\pi\epsilon} \iint_0^\infty dr_1 dr_2 dq \iint_0^{2\pi} d\theta_1 d\theta_2 d\alpha r_1 r_2 \psi_{\mathbf{P}}^*(\mathbf{r}_1) \psi_{\mathbf{Q}}^*(\mathbf{r}_2) \psi_{\mathbf{S}}(\mathbf{r}_1) \psi_{\mathbf{R}}(\mathbf{r}_2) \times e^{iq(r_1 \cos(\alpha - \theta_1) - r_2 \cos(\alpha - \theta_2))}. \quad (2.A.3)$$

First, the integration w.r.t. the polar variable α is performed. The result of this integration is a manifestation of the conservation of angular momentum due to the appearance of the delta function $\delta_{P_2 - S_2, R_2 - Q_2}$. Note that due to the conservation of angular momentum, the choice of $P_2 - S_2 \geq 0$ also implies that $R_2 - Q_2 \geq 0$. The expression after integrating out the polar degree of freedom reads

$$h_{\mathbf{P}\mathbf{Q}\mathbf{R}\mathbf{S}} = \frac{e^2 \mathcal{C}}{\epsilon} \int_0^\infty dq K_{\mathbf{P},\mathbf{S}}(q) K_{\mathbf{R},\mathbf{Q}}(q)^* \delta_{P_2 - S_2, R_2 - Q_2}, \quad (2.A.4)$$

where $K_{\mathbf{P},\mathbf{S}}(q)$ is defined as

$$K_{\mathbf{P},\mathbf{S}}(q) = \int_0^\infty dr_1 \int_0^{2\pi} d\tilde{\theta}_1 r_1^{P_2 + S_2 + 1} e^{i\tilde{\theta}_1(P_2 - S_2)} e^{-\frac{1}{2}r_1^2} L_{P_1}^{(P_2)}\left(\frac{r_1^2}{2}\right) L_{S_1}^{(S_2)}\left(\frac{r_1^2}{2}\right) e^{iqr_1 \cos(\tilde{\theta}_1)}. \quad (2.A.5)$$

We use the integral representation of the Bessel function of Eq. (2.2.17) to rewrite Eq. (2.A.5). For $y > 0$ and complex parameters α and ν , satisfying $\text{Re}\{\alpha\} > 0$ and $\text{Re}\{\nu\} > -1$ [185, 186] (the results for this type of integral given in standard literature [187, 188] are incorrect as they contain sign errors), we have

$$\int_0^\infty dx x^{\nu+1} e^{-\alpha x^2} L_m^{(\nu-\sigma)}(\alpha x^2) L_n^{(\sigma)}(\alpha x^2) J_\nu(xy) = (-1)^{m+n} (2\alpha)^{-\nu-1} y^\nu e^{-\frac{y^2}{4\alpha}} L_m^{(\sigma-m+n)}\left(\frac{y^2}{4\alpha}\right) \times L_n^{(\nu-\sigma+m-n)}\left(\frac{y^2}{4\alpha}\right), \quad (2.A.6)$$

⁶In this work, \mathbf{P} (\mathbf{Q}) and \mathbf{S} (\mathbf{R}) belong to particle 'one' ('two').

which has the same functional form as the integral in Eq. (2.2.16). We will further need the following identity for Laguerre polynomials, for $a, b \in \mathbb{Z}$,

$$\frac{(-x)^a}{a!} L_b^{(a-b)}(x) = \frac{(-x)^b}{b!} L_a^{(b-a)}(x), \quad (2.A.7)$$

which can be proven by simply inserting the definition of Laguerre polynomials into Eq. (2.2.6). Using the integral identity of Eq. (2.A.6), we can bring Eq. (2.A.5) into the form displayed in Eq. (2.2.18) and the explicit form of the integral for case (i) in Eq. (2.A.4) reduces to

$$h_{\mathbf{PQRS}}^{(i)} = \frac{e^2 \mathcal{C}^{(i)}}{\epsilon} \int_0^\infty dq f^{(1)}(q) f^{(2)}(q) \delta_{P_2 - S_2, R_2 - Q_2}, \quad (2.A.8)$$

where the constant $\mathcal{C}^{(i)}$ is given in Table 2.3.

Table 2.A.1: In order to use the integral formula of Eq. (2.A.12), one has to study the various parameter regimes and apply the transformation given in Eq. (2.A.7) to ensure that all requirements for using the integral formula are met. Note that the argument of the Laguerre polynomials are omitted.

case	parameter regime	integrand substitutions
(i.i)	$(S_1 - P_1 \geq 0) \wedge (P_\Sigma - S_\Sigma < 0)$ $\wedge (Q_1 - R_1 \geq 0) \wedge (R_\Sigma - Q_\Sigma < 0)$	$L_{S_\Sigma}^{(P_\Sigma - S_\Sigma)} = \frac{P_\Sigma!}{S_\Sigma!} (-x)^{S_\Sigma - P_\Sigma} L_{P_\Sigma}^{(S_\Sigma - P_\Sigma)}$, $L_{Q_\Sigma}^{(R_\Sigma - Q_\Sigma)} = \frac{R_\Sigma!}{Q_\Sigma!} (-x)^{Q_\Sigma - R_\Sigma} L_{R_\Sigma}^{(Q_\Sigma - R_\Sigma)}$
(i.ii)	$(S_1 - P_1 \geq 0) \wedge (P_\Sigma - S_\Sigma < 0)$ $\wedge (Q_1 - R_1 \geq 0) \wedge (R_\Sigma - Q_\Sigma \geq 0)$	$L_{S_\Sigma}^{(P_\Sigma - S_\Sigma)} = \frac{P_\Sigma!}{S_\Sigma!} (-x)^{S_\Sigma - P_\Sigma} L_{P_\Sigma}^{(S_\Sigma - P_\Sigma)}$
(i.iii)	$(S_1 - P_1 \geq 0) \wedge (P_\Sigma - S_\Sigma < 0)$ $\wedge (Q_1 - R_1 < 0)$	$L_{S_\Sigma}^{(P_\Sigma - S_\Sigma)} = \frac{P_\Sigma!}{S_\Sigma!} (-x)^{S_\Sigma - P_\Sigma} L_{P_\Sigma}^{(S_\Sigma - P_\Sigma)}$, $L_{R_1}^{(Q_1 - R_1)}(x) = \frac{Q_1!}{R_1!} (-x)^{R_1 - Q_1} L_{Q_1}^{(R_1 - Q_1)}$
(i.iv)	$(S_1 - P_1 \geq 0) \wedge (P_\Sigma - S_\Sigma \geq 0)$ $\wedge (Q_1 - R_1 \geq 0) \wedge (R_\Sigma - Q_\Sigma < 0)$	$L_{Q_\Sigma}^{(R_\Sigma - Q_\Sigma)} = \frac{R_\Sigma!}{Q_\Sigma!} (-x)^{Q_\Sigma - R_\Sigma} L_{R_\Sigma}^{(Q_\Sigma - R_\Sigma)}$
(i.v)	$(S_1 - P_1 \geq 0) \wedge (P_\Sigma - S_\Sigma \geq 0)$ $\wedge (Q_1 - R_1 \geq 0) \wedge (R_\Sigma - Q_\Sigma \geq 0)$	no substitution necessary
(i.vi)	$(S_1 - P_1 \geq 0) \wedge (P_\Sigma - S_\Sigma \geq 0)$ $\wedge (Q_1 - R_1 < 0)$	$L_{R_1}^{(Q_1 - R_1)} = \frac{Q_1!}{R_1!} (-x)^{R_1 - Q_1} L_{Q_1}^{(R_1 - Q_1)}$
(i.vii)	$(S_1 - P_1 < 0) \wedge (Q_1 - R_1 \geq 0)$ $\wedge (R_\Sigma - Q_\Sigma < 0)$	$L_{P_1}^{(S_1 - P_1)} = \frac{S_1!}{P_1!} (-x)^{P_1 - S_1} L_{S_1}^{(P_1 - S_1)}$, $L_{Q_\Sigma}^{(R_\Sigma - Q_\Sigma)} = \frac{R_\Sigma!}{Q_\Sigma!} (-x)^{Q_\Sigma - R_\Sigma} L_{R_\Sigma}^{(Q_\Sigma - R_\Sigma)}$
(i.viii)	$(S_1 - P_1 < 0) \wedge (Q_1 - R_1 \geq 0)$ $\wedge (R_\Sigma - Q_\Sigma \geq 0)$	$L_{P_1}^{(S_1 - P_1)} = \frac{S_1!}{P_1!} (-x)^{P_1 - S_1} L_{S_1}^{(P_1 - S_1)}$
(i.ix)	$(S_1 - P_1 < 0) \wedge (Q_1 - R_1 < 0)$	$L_{P_1}^{(S_1 - P_1)} = \frac{S_1!}{P_1!} (-x)^{P_1 - S_1} L_{S_1}^{(P_1 - S_1)}$, $L_{R_1}^{(Q_1 - R_1)} = \frac{Q_1!}{R_1!} (-x)^{R_1 - Q_1} L_{Q_1}^{(R_1 - Q_1)}$

In order to compute the integrals in Eq. (2.A.8), we will have to make a small detour into the properties of hypergeometric functions. We define the Pochhammer symbol (also known as the rising factorial) $(\lambda)_n = \Gamma(\lambda + n)/\Gamma(\lambda)$ and the generalized hypergeometric series [189]

$${}_pF_q \left[\begin{matrix} a_1, \dots, a_p; \\ b_1, \dots, b_q; \end{matrix} z \right] = \sum_{j=0}^{\infty} \frac{(a_1)_j \cdots (a_p)_j}{(b_1)_j \cdots (b_q)_j} \frac{z^j}{j!}. \quad (2.A.9)$$

Two properties of the generalized hypergeometric series are noteworthy: First, as soon as at least one of the numerator parameters a_k is a non-positive integer, the series terminates and becomes a finite polynomial in z . Second, if one of the denominator parameters b_l is non-positive, there will appear a zero in the denominator due to the properties of the Pochhammer symbol and Eq. (2.A.9) is no longer a well defined expression. As we will see, the requirement $b_l > 0$ is the reason our final expression for $h_{\mathbf{PQRS}}$ will be case-sensitive to the values of the quantum numbers (it is the reason we have to consider all the various parameter regimes in Table 2.A.1). There are also higher-order hypergeometric functions which possess more than one variable, such as the first Lauricella function [190, 191]

$$F_A^{(r)} \left[\begin{matrix} a, b_1, \dots, b_r; \\ c_1, \dots, c_r; \end{matrix} z_1, \dots, z_r \right] = \sum_{k_1, \dots, k_r=0}^{\infty} \frac{(a)_{k_1+\dots+k_r} (b_1)_{k_1} \cdots (b_r)_{k_r}}{(c_1)_{k_1} \cdots (c_r)_{k_r}} \frac{z_1^{k_1}}{k_1!} \cdots \frac{z_r^{k_r}}{k_r!}, \quad (2.A.10)$$

with the constraint $(|z_1| + \dots + |z_r|) < 1$, which is only of significance for the convergence of non-terminating hypergeometric series. As we shall see, all hypergeometric sums we encounter terminate, meaning that we do not need to worry about any convergence issues.

We simplify the following expressions by omitting the arguments of the Laguerre polynomials, writing $L_n^{(\alpha)}(x) = L_n^{(\alpha)}$, where $x = q^2/2$ is a substitution used for q in Eq. (2.A.8). This allows us to simplify Eq. (2.A.8), after substitution of the integration variable, to

$$h_{\mathbf{PQRS}} = \frac{e^2}{\epsilon} \sqrt{\frac{P_1! Q_{\Sigma}! S_{\Sigma}! R_1!}{2P_{\Sigma}! Q_1! S_1! R_{\Sigma}!}} \int_0^{\infty} dx x^{(P_2-S_2+1/2)-1} e^{-2x} L_{[P_1, S_{\Sigma}, R_1, Q_{\Sigma}]} \delta_{P_2-S_2, R_2-Q_2}. \quad (2.A.11)$$

The integral in Eq. (2.A.11) is a Laplace transform of a product of Laguerre polynomials which has been thoroughly studied, e.g. in Refs. [191–194]. For $\text{Re}\{p\} > 0$, $\text{Re}\{s\} > 0$, and $n_j \in \mathbb{N}_0$ for $j = 1, \dots, r$, the following integral identity holds [191],

$$\begin{aligned} & \int_0^{\infty} dx x^{p-1} e^{-sx} L_{n_1}^{(\alpha_1)}(\lambda_1 x) \cdots L_{n_r}^{(\alpha_r)}(\lambda_r x) \\ &= \frac{\Gamma(p)}{s^p} \left(\prod_{k=1}^r \binom{n_k + \alpha_k}{n_k} \right) F_A^{(r)} \left[\begin{matrix} p, -n_1, \dots, -n_r; \\ \alpha_1 + 1, \dots, \alpha_r + 1; \end{matrix} \frac{\lambda_1}{s}, \dots, \frac{\lambda_r}{s} \right]. \end{aligned} \quad (2.A.12)$$

By setting $r = 4$, $s = 2$ and $\lambda_j = 1$ for $j = 1, \dots, 4$, we can use Eq. (2.A.12) to solve the integral in Eq. (2.A.11). One has to exert caution, since just as hypergeometric series must not have negative integers in its lower set of parameters, the same holds for Lauricella functions—they are a generalization of the former. A further prerequisite for applying the integral formula is that the real part of p must be larger than zero and $n_j \in \mathbb{N}_0$. In addition, the Lauricella function is well defined only for $\alpha_j \geq 0$. Using again the transformation between Laguerre polynomials given in Eq. (2.A.7), we can bring the integral (2.A.11) to a form which meets all prerequisites for using the integral formula.

In Table 2.A.1 we consider all possible parameter regimes for $\mathbf{P}, \mathbf{Q}, \mathbf{R}, \mathbf{S}$ that would allow the order of any Laguerre polynomial appearing in Eq. (2.A.11) to become negative. Then, one can flip the sign of the negative exponent via Eq. (2.A.7) and the additional polynomial in x ensures that the power p of the final polynomial is positive. The results for all possible parameter regime choices are summarized in Tables 2.3 and 2.A.1. Eq. (2.A.11) can thus be recast into the following form,

$$h_{\mathbf{PQRS}}^{(i)} = \frac{e^{2\mathcal{C}^{(i)}}}{\epsilon} \int_0^\infty dx x^{p-1} e^{-2x} L_{[n_1, n_2, n_3, n_4]}(x) \delta_{P_2 - S_2, R_2 - Q_2}, \quad (2.A.13)$$

where $\mathcal{C}^{(i)}$, p and $[n_1, n_2, n_3, n_4]$ are given in Table (2.3) and the solution of the integral is given by Eq. (2.A.12).

The numerical challenge is thus to either find a fast and reliable implementation of the Lauricella function, or to break the Lauricella function down into lower-order hypergeometric functions. We will give an explicit solution which computes a Coulomb matrix element as a simple scalar product between two vectors. One of these two vectors contains entries which are the results of sums of fractions of rising factorials. There are numerous ways to break down the Lauricella function into lower-order hypergeometric expressions, and which one to pick should depend on which expressions one can compute fast and reliable. In the last part of this section, we give an example on how this can be achieved. We consider the integral representation of the Lauricella function $F_A^{(r)}$ [148]

$$F_A^{(r)} \left[\begin{matrix} a, b_1, \dots, b_r; \\ c_1, \dots, c_r; \end{matrix} x_1, \dots, x_r \right] = \int_0^\infty dt e^{-t} \frac{t^{a-1}}{\Gamma(a)} \prod_{j=1}^r {}_1F_1 \left[\begin{matrix} b_j; \\ c_j; \end{matrix} x_j t \right], \quad (2.A.14)$$

where $\text{Re}\{a\} > 0$ and $\text{Re}\{x_1 + \dots + x_r\} < 1$ and we are only interested in the case where $r = 4$. Clearly, in order to avoid zeros in the denominator, we require all c_j to be positive integers. Using the definition of the hypergeometric function and the Cauchy-product formula, the product of two such hypergeometric functions results in

$${}_1F_1 \left[\begin{matrix} b_1; \\ c_1; \end{matrix} x_1 t \right] {}_1F_1 \left[\begin{matrix} b_2; \\ c_2; \end{matrix} x_2 t \right] = \sum_{k=0}^\infty \left(\sum_{l=0}^k \frac{(b_1)_{k-l} (b_2)_l}{(c_1)_{k-l} (c_2)_l} \frac{x_1^{k-l} x_2^l}{(k-l)! l!} \right) t^k. \quad (2.A.15)$$

We perform the product of all four hypergeometric functions of Eq. (2.A.14) in the above manner and the resulting coefficient of the resulting polynomial is given by

$$\xi_k = \sum_{p=0}^k \sum_{q=0}^p \sum_{r=0}^q \frac{(-n_1)_r (-n_2)_{q-r} (-n_3)_{p-q} (-n_4)_{k-p}}{(\alpha_1 + 1)_r (\alpha_2 + 1)_{q-r} (\alpha_3 + 1)_{p-q} (\alpha_4 + 1)_{k-p} r! (q-r)! (p-q)! (k-p)! 2^k}, \quad (2.A.16)$$

which we call the convolution coefficient. This coefficient allows us to compute the expression in Eq. (2.A.14),

$$\begin{aligned} \int_0^\infty dt e^{-t} \frac{t^{p-1}}{\Gamma(p)} \prod_{i=1}^4 {}_1F_1 \left[\begin{matrix} -n_i; \\ \alpha_i + 1; \end{matrix} \frac{t}{2} \right] &= \sum_{k=0}^{n_1+n_2+n_3+n_4} \frac{\xi_k}{\Gamma(p)} \int_0^\infty dt e^{-t} t^{p+k-1} \\ &= \sum_{k=0}^{n_1+n_2+n_3+n_4} \xi_k (p)_k. \end{aligned} \quad (2.A.17)$$

We recognize the definition of the Gamma function in the last integral. By using the definition of the Pochhammer symbol and defining the two column vectors

$$\boldsymbol{\xi} = (\xi_0, \xi_1, \dots, \xi_{n_1+n_2+n_3+n_4})^T \quad (2.A.18)$$

$$\mathbf{(p)} = ((p)_0, (p)_1, \dots, (p)_{n_1+n_2+n_3+n_4})^T, \quad (2.A.19)$$

we have

$$F_A^{(r)} \left[\begin{array}{l} a, b_1, \dots, b_r; \\ c_1, \dots, c_r; \end{array} x_1, \dots, x_r \right] = \boldsymbol{\xi} \cdot (\boldsymbol{p}). \quad (2.A.20)$$

As a sanity check, we compared the values of Eq. (2.A.20) combined with the additional prefactors appearing in Eqs. (2.A.12) and (2.A.13), with M_{mn}^l in Eq. (2.2.31) for the LLL for various system sizes and they are—up to numerical precision error—in exact agreement with one another.

Section 2.B

HAMILTONIAN SIMULATION THROUGH LINEAR COMBINATION OF UNITARIES USING THE SELF-INVERSE MATRIX DECOMPOSITION STRATEGY

In this section, we use an alternative algorithm to the low rank factorization algorithm of [143] used in Section 2.3 to sample the eigenspectrum of the Hamiltonian H . This algorithm is based on the self-inverse matrix decomposition strategy first described in Ref. [144] and makes use of the fact that we have an analytical form for the matrix elements of the Hamiltonian as derived in Section 2.2 and that within the LLL, the largest matrix element $\max_{\ell}(\omega_{\ell})$ of the Hamiltonian elements is constant $O(1)$ ⁷. As we will see, this approach scales considerably worse than the low rank factorization method of Ref. [143] presented in Section 2.3, unless one is able to considerably lower the computational cost of evaluating sums of products of factorials as given by Eq. (2.2.19) and Eqs. (2.2.31)-(2.2.34). In the approach used in this section one will dramatically increase the number of terms in the Hamiltonian but with the advantage that the coefficients of the state $\text{PREPARE } |0\rangle^{\otimes \log(L)}$ will only be 1 or i , which makes the state much simpler to prepare.

As described in Section 2.2, for the Hamiltonians of interest in this paper we are able to compute the ω_{ℓ} efficiently from ℓ (which is essentially equivalent to computing the $h_{\mathbf{PQRS}}$ from the indices $\mathbf{P}, \mathbf{Q}, \mathbf{R}$ and \mathbf{S}). Several steps are required in order to go from computing these coefficients to implementing the PREPARE operator. The ability to compute the coefficients essentially allows us to prepare the state

$$\sqrt{\frac{1}{L}} \sum_{\ell=1}^L |\ell\rangle |\omega_{\ell}\rangle. \tag{2.B.1}$$

But to translate this into the desired state,

$$\sqrt{\frac{1}{\lambda}} \sum_{\ell=1}^L \sqrt{\omega_{\ell}} |\ell\rangle, \tag{2.B.2}$$

we will use the self-inverse matrix decomposition strategy first described in Ref. [144]. This is described in the context of simulating electronic structure in Section 4.4 of Ref. [50]. The result is that C_P (again, the cost to implement PREPARE) ends up scaling like the cost to compute the coefficient mentioned above, but the λ value is increased to scaling like $O(L \max_{\ell}(\omega_{\ell}))$.

Essentially, the strategy which leads to this scaling is as follows. First, one re-imagines the Hamiltonian as being a sum of a very large number of terms where each term has a coefficient that is the same magnitude; specifically, the coefficient of each term is either $+\zeta$ or $-\zeta$ where ζ is thus chosen to limit the precision of the Hamiltonian representation,

⁷The largest matrix element at fixed filling factor $\nu = 1/3$ turns out to be identical in the LLL for all studied angular momenta cutoffs.

i.e., $\zeta = O(\epsilon)$. The largest term in the original Hamiltonian will have the property that each of the subterms into which it is decomposed in the new Hamiltonian has the same coefficient; thus, each term consists of $O(\max_\ell(\omega_\ell)/\epsilon)$ subterms of magnitude ζ . The advantage of this is that the coefficients of the state that we must realize with PREPARE are now all either 1 or i . For details of how this is realized, see Ref. [144]. In order to determine whether we should phase by 1 or i we need to compute the coefficient and compare it to a coin register. Essentially, we are dramatically increasing the number of terms in the Hamiltonian by decomposing each term into a sum of small terms that all have the same coefficient up to a sign. The difficult part of realizing prepare is thus simply to decide which sign is associated with each computational basis.

In the following, we will give an estimate of the time complexity for computing the two-body Hamiltonian matrix elements $h_{\mathbf{PQRS}}$ for the LLL approximation, where $h_{\mathbf{PQRS}} = M_{mn}^l$ is given by Eq. (2.2.31). These coefficients correspond to the ω_ℓ mentioned before. We ignore the complexity of the one-body matrix elements of Eq. (2.2.12), as they are trivial to compute and there are many fewer of them so they are easier to simulate as well. We expect a slightly worse, but similar complexity scaling when computing $h_{\mathbf{PQRS}}$ via Eq. (2.2.19), which takes into account LL mixing. We consider Eq. (2.2.31), and observe that it is sufficient to only consider the complexity of computing C_{lmn} , A_{mn}^l and B_{nm}^l as given by Eqs. (2.2.32)-(2.2.34), respectively and then singling out the term possessing the largest complexity. While C_{lmn} is dominated by the cost of computing the Gamma function, A_{mn}^l and B_{nm}^l require the evaluation of a finite sum of division and products of Gamma functions, which will thus have a larger complexity than C_{lmn} .

The problem is that to compute the value of A_{mn}^l and B_{nm}^l to within precision ϵ it is required to use a number of bits that scales as $O(M \log M)$ where M is the cutoff in angular momentum). This is because A_{mn}^l and B_{nm}^l involve computing factorials of M and we know from Stirling's approximation that $\log(M!) = O(M \log M)$. We then need to multiply these numbers together, which gives us complexity $O(M^2 \text{polylog}(M)) = \tilde{O}(M^2)$. If one then explicitly evaluates the sum the complexity becomes $\tilde{O}(M^3)$, which is very bad. One could choose to expand the sum using LCU methods (by which we mean, one can consider each term in the sum as a distinct term in the Hamiltonian) but this will dramatically increase λ . We note that the complexity bound on computing the coefficients ω_ℓ can in principle be further reduced using algorithms designed for computing linearly convergent series as in Refs. [195, 196].

Unfortunately, the number of times we must repeat this primitive is $\lambda = O(L \max_\ell(\omega_\ell))$. In our context, when restricting ourselves to the LLL, $\omega_\ell = O(1)$ for the two-body operator and we have $L = O(M^3)$ for that operator⁸. For the one-body operator $\max_\ell \omega_\ell = O(N)$, where N is the number of LLs, but there are only $L = O(N_{\text{so}})$ terms due to the delta function in Eq. (2.2.12). Furthermore, when considering only the LLL, we have $\max_\ell \omega_\ell = O(1)$ and $L = O(M)$ for the two-body operators. Thus, overall we have that $\lambda = O(N_{\text{so}}N + M^3) = O(M^3)$ when restricting ourselves to the LLL.

Putting this all together then, we see that using Eq. (2.3.5), the total complexity of deploying phase estimation to estimate the ground state energy within precision ΔE is $\tilde{O}(M^6/\Delta E)$ up to log factor, which is considerably worse than the $O(N^{4.35}/\Delta E)$ T gate complexity obtained when using the low rank factorization of Ref. [143] presented in Section 2.3. Overall, it is surprising that we cannot exceed this more generic strategy

⁸The scaling in terms of N (which we neglect in that section) and M is however not $O(N^3M^3)$, as one might think, but $O(N^4M^3)$, since the conservation of angular momentum only reduces the M scaling by one order.

despite having a closed form for the coefficients. The reason is ultimately because of the extremely high precision required to compute the Gamma functions in the coefficients.

Section 2.C

DERIVATIONS FOR THE EQUATIONS OF MOTION OF THE CM

For FGS (as defined in Eq. (2.4.2)) with fixed particle number, Wick's theorem gives

$$\langle c_p^\dagger c_i^\dagger c_q c_j \rangle = -\Gamma_{qp}\Gamma_{ji} + \Gamma_{jp}\Gamma_{qi} \quad (2.C.1)$$

$$\langle c_p^\dagger c_q^\dagger c_i^\dagger c_r c_s c_j \rangle = \Gamma_{jp}\Gamma_{sq}\Gamma_{ri} - \Gamma_{jp}\Gamma_{rq}\Gamma_{si} + \Gamma_{sp}\Gamma_{rq}\Gamma_{ji} - \Gamma_{sp}\Gamma_{jq}\Gamma_{ri} + \Gamma_{rp}\Gamma_{jq}\Gamma_{si} - \Gamma_{rp}\Gamma_{sq}\Gamma_{ji}, \quad (2.C.2)$$

since the pairing terms $\langle c_i c_j \rangle$ and $\langle c_i^\dagger c_j^\dagger \rangle$ vanish [170]. We compute the quadratic contribution to the imaginary-time evolution of Eq. (2.4.8) using Eqs. (2.C.1) and (2.C.2),

$$\begin{aligned} \text{tr}\{\{T, c_i^\dagger c_j\}\rho\} &= \sum_{p,q} f_{pq} \langle c_p^\dagger c_q c_i^\dagger c_j + c_i^\dagger c_j c_p^\dagger c_q \rangle \\ &= -2 \sum_{p,q} f_{pq} \langle c_p^\dagger c_i^\dagger c_q c_j \rangle + \sum_p f_{pi} \langle c_p^\dagger c_j \rangle + \sum_q f_{jq} \langle c_i^\dagger c_q \rangle \\ &= 2 \sum_{p,q} f_{pq} (\Gamma_{qp}\Gamma_{ji} - \Gamma_{jp}\Gamma_{qi}) + \sum_p f_{pi} \Gamma_{jp} + \sum_q f_{jq} \Gamma_{qi} \\ &= 2\text{tr}[f\Gamma]\Gamma_{ji} - 2[\Gamma f\Gamma]_{ji} + [\{\Gamma, f\}]_{ji}. \end{aligned} \quad (2.C.3)$$

The contributions from the quartic interaction term are given by

$$\begin{aligned} \text{tr}\{\{V, c_i^\dagger c_j\}\rho\} &= \sum_{p,q,r,s} h_{pqrs} \langle c_p^\dagger c_q^\dagger c_i^\dagger c_r c_s c_j \rangle - \frac{1}{2} \sum_{p,q,s} h_{pqis} \langle c_p^\dagger c_q^\dagger c_s c_j \rangle + \frac{1}{2} \sum_{p,q,r} h_{pqri} \langle c_p^\dagger c_q^\dagger c_r c_j \rangle \\ &\quad - \frac{1}{2} \sum_{p,r,s} h_{pjrs} \langle c_i^\dagger c_p^\dagger c_r c_s \rangle + \frac{1}{2} \sum_{q,r,s} h_{jqr s} \langle c_i^\dagger c_q^\dagger c_r c_s \rangle. \end{aligned} \quad (2.C.4)$$

The first term on the right-hand side of Eq. (2.C.4) gives

$$\begin{aligned} \sum_{p,q,r,s} h_{pqrs} \langle c_p^\dagger c_q^\dagger c_i^\dagger c_r c_s c_j \rangle &= \sum_{p,q,r,s} h_{pqrs} (\Gamma_{jp}\Gamma_{sq}\Gamma_{ri} - \Gamma_{jp}\Gamma_{rq}\Gamma_{si} + \Gamma_{sp}\Gamma_{rq}\Gamma_{ji} - \Gamma_{sp}\Gamma_{jq}\Gamma_{ri} \\ &\quad + \Gamma_{rp}\Gamma_{jq}\Gamma_{si} - \Gamma_{rp}\Gamma_{sq}\Gamma_{ji}) \\ &= -4[\Gamma \text{tr}_{1,4}[h\Gamma]\Gamma]_{ji} + 2\text{tr}[\text{tr}_{1,4}[h\Gamma]\Gamma]\Gamma_{ji}, \end{aligned} \quad (2.C.5)$$

the remaining terms contribute

$$\begin{aligned} 2[\{\Gamma, \text{tr}_{1,4}[h\Gamma]\}]_{ji} &= -\frac{1}{2} \sum_{p,q,s} h_{pqis} \langle c_p^\dagger c_q^\dagger c_s c_j \rangle + \frac{1}{2} \sum_{p,q,r} h_{pqri} \langle c_p^\dagger c_q^\dagger c_r c_j \rangle - \frac{1}{2} \sum_{p,r,s} h_{pjrs} \langle c_i^\dagger c_p^\dagger c_r c_s \rangle \\ &\quad + \frac{1}{2} \sum_{q,r,s} h_{jqr s} \langle c_i^\dagger c_q^\dagger c_r c_s \rangle. \end{aligned} \quad (2.C.6)$$

The second term in Eq. (2.4.8) simplifies to

$$2\Gamma_{ji}\text{tr}[H\rho(t)] = 2\text{tr}[f\Gamma]\Gamma_{ji} + 2\text{tr}[\text{tr}_{1,4}[h\Gamma]\Gamma]\Gamma_{ji}. \quad (2.C.7)$$

Defining the mean field term

$$h_m(\Gamma) = f + 2\text{tr}_{1,4}[h\Gamma], \quad (2.C.8)$$

the imaginary-time evolution of the CM is given by

$$d_\tau \Gamma_{ji} = -[\{\Gamma, h_m(\Gamma)\}]_{ji} + 2[\Gamma h_m(\Gamma)\Gamma]_{ji}. \quad (2.C.9)$$

Section 2.D

ROTATING THE SYSTEM HAMILTONIAN INTO THE EIGENBASIS OF THE CM

For the ASCI algorithm described in Section 2.4.2, we rotate the Hamiltonian of Eq. (2.2.9) into the eigenbasis of the CM Γ at the end of the imaginary time evolution. Let O be the matrix which diagonalizes the CM as in Eq. (2.4.14) and corresponding orbital rotations as given in Eq. (2.4.16). The system Hamiltonian in the mean-field eigenbasis is then given by

$$H = \sum_{i,j} \tilde{f}_{ij} \tilde{c}_i^\dagger \tilde{c}_j + \frac{1}{2} \sum_{i,j,k,l} \tilde{h}_{ijkl} \tilde{c}_i^\dagger \tilde{c}_j^\dagger \tilde{c}_k \tilde{c}_l \quad (2.D.1)$$

where

$$\tilde{f}_{ij} = \sum_{p,q} O_{pi} f_{pq} O_{qj} \quad (2.D.2)$$

$$\tilde{h}_{ijkl} = \sum_{p,q,r,s} h_{pqrs} O_{pi} O_{qj} O_{rk} O_{sl}. \quad (2.D.3)$$

Note that under the orthogonal transformations, the integrals are still anti-symmetric w.r.t. index permutation (however, the conservation of angular momentum is now no longer visible in the indices).

Section 2.E

IMAGINARY TIME EVOLUTION - FORMAL INTEGRATION

We aim to solve the differential equation

$$d_\tau \Gamma = -h_m(\Gamma) - \Gamma h_m(\Gamma) \Gamma.$$

Following Ref. [63], we formally integrate the equation of motion, which result in

$$\Gamma(\tau) = O(\tau) \Gamma(0) O(\tau)^T, \quad (2.E.1)$$

where $O(\tau)$ is an orthogonal matrix (if the transformation were not orthogonal, it could take us out of the family of FGS, where Wick's theorem no longer applies) given by

$$O(\tau) = \mathcal{T} \exp \left(\int_0^\tau d\tau' A(\Gamma(\tau')) \right), \quad (2.E.2)$$

with \mathcal{T} denoting the time-ordering operator. For a small time step $\Delta\tau$, we can expand $O(\tau)$ using the orthogonality property $A(\Gamma(\tau))^T = -A(\Gamma(\tau))$ to get in first order $\Delta\tau$

$$\begin{aligned} \Gamma(\tau + \Delta\tau) &= \Gamma(\tau) - \Gamma(\tau) A(\Gamma(\tau)) \Delta\tau \\ &\quad + A(\Gamma(\tau)) \Gamma(\tau) \Delta\tau + O(\Delta\tau^2). \end{aligned} \quad (2.E.3)$$

This leads to

$$\frac{\Gamma(\tau + \Delta\tau) - \Gamma(\tau)}{\Delta\tau} = [A(\Gamma(\tau)), \Gamma(\tau)] \quad (2.E.4)$$

which in the limit of small $\Delta\tau$ should be equal to the right-hand side of Eq.(2.4.8). Together with the fact that $\Gamma^2 = -\mathbb{1}_{2N_f}$, this allows us to find an explicit expression for $A(\Gamma(\tau))$, namely

$$A(\Gamma(\tau)) = \frac{1}{2} [h_m(\Gamma(\tau)), \Gamma(\tau)], \quad (2.E.5)$$

since

$$\begin{aligned} [A, \Gamma] &= \frac{1}{2} (-h_m - \Gamma h_m \Gamma - \Gamma h_m \Gamma - h_m) \\ &= -h_m - \Gamma h_m \Gamma. \end{aligned} \quad (2.E.6)$$

Thus, for small time steps $\Delta\tau$, we can compute the CM by an orthogonal transformation of the prior CM via

$$\begin{aligned} \Gamma(\tau + \Delta\tau) &\approx \exp(A(\Gamma(\tau)) \Delta\tau) \Gamma(\tau) \\ &\quad \times \exp(-A(\Gamma(\tau)) \Delta\tau). \end{aligned} \quad (2.E.7)$$

While in the Dirac representation of fermionic creation and annihilation operators the particle number is conserved, small numerical fluctuations will lower the number of particles when working in a Majorana representation, where only parity is a conserved quantity. One would have to introduce a chemical potential in order to enforce particle number conservation during the iterative process in the latter case. As shown in Fig. 2.E.1, particle number is conserved when solving the equations of motions in the fermionic basis following the imaginary time evolution as defined in Eq. (2.E.7).

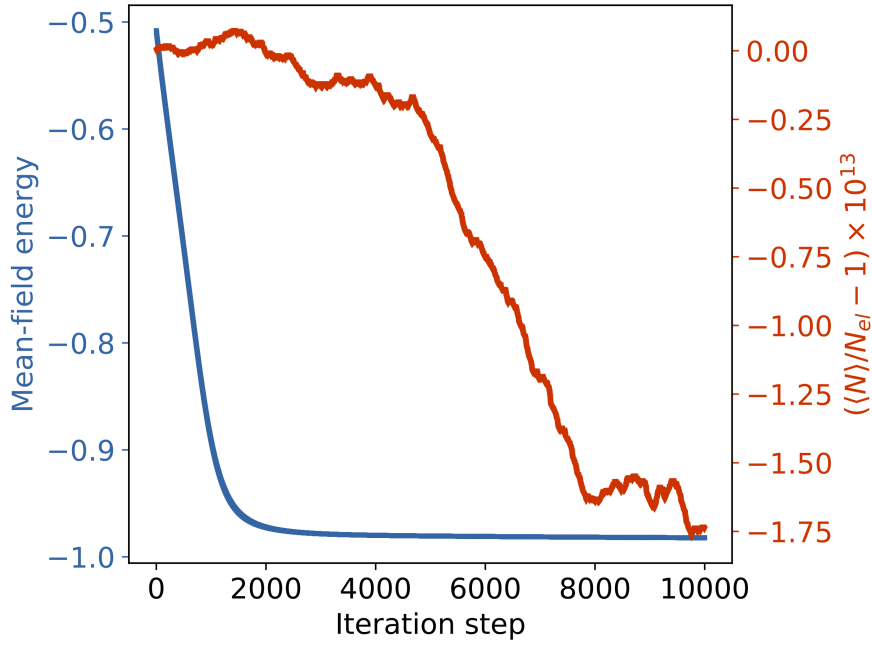


Figure 2.E.1: Scatter plot of the monotonic decrease of the mean-field energy (blue, left y -axis) as defined in Eq. (2.4.12) at each iteration step for the imaginary time evolution of the CM for a system of $N_{el} = 4$ electrons distributed among $N_{so} = 12$ spin-orbitals at a chemical potential $\mu = 1$ and step size $\Delta\tau = 0.01$. We also calculated the deviation of the number of particles present at each iteration step by plotting $(\langle N \rangle / N_{el} - 1) \times 10^{13}$ (red line, right y -axis), where $\langle N \rangle = \text{tr}(\Gamma)$ is the expectation value of the particle number operator. Note that the particle number changes from $N_{el} = 4$ only in the 13. decimal place.

Section 2.F

COMPUTING CORRELATION FUNCTIONS FOR SINGLE-REFERENCE AND MULTI-REFERENCE STATES

If the initial state is a single Slater determinant $|\Psi_{\text{GS}}\rangle$, we have

$$\begin{aligned} G_1(\mathbf{r}, \mathbf{r}') &= \sum_{p,q} \eta_p^*(\mathbf{r}) \eta_q(\mathbf{r}') \langle \Psi_{\text{GS}} | c_p^\dagger c_q | \Psi_{\text{GS}} \rangle \\ &= \boldsymbol{\eta}(\mathbf{r}')^T \Gamma \boldsymbol{\eta}^*(\mathbf{r}) \end{aligned} \quad (2.F.1)$$

where $\boldsymbol{\eta}(\mathbf{r}) = (\eta_0(\mathbf{r}), \eta_1(\mathbf{r}), \dots, \eta_M(\mathbf{r}))^T$ is a vector of the basis functions chosen and M is the angular momentum cutoff. The mean-field density correlations are given by

$$G_2(\mathbf{r}, \mathbf{r}') = \boldsymbol{\eta}(\mathbf{r})^T \Gamma \boldsymbol{\eta}^*(\mathbf{r}) \boldsymbol{\eta}(\mathbf{r}')^T \Gamma \boldsymbol{\eta}^*(\mathbf{r}') - \boldsymbol{\eta}(\mathbf{r}')^T \Gamma \boldsymbol{\eta}^*(\mathbf{r}) \boldsymbol{\eta}(\mathbf{r})^T \Gamma \boldsymbol{\eta}^*(\mathbf{r}'). \quad (2.F.2)$$

Both correlation functions in Eqs. (2.F.1) and (2.F.2) can be computed efficiently. One can compute the one-particle reduced density matrix and the pair correlation function also for the multi-reference state of Eq. (2.4.17) through the CM using Wick's theorem.

In Fig. 2.F.1, we show the pair correlation function for the FGS solution obtained from imaginary time evolution as introduced in Section 2.4.1 for $N_{\text{so}} = 138$ spin-orbitals at filling $\nu = 1/3$ in the LLL, with an apparent crystal-like structure emerging as expected for mean-field solutions (see e.g. chapter 4 in Ref. [197]).

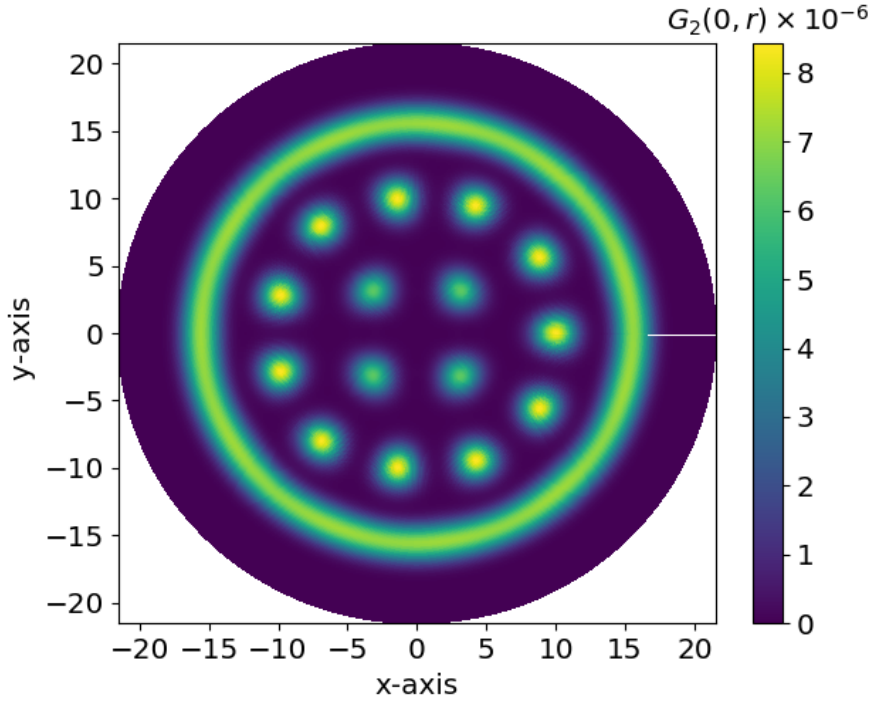


Figure 2.F.1: Plot of the pair correlation function $G_2(\mathbf{0}, \mathbf{r})$, with $N_{\text{so}} = 138$ at filling $\nu = 1/3$ ($N_{\text{el}} = 46$), as defined in Eq. (2.6.4) and expressed in terms of the CM as outlined in Section 2.F for a single-reference state $|\Psi_{\text{GS}}\rangle$ obtained from imaginary time evolution via the formal integration method of Section 2.E, with 10^4 steps and step size $\Delta t = 0.1$. The first particle was located at the origin. A crystal structure emerges in the bulk, while the disk boundary is centered at a circle of radius $R_d = \sqrt{2(M+1)} \approx 16.6$, where the magnetic length is set to unity.

Section 2.G

 COMPUTING THE OVERLAPS

The following subsections detail some of the computational steps that were used for implementing the ASCI algorithm.

2.G.1 Computing the diagonal terms

One of the computationally more costly steps in the ASCI algorithm is getting the diagonal Hamiltonian term

$$H_{ii} = \langle A_i | H | A_i \rangle, \quad (2.G.1)$$

where $|A_i\rangle$ is a determinant from the set of determinants $H|\{C\}\rangle$ (which are all unique determinants that have non-zero coefficients when acting with H on the core determinants $|\{C\}\rangle = \{|C_1\rangle, \dots, |C_{\text{cdets}}\rangle\}$). By inserting the Hamiltonian of Eq. (2.D.1), we get

$$H_{ii} = \sum_{p,q} \tilde{f}_{pq} \langle A_i | \tilde{c}_p^\dagger \tilde{c}_q | A_i \rangle + \frac{1}{2} \sum_{p,q,r,s} \tilde{h}_{pqrs} \langle A_i | \tilde{c}_p^\dagger \tilde{c}_q^\dagger \tilde{c}_r \tilde{c}_s | A_i \rangle. \quad (2.G.2)$$

Let us first look at the second term in Eq. (2.G.2). The only non-vanishing terms are given by either (i) $p = r$ and $q = s$, or (ii) $p = s$ and $q = r$. Furthermore, both p and q must be occupied in $|A_i\rangle$, a condition we will denote as $\{p, q\} \in A_{i,\text{occ}}$. Since $p \neq q$ (a fermionic mode can only contain zero or one particle) and due to the anti-symmetry in \tilde{h}_{pqrs} , we can unify cases (i) and (ii), which results in a factor of two,

$$\begin{aligned} \sum_{p,q,r,s} \frac{\tilde{h}_{pqrs}}{2} \langle A_i | \tilde{c}_p^\dagger \tilde{c}_q^\dagger \tilde{c}_r \tilde{c}_s | A_i \rangle &= \sum_{p,q} \tilde{h}_{ppqq} \langle A_i | \tilde{c}_p^\dagger \tilde{c}_q^\dagger \tilde{c}_p \tilde{c}_q | A_i \rangle \\ &= -2 \sum_{\substack{\{p,q\} \in A_{i,\text{occ}} \\ p < q}} \tilde{h}_{ppqq}, \end{aligned} \quad (2.G.3)$$

where the minus sign is due to the Jordan-Wigner transformation (which we will explain a couple of lines below in more detail). Similarly, we get for the first term in Eq.(2.G.2)

$$\sum_{p,q} \tilde{f}_{pq} \langle A_i | \tilde{c}_p^\dagger \tilde{c}_q | A_i \rangle = \sum_{p \in A_{i,\text{occ}}} \tilde{f}_{pp}, \quad (2.G.4)$$

thus resulting in a simple formula for the diagonal terms

$$H_{ii} = \sum_{p \in A_{i,\text{occ}}} \tilde{f}_{pp} - 2 \sum_{\substack{\{p,q\} \in A_{i,\text{occ}} \\ p < q}} \tilde{h}_{ppqq}, \quad (2.G.5)$$

where the dependence on index i results in the task of finding all occupied orbitals $A_{i,\text{occ}}$ in the determinant $|A_i\rangle$, which is a simple problem.

2.G.2 Computing the off-diagonal terms

We consider the action of the Hamiltonian operator on a core determinant

$$H |C_j\rangle = \left(\sum_{p,q} \tilde{f}_{pq} \tilde{c}_p^\dagger \tilde{c}_q + \frac{1}{2} \sum_{p,q,r,s} \tilde{h}_{pqrs} \tilde{c}_p^\dagger \tilde{c}_q^\dagger \tilde{c}_r \tilde{c}_s \right) |C_j\rangle. \quad (2.G.6)$$

We are going to treat the one- and two-body operators separately.

One-body terms

Since we have real coefficients, $\tilde{f}_{pq} = \tilde{f}_{qp}$ and thus

$$\sum_{p,q} \tilde{f}_{pq} \tilde{c}_p^\dagger \tilde{c}_q |C_j\rangle = 2 \sum_{p<q} \tilde{f}_{pq} \tilde{c}_p^\dagger \tilde{c}_q |C_j\rangle + \sum_p \tilde{f}_{pp} \tilde{c}_p^\dagger \tilde{c}_p |C_j\rangle. \quad (2.G.7)$$

In our case, the one-body terms are diagonal, therefore

$$\sum_{p,q} \tilde{f}_{pq} \tilde{c}_p^\dagger \tilde{c}_q |C_j\rangle = \sum_p \tilde{f}_{pp} \tilde{c}_p^\dagger \tilde{c}_p |C_j\rangle. \quad (2.G.8)$$

Two-body terms

Due to the real-valued, anti-symmetric nature of the two-body coefficients, we have

$$\sum_{p,q,r,s} \frac{\tilde{h}_{pqrs}}{2} \tilde{c}_p^\dagger \tilde{c}_q^\dagger \tilde{c}_r \tilde{c}_s |C_j\rangle = 2 \sum_{p<q;r<s} \tilde{h}_{pqrs} \tilde{c}_p^\dagger \tilde{c}_q^\dagger \tilde{c}_r \tilde{c}_s |C_j\rangle, \quad (2.G.9)$$

since $p \neq q$ and $r \neq s$ (otherwise, they would give zero). We can break the operator into parts of single and double excitations (we will omit the factor 2 in front of the sum for now and add it later),

$$\begin{aligned} \sum_{p<q,r<s} \tilde{h}_{pqrs} \tilde{c}_p^\dagger \tilde{c}_q^\dagger \tilde{c}_r \tilde{c}_s &= \sum_{\substack{p<q;r<s \\ p \neq \{r,s\}; q \neq \{r,s\}}} \tilde{h}_{pqrs} \tilde{c}_p^\dagger \tilde{c}_q^\dagger \tilde{c}_r \tilde{c}_s + \sum_{\substack{q<p;r<p \\ q \neq r}} \tilde{h}_{qprp} \tilde{c}_q^\dagger \tilde{c}_p^\dagger \tilde{c}_r \tilde{c}_p + \sum_{q<p<r} \tilde{h}_{qprr} \tilde{c}_q^\dagger \tilde{c}_p^\dagger \tilde{c}_p \tilde{c}_r \\ &+ \sum_{r<p<q} \tilde{h}_{pqrp} \tilde{c}_p^\dagger \tilde{c}_q^\dagger \tilde{c}_r \tilde{c}_p + \sum_{\substack{p<q;p<r \\ q \neq r}} \tilde{h}_{pqp r} \tilde{c}_p^\dagger \tilde{c}_q^\dagger \tilde{c}_p \tilde{c}_r + \sum_{p<q} \tilde{h}_{pqqq} \tilde{c}_p^\dagger \tilde{c}_q^\dagger \tilde{c}_p \tilde{c}_q. \end{aligned} \quad (2.G.10)$$

In the following, we will discuss how we can numerically get the matrix elements.

2.G.3 Identity map

Some terms in H map the input determinant $|D_i\rangle$ back onto itself. Since in Eq. (2.4.18), the sum is taken over all input core determinants that do not map onto itself via the action of H , such amplitudes will be set to zero.

2.G.4 One-body excitations

We have four terms in Eq. (2.G.10) that create single excitations. Such determinants which differ only by a single creation-annihilation pair are called single-connected. In order to know which terms to take into account when going from a determinant $|D_1\rangle$ to a single-connected determinant $|D_2\rangle$ via application of $H |D_1\rangle$, we first determine the pair

$[i, j]$, with $i < j$, that indicates the spin-orbitals i and j where the two determinants differ. We let $k_j \in \{0, 1\}$ denote the occupation of spin-orbital j and consider two distinct cases.

We denote with α_{JW} the phase factor due to the Jordan-Wigner transformation as introduced in Section 2.2.4 and first let $k_i = 0$ in $|D_1\rangle$, which leads to $k_j = 1$ in $|D_2\rangle$ and thus

$$\sum_{\substack{q < p; r < p \\ q \neq r}} \tilde{h}_{qprp} \tilde{c}_q^\dagger \tilde{c}_p^\dagger \tilde{c}_r \tilde{c}_p = \sum_{\substack{p > j \\ p \in N_{\text{el}}^{|D_1\rangle}}} \tilde{h}_{ipjp} \tilde{c}_i^\dagger \tilde{c}_p^\dagger \tilde{c}_j \tilde{c}_p, \quad \alpha_{\text{JW}} = -(-1)^{k_{i+1} + \dots + k_{j-1}} \quad (2.G.11)$$

$$\sum_{q < p < r} \tilde{h}_{qppr} \tilde{c}_q^\dagger \tilde{c}_p^\dagger \tilde{c}_p \tilde{c}_r = \sum_{\substack{i < p < j \\ p \in N_{\text{el}}^{|D_1\rangle}}} \tilde{h}_{ippj} \tilde{c}_i^\dagger \tilde{c}_p^\dagger \tilde{c}_p \tilde{c}_j, \quad \alpha_{\text{JW}} = (-1)^{k_{i+1} + \dots + k_{p-1} + 1_p + k_{p+1} + \dots + k_{j-1}} \quad (2.G.12)$$

$$\sum_{r < p < q} \tilde{h}_{pqrp} \tilde{c}_p^\dagger \tilde{c}_q^\dagger \tilde{c}_r \tilde{c}_p = \sum_{\substack{j < p < i \\ p \in N_{\text{el}}^{|D_1\rangle}}} \tilde{h}_{pijp} \tilde{c}_p^\dagger \tilde{c}_i^\dagger \tilde{c}_j \tilde{c}_p = 0, \quad \text{since } i < j \quad (2.G.13)$$

$$\sum_{\substack{p < q; p < r \\ q \neq r}} \tilde{h}_{pqrp} \tilde{c}_p^\dagger \tilde{c}_q^\dagger \tilde{c}_p \tilde{c}_r = \sum_{\substack{p < i \\ p \in N_{\text{el}}^{|D_1\rangle}}} \tilde{h}_{pipj} \tilde{c}_p^\dagger \tilde{c}_i^\dagger \tilde{c}_p \tilde{c}_j, \quad \alpha_{\text{JW}} = -(-1)^{k_{i+1} + \dots + k_{j-1}}, \quad (2.G.14)$$

where $p \in N_{\text{el}}^{|D_1\rangle}$ are indices belonging to occupied spin orbitals. Second, if $k_i = 1$ in $|D_1\rangle$, we have $k_j = 1$ in $|D_2\rangle$ and thus

$$\sum_{\substack{q < p; r < p \\ q \neq r}} \tilde{h}_{qprp} \tilde{c}_q^\dagger \tilde{c}_p^\dagger \tilde{c}_r \tilde{c}_p = \sum_{\substack{p > j \\ p \in N_{\text{el}}^{|D_1\rangle}}} \tilde{h}_{jpip} \tilde{c}_j^\dagger \tilde{c}_p^\dagger \tilde{c}_i \tilde{c}_p, \quad \alpha_{\text{JW}} = -(-1)^{k_{i+1} + \dots + k_{j-1}} \quad (2.G.15)$$

$$\sum_{q < p < r} \tilde{h}_{qppr} \tilde{c}_q^\dagger \tilde{c}_p^\dagger \tilde{c}_p \tilde{c}_r = \sum_{\substack{j < p < i \\ p \in N_{\text{el}}^{|D_1\rangle}}} \tilde{h}_{jppi} \tilde{c}_j^\dagger \tilde{c}_p^\dagger \tilde{c}_p \tilde{c}_i = 0, \quad \text{since } i < j \quad (2.G.16)$$

$$\sum_{r < p < q} \tilde{h}_{pqrp} \tilde{c}_p^\dagger \tilde{c}_q^\dagger \tilde{c}_r \tilde{c}_p = \sum_{\substack{i < p < j \\ p \in N_{\text{el}}^{|D_1\rangle}}} \tilde{h}_{pjip} \tilde{c}_p^\dagger \tilde{c}_j^\dagger \tilde{c}_i \tilde{c}_p, \quad \alpha_{\text{JW}} = -(-1)^{k_{i+1} + \dots + k_{p-1} + k_{p+1} + \dots + k_{j-1}} \quad (2.G.17)$$

$$\sum_{\substack{p < q; p < r \\ q \neq r}} \tilde{h}_{pqrp} \tilde{c}_p^\dagger \tilde{c}_q^\dagger \tilde{c}_p \tilde{c}_r = \sum_{\substack{p < i \\ p \in N_{\text{el}}^{|D_1\rangle}}} \tilde{h}_{pjpi} \tilde{c}_p^\dagger \tilde{c}_j^\dagger \tilde{c}_p \tilde{c}_i, \quad \alpha_{\text{JW}} = -(-1)^{k_{i+1} + \dots + k_{j-1}}. \quad (2.G.18)$$

Eqs. (2.G.11)-(2.G.18) give six non-vanishing terms from Eq. (2.G.10) which can create single-connected determinants.

2.G.5 Two-body excitations

Only the first term on the right hand side of Eq. (2.G.10) gives rise to two-body excitations. There are $\binom{4}{2}$ possible non-vanishing determinants, all having an identical form for the Jordan-Wigner phase factors.

Section 2.H

CONVERGENCE OF THE ASCI ENERGY

In Fig 2.H.1, we show how the energy of the reduced system Hamiltonian converges to the exact ground state energy when increasing the number of determinants in the ASCI expansion. A monotonic decreasing behavior as well as a convergence after about five ASCI iterations for all system sizes is clearly visible.

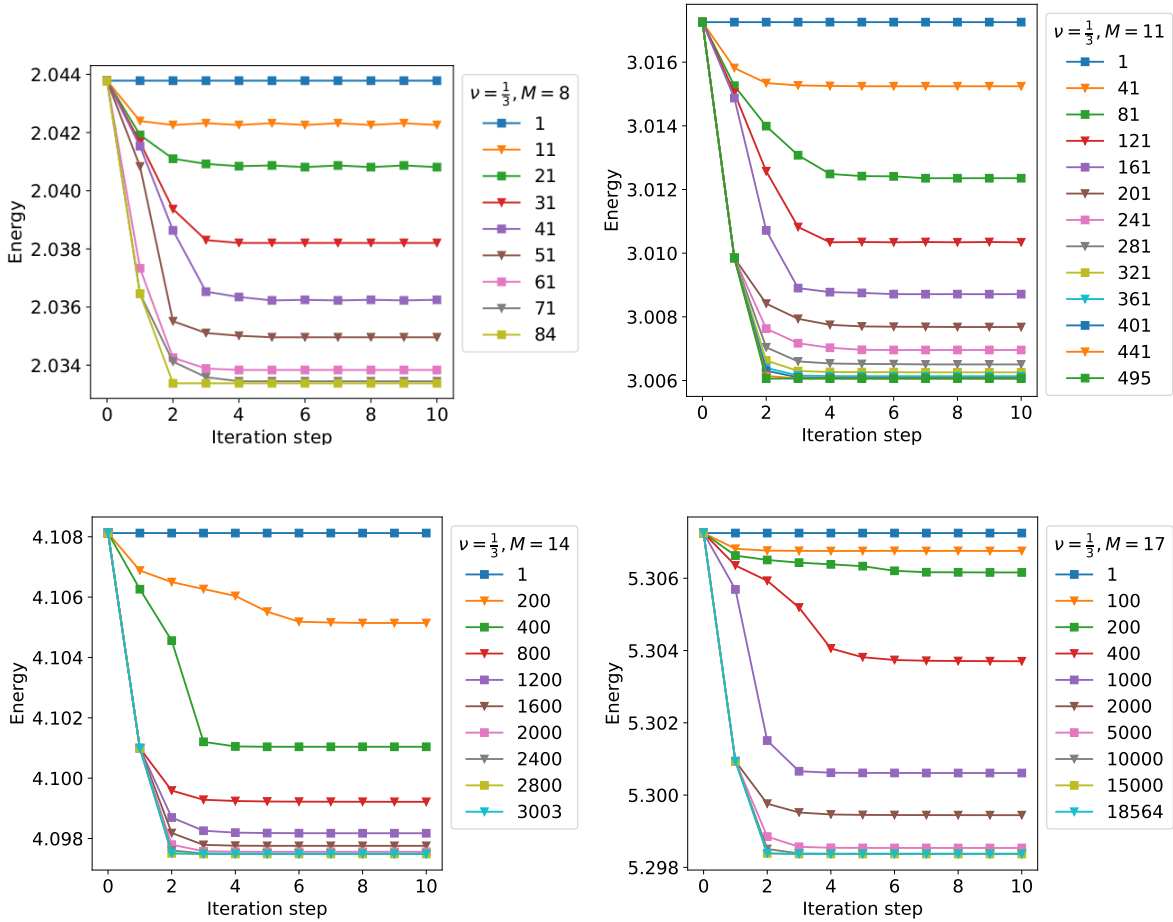


Figure 2.H.1: Scatter plots of the convergence of the energy obtained from diagonalizing the reduced system Hamiltonian at each of the ASCI iteration steps for a filling factor $\nu = 1/3$ and systems containing $N_{so} = 9, 12, 15, 18$ spin-orbitals in the LLL. The top-most data points in each plot belong to the single reference state $|\Psi_{GS}\rangle$ obtained from the method presented in Section 2.4.1, while the bottom-most correspond to the FCI ASCI expansion, i.e. the exact solution, where $t\dets = c\dets = \binom{N_{so}}{N_{el}}$.

CHAPTER 3

EFFICIENTLY COMPUTABLE APPROXIMATE GROUND STATES FOR STRONGLY INTERACTING FERMIONIC SYSTEMS BEYOND GAUSSIAN STATES

One of the greatest challenges of modern condensed matter theory and theoretical chemistry lies in understanding the behavior of strongly correlated electrons at low temperatures, where the properties of the quantum system can be extracted from its ground state and low lying excitations. A recently proposed variational method to study fermionic and bosonic systems presents a new approximate method that extends generalized Hartree-Fock methods beyond mean-field and is able to create non-factorizable correlations between fermionic and bosonic systems with a polynomial computational complexity [41]. While a variety of systems has been studied using this approach, a treatment of general long-range interacting fermionic systems has yet to be realized. In this work, we present all explicit analytical formulas required to apply the variational method to long-range interacting fermionic systems and propose an alternative approach for approximating the ground state in a hybrid scheme which is based on imaginary time evolution and gradient descent. We give an explicit quantum circuit to realize the beyond-mean-field part of the many-body wave function and propose its application as an initial/reference state on a quantum computer.

Section 3.1

MOTIVATION AND MAIN RESULT

The study of strongly interacting fermionic systems can be considered one of the main tasks of modern condensed matter physics. In this work, we will focus on the objective of approximating the ground state of a given Hamiltonian describing N_f interacting fermions, where—due to the exponential growth of the Hilbert space with system size—an exact solution for the ground state quickly becomes intractable on a classical computer. Instead, one has to draw on approximate solutions in order to be able to handle system sizes beyond a few tens of fermionic modes. Various numerical methods, such as density matrix renormalization group [198], attempt to use a relatively small number of states (compared to full configuration interaction) to expand the ground state in order to keep the computational costs manageable. Other approaches, such as generalized Hartree-Fock theory [63], search for approximate solutions to the ground state within a restricted family of states that can be efficiently computed. In generalized Hartree-Fock theory this is possible, since the underlying Fermionic Gaussian States (FGS) can be fully described by their respective $(2N_f \times 2N_f)$ Covariance Matrix (CM). Since FGS obey Wick’s theorem, expectation values of arbitrary products of fermionic operators can be computed from its CM [199] and are thus efficiently simulatable on classical computers. However, FGS do not contain sufficient entanglement to successfully describe most strongly entangled fermionic systems, as they describe the ground state of non-interacting particles.

The recently introduced variational study of fermionic and bosonic systems [41]—which we will from now on call the Variational Method (VM)—expands the family of efficiently computable states to include specific types of Non-Gaussian States (NGS), which are generated by an entangling unitary operator U_S ,

$$|\Psi_{\text{NGS}}\rangle = U_S |\Psi_{\text{GS}}\rangle, \tag{3.1.1}$$

where $|\Psi_{\text{GS}}\rangle = U_{\text{GS}} |0\rangle$ denotes a FGS generated by the unitary operator U_{GS} . In contrast to previous approaches, both U_S and U_{GS} (to be defined in Section 3.3) contain time-dependent variational parameters ω and ξ , respectively. The role of U_S is to introduce entanglement between bosonic and fermionic fields, and in context of purely fermionic or bosonic systems, to take the variational state beyond the family of Gaussian states. While U_S can include various types of unitary operators, we focus on a very specific Ansatz proposed in Ref. [41] for the study of purely fermionic systems. The unitary operator U_S can have a more general form (see Chapter 2 in Ref. [41]), we choose $U_S = U_{\text{FA}}(\omega)$, which is a specific Ansatz for a unitary operator that depends on a set of time-dependent variational parameters ω and whose role is to introduce non-factorizable correlations. Broadly speaking, the reason why one can handle the NGS Ansatz of Eq. (3.1.1) on classical computers is owed to the fact that the specific choice of U_S leads to expectation values of a particular type, for which Ref. [41] presents a generating function.

The VM is based on the time-dependent variational principle [200] that allows for the study of ground state properties of bosonic and fermionic systems, while preserving intrinsic symmetries (such as particle number) and is limited by the type of states that can be reached by the Ansatz of Eq. (3.1.1). By following an imaginary-time evolution of

Table 3.1.1: List containing all abbreviations used in this chapter.

Abbreviation	
Fermionic Gaussian State	FGS
Covariance Matrix	CM
Variational Method	VM
Non-Gaussian State	NGS
Hybrid Imaginary-Time-Gradient-Descent	(HITGD)

the variational parameters, one obtains the lowest energy solution within the variational family and thus an approximate ground state for the system Hamiltonian. This allows one to study not only ground state properties, but also collective excitations and out-of-equilibrium dynamics [41, 201, 202]—which we will however not be concerned with in this work.

The VM was introduced in 2017 and has so far been applied to study (1+1)-dimensional lattice Gauge models [203], the Anderson-Holstein model [204], the quantum Rydberg central spin model [205], the dynamics of a spatially extended bosonic Kondo model [206], the Bose-Hubbard model [207], a trapped Bose-Einstein condensate with attractive s-wave interaction [208], Fermi polarons in one- and two-dimensional lattices [209], and the self-bound states of dipolar Dysprosium condensates [210]. Furthermore, a new method combining the VM with exact diagonalization has been used to study the Hubbard-Holstein model [211]. However, the VM has yet to be applied to purely fermionic systems since the interaction terms consisting of the tensor product of four fermionic operators make the parameter space much more complicated. We will fill this gap by presenting a detailed derivation of all required expressions for implementing the VM to fermionic systems interacting via long-range interactions¹ (such as the Coulomb interaction) in this work.

This chapter is structured as follows. In Section 3.2 we introduce notation. In Section 3.3 we motivate how the VM boils down to the computation of a particular kind of expectation value, which is the subject of Section 3.4. These expressions constitute the pillar the VM is build on. We re-derive the implicit expression for evaluating these type of expectation values as presented in Ref. [41] in Section 3.4.1 and—based on the findings of Section 3.4.2—present a general explicit expression in Section 3.4.3. In Section 3.5 we give an analytical expression for the energy expectation value in terms of the variational parameters ξ and ω . Section 3.6 details the equations of motions of the variational parameters following an imaginary-time evolution and adds analytical expressions for all required expressions when treating general long-range interacting fermionic systems. We show how the energy changes in time for the NGS Ansatz and introduce an alternative algorithm for computing the time evolution of the NGS parameters. This algorithm is based on a hybrid between imaginary time evolution and a gradient descent Ansatz and is introduced in Section 3.7. Using the final set of variational parameters ξ and ω obtained from applying the VM, we give an explicit quantum circuit that shows how the non-Gaussian unitary U_{FA} can be efficiently implemented on a quantum computer using basic N_f single-qubit Pauli- Z rotations and $N_f(N_f + 1)/2$ two-qubit ZZ rotations in logarithmic circuit depth in Section 3.8. Combined with the fact, that FGS can be efficiently implemented on a quantum computer [168], the framework of the VM allows one not only to study strongly interacting fermionic systems on classical computers, but also

¹Long-range interactions are interactions, where the two-body potential that describes the interaction decays algebraically at large distances with a power smaller than the spatial dimension.

to efficiently compute initial/reference states for digital quantum simulations of strongly interacting fermionic systems, going beyond the limitations of generalized Hartree-Fock theory. Since we want this work on fermionic systems to be self-contained, we add extensive derivations both in the main text and Appendices 3.A-3.P, with the hope of making the VM more accessible to a broader audience. In Section 3.9 we summarize our findings and give an outlook for future research directions.

Section 3.2

NOTATIONS

With some minor exceptions, we adapt the notation introduced in Ref. [41]. Throughout this work, we follow the convention to neglect the hat symbol "^\wedge" which is usually used to highlight that a given symbol denotes an operator. We define c_j and c_j^\dagger as the fermionic annihilation and creation operators, $n_j^f = c_j^\dagger c_j$ as the fermionic number operator and let N_f denote the number of fermionic modes in the system. The Majorana operators are defined as [3]

$$a_{1,j} = c_j^\dagger + c_j \tag{3.2.1}$$

$$a_{2,j} = i(c_j^\dagger - c_j). \tag{3.2.2}$$

The respective canonical anti-commutation relations are given by

$$\{c_i, c_j^\dagger\} = \delta_{i,j} \tag{3.2.3}$$

$$\{a_{\alpha,i}, a_{\beta,j}\} = 2\delta_{\alpha,\beta}\delta_{i,j}, \tag{3.2.4}$$

with $\alpha, \beta = 1, 2$ and $i, j = 1, 2, \dots, N_f$. We define the column vectors

$$C = (c_1, \dots, c_{N_f}, c_1^\dagger, \dots, c_{N_f}^\dagger)^T \tag{3.2.5}$$

$$A = (a_{1,1}, \dots, a_{1,N_f}, a_{2,1}, \dots, a_{2,N_f})^T. \tag{3.2.6}$$

Due to the above definitions, for some $(2N_f \times 2N_f)$ -matrix M , we can equivalently write

$$A^T M A = \sum_{j,k=1}^{2N_f} A_j M_{jk} A_k. \tag{3.2.7}$$

We define three column vectors $\mathbf{1}_k$, \mathbf{i}_k and $\mathbf{0}$ of length N_f as

$$\mathbf{1}_k = (0_1, \dots, 0_{k-1}, 1_k, 0_{k+1}, \dots, 0_{N_f})^T \tag{3.2.8}$$

$$\mathbf{i}_k = (0_1, \dots, 0_{k-1}, i_k, 0_{k+1}, \dots, 0_{N_f})^T \tag{3.2.9}$$

$$\mathbf{0} = (0_1, \dots, 0_{N_f})^T, \tag{3.2.10}$$

which will simplify some reoccurring expressions later on. The above definitions provide us with a simple way of writing vectors of length $2N_f$. For instance, for any $p, q \in \{1, \dots, N_f\}$, we can use the short-hand notation

$$(0_1, \dots, 1_p, 0_{p+1}, \dots, i_{q+N_f}, 0_{q+N_f+1}, \dots)^T = \begin{pmatrix} \mathbf{1}_p \\ \mathbf{i}_q \end{pmatrix}.$$

We will use the following notation for the right-hand side sums of expectation values w.r.t. some $(2N_f \times 2N_f)$ -matrix M ,

$$M_{\mathbf{p}^\dagger \mathbf{q}} = \begin{pmatrix} \mathbf{1}_q^T & \mathbf{i}_q^T \\ & -\mathbf{i}_p \end{pmatrix} M \begin{pmatrix} \mathbf{1}_p \\ -\mathbf{i}_p \end{pmatrix} \quad (3.2.11)$$

$$M_{\mathbf{p} \mathbf{q}^\dagger} = \begin{pmatrix} \mathbf{1}_q^T & -\mathbf{i}_q^T \\ & \mathbf{i}_p \end{pmatrix} M \begin{pmatrix} \mathbf{1}_p \\ \mathbf{i}_p \end{pmatrix} \quad (3.2.12)$$

$$M_{\mathbf{p}^\dagger \mathbf{q}^\dagger} = \begin{pmatrix} \mathbf{1}_q^T & -\mathbf{i}_q^T \\ & -\mathbf{i}_p \end{pmatrix} M \begin{pmatrix} \mathbf{1}_p \\ -\mathbf{i}_p \end{pmatrix} \quad (3.2.13)$$

$$M_{\mathbf{p} \mathbf{q}} = \begin{pmatrix} \mathbf{1}_q^T & \mathbf{i}_q^T \\ & \mathbf{i}_p \end{pmatrix} M \begin{pmatrix} \mathbf{1}_p \\ \mathbf{i}_p \end{pmatrix}. \quad (3.2.14)$$

We use boldfaced indices $\mathbf{p}, \mathbf{p}^\dagger$ in order to distinguish the above expressions from regular matrix entries of M , which are denoted by $M_{p,q}$ (or equivalently M_{pq} if this notation does not cause ambiguity). Note that expressions like

$$M_{\mathbf{p}^\dagger \mathbf{q}} = M_{q,p} - iM_{q,N_f+p} + iM_{N_f+q,p} + M_{N_f+q,N_f+p}$$

represent the sum of four matrix elements. When there are expressions as above of sums of products of matrices $A^{\tilde{\alpha}}, B^{\tilde{\alpha}}, \dots$ which all depend on the same $\tilde{\alpha}$, we will use the following compact notation

$$A_{\mathbf{p}^\dagger \mathbf{q}}^{\tilde{\alpha}} + B_{\mathbf{p}^\dagger \mathbf{q}}^{\tilde{\alpha}} = [A_{\mathbf{p}^\dagger \mathbf{q}} + B_{\mathbf{p}^\dagger \mathbf{q}}]^{\tilde{\alpha}}, \quad A_{\mathbf{p}^\dagger \mathbf{q}}^{\tilde{\alpha}} B_{\mathbf{p}^\dagger \mathbf{q}}^{\tilde{\alpha}} = [A_{\mathbf{p}^\dagger \mathbf{q}} B_{\mathbf{p}^\dagger \mathbf{q}}]^{\tilde{\alpha}}. \quad (3.2.15)$$

The symbol $: * :$ indicates normal-ordering of a fermionic / Majorana operator expression $*$, for instance $: n_i n_j := -c_i^\dagger c_j^\dagger c_i c_j$. We will use the short-hand notation $\langle * \rangle_{\text{GS}} = \langle \Psi_{\text{GS}} | * | \Psi_{\text{GS}} \rangle$ to express expectation values w.r.t. FGS.

We denote the Hadamard product as

$$(A \odot B)_{ij} = A_{ij} B_{ij}, \quad (3.2.16)$$

where A and B are here two equally-sized matrices.

For some invertible matrix M , we will equivalently use $M^{-1} = \frac{1}{M}$ to denote its inverse. We denote the Moore-Penrose inverse of a $(m \times n)$ -matrix A of real or complex numbers with A^+ , which by definition satisfies the Moore-Penrose conditions [212, 213]

1. $A^+ A A^+ = A^+$
2. $A A^+ A = A$
3. $(A^+ A)^* = A^+ A$
4. $(A A^+)^* = A A^+$,

where the raised asterisk denotes the adjoint of the respective matrix. A basic property of the pseudo-inverse is that it commutes with the transposition operation, i.e. $(A^+)^T = (A^T)^+$. If A has full rank, the pseudo-inverse is identical to the inverse, i.e. $A^+ = A^{-1}$.

The most important symbols and their respective explanation and appearance are summarized in Table 3.2.1, while Table 3.1.1 lists all used abbreviations.

Table 3.2.1: This table lists and explains the most important symbols appearing in the main text and refers to their respective definition in the third column. Symbols which are explained in the notation section or which only appear in the appendix are not included.

Symbol	Explanation	Equation
$U_{\text{GS}}(\xi) = U_{\text{GS}}$	Unitary operator creating fermionic Gaussian state (FGS)	(3.3.4)
$U_{\text{FA}}(\omega) = U_{\text{FA}}$	Unitary operator creating non-Gaussian state (NGS)	(3.3.5)
ξ	Hermitian and anti-symmetric matrix, variational parameters describing GS	
ω	Symmetric matrix, variational parameters describing NGS	
$ 0\rangle$	State vector, fermionic vacuum state = simultaneous eigenstate of all c_j	
$ \Psi_{\text{NGS}}\rangle$	State vector, non-Gaussian variational state Ansatz $ \Psi_{\text{NGS}}\rangle = U_{\text{FA}}(\omega)U_{\text{GS}}(\xi) 0\rangle$	(3.1.1)
$ \Psi_{\text{GS}}\rangle = U_{\text{GS}} 0\rangle$	State vector, fermionic Gaussian state	
H	Operator, system Hamiltonian in second quantization	(3.3.1)
f_{pq}, h_{pqrs}	Real-valued coefficients, one- and two-body integrals of H	(3.A.2),(3.A.3)
$U_m = e^{i\xi m}$	Orthogonal matrix describing the linear transformation due to the unitary FGS operator	(3.3.7)
Γ_m	Skew-symmetric matrix, covariance matrix (CM)	(3.3.8)
$\Upsilon = \sigma \otimes \mathbf{1}_{N_f}$	Skew-symmetric matrix, relates CM with U_m	(3.3.11)
W_m	Matrix, transforms between fermionic creation and annihilation and the Majorana operators	(3.3.13)
$E = E(\xi, \omega)$	Expectation value of Hamiltonian operator w.r.t. NGS Ansatz	(3.3.17),(3.3.26), (3.3.29),(3.5.1)
$\bar{H} = U_{\text{FA}}^\dagger H U_{\text{FA}}$	Hermitian operator, rotated system Hamiltonian, $\bar{H} = \bar{H}_1 + \bar{H}_2$	(3.3.18)
$f_{pq}^{\text{FA}} = f_{pq}^{\text{FA}}(\omega)$	Coefficient, rotated one-body Hamiltonian coefficients	(3.3.22)
$h_{pqrs}^{\text{FA}} = h_{pqrs}^{\text{FA}}(\omega)$	Coefficient, rotated two-body Hamiltonian coefficients	(3.3.23)
$\alpha_{pq}^{\text{FA}}(k), k = 1, \dots, N_f$	Coefficient, argument of exponential sum of of number operators for \bar{H}_1	(3.3.24)
$\beta_{pqrs}^{\text{FA}}(k), k = 1, \dots, N_f$	Coefficient, argument of exponential sum of of number operators for \bar{H}_2	(3.3.25)
$\mathcal{A}_{j_1 \dots j_a k_1 \dots k_b}^{\tilde{\alpha}}$	Coefficient, expectation values w.r.t. FGS of exponential of $\sum_j \tilde{\alpha}(j)n_j$ multiplied by a (b)-many raising (lowering) operators	(3.3.27),(3.4.70), (3.4.71),(3.4.72)
\mathcal{F}_J	Operator, differential operator w.r.t. Grassmann variables	(3.4.25)
$\tilde{\alpha}$	Vector, contains k entries, each entry $\tilde{\alpha}(j)$ being the weight factor of n_j	
$\Gamma_F^{\tilde{\alpha}}$	Matrix, altered CM	(3.4.62)
$X^{\tilde{\alpha}}$	Polynomial of Grassmann variables	(3.4.66)
$\mathcal{G}^{\tilde{\alpha}}$	Matrix, depends on Γ_m and ω	(3.4.69)
$\mathcal{G}^0 = \Gamma_m + \Upsilon$	Matrix, depends only on Γ_m	(3.7.4)
θ	Vector containing all variational parameters of ξ and ω	(3.6.4)
$ \mathbf{R}_\Psi\rangle$	State vector required for evaluating $d_\tau \theta_j$	(3.6.5)
$ \Psi_j\rangle = \frac{d}{d\theta_j} \Psi_{\text{NGS}}\rangle$	State vector, tangent vector	(3.6.6)
$G_{jk} = \langle \Psi_j \Psi_k \rangle$	Gram matrix, containing the overlaps of the tangent vectors	(3.6.7)
\bar{H}_m	Anti-symmetric matrix, mean-field of rotated system Hamiltonian	(3.6.19)
$\mathcal{L}^{\tilde{\alpha}}$	Matrix, required for evaluating the derivative of $\mathcal{G}^{\tilde{\alpha}}$ w.r.t. the CM	(3.6.20)
$\mathcal{F}^{\tilde{\alpha}}$	Matrix, NOT to be confused with the operator \mathcal{F}_J defined in Eq. (3.4.25)	(3.6.21)
H_μ	Hermitian operator, chemical potential—can be tuned by the free parameter μ	(3.6.29)
O_m	Anti-symmetric imaginary matrix, mean-field matrix of NGS Ansatz	(3.6.33)
g	Vector, contains diagonal elements of off-diagonal block matrix of \mathcal{G}^0	(3.6.35)
B	Tensor of dimensions $(N_f \times N_f \times N_f \times N_f)$	(3.7.5)
$\frac{dE}{d\omega}$	Symmetric matrix, gradient of the energy w.r.t. ω	(3.7.9)

Section 3.3

EMERGING EXPECTATION VALUES

We focus on interacting fermionic systems, whose second quantized Hamiltonian with general two-body interactions is of the general form [2]

$$H = \sum_{p,q=1}^{N_f} f_{pq} c_p^\dagger c_q + \frac{1}{2} \sum_{p,q,r,s=1}^{N_f} h_{pqrs} c_p^\dagger c_q^\dagger c_r c_s, \quad (3.3.1)$$

where N_f is the number of fermionic modes p, q, r, s which index the single-particle bases functions, f_{pq} and h_{pqrs} are the so-called one- and two-body coefficients, as defined in Appendix 3.A. The coefficients possess the symmetries $f_{pq}^* = f_{qp}$ and $h_{pqrs}^* = h_{srqp}$ by definition, and—due to the anti-commutation relations of the fermionic operators—the two-body coefficients can always be chosen such that

$$h_{pqrs} = -h_{qprs} = -h_{pqsr} = h_{qpsr}. \quad (3.3.2)$$

Note that Hamiltonians including higher order polynomials of fermionic operators can also be efficiently studied using the VM since the formula for the appearing expectation values holds for arbitrary (even) order monomials [41].

Our aim is to minimize the energy of the Hamiltonian in Eq. (3.3.1) w.r.t. the variational state²

$$|\Psi_{\text{NGS}}\rangle = U_{\text{FA}} U_{\text{GS}} |0\rangle, \quad (3.3.3)$$

thus obtaining an approximate ground state to the Hamiltonian in Eq. (3.3.1). Here, $|0\rangle \equiv |0^{N_f}\rangle$ denotes the fermionic vacuum state and

$$U_{\text{GS}} = e^{\frac{i}{4} A^T \xi A} = e^{\frac{i}{4} \sum_{j,k=1}^{2N_f} A_j \xi_{jk} A_k} \quad (3.3.4)$$

$$U_{\text{FA}} = e^{\frac{i}{2} \sum_{j,k=1}^{N_f} \omega_{jk} :n_j^f n_k^f:} \quad (3.3.5)$$

are unitary operators which depend on a set of time-dependent variational parameters ξ_{jk} and ω_{jk} , which are collected in an anti-symmetric and Hermitian ($2N_f \times 2N_f$) matrix ξ and a symmetric ($N_f \times N_f$) matrix ω , respectively. State of the form of Eq. (3.3.3), where a linear transformation of the fermionic modes under the action of U_{FA} exists, are known as *generalized Gaussian states* [41]. We will set all diagonal entries $\omega_{ii} = 0$, since they

²As suggested in Ref. [41], one can add another FGS unitary to the variational state, i.o.w. one can choose

$$|\Psi_{\text{NGS}}\rangle = \bar{U}_{\text{GS}}(\bar{\xi}) U_{\text{FA}}(\omega) U_{\text{GS}}(\xi) |0\rangle,$$

where $\bar{\xi}$ is an anti-symmetric and Hermitian matrix and \bar{U}_{GS} is of the form given by Eq. (3.3.4). This will make the Ansatz more flexible, as it introduces additional degrees of freedom to the VM. The VM remains still efficient to compute under this more general Ansatz, since the additional Gaussian unitary just leads to a linear transformation of the orbitals in H before acting with the NGS operator U_{FA} . The appearing expectation values are still identical to the ones derived in Section 3.3. The time evolution of the new set of variational parameters $\bar{\xi}$ can be computed via Eq. (3.6.4). We will restrict ourselves in this work to the simpler Ansatz given by Eq. (3.3.3).

are one-body operators and therefore contained in the Gaussian state. Allowing for the diagonal of ω to be nonzero will lead to some minor corrections to the form of the rotated Hamiltonian and mean-field expressions. For brevity, we will often omit the dependence of the operators on the variational parameters as in Eq. (3.3.3) and ensuing equations.

In the rest of this chapter, we will show how the unitary operators in Eq. (3.3.3) transform fermionic and Majorana modes in Section 3.3.1 and use these results to compute the energy expectation value w.r.t. the NGS Ansatz of Eq. (3.3.3) in Section 3.3.2. We show how this results in expectation values w.r.t. the FGS of operators of a particular form, whose solution is given in the ensuing section.

3.3.1 Unitary transformations of fermionic and Majorana operators

When applied to the fermionic vacuum, the unitary operator U_{GS} creates a pure FGS [3, 24, 41, 63]

$$|\Psi_{\text{GS}}\rangle = U_{\text{GS}}|0\rangle, \quad (3.3.6)$$

that describes e.g. Bardeen-Cooper-Schrieffer states and Slater determinants (the latter possess a fixed particle number) [63]. The generalized canonical transformation defined by the action of U_{GS} on a Majorana operator is described by the following linear operation [3]

$$U_{\text{GS}}^\dagger A_j U_{\text{GS}} = \sum_{k=1}^{2N_f} (U_m)_{jk} A_k, \quad (3.3.7)$$

known as a Bogoliubov transformation, where $U_m = e^{i\xi}$ is an orthogonal matrix. The choice of ξ is not unique since there is a Gauge degree of freedom in the definition of U_{GS} which results in an equivalent class for different parameters $\tilde{\xi}$ describing the same Gaussian transformation. However, the (real and anti-symmetric) CM with matrix entries

$$(\Gamma_m)_{ij} = \frac{i}{2} \langle [A_i, A_j] \rangle_{\text{GS}}, \quad (3.3.8)$$

uniquely defines the FGS. Due to the definition of A in Eq. (3.2.6) the CM has a simple block structure of four $(N_f \times N_f)$ -matrices,

$$\Gamma_m = \begin{pmatrix} (\Gamma_m)_{11} & (\Gamma_m)_{12} \\ (\Gamma_m)_{21} & (\Gamma_m)_{22} \end{pmatrix}, \quad (3.3.9)$$

where $(\Gamma_m)_{21} = -(\Gamma_m)_{12}^T$. Note that we will only consider pure FGS, which are characterized by $\Gamma_m^2 = -\mathbb{1}_{2N_f}$ [63]. The connection between U_m and Γ_m is given by [41]

$$\Gamma_m = -U_m \Upsilon U_m^T, \quad (3.3.10)$$

where we defined

$$\Upsilon = \sigma \otimes \mathbb{1}_{N_f}, \quad \sigma = \begin{pmatrix} 0 & 1 \\ -1 & 0 \end{pmatrix}, \quad (3.3.11)$$

with $\mathbb{1}_{N_f}$ denoting the $(N_f \times N_f)$ -identity matrix. One can obtain ξ from Γ_m through Eq. (3.3.10) numerically by performing a Schur decomposition and applying appropriate

permutation matrices. Similarly, one can define the CM Γ_f in the representation of fermionic creation and annihilation operators using the definition of C in Section 3.2,

$$\Gamma_f = \langle CC^\dagger \rangle_{\text{GS}} = \frac{1}{2} \mathbb{1}_{2N_f} - \frac{i}{4} W_m^\dagger \Gamma_m W_m, \quad (3.3.12)$$

where

$$W_m = \begin{pmatrix} \mathbb{1}_{N_f} & \mathbb{1}_{N_f} \\ -i\mathbb{1}_{N_f} & i\mathbb{1}_{N_f} \end{pmatrix} \quad (3.3.13)$$

is the matrix that transforms between the Majorana and fermionic representation through

$$A = W_m C. \quad (3.3.14)$$

An important property of FGS is that Wick's theorem applies, namely, expectation values of products of fermionic (or Majorana-) operators split into sums of products of the CM elements [3], which we will discuss in more detail in Appendix 3.B.

The unitary operator U_{FA} in Eq. (3.3.5) takes us beyond the family of FGS, as its exponential includes fermionic operators of quartic order. As shown in Appendix 3.C, U_{FA} transforms a fermionic operator as follows,

$$U_{\text{FA}}^\dagger c_k U_{\text{FA}} = e^{i \sum_j \omega_{jk} n_j^f} c_k, \quad (3.3.15)$$

attaching a phase depending on particles in modes j to the fermionic mode k , reminiscent of the flux attachment (FA) procedure known from composite fermion theory [9]. We can commute the single fermionic operator on the right-hand side of Eq. (3.3.15) past the exponential term through

$$e^{i \sum_j \omega_{jq} n_j^f} c_p = e^{-i \omega_{pq}} c_p e^{i \sum_j \omega_{jq} n_j^f}. \quad (3.3.16)$$

In the rest of this paper, we shall make extensive use of Eq. (3.3.7) and Eqs. (3.3.15)-(3.3.16), as well as the properties of the CM.

3.3.2 Deriving the energy functional

At the heart of the variational method lies the ability to efficiently compute the energy expectation value of H w.r.t. the NGS of Eq. (3.1.1),

$$E = \langle \Psi_{\text{NGS}} | H | \Psi_{\text{NGS}} \rangle. \quad (3.3.17)$$

Using Eqs. (3.3.15) and (3.3.16), we can equivalently compute the energy of the variational state Ansatz of Eq. (3.3.3) as an expectation value of the rotated Hamiltonian

$$\bar{H} = U_{\text{FA}}^\dagger H U_{\text{FA}} \quad (3.3.18)$$

w.r.t. the FGS, where \bar{H} is given by

$$\bar{H} = \bar{H}_1 + \bar{H}_2, \quad (3.3.19)$$

where

$$\bar{H}_1 = \sum_{p,q=1}^{N_f} f_{pq}^{\text{FA}} e^{i \sum_k \alpha_{pq}^{\text{FA}}(k) n_k^f} c_p^\dagger c_q, \quad (3.3.20)$$

$$\bar{H}_2 = \frac{1}{2} \sum_{p,q,r,s=1}^{N_f} h_{pqrs}^{\text{FA}} e^{i \sum_k \beta_{pqrs}^{\text{FA}}(k) n_k^f} c_p^\dagger c_q^\dagger c_r c_s, \quad (3.3.21)$$

are the rotated one- and two-body Hamiltonian terms—as detailed in Appendix 3.D— $k = 1, \dots, N_f$. We also defined the ω -dependent coefficients

$$f_{pq}^{\text{FA}} = f_{pq} e^{-i\omega pq}, \quad (3.3.22)$$

$$h_{pqrs}^{\text{FA}} = h_{pqrs} e^{i(\omega_{rs} + \omega_{pq} - \omega_{qr} - \omega_{pr} - \omega_{qs} - \omega_{ps})}, \quad (3.3.23)$$

and phase factors belonging to the argument of the exponential of the sum of fermionic number operators of the unitary operator U_{FA} ,

$$\alpha_{pq}^{\text{FA}}(k) = \omega_{kq} - \omega_{kp}, \quad (3.3.24)$$

$$\beta_{pqrs}^{\text{FA}}(k) = \omega_{kr} + \omega_{ks} - \omega_{kp} - \omega_{kq}. \quad (3.3.25)$$

It is important to note that the operators of the rotated Hamiltonian \bar{H} in Eq. (3.3.19) differ considerably from the operators in Eq. (3.3.1). Meanwhile, the total number of terms appearing in \bar{H} is identical to the number of terms in the unrotated Hamiltonian H . Combining Eq. (3.3.17) with Eq. (3.3.18), we can write

$$E = \langle \bar{H} \rangle_{\text{GS}}. \quad (3.3.26)$$

Since taking the expectation value is a linear operation, one is able to compute Eq. (3.3.26) (even for higher-order Hamiltonians), given that one is able to compute expressions of the following form,

$$\mathcal{A}_{j_1^\dagger \dots j_a^\dagger k_1 \dots k_b}^{\tilde{\alpha}} = \left\langle e^{i \sum_j \tilde{\alpha}(j) n_j^f} c_{j_1}^\dagger \dots c_{j_a}^\dagger c_{k_1} \dots c_{k_b} \right\rangle_{\text{GS}} \quad (3.3.27)$$

Here, $a, b = 1, \dots, N_f$ and we let $j_1^\dagger, \dots, j_a^\dagger$ denote the modes of the creation operators, k_1, \dots, k_b the modes of the annihilation operators and $\tilde{\alpha}(j) \in \mathbb{R}$ with $j = 1, \dots, N_f$ correspond to the various phase factors appearing in the argument of the exponential of the sum of fermionic number operators³. In particular, we have

$$\mathcal{A}^{\tilde{\alpha}} = \left\langle e^{i \sum_j \tilde{\alpha}(j) n_j^f} \right\rangle_{\text{GS}}, \quad (3.3.28)$$

which will depend on the form of the coefficients in the arguments $\tilde{\alpha}$. We note that explicit analytical formulas for Eq. (3.3.27) were given in Ref. [41] only for Eq. (3.3.28), as well as for $\mathcal{A}_{p^\dagger q}^{\tilde{\alpha}}$. Throughout this chapter we will see that expressions as in Eq. (3.3.27) will not only arise when computing the energy expectation value, but also when using various methods designed to find the ground state. Using the above definitions, Eq. (3.3.26) can be written as

$$E(\xi, \omega) = \sum_{p,q} f_{pq}^{\text{FA}} \mathcal{A}_{p^\dagger q}^{\alpha_{pq}^{\text{FA}}} + \frac{1}{2} \sum_{p,q,r,s} h_{pqrs}^{\text{FA}} \mathcal{A}_{p^\dagger q^\dagger rs}^{\beta_{pqrs}^{\text{FA}}}, \quad (3.3.29)$$

³The modes $j_1^\dagger, \dots, j_a^\dagger$ and k_1, \dots, k_b do not necessarily have to be ordered, meaning we do not require $j_1 < \dots < j_a$.

which depends implicitly on the variational parameters ξ and ω in both, the Hamiltonian coefficients f_{pq}^{FA} and h_{pqrs}^{FA} and the expectation values $\mathcal{A}_{p^\dagger q}^{\alpha_{pq}^{\text{FA}}}$ and $\mathcal{A}_{p^\dagger q^\dagger rs}^{\beta_{pqrs}^{\text{FA}}}$, which we emphasize by writing $E = E(\xi, \omega)$. As derived in Appendix 3.F, the terms in Eq. (3.3.29) possess the following symmetry,

$$\left(f_{pq}^{\text{FA}}\right)^* \left(\mathcal{A}_{p^\dagger q}^{\alpha_{pq}^{\text{FA}}}\right)^* = f_{qp}^{\text{FA}} \mathcal{A}_{q^\dagger p}^{\alpha_{pq}^{\text{FA}}} \quad (3.3.30)$$

$$\left(h_{pqrs}^{\text{FA}}\right)^* \left(\mathcal{A}_{p^\dagger q^\dagger rs}^{\beta_{pqrs}^{\text{FA}}}\right)^* = h_{srqp}^{\text{FA}} \mathcal{A}_{s^\dagger r^\dagger qp}^{\beta_{pqrs}^{\text{FA}}}, \quad (3.3.31)$$

where the raised asterisk denotes the Hermitian conjugate. In the following subsection we will show, that for arbitrary order a, b , computing Eq. (3.3.27) only involves the expressions $\mathcal{A}^{\tilde{\alpha}}, \mathcal{A}_{p^\dagger q}^{\tilde{\alpha}}, \mathcal{A}_{p^\dagger q^\dagger}^{\tilde{\alpha}}$ and $\mathcal{A}_{pq}^{\tilde{\alpha}}$. For practical reasons we restrict ourselves to the case $a = b$ in our derivations, but our results can also be applied to $a \neq b$ for $a, b > 1$.

Section 3.4

ANALYTICAL FORMULAS FOR EMERGING EXPECTATION VALUES

In this section we will provide an explicit analytical formula for the expectation values of arbitrary products of fermionic operators of the type given by Eq. (3.3.27). In order to simplify notation, we will mostly omit the vector $\tilde{\alpha}$ containing the information about the phase factors in Eq. (3.3.27) in the following two sections.

This section is structured as follows. In Section 3.4.1 we provide a detailed derivation of the generating function for expressions of Eq. (3.3.27), reproducing the result of Ref. [41]. Most of the derivation that leads to Eq. (3.4.67) in Ref. [41] is left to the reader, which is why it is included here in detail. This makes Section 3.4.1 rather technical, but nevertheless important, as it provides the foundation that the rest of this work (and the VM) is build on. Section 3.4.2 displays the types of expectation values which are needed to evaluate the general expression through the explicit formula we provide in Section 3.4.3. The derivation for the expectation values is provided in Appendix 3.E.

3.4.1 Derivation of implicit formula

At the heart of Ref.[41] lies the ability to efficiently compute expectation values of polynomials of fermionic operators of the general form

$$\left\langle e^{i \sum_j \tilde{\alpha}(j) n_j^f} \text{poly}(C) \right\rangle_{\text{GS}}, \quad (3.4.1)$$

where $\tilde{\alpha}(j)$ denotes some real-valued parameter and the lower case index reminds us that the expectation value is to be taken w.r.t. the FGS, i.e. $|\Psi_{\text{GS}}\rangle = U_{\text{GS}} |0\rangle$, and C contains fermionic creation and annihilation operators. In principle, the above sum over j can include all N_f fermionic modes, but some (or even all) of the $\tilde{\alpha}(j)$'s might be zero, in which case the sum will be shorter, or even vanish. As it turns out, these expectation values can be computed analytically by means of the coherent representation of Gaussian states, which we introduced in Section 1.3. This section will present a detailed derivation of the result of Eq. (D.8) in Ref. [41], which presents a generating function for evaluating expression of the form of Eq. (3.3.27), or—since taking the expectation value is a linear operation—equivalently Eq. (3.4.1).

The characteristic function $\chi(\boldsymbol{\eta})$ of a vector of Grassmann variables $\boldsymbol{\eta} = (\eta_1, \eta_2, \dots, \eta_{N_f})^T$ for a state ρ of a collection of N_f fermionic modes is defined as

$$\chi(\boldsymbol{\eta}) = \text{tr} [\rho D(\boldsymbol{\eta})], \quad (3.4.2)$$

where $D(\boldsymbol{\eta})$ is the fermionic displacement operator

$$D(\boldsymbol{\eta}) = \exp \left(\sum_i (c_i^\dagger \eta_i + \eta_i^* c_i) \right), \quad (3.4.3)$$

defining the coherent state $D(\boldsymbol{\eta})|0\rangle = |\boldsymbol{\eta}\rangle$ with $c_i|\boldsymbol{\eta}\rangle = \eta_i|\boldsymbol{\eta}\rangle$ and η_i, η_i^* are Grassmann variables, where η_i^* is the conjugated variable of η_i ⁴ and Grassmann variables satisfy

$$\{\eta_i, \eta_j\} = 0, \quad (3.4.4)$$

$$\{\eta_i^*, \eta_j\} = 0, \quad (3.4.5)$$

$$\{\eta_i^*, \eta_j^*\} = 0. \quad (3.4.6)$$

Any fermionic density operator ρ may be expanded in terms of the characteristic function through [4]

$$\rho = \int d^2\boldsymbol{\eta} \chi(\boldsymbol{\eta}) E(\boldsymbol{\eta}) e^{\frac{1}{2} \sum_n \eta_n^* \eta_n}, \quad (3.4.7)$$

where $E(\boldsymbol{\eta})$ is the Fourier transform of a coherent state dyadic

$$E(\boldsymbol{\eta}) = \int d^2\boldsymbol{\mu} e^{\sum_n (\eta_n \mu_n^* - \mu_n \eta_n^*)} |\boldsymbol{\mu}\rangle \langle -\boldsymbol{\mu}|. \quad (3.4.8)$$

For the density matrix of a FGS,

$$\rho_{\text{GS}} = \int d^2\boldsymbol{\eta} \int d^2\boldsymbol{\mu} \chi(\boldsymbol{\eta}) e^{\frac{1}{2} \sum_n \eta_n^* \eta_n} e^{\sum_m (\mu_m \eta_m^* - \eta_m \mu_m^*)} |\boldsymbol{\mu}\rangle \langle -\boldsymbol{\mu}|, \quad (3.4.9)$$

the characteristic function χ is given by the CM through [41]

$$\chi(\boldsymbol{\eta}) = \text{tr} [\rho_{\text{GS}} D(\boldsymbol{\eta})] = e^{\frac{i}{8} (\boldsymbol{\eta}_1, \boldsymbol{\eta}_2) \Gamma_m (\boldsymbol{\eta}_1, \boldsymbol{\eta}_2)^T}, \quad (3.4.10)$$

where $\boldsymbol{\eta}_1 = \boldsymbol{\eta}^* + \boldsymbol{\eta}$ and $\boldsymbol{\eta}_2 = i(\boldsymbol{\eta}^* - \boldsymbol{\eta})$ are vectors of real Grassmann variables, where "real" refers to them remaining invariant under complex conjugation. We will often use the peculiar way of writing column or row vectors of length $2N_f$ as a vector of length 2, where each entry is itself a vector of length N_f —this is how the above terms $(\boldsymbol{\eta}_1, \boldsymbol{\eta}_2)$ and $(\boldsymbol{\eta}_1, \boldsymbol{\eta}_2)^T$ should be understood.

Any term in Eq. (3.4.1) that will be of interest to us will be of the form

$$\mathcal{A}_{j_1^\dagger \dots j_a^\dagger k_1 \dots k_a} = \left\langle e^{i \sum_j \tilde{\alpha}^{(j)} n_j^f} c_{j_1}^\dagger \dots c_{j_a}^\dagger c_{k_1} \dots c_{k_a} \right\rangle_{\text{GS}} = \text{tr} \left[\rho_{\text{GS}} e^{i \sum_j \tilde{\alpha}^{(j)} n_j^f} c_{j_1}^\dagger \dots c_{j_a}^\dagger c_{k_1} \dots c_{k_a} \right], \quad (3.4.11)$$

which is here only considering number conserving terms (otherwise we would have j_1, \dots, j_a and k_1, \dots, k_b indices with $a \neq b$)⁵. We will ignore the superscript $\tilde{\alpha}$ throughout this chapter, as we are considering just a single expression. In order to compute this expression, we will have to use some properties of the Grassmann calculus properties introduced in Section 1.3.2.

The identity operator and trace over an operator B in terms of Grassmann variables are given by

$$\mathbb{1} = \int d^2\boldsymbol{\eta} |\boldsymbol{\eta}\rangle \langle \boldsymbol{\eta}| \quad (3.4.12)$$

$$\text{tr} [B] = \int d^2\boldsymbol{\alpha} \langle -\boldsymbol{\alpha} | B | \boldsymbol{\alpha} \rangle = \int d^2\boldsymbol{\alpha} \langle \boldsymbol{\alpha} | B | -\boldsymbol{\alpha} \rangle. \quad (3.4.13)$$

⁴Conjugated Grassmann variables satisfy $(\eta_i \eta_i^*)^* = \eta_i \eta_i^*$ and $(\eta_i^* \eta_i)^* = \eta_i^* \eta_i$, conjugated pairs are thus Hermitian.

⁵The case $a \neq b$ can be treated analogously.

Note that the "tilde" is often used to distinguish between the real coefficients $\tilde{\alpha}(j)$ in the exponent and Grassmann numbers α_j from the Grassmann vector $|\boldsymbol{\alpha}\rangle$. In the following, we will calculate expressions like

$$\langle \boldsymbol{\alpha} | e^{i \sum_{j=1}^{N_f} \tilde{\alpha}(j) n_j} | \boldsymbol{\beta} \rangle = \langle \boldsymbol{\alpha} | \prod_{j=1}^{N_f} e^{i \tilde{\alpha}(j) n_j} | \boldsymbol{\beta} \rangle, \quad (3.4.14)$$

where we will be inserting the identity of Eq. (3.4.12) and using the coherent state overlap given by

$$\langle \boldsymbol{\alpha} | \boldsymbol{\gamma} \rangle = e^{\sum_i (\alpha_i^* \gamma_i - \frac{1}{2} \alpha_i^* \alpha_i - \frac{1}{2} \gamma_i^* \gamma_i)} \quad (3.4.15)$$

as well as the relation

$$e^{i \tilde{\alpha}(j) n_j} = \mathbb{1} + (e^{i \tilde{\alpha}(j)} - 1) n_j. \quad (3.4.16)$$

We can normal order the operator product by factoring out all terms and writing them as a sum over all partitions \mathcal{S} containing k elements,

$$e^{i \sum_{j=1}^{N_f} \tilde{\alpha}(j) n_j} = \prod_{j=1}^{N_f} (\mathbb{1} + (e^{i \tilde{\alpha}(j)} - 1) c_j^\dagger c_j) = \sum_{k=1}^{N_f} \sum_{\substack{\mathcal{S} \subseteq \mathbb{N}_{N_f} \\ |\mathcal{S}|=k}} : \prod_{l \in \mathcal{S}} (\mathbb{1} + (e^{i \tilde{\alpha}(l)} - 1) c_l^\dagger c_l) :, \quad (3.4.17)$$

where $::$ denotes normal ordering. Thus, Eq. (3.4.14) reads

$$\begin{aligned} \langle \boldsymbol{\alpha} | e^{i \sum_{j=1}^{N_f} \tilde{\alpha}(j) n_j} | \boldsymbol{\beta} \rangle &= \prod_{j=1}^{N_f} (1 + (e^{i \tilde{\alpha}(j)} - 1) \alpha_j^* \beta_j) \langle \boldsymbol{\alpha} | \boldsymbol{\beta} \rangle \\ &= \left(\prod_{j=1}^{N_f} e^{(e^{i \tilde{\alpha}(j)} - 1) \alpha_j^* \beta_j} \right) e^{\sum_i (\alpha_i^* \beta_i - \frac{1}{2} \alpha_i^* \alpha_i - \frac{1}{2} \beta_i^* \beta_i)} \end{aligned} \quad (3.4.18)$$

and specifically

$$\langle \boldsymbol{\alpha} | e^{i \sum_{j=1}^{N_f} \tilde{\alpha}(j) n_j} | \boldsymbol{\alpha} \rangle = \left(\prod_{j=1}^{N_f} e^{(e^{i \tilde{\alpha}(j)} - 1) \alpha_j^* \alpha_j} \right). \quad (3.4.19)$$

For a normal-ordered fermionic operator O , the following relation holds [4]

$$\text{tr} [\rho O] = \int d^2 \boldsymbol{\alpha} P(\boldsymbol{\alpha}) \langle \boldsymbol{\alpha} | O | \boldsymbol{\alpha} \rangle, \quad (3.4.20)$$

where $P(\boldsymbol{\alpha})$ is the normal-ordered weight function

$$P(\boldsymbol{\alpha}) = \int d^2 \boldsymbol{\xi} e^{\sum_n (\alpha_n \xi_n^* - \xi_n \alpha_n^*)} \chi(\boldsymbol{\xi}) e^{\frac{1}{2} \sum_n \xi_n^* \xi_n}. \quad (3.4.21)$$

By using the commutation relations (which can be proven via e.g. Baker-Campbell-Hausdorff expansion)

$$e^{i \sum_j \tilde{\alpha}(j) n_j^f} c_p = e^{-i \tilde{\alpha}(p)} c_p e^{i \sum_j \tilde{\alpha}(j) n_j^f} \quad (3.4.22)$$

$$c_p^\dagger e^{-i \sum_j \tilde{\alpha}(j) n_j^f} = e^{i \tilde{\alpha}(p)} e^{-i \sum_j \tilde{\alpha}(j) n_j^f} c_p^\dagger, \quad (3.4.23)$$

one obtains

$$\langle \boldsymbol{\alpha} | e^{i \sum_{j=1}^{N_f} \tilde{\alpha}(j) n_j} c_{j_1}^\dagger \cdots c_{j_a}^\dagger c_{k_1} \cdots c_{k_a} | \boldsymbol{\alpha} \rangle = \langle \boldsymbol{\alpha} | \left(\prod_{l=1}^a e^{i \tilde{\alpha}(j_l)} \alpha_{j_l}^* \right) \left(\prod_{m=1}^a \alpha_{k_m} \right) e^{i \sum_{j=1}^{N_f} \tilde{\alpha}(j) n_j} | \boldsymbol{\alpha} \rangle, \quad (3.4.24)$$

see Appendix 3.P.1. Inserting Eq. (3.4.24) into (3.4.20) and defining the differential operator w.r.t. the Grassmann variables J_{j_1}, \dots, J_{j_a} [41],

$$\mathcal{F}_J = \lim_{J \rightarrow 0} \frac{d}{dJ_{j_1}} \cdots \frac{d}{dJ_{j_a}} \frac{d}{dJ_{k_1}^*} \cdots \frac{d}{dJ_{k_a}^*}, \quad (3.4.25)$$

we can rewrite the second line of Eq. (3.4.11) as

$$\begin{aligned} \mathcal{A}_{j_1^\dagger \dots j_a^\dagger k_1 \dots k_a} &= \mathcal{F}_J \int d^2 \boldsymbol{\xi} \chi(\boldsymbol{\xi}) e^{\frac{1}{2} \sum_n \xi_n^* \xi_n} \\ &\times \int d^2 \boldsymbol{\alpha} \left(\prod_{n=1}^{N_f} e^{-(1-e^{i\tilde{\alpha}(n)}) \alpha_n^* \alpha_n + (J_n^* - \xi_n^*) \alpha_n + (J_n e^{i\tilde{\alpha}(n)} - \xi_n) \alpha_n^*} \right), \end{aligned} \quad (3.4.26)$$

see Appendix 3.P.2. We will be focusing on performing the above integral w.r.t. $\boldsymbol{\alpha}$ in the following part. Since all terms appearing in the argument of the second integral of Eq. (3.4.26) commute, we can rewrite it as

$$\begin{aligned} &\prod_{n=1}^{N_f} e^{-(1-e^{i\tilde{\alpha}(n)}) \alpha_n^* \alpha_n + (J_n^* - \xi_n^*) \alpha_n + (J_n e^{i\tilde{\alpha}(n)} - \xi_n) \alpha_n^*} \\ &= e^{\sum_{n=1}^{N_f} \left(-(1-e^{i\tilde{\alpha}(n)}) \alpha_n^* \alpha_n + (J_n^* - \xi_n^*) \alpha_n + (J_n e^{i\tilde{\alpha}(n)} - \xi_n) \alpha_n^* \right)}. \end{aligned} \quad (3.4.27)$$

Given a vector of Grassmann variables $\boldsymbol{\alpha}$, a fixed Grassmann vector $\boldsymbol{\xi}$ and an anti-symmetric matrix A , the Gaussian Grassmann integral is given by [214]

$$\int d\boldsymbol{\alpha} e^{-\frac{1}{2} \boldsymbol{\alpha}^T A \boldsymbol{\alpha} + \boldsymbol{\xi}^T \boldsymbol{\alpha}} = \text{Pf}(A) e^{-\frac{1}{2} \boldsymbol{\xi}^T A^{-1} \boldsymbol{\xi}}, \quad (3.4.28)$$

where $\text{Pf}(A) = \sqrt{\det(A)}$ denotes the Pfaffian of the matrix A [215]. In order to apply Eq. (3.4.28) to the second integral in Eq. (3.4.27), we introduce the Grassmann vectors

$$\vec{\boldsymbol{\alpha}} = \begin{pmatrix} \alpha_1 \\ \vdots \\ \alpha_{N_f} \\ \alpha_1^* \\ \vdots \\ \alpha_{N_f}^* \end{pmatrix}, \quad \vec{\boldsymbol{\xi}} = \begin{pmatrix} J_1^* - \xi_1^* \\ \vdots \\ J_{N_f}^* - \xi_{N_f}^* \\ J_1 e^{i\tilde{\alpha}_1} - \xi_1 \\ \vdots \\ J_{N_f} e^{i\tilde{\alpha}_{N_f}} - \xi_{N_f} \end{pmatrix},$$

where $\vec{\boldsymbol{\xi}}$ is chosen such that

$$\vec{\boldsymbol{\xi}}^T \vec{\boldsymbol{\alpha}} = \sum_{n=1}^{N_f} \left((J_n^* - \xi_n^*) \alpha_n + (J_n e^{i\tilde{\alpha}(n)} - \xi_n) \alpha_n^* \right) \quad (3.4.29)$$

and the form of A is determined by the equation

$$-\frac{1}{2}\vec{\alpha}^T A \vec{\alpha} = -\sum_{n=1}^{N_f} (1 - e^{i\tilde{\alpha}(n)}) \alpha_n^* \alpha_n. \quad (3.4.30)$$

Eq. (3.4.30) leads to

$$A = \sigma \otimes \text{diag} (1 - e^{i\tilde{\alpha}}), \quad (3.4.31)$$

where σ is defined in Eq. (3.3.11) and we defined the $(N_f \times N_f)$ -diagonal matrix

$$\text{diag} (1 \pm e^{i\tilde{\alpha}}) = \begin{pmatrix} 1 \pm e^{i\tilde{\alpha}(1)} & & \\ & \ddots & \\ & & 1 \pm e^{i\tilde{\alpha}(N_f)} \end{pmatrix}. \quad (3.4.32)$$

The matrix A we used here has unfortunately the same letter as the vector of Majorana operators. Since the latter does not appear in this section and we merely use the matrix A in an intermediate step, we do not change notation, sticking to the conventions used in the literature. We compute the determinant of the matrix A next, using the property that for two square matrices B, C , of sizes $(m_B \times m_B)$ and $(m_C \times m_C)$ respectively, the determinant of the tensor product may be written as

$$\det(B \otimes C) = \det(B)^{m_C} \det(C)^{m_B}, \quad (3.4.33)$$

thus

$$\det(A) = \det(\sigma)^{N_f} \det(\text{diag}(1 - e^{i\tilde{\alpha}}))^2 = \prod_{j=1}^{N_f} (1 - e^{i\tilde{\alpha}(j)})^2 \quad (3.4.34)$$

and therefore

$$\sqrt{\det(A)} = \prod_{j=1}^{N_f} (1 - e^{i\tilde{\alpha}(j)}), \quad (3.4.35)$$

which is the prefactor in front of the exponential term in Eq. (3.4.28). Next, we compute the inverse of the matrix A , by using the property, that for any tensor product of two matrices B and C we have

$$(B \otimes C)^{-1} = B^{-1} \otimes C^{-1}. \quad (3.4.36)$$

The inverse of σ is given by $\sigma^{-1} = -\sigma$, and we have

$$A^{-1} = \sigma^{-1} \otimes \text{diag} (1 - e^{i\tilde{\alpha}})^{-1}. \quad (3.4.37)$$

We rewrite the argument of the exponential in Eq. (3.4.28),

$$-\frac{1}{2}\vec{\xi}^T A^{-1} \vec{\xi} = \sum_{k=1}^{N_f} \frac{J_k (J_k^* - \xi_k^*) e^{i\tilde{\alpha}(k)} - \xi_k (J_k^* - \xi_k^*)}{1 - e^{i\tilde{\alpha}(k)}}. \quad (3.4.38)$$

Inserting Eqs. (3.4.35) and (3.4.38), with the replacement Eq. (3.4.27), into the second integral in Eq. (3.4.26) gives

$$\begin{aligned} & \int d^2 \boldsymbol{\alpha} e^{\sum_{n=1}^{N_f} (-1 - e^{i\tilde{\alpha}(n)}) \alpha_n^* \alpha_n + (J_n^* - \xi_n^*) \alpha_n + (J_n e^{i\tilde{\alpha}(n)} - \xi_n) \alpha_n^*} \\ &= \left(\prod_{j=1}^{N_f} (1 - e^{i\tilde{\alpha}(j)}) \right) e^{\sum_{k=1}^{N_f} \frac{J_k J_k^* e^{i\tilde{\alpha}(k)} - J_k \xi_k^* e^{i\tilde{\alpha}(k)} - \xi_k J_k^* + \xi_k \xi_k^*}{1 - e^{i\tilde{\alpha}(k)}}} \end{aligned} \quad (3.4.39)$$

which leads to

$$\begin{aligned} \mathcal{A}_{j_1^\dagger \dots j_a^\dagger k_1 \dots k_a} &= \left(\prod_{j=1}^{N_f} (1 - e^{i\tilde{\alpha}(j)}) \right) \mathcal{F}_J e^{\sum_{k=1}^{N_f} \frac{J_k J_k^* e^{i\tilde{\alpha}(k)}}{1 - e^{i\tilde{\alpha}(k)}}} \\ &\times \int d^2 \boldsymbol{\xi} \chi(\boldsymbol{\xi}) e^{\sum_{k=1}^{N_f} \frac{-J_k \xi_k^* e^{i\tilde{\alpha}(k)} - \xi_k J_k^* - \frac{1}{2} \xi_k^* \xi_k (1 + e^{i\tilde{\alpha}(k)})}{1 - e^{i\tilde{\alpha}(k)}}}, \end{aligned} \quad (3.4.40)$$

which is identical to Eq. (D.6) in Ref. [41]. Using the real Grassmann vectors $\boldsymbol{\xi}_1 = \boldsymbol{\xi}^* + \boldsymbol{\xi}$, $\boldsymbol{\xi}_2 = i(\boldsymbol{\xi}^* - \boldsymbol{\xi})$, where $\boldsymbol{\xi} = \frac{1}{2}(\boldsymbol{\xi}_1 + i\boldsymbol{\xi}_2)$, $\boldsymbol{\xi}^* = \frac{1}{2}(\boldsymbol{\xi}_1 - i\boldsymbol{\xi}_2)$, we want to perform a change of integration variables next. For a set of Grassmann variables $\boldsymbol{\theta} = (\theta_1, \dots, \theta_{2N_f})$, a change of integration variables $\theta'_i = \sum_{j=1}^{2N_f} R_{ij} \theta_j$ is given by

$$\int d\boldsymbol{\theta} f(\boldsymbol{\theta}) = \det(R) \int d\boldsymbol{\theta}' f(\boldsymbol{\theta}'), \quad (3.4.41)$$

where $f(\boldsymbol{\theta})$ is a polynomial function of the Grassmann variables. In order to go from the set of Grassmann variables $(\boldsymbol{\xi}, \boldsymbol{\xi}^*)$ to $(\boldsymbol{\xi}_1, \boldsymbol{\xi}_2)$, we first find the explicit form of the rotation matrix R from $(\boldsymbol{\xi}_1, \boldsymbol{\xi}_2)^T = R(\boldsymbol{\xi}, \boldsymbol{\xi}^*)^T$,

$$\begin{pmatrix} \xi_{1,1} \\ \xi_{1,2} \\ \vdots \\ \xi_{1,N_f} \\ \xi_{2,1} \\ \xi_{2,2} \\ \vdots \\ \xi_{2,N_f} \end{pmatrix} = \begin{pmatrix} 1 & 0 & \dots & 0 & 1 & 0 & \dots & 0 \\ 0 & 1 & \dots & 0 & 0 & 1 & \dots & 0 \\ \vdots & \vdots & \ddots & \vdots & \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \dots & 1 & 0 & 0 & \dots & 1 \\ -i & 0 & \dots & 0 & i & 0 & \dots & 0 \\ 0 & -i & \dots & 0 & 0 & i & \dots & 0 \\ \vdots & \vdots & \ddots & \vdots & \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \dots & -i & 0 & 0 & \dots & i \end{pmatrix} \begin{pmatrix} \xi_1 \\ \xi_2 \\ \vdots \\ \xi_{N_f} \\ \xi_1^* \\ \xi_2^* \\ \vdots \\ \xi_{N_f}^* \end{pmatrix}. \quad (3.4.42)$$

The determinant of R can be simply derived by the following formula, which holds for square matrices A, B, C, D of equal lengths with $[C, D] = 0$,

$$\det \begin{pmatrix} A & B \\ C & D \end{pmatrix} = \det(AD - BC), \quad (3.4.43)$$

which gives

$$\det(R) = (2i)^{N_f}. \quad (3.4.44)$$

There is a crucial detail that can get lost in the calculation when performing the change of variables. To see this, note how the order of differentials in the integral of the right-hand side of Eq. (3.4.40) is ordered according to [4],

$$d^2 \boldsymbol{\xi} = \int \prod_{j=1}^{N_f} d\xi_j^* d\xi_j, \quad (3.4.45)$$

while we have a different order in transformation matrix in Eq. (3.4.42). To reorder $\xi_1, \dots, \xi_{N_f}, \xi_1^*, \dots, \xi_{N_f}^*$ to $\xi_1^*, \xi_1, \xi_2^*, \xi_2, \dots, \xi_{N_f}^*, \xi_{N_f}$, we pick up $N_f + (N_f - 1) + \dots + 2 + 1 = N_f(N_f + 1)/2$ minus-signs, leading to an additional factor of $(-1)^{N_f(N_f + 1)/2}$, which leads to an overall phase factor s_{N_f} which depends on whether N_f is even or odd. More precisely, if N_f is *even*, i.e. $N_f = 2M$ where $M \in \mathbb{N}$, we have

$$s_{N_f=\text{even}} = (-1)^{\frac{N_f(N_f+1)}{2}} = (-1)^{\frac{N_f}{2}}, \quad (3.4.46)$$

whereas if N_f is *odd*, i.e. $N_f = 2M + 1$, we have

$$s_{N_f=\text{odd}} = (-1)^{\frac{N_f(N_f+1)}{2}} = (-1)^{\frac{N_f+1}{2}}. \quad (3.4.47)$$

Rewriting the argument of the exponential inside the integral of Eq. (3.4.40) by substituting the new set of Grassmann variables gives

$$\begin{aligned} & e^{\sum_{k=1}^{N_f} \frac{-J_k \xi_k^* e^{i\tilde{\alpha}(k)} - \xi_k J_k^* - \frac{1}{2} \xi_k^* \xi_k (1 + e^{i\tilde{\alpha}(k)})}{1 - e^{i\tilde{\alpha}(k)}}} \\ &= e^{\frac{1}{2} \sum_{k=1}^{N_f} \frac{-J_k \xi_{1k} e^{i\tilde{\alpha}(k)} - i J_k \xi_{2k} e^{i\tilde{\alpha}(k)} + J_k^* \xi_{1k} - i J_k^* \xi_{2k} - \frac{i}{2} \xi_{1k} \xi_{2k} (1 + e^{i\tilde{\alpha}(k)})}{1 - e^{i\tilde{\alpha}(k)}}}. \end{aligned} \quad (3.4.48)$$

In addition, we define two new sets of Grassmann variables

$$\eta_k = \frac{\xi_k}{\sqrt{1 - e^{i\tilde{\alpha}(k)}}}, \quad \eta_k^* = \frac{\xi_k^*}{\sqrt{1 - e^{i\tilde{\alpha}(k)}}}, \quad (3.4.49)$$

where the determinant $\det(R')$ of the rotation matrix $\boldsymbol{\eta} = R' \boldsymbol{\xi}$ gives a factor

$$\det(R') = \prod_{k=1}^{N_f} \frac{1}{1 - e^{i\tilde{\alpha}(k)}}. \quad (3.4.50)$$

The characteristic function of Eq. (3.4.10) can be expressed in the new basis as

$$\chi(\boldsymbol{\eta}) = e^{\frac{i}{8} (\boldsymbol{\eta}_1, \boldsymbol{\eta}_2) \text{sqrt}(1 - e^{i\tilde{\alpha}}) \Gamma_m \text{sqrt}(1 - e^{i\tilde{\alpha}}) (\boldsymbol{\eta}_1, \boldsymbol{\eta}_2)^T}, \quad (3.4.51)$$

where we defined the $(2N_f \times 2N_f)$ diagonal matrix

$$\text{sqrt}(1 - e^{i\tilde{\alpha}}) = \mathbf{1}_2 \otimes \begin{pmatrix} \sqrt{1 - e^{i\tilde{\alpha}(1)}} & & \\ & \ddots & \\ & & \sqrt{1 - e^{i\tilde{\alpha}(N_f)}} \end{pmatrix} \quad (3.4.52)$$

and we also define its square

$$(1 - e^{i\tilde{\alpha}}) = \text{sqrt}(1 - e^{i\tilde{\alpha}})^2. \quad (3.4.53)$$

The last equation in Eq (3.4.40) can be rewritten in terms of the new Grassmann variables as

$$\mathcal{A}_{j_1^\dagger \dots j_a^\dagger k_1 \dots k_a} = \mathcal{F}_J e^{\sum_{k=1}^{N_f} \frac{J_k J_k^* e^{i\tilde{\alpha}(k)}}{1 - e^{i\tilde{\alpha}(k)}}} \int d^2 \boldsymbol{\eta} \chi(\boldsymbol{\eta}) e^{\sum_{k=1}^{N_f} \frac{-J_k \eta_k^* e^{i\tilde{\alpha}(k)} - \eta_k J_k^* - \frac{1}{2} \eta_k^* \eta_k (1 + e^{i\tilde{\alpha}(k)})}{\sqrt{1 - e^{i\tilde{\alpha}(k)}}}}. \quad (3.4.54)$$

We can write

$$-\frac{1}{2} \sum_{k=1}^{N_f} \eta_k^* \eta_k \left(1 + e^{i\tilde{\alpha}(k)}\right) = -\frac{i}{4} \sum_{k=1}^{N_f} \eta_{1,k} \eta_{2,k} \left(1 + e^{i\tilde{\alpha}(k)}\right) \quad (3.4.55)$$

and analogous to Eq. (3.4.30), we want to find the matrix B such that

$$-\frac{i}{8} (\boldsymbol{\eta}_1, \boldsymbol{\eta}_2) B \begin{pmatrix} \boldsymbol{\eta}_1 \\ \boldsymbol{\eta}_2 \end{pmatrix} = -\frac{i}{4} \sum_{k=1}^{N_f} \eta_{1,k} \eta_{2,k} \left(1 + e^{i\tilde{\alpha}(k)}\right). \quad (3.4.56)$$

By inspection, B must have the form

$$B = \sigma \otimes \text{diag} \left(1 + e^{i\tilde{\alpha}}\right). \quad (3.4.57)$$

Thus, using the results of Eqs. (3.4.51) and (3.4.56), one can write Eq. (3.4.54) as

$$\begin{aligned} \mathcal{A}_{j_1^\dagger \dots j_a^\dagger k_1 \dots k_a} &= \mathcal{F}_J e^{\sum_{k=1}^{N_f} \frac{J_k J_k^* e^{i\tilde{\alpha}(k)}}{1 - e^{i\tilde{\alpha}(k)}}} \int d^2 \boldsymbol{\eta} e^{\frac{i}{8} (\boldsymbol{\eta}_1, \boldsymbol{\eta}_2) (\text{sqrt}(1 - e^{i\tilde{\alpha}}) \Gamma_m \text{sqrt}(1 - e^{i\tilde{\alpha}}) - \sigma \otimes \text{diag}(1 + e^{i\tilde{\alpha}})) \begin{pmatrix} \boldsymbol{\eta}_1 \\ \boldsymbol{\eta}_2 \end{pmatrix}} \\ &\quad \times e^{\sum_{k=1}^{N_f} \frac{-J_k \eta_k^* e^{i\tilde{\alpha}(k)} - \eta_k J_k^*}{\sqrt{1 - e^{i\tilde{\alpha}(k)}}}} \\ &= \mathcal{F}_J e^{\sum_{k=1}^{N_f} \frac{J_k J_k^* e^{i\tilde{\alpha}(k)}}{1 - e^{i\tilde{\alpha}(k)}}} \int d^2 \boldsymbol{\eta} e^{-\frac{1}{2} (\boldsymbol{\eta}_1, \boldsymbol{\eta}_2) B' \begin{pmatrix} \boldsymbol{\eta}_1 \\ \boldsymbol{\eta}_2 \end{pmatrix}} e^{\sum_{k=1}^{N_f} \frac{-J_k \eta_k^* e^{i\tilde{\alpha}(k)} - \eta_k J_k^*}{\sqrt{1 - e^{i\tilde{\alpha}(k)}}}}, \end{aligned} \quad (3.4.58)$$

where we introduce a couple of definition to cast the above equation into a more recognizable form, namely, we set

$$B' = -\frac{i}{4} \text{sqrt}(1 - e^{i\tilde{\alpha}}) \Gamma_m \text{sqrt}(1 - e^{i\tilde{\alpha}}) + \frac{i}{4} \sigma \otimes \text{diag} \left(1 + e^{i\tilde{\alpha}}\right). \quad (3.4.59)$$

and defined the Grassmann vector $\vec{\boldsymbol{\eta}} = (\boldsymbol{\eta}_1, \boldsymbol{\eta}_2)^T$, as well as

$$\vec{\mathbf{K}} = \frac{1}{2} \begin{pmatrix} \frac{-J_1 e^{i\tilde{\alpha}(1)} + J_1^*}{\sqrt{1 - e^{i\tilde{\alpha}(1)}}} \\ \vdots \\ \frac{-J_{N_f} e^{i\tilde{\alpha}(N_f)} + J_{N_f}^*}{\sqrt{1 - e^{i\tilde{\alpha}(N_f)}}} \\ \frac{iJ_1 e^{i\tilde{\alpha}(1)} + iJ_1^*}{\sqrt{1 - e^{i\tilde{\alpha}(1)}}} \\ \vdots \\ \frac{iJ_{N_f} e^{i\tilde{\alpha}(N_f)} + iJ_{N_f}^*}{\sqrt{1 - e^{i\tilde{\alpha}(N_f)}}} \end{pmatrix}. \quad (3.4.60)$$

With these definition, Eq. (3.4.58) reads

$$\mathcal{A}_{j_1^\dagger \dots j_a^\dagger k_1 \dots k_a} = \frac{S_{N_f}}{2^{N_f}} \mathcal{F}_J e^{\sum_{k=1}^{N_f} \frac{J_k J_k^* e^{i\tilde{\alpha}(k)}}{1 - e^{i\tilde{\alpha}(k)}}} \text{Pf}(\Gamma_F) e^{2i\vec{\mathbf{K}}^T \Gamma_F^{-1} \vec{\mathbf{K}}}, \quad (3.4.61)$$

see Appendix 3.P.3. We now reintroduce the dependence on $\tilde{\alpha}$ in the expressions. We defined the altered CM

$$\Gamma_F^{\tilde{\alpha}} = \text{sqrt}(1 - e^{i\tilde{\alpha}}) \Gamma_m \text{sqrt}(1 - e^{i\tilde{\alpha}}) - \sigma \otimes \text{diag} \left(1 + e^{i\tilde{\alpha}}\right) \quad (3.4.62)$$

and summarized the sign factors of Eqs. (3.4.46) and (3.4.47) to

$$s_{N_f} = \begin{cases} (-1)^{\frac{N_f}{2}}, & \text{if } N_f \text{ is even} \\ (-1)^{\frac{N_f+1}{2}}, & \text{if } N_f \text{ is odd.} \end{cases} \quad (3.4.63)$$

In order to recognize that our result of Eq. (3.4.61) is identical to Eq. (D.8) in Ref. [41], we define the $(2N_f \times 2N_f)$ matrix

$$\Sigma_F^{\tilde{\alpha}} = \begin{pmatrix} \frac{e^{i\tilde{\alpha}(1)}}{1-e^{i\tilde{\alpha}(1)}} & & & \frac{-1}{1-e^{i\tilde{\alpha}(1)}} & & \\ & \ddots & & & \ddots & \\ & & \frac{e^{i\tilde{\alpha}(N_f)}}{1-e^{i\tilde{\alpha}(N_f)}} & & & \frac{-1}{1-e^{i\tilde{\alpha}(N_f)}} \\ \frac{-ie^{i\tilde{\alpha}(1)}}{1-e^{i\tilde{\alpha}(1)}} & & & \frac{-i}{1-e^{i\tilde{\alpha}(1)}} & & \\ & \ddots & & & \ddots & \\ & & \frac{-ie^{i\tilde{\alpha}(N_f)}}{1-e^{i\tilde{\alpha}(N_f)}} & & & \frac{-i}{1-e^{i\tilde{\alpha}(N_f)}} \end{pmatrix} \quad (3.4.64)$$

as well as the $2N_f$ Grassmann column vector

$$\vec{J} = \begin{pmatrix} J_1 \\ \vdots \\ J_{N_f} \\ J_1^* \\ \vdots \\ J_{N_f}^* \end{pmatrix}. \quad (3.4.65)$$

We further define the following quadratic polynomial of Grassmann variables,

$$X^{\tilde{\alpha}} = \sum_{k=1}^{N_f} \frac{J_k^* J_k}{1 - e^{-i\tilde{\alpha}(k)}} - \frac{i}{2} \vec{J}^\dagger \Sigma_F^{\tilde{\alpha} \dagger} \text{sqrt}(1 - e^{i\tilde{\alpha}}) \Gamma_F^{\tilde{\alpha} - 1} \text{sqrt}(1 - e^{i\tilde{\alpha}}) \Sigma_F^{\tilde{\alpha}} \vec{J}. \quad (3.4.66)$$

With these definitions, the implicit formula for computing expressions as given in Eq. (3.3.27) is

$$\mathcal{A}_{j_1^\dagger \dots j_a^\dagger k_1 \dots k_a}^{\tilde{\alpha}} = \mathcal{A}^{\tilde{\alpha}} \mathcal{F}_J e^{X^{\tilde{\alpha}}}, \quad (3.4.67)$$

which is identical to Eq. (D.8) in Ref. [41]. The first term in Eq. (3.4.67),

$$\mathcal{A}^{\tilde{\alpha}} = s_{N_f} \left(\frac{1}{2}\right)^{N_f} \text{Pf}(\Gamma_F^{\tilde{\alpha}}), \quad (3.4.68)$$

is identical to the zeroth-order expression in Eq (3.3.28).

3.4.2 Derivation of explicit expressions

In order to give explicit expressions for the expectation values of the form of Eq. (3.4.67), we define

$$\mathcal{G}^{\tilde{\alpha}} = (\Gamma_m + \Upsilon) \frac{1}{\mathbb{1}_{2N_f} + \frac{1}{2}(1 - e^{i\tilde{\alpha}})(\Upsilon \Gamma_m - \mathbb{1}_{2N_f})}, \quad (3.4.69)$$

that has the property of being anti-symmetric under transposition, $(\mathcal{G}^{\tilde{\alpha}})^T = -\mathcal{G}^{\tilde{\alpha}}$, as shown in Appendix 3.E.3 and which (along with $\mathcal{A}^{\tilde{\alpha}}$ as defined in Eq. (3.4.68)) will play a central role in our work. As detailed in Appendix 3.E, for $p, q = 1, \dots, N_f$, the second-order expectation values are given by

$$\mathcal{A}_{p^\dagger q}^{\tilde{\alpha}} = \frac{i}{4} e^{i\tilde{\alpha}(p)} \mathcal{A}^{\tilde{\alpha}} \mathcal{G}_{\mathbf{p}^\dagger \mathbf{q}}, \quad (3.4.70)$$

$$\mathcal{A}_{p^\dagger q^\dagger}^{\tilde{\alpha}} = \frac{i}{4} e^{i(\tilde{\alpha}(p) + \tilde{\alpha}(q))} \mathcal{A}^{\tilde{\alpha}} \mathcal{G}_{\mathbf{p}^\dagger \mathbf{q}^\dagger}^{\tilde{\alpha}}, \quad (3.4.71)$$

$$\mathcal{A}_{pq}^{\tilde{\alpha}} = \frac{i}{4} \mathcal{A}^{\tilde{\alpha}} \mathcal{G}_{\mathbf{p}^\dagger \mathbf{q}^\dagger}^{\tilde{\alpha}}, \quad (3.4.72)$$

where we made use of the notation introduced in Section 3.2. This representation in terms of $\mathcal{G}^{\tilde{\alpha}}$ is preferable for numerical implementation, as it avoids divergency issues caused by numerical precision error when e.g. $\tilde{\alpha}$ contains only zeros⁶. Note that Eqs. (3.4.68) and (3.4.70) are identical to the expressions given in Ref. [41].

3.4.3 General explicit formula

Since for Grassmann variables, left differentiation is identical to integration and $X^{\tilde{\alpha}}$ in Eq. (3.4.67) is a quadratic polynomial, Wick's theorem applies, meaning that higher(-even)-order expressions can be calculated from second-order expressions as provided by Eqs. (3.4.70)-(3.4.72). For the general case, this leads to

$$\frac{\mathcal{A}_{j_1^\dagger \dots j_a^\dagger k_1 \dots k_a}^{\tilde{\alpha}}}{\mathcal{A}^{\tilde{\alpha}}} = \sum_{i_1 < \dots < i_{2a}} (-1)^{\mathcal{X}} \frac{\mathcal{A}_{i_1 i_2}^{\tilde{\alpha}}}{\mathcal{A}^{\tilde{\alpha}}} \dots \frac{\mathcal{A}_{i_{2a-1} i_{2a}}^{\tilde{\alpha}}}{\mathcal{A}^{\tilde{\alpha}}}, \quad (3.4.73)$$

where \mathcal{X} is identical to the number of crossing contractions for each individual term and the indices i_1, \dots, i_{2a} can be any permutation of the combined indices $j_1^\dagger, \dots, j_a^\dagger$ and k_1, \dots, k_a . For instance, as explicitly shown in Appendix 3.E.4, the fourth-order expression is given by

$$\frac{\mathcal{A}_{p^\dagger q^\dagger r s}^{\tilde{\alpha}}}{\mathcal{A}^{\tilde{\alpha}}} = \frac{\mathcal{A}_{p^\dagger s}^{\tilde{\alpha}}}{\mathcal{A}^{\tilde{\alpha}}} \frac{\mathcal{A}_{q^\dagger r}^{\tilde{\alpha}}}{\mathcal{A}^{\tilde{\alpha}}} - \frac{\mathcal{A}_{p^\dagger r}^{\tilde{\alpha}}}{\mathcal{A}^{\tilde{\alpha}}} \frac{\mathcal{A}_{q^\dagger s}^{\tilde{\alpha}}}{\mathcal{A}^{\tilde{\alpha}}} + \frac{\mathcal{A}_{p^\dagger q^\dagger}^{\tilde{\alpha}}}{\mathcal{A}^{\tilde{\alpha}}} \frac{\mathcal{A}_{r s}^{\tilde{\alpha}}}{\mathcal{A}^{\tilde{\alpha}}}, \quad (3.4.74)$$

and the sign is due to the number of crossing contractions, more specifically

$$\overline{\overline{\square}} \rightarrow \mathcal{X} = 0; \quad \overline{\overline{\square}} \rightarrow \mathcal{X} = 1; \quad \overline{\square} \overline{\square} \rightarrow \mathcal{X} = 0.$$

We want to stress that Eq. (3.4.73) also holds for the special case $U_{\text{FA}} = \mathbb{1}_{2N_f}$, where the formula reduces to the well-known Wick factorization of a product of fermionic raising and lowering operators [3]. However, the indices j_p and k_q appearing in Eq. (3.4.73) do not need to be ordered (in the sense that e.g. $j_p < j_{p+1}$), which makes our formula especially useful.

By giving an explicit formula for the rotated Hamiltonian in Eq. (3.3.19) and providing an explicit for general expectation values in Eq. (3.4.73), we have presented all tools required to efficiently compute the variational energy $E(\xi, \omega)$ of Eq. (3.3.17) on a classical

⁶As we have seen in Appendix 3.E, Eqs. (3.4.70)-(3.4.72) can also be calculated through expressions involving the matrix \mathcal{M} . These expression however contain divisions of zeros for $\tilde{\alpha}(j) = 0$ which are usually not treated well in numerical simulations.

computer. We will provide an explicit expressions for the energy expectation value using the findings of Sections 3.4.2-3.4.3 in the following section. Using an optimization algorithm of one's choice, one can then search for the optimal set of values ξ, ω that minimize the energy. One has to be careful, however, since a generic minimization algorithm must not violate the symmetries of the matrices that describe the variational parameters.

Section 3.5

ENERGY EXPECTATION VALUE

Using Eq. (3.4.74) and inserting Eqs. (3.3.22)-(3.3.25) and Eqs. (3.4.70)-(3.4.72) into Eq. (3.3.29), we obtain a Wick-factorized expression of the energy expectation value w.r.t. the NGS, i.e. Eq. (3.3.29) can be written as

$$E(\xi, \omega) = E_1(\xi, \omega) + E_2(\xi, \omega), \quad (3.5.1)$$

where we defined the one- and two-particle energy expectation values

$$E_1(\xi, \omega) = \frac{i}{4} \sum_{p,q=1}^{N_f} f_{pq} \mathcal{A}^{\alpha_{pq}^{\text{FA}}} \mathcal{G}_{\mathbf{p}^\dagger \mathbf{q}}^{\alpha_{pq}^{\text{FA}}} \quad (3.5.2)$$

$$E_2(\xi, \omega) = -\frac{1}{32} \sum_{p,q,r,s=1}^{N_f} h_{pqrs} e^{i(\omega_{rs} - \omega_{pq})} \mathcal{A}^{\beta_{pqrs}^{\text{FA}}} \left[2\mathcal{G}_{\mathbf{p}^\dagger \mathbf{s}} \mathcal{G}_{\mathbf{q}^\dagger \mathbf{r}} + \mathcal{G}_{\mathbf{p}^\dagger \mathbf{q}^\dagger} \mathcal{G}_{\mathbf{r} \mathbf{s}} \right]^{\beta_{pqrs}^{\text{FA}}} \quad (3.5.3)$$

and we used the notation introduced in Eq. (3.2.15).

Section 3.6

MINIMIZING THE ENERGY THROUGH IMAGINARY-TIME EVOLUTION

The imaginary-time evolution, which was introduced in Section 1.4, is a method where the real time variable t is replaced by an imaginary-time variable (which we will denote as τ), known as a Wick rotation [61]. Given a certain quantum state $|\varphi(\tau)\rangle$, which can be expanded in terms of the eigenfunctions of the system Hamiltonian, its imaginary-time evolution follows

$$|\varphi(\tau)\rangle = \frac{e^{-H\tau} |\varphi(0)\rangle}{\sqrt{\langle\varphi(0)|e^{-2H\tau}|\varphi(0)\rangle}}. \quad (3.6.1)$$

All eigenstate amplitudes will decay exponentially in time, however, all eigenstates possessing a higher energy than the ground state will decay exponentially faster than the ground state. Therefore, in the limit $\tau \rightarrow \infty$, $|\varphi(\tau)\rangle$ will end up in a ground state, provided that the overlap of the latter with the initial state $|\varphi(0)\rangle$ is non-zero [216]. Depending on the initial state, the imaginary-time evolution can get stuck in local minima, therefore one will have to repeat the minimization procedure with a number of distinct initial states $|\varphi(0)\rangle$.

Defining $d_\tau = d/d\tau$, the state given by Eq. (3.6.1) fulfills the differential equation

$$d_\tau |\varphi(\tau)\rangle = -(H - \langle\varphi(\tau)|H|\varphi(\tau)\rangle) |\varphi(\tau)\rangle. \quad (3.6.2)$$

We summarize all variational parameters appearing in the matrices ξ and ω of the NGS Ansatz in Eq. (3.3.3) by flattening the two matrices to a vector $\theta = \theta_1, \theta_2, \dots, \theta_{N_{\text{parms}}}$, where N_{parms} is identical to the number of total variational parameters. By making all variational parameters time-dependent and identifying

$$|\varphi(\tau)\rangle = |\Psi_{\text{NGS}}(\theta(\tau))\rangle, \quad (3.6.3)$$

one can derive from Eq. (3.6.2) a formula which describes how the variational parameters change in each step of the imaginary-time evolution [41],

$$d_\tau \theta_j = \sum_{k=1}^{N_{\text{var}}} (G^{-1})_{jk} \langle\Psi_k|\mathbf{R}_\Psi\rangle. \quad (3.6.4)$$

Eq. (3.6.4) follows from McLachlan's minimal error principal [217] as detailed in Ref. [218]⁷.

⁷A detailed discussion of the theory leading to Eq. (3.6.4) in a geometrical picture can be found in Ref. [218]. McLachlan's principle describes a time-dependent variational principle which is purely routed in geometric ideas [219] and is given by the stationary condition

$$\Im \{ \langle \delta\dot{\Psi} | i\hbar\partial_t - H | \Psi \rangle \} = 0,$$

where $|\delta\dot{\Psi}\rangle$ describes the time-variation of the variational Ansatz in the tangent plane. There does not exist a formulation of imaginary time evolution in terms of an action principle, so McLachlan's minimal error principle is the only projection method that can be used here.

Here, we defined the states

$$|\mathbf{R}_\Psi\rangle = -(H - E) |\Psi_{\text{NGS}}\rangle \quad (3.6.5)$$

$$|\Psi_j\rangle = d_{\theta_j} |\Psi_{\text{NGS}}\rangle, \quad (3.6.6)$$

as well as the Gram matrix

$$G_{jk} = \langle \Psi_j | \Psi_k \rangle \quad (3.6.7)$$

and the variational energy E is given by Eq. (3.3.17). In each time step, $\langle \Psi_k | \mathbf{R}_\Psi \rangle$ and the Gram matrix has to be computed. As pointed out in Ref. [41], in order for the Gram matrix to be invertible, it might be necessary to fix some of the parameters θ_j within a single time slice, in which case the number of parameters in Eq. (3.6.4) decreases.

As derived in Ref. [41] and detailed in Appendix 3.N, the imaginary-time evolution of the NGS Ansatz in Eq. (3.3.3) leads to an equation of motion of the CM through

$$d_\tau \Gamma_m = -\bar{H}_m - \Gamma_m \bar{H}_m \Gamma_m + i [\Gamma_m, O_m], \quad (3.6.8)$$

where \bar{H}_m is the mean-field matrix of the rotated Hamiltonian and O_m is the mean-field matrix of the NGS-dependent Ansatz, whose exact expressions for a general system Hamiltonian as given in Eq. (3.3.1) are presented in Sections 3.6.2 and 3.6.4, respectively. One can solve Eq. (3.6.8) numerically by means of a formal integration for small Δt ,

$$\Gamma_m(t + \Delta t) \approx \exp\left(\left(\frac{1}{2} [\bar{H}_m, \Gamma_m] - iO_m\right) \Delta t\right) \Gamma_m(t) \exp\left(-\left(\frac{1}{2} [\bar{H}_m, \Gamma_m] - iO_m\right) \Delta t\right). \quad (3.6.9)$$

Note that Eq. (3.6.8) reduces to the imaginary-time evolution of the CM as reported in Ref. [63] whenever $U_{\text{FA}} = \mathbf{1}_{2^{N_f}}$, i.e. when only considering variational states restricted to the family of FGS.

While the time evolution of the Gaussian parameters ξ_{ij} is contained in the CM on the left-hand side of Eq. (3.6.8), the time evolution of the NGS parameters ω_{ij} has to be computed at each time step $\Delta\tau$ through Eq. (3.6.4). We note that in order to apply said equation, the tuple-indexed variational parameters ξ_{ij} and ω_{ij} need to be cast to a single index through flattening of the matrix into a vector.

Note that Eq. (3.6.8) in general still describes a beyond mean-field state even if the mean-field term $O_m = 0$, since the NGS variational parameters are also contained in \bar{H}_m . In order to be able to compute Eq. (3.6.8) using Eq. (3.6.4), we derive general expressions for G , \bar{H}_m and O_m in the following subsections.

3.6.1 Computing the Gram matrix

In this section, we will compute the Gram matrix $G_{ij} = \langle \Psi_i | \Psi_j \rangle$ for a variational state Ansatz as given by Eq. (3.3.3), where $|\Psi_j\rangle = d_{\theta_j} |\Psi_{\text{NGS}}\rangle$ is the derivative w.r.t. θ_j either being a Gaussian parameter (and therefore an element of the matrix ξ), or a non-Gaussian parameter (an element of the matrix ω), which leads to three different types of matrix elements—the indices i and j can belong to Gaussian, non-Gaussian or mixed Gaussian and non-Gaussian parameters. We will derive in the following the individual state vectors whose overlaps need to be evaluated using the findings of Section 3.4 in order to be able to construct the Gram matrix.

Note, that one also has to compute expressions of the form $\langle \Psi_k | \mathbf{R}_\Psi \rangle$ in order to compute Eq.(3.6.4). These follow naturally from the results of the following two subsections and the definition of $|\mathbf{R}_\Psi\rangle$ in Eq. (3.6.5).

Derivative of variational state w.r.t. Gaussian parameters

We consider the derivative of the variational state Ansatz w.r.t. a Gaussian parameter ξ_{kl} , which leads to

$$d_{\xi_{kl}} |\Psi_{\text{NGS}}\rangle = U_{\text{FA}} (d_{\xi_{kl}} U_{\text{GS}}) |0\rangle, \quad (3.6.10)$$

since $d_{\xi_{kl}}$ does not affect the unitary U_{FA} . Analogue to strategy that leads to Eq. (3.I.7), it follows that

$$d_{\xi_{kl}} U_{\text{GS}} = \frac{1}{4} A^T (d_{\xi_{kl}} U_m) U_m^T A U_{\text{GS}}. \quad (3.6.11)$$

Thus, Eq. (3.6.11) results in a quadratic polynomial of fermionic operators multiplied from the left to the Gaussian unitary U_{GS} . One can evaluate $d_{\xi_{kl}} U_m = d_{\xi_{kl}} e^{i\xi}$ either using Wilcox's formula, or through diagonalization of the matrix ξ [220]⁸. Let $d_{\xi_1}, \dots, d_{\xi_{2N_f}}$ denote the eigenvalues of ξ , so that [221]

$$e^{i\xi} = X_\xi \begin{pmatrix} e^{id_{\xi_1}} & & \\ & \ddots & \\ & & e^{id_{\xi_{2N_f}}} \end{pmatrix} X_\xi^{-1}, \quad (3.6.12)$$

where X_ξ is an $(2N_f \times 2N_f)$ -dimensional matrix whose j -th column vector belongs to the j -th eigenvalue d_j . Since ξ is a structured matrix (it is anti-symmetric and Hermitian), the matrix derivative has to be altered, in order to ensure that $d_{\xi_{kl}} U_m = -d_{\xi_{lk}} U_m$ [222]. One can compute the derivative through

$$d_{\xi_{kl}} U_m = X_\xi V^{(kl)} X_\xi^{-1}, \quad (3.6.13)$$

where $V^{(kl)}$ is a $(2N_f \times 2N_f)$ -dimensional matrix whose (m, n) -th entry is given by

$$V_{mn}^{(kl)} = \frac{1}{2} \left(X_\xi^{-1} (|k\rangle \langle l| - |l\rangle \langle k|) X_\xi \right)_{mn} \times \begin{cases} \frac{e^{id_{\xi_m}} - e^{id_{\xi_n}}}{d_{\xi_m} - d_{\xi_n}} & , \text{ if } m \neq n \\ ie^{id_{\xi_m}} & , \text{ if } m = n \end{cases} \quad (3.6.14)$$

and $|k\rangle$ ($\langle l|$) is a column (row) vector of length $2N_f$ with a 1 at its k (l)-th entry and 0s everywhere else. In order to arrive at Eq. (3.6.13), we used that $\xi^T = -\xi$ and $d_{\xi_{kl}} e^{i\xi} = -d_{\xi_{lk}} e^{i\xi}$. In the following, we bring Eq. (3.6.11) into a form that allows for the use of the analytical formula of Eq. (3.4.73). We define the matrix

$$\left(\tilde{U}_m^{\xi_{kl}} \right)_{pq} = \frac{1}{4} \sum_{r=1}^{2N_f} (d_{\xi_{kl}} U_m)_{pr} \left(U_m^T \right)_{rq}. \quad (3.6.15)$$

Using the transformation $A = W_m C$ and the definitions from Eqs. (3.2.11)-(3.2.14), Eq. (3.6.11) can be written as

$$d_{\xi_{kl}} U_{\text{GS}} = \sum_{p,q=1}^{N_f} \left[\left(\tilde{U}_m^{\xi_{kl}} \right)_{\mathbf{q}\mathbf{p}} c_p^\dagger c_q^\dagger + \left(\tilde{U}_m^{\xi_{kl}} \right)_{\mathbf{q}^\dagger \mathbf{p}} c_p^\dagger c_q + \left(\tilde{U}_m^{\xi_{kl}} \right)_{\mathbf{q}\mathbf{p}^\dagger} c_p c_q^\dagger + \left(\tilde{U}_m^{\xi_{kl}} \right)_{\mathbf{q}^\dagger \mathbf{p}^\dagger} c_p c_q \right] U_{\text{GS}}. \quad (3.6.16)$$

This form allows for a straightforward application of Eq. (3.4.73) for the evaluation of respective Gram matrix entries.

⁸We are assuming that ξ is in fact diagonalizable, which can always be arranged by fixing certain parameters.

Derivative of variational state with respect to non-Gaussian parameters

We now consider the derivative w.r.t. a non-Gaussian variational parameter ω_{kl} , which leads to

$$d_{\omega_{kl}} |\Psi_{\text{NGS}}\rangle = (d_{\omega_{kl}} U_{\text{FA}}) U_{\text{GS}} |0\rangle. \quad (3.6.17)$$

Using the definition of Eq. (3.3.5), we get

$$d_{\omega_{kl}} U_{\text{FA}} = \frac{i}{2} : n_k n_l : U_{\text{FA}} = \frac{i}{2} U_{\text{FA}} : n_k n_l : . \quad (3.6.18)$$

This is a quartic monomial of fermionic operators multiplied from the left to the non-Gaussian unitary U_{FA} . The form of Eq. (3.6.17) allows for a straightforward application of Eq. (3.4.73) for the evaluation of respective Gram matrix entries.

3.6.2 Mean-field matrix of rotated Hamiltonian

In this section, we present an analytical form for \bar{H}_m from Eq. (3.6.8) for a system of interacting fermions described by the general Hamiltonian H in Eq. (3.3.1). From the derivation of the time-evolution of the CM in Appendix 3.N—which is based on the normal-ordering expansion of Majorana operators as introduced in Appendix 3.H—the mean-field matrix of the rotated Hamiltonian is defined as

$$\left(\bar{H}_m\right)_{ij} = 4 \frac{d \langle \bar{H} \rangle_{\text{GS}}}{d (\Gamma_m)_{ij}}, \quad (3.6.19)$$

which can be evaluated by inserting $E(\xi, \omega) = \langle \bar{H} \rangle_{\text{GS}}$ from Eq. (3.5.1) and taking its derivative w.r.t. the CM, as explicitly shown in Appendix 3.G.2 and 3.G.3. We define⁹

$$\mathcal{L}^{\tilde{\alpha}} = \mathbb{1}_{2N_f} - \frac{1}{2} \mathcal{G}^{\tilde{\alpha}} (1 - e^{i\tilde{\alpha}}) \Upsilon, \quad (3.6.20)$$

$$\mathcal{F}^{\tilde{\alpha}} = -\frac{1}{2} \text{sqrt} (1 - e^{i\tilde{\alpha}}) (\Gamma_F^{\tilde{\alpha}})^{-1} \text{sqrt} (1 - e^{i\tilde{\alpha}}), \quad (3.6.21)$$

where $(1 - e^{i\tilde{\alpha}})$ is the $(2N_f \times 2N_f)$ -diagonal matrix defined in Eq. (3.4.53) and $\Gamma_F^{\tilde{\alpha}}$ is the skew-symmetric $(2N_f \times 2N_f)$ CM defined in Eq. (3.4.62). By definition, $\mathcal{F}^{\tilde{\alpha}}$ is anti-symmetric under transposition, $(\mathcal{F}^{\tilde{\alpha}})^T = -\mathcal{F}^{\tilde{\alpha}}$. Using the results of Appendix 3.L and Eqs. (3.5.2)-(3.5.3), Eq. (3.6.19) can be evaluated through

$$\left(\bar{H}_m\right)_{ij} = \left(\bar{H}_{1,m}\right)_{ij} + \left(\bar{H}_{2,m}\right)_{ij}, \quad (3.6.22)$$

where we defined the one- and two-body terms

$$\left(\bar{H}_{1,m}\right)_{ij} = i \sum_{p,q=1}^{N_f} f_{pq} \mathcal{A}^{\alpha_{pq}^{\text{FA}}} \left[\mathcal{F}_{ij} \mathcal{G}_{\mathbf{p}^\dagger \mathbf{q}} + \frac{1}{2} \left(\mathcal{L}(|i\rangle \langle j| - |j\rangle \langle i|) \mathcal{L}^T \right)_{\mathbf{p}^\dagger \mathbf{q}} \right]^{\alpha_{pq}^{\text{FA}}}, \quad (3.6.23)$$

$$\begin{aligned} \left(\bar{H}_{2,m}\right)_{ij} = & -\frac{1}{16} \sum_{p,q,r,s=1}^{N_f} h_{pqrs} e^{i(\omega_{rs} - \omega_{pq})} \mathcal{A}^{\beta_{pqrs}^{\text{FA}}} \left[4 \mathcal{F}_{ij} \mathcal{G}_{\mathbf{p}^\dagger \mathbf{s}} \mathcal{G}_{\mathbf{q}^\dagger \mathbf{r}} + 2 \mathcal{F}_{ij} \mathcal{G}_{\mathbf{p}^\dagger \mathbf{q}^\dagger} \mathcal{G}_{\mathbf{r}\mathbf{s}} \right. \\ & + 4 \left(\mathcal{L}(|i\rangle \langle j| - |j\rangle \langle i|) \mathcal{L}^T \right)_{\mathbf{p}^\dagger \mathbf{s}} \mathcal{G}_{\mathbf{q}^\dagger \mathbf{r}} + \left. \left(\mathcal{L}(|i\rangle \langle j| - |j\rangle \langle i|) \mathcal{L}^T \right)_{\mathbf{p}^\dagger \mathbf{q}^\dagger} \mathcal{G}_{\mathbf{r}\mathbf{s}} \right. \\ & \left. + \mathcal{G}_{\mathbf{p}^\dagger \mathbf{q}^\dagger} \left(\mathcal{L}(|i\rangle \langle j| - |j\rangle \langle i|) \mathcal{L}^T \right)_{\mathbf{r}\mathbf{s}} \right]^{\beta_{pqrs}^{\text{FA}}}. \end{aligned} \quad (3.6.24)$$

⁹ $\mathcal{F}^{\tilde{\alpha}}$ is a matrix and should not be confused with the operator \mathcal{F}_J in Eq. (3.4.25).

We made use of the notation introduced in Eq. (3.2.15) in both equations. The terms containing derivatives of \mathcal{G} w.r.t. the CM can be rewritten as the (i, j) -th matrix element of the following matrices,

$$\left(\mathcal{L}(|i\rangle\langle j| - |j\rangle\langle i|)\mathcal{L}^T\right)_{\mathbf{p}^\dagger\mathbf{s}} = \langle i|\mathcal{L}^T\left(\begin{pmatrix} \mathbf{1}_s \\ \mathbf{i}_s \end{pmatrix}(\mathbf{1}_p^T, -\mathbf{i}_p^T) - \begin{pmatrix} \mathbf{1}_p \\ -\mathbf{i}_p \end{pmatrix}(\mathbf{1}_s^T, \mathbf{i}_s^T)\right)\mathcal{L}|j\rangle, \quad (3.6.25)$$

$$\left(\mathcal{L}(|i\rangle\langle j| - |j\rangle\langle i|)\mathcal{L}^T\right)_{\mathbf{p}^\dagger\mathbf{q}^\dagger} = \langle i|\mathcal{L}^T\left(\begin{pmatrix} \mathbf{1}_q \\ -\mathbf{i}_q \end{pmatrix}(\mathbf{1}_p^T, -\mathbf{i}_p^T) - \begin{pmatrix} \mathbf{1}_p \\ -\mathbf{i}_p \end{pmatrix}(\mathbf{1}_q^T, -\mathbf{i}_q^T)\right)\mathcal{L}|j\rangle, \quad (3.6.26)$$

$$\left(\mathcal{L}(|i\rangle\langle j| - |j\rangle\langle i|)\mathcal{L}^T\right)_{\mathbf{rs}} = \langle i|\mathcal{L}^T\left(\begin{pmatrix} \mathbf{1}_s \\ \mathbf{i}_s \end{pmatrix}(\mathbf{1}_r^T, \mathbf{i}_r^T) - \begin{pmatrix} \mathbf{1}_s \\ \mathbf{i}_s \end{pmatrix}(\mathbf{1}_r^T, \mathbf{i}_r^T)\right)\mathcal{L}|j\rangle. \quad (3.6.27)$$

Eqs. (3.6.25)-(3.6.27) allow for a matrix representation of the mean-field matrix \bar{H}_m and is therefore well-suited for numerical implementation. The rotated mean-field expression is anti-symmetric under transposition,

$$\left(\bar{H}_m\right)^T = -\bar{H}_m. \quad (3.6.28)$$

3.6.3 The role of the chemical potential

In numerical simulations, small numerical errors can cause the particle number to jump by even integers (parity is conserved in the Majorana picture, so one will not encounter odd integer jumps). In order to stabilize the numerical simulations one introduces a chemical potential

$$H_\mu = -\mu \sum_{i=1}^{N_f} c_i^\dagger c_i, \quad (3.6.29)$$

with $\mu \in \mathbb{R}$ being a free parameter to tune the chemical potential. The generalized canonical transformation through U_{FA} does not affect the chemical potential, $\bar{H}_\mu = H_\mu$. As we show in Appendix 3.A, the expectation value of the chemical potential (given in Majorana representation by Eq. (3.A.20)) w.r.t. the FGS can be written as

$$\langle H_\mu \rangle_{\text{GS}} = -\frac{\mu}{2} \left(N_f - \frac{1}{2} \text{tr}(\Upsilon \Gamma_m) \right), \quad (3.6.30)$$

i.e. it only depends upon the Gaussian parameters ξ , or equivalently its CM Γ_m . The contributions to the mean-field matrix of the rotated Hamiltonian in Eq. (3.6.22) is thus given by

$$\left(\bar{H}_{\mu,m}\right)_{ij} = -\mu \Upsilon_{ij} \quad (3.6.31)$$

and Eq. (3.6.22) will have to be adjusted to

$$\left(\bar{H}_m\right)_{ij} = \left(\bar{H}_{1,m}\right)_{ij} + \left(\bar{H}_{2,m}\right)_{ij} + \left(\bar{H}_{\mu,m}\right)_{ij}, \quad (3.6.32)$$

where the first and second term in Eq. (3.6.32) are given by Eqs. (3.6.23)-(3.6.24).

3.6.4 Mean-field matrix of NGS Ansatz

As derived in Appendix 3.J—and using the notation introduced in Eq. (3.3.9)—the mean-field matrix O_m of the NGS-dependent Ansatz is given by

$$O_m = \frac{i}{4} \left(\begin{pmatrix} \mathbb{0}_{N_f} & \text{diag}((d_\tau \omega + d_\tau \bar{\omega})g) \\ -\text{diag}((d_\tau \omega + d_\tau \bar{\omega})g) & \mathbb{0}_{N_f} \end{pmatrix} + (d_\tau \Omega + d_\tau \bar{\Omega}) \odot \begin{pmatrix} -\mathcal{G}_{22} & \mathcal{G}_{21} \\ \mathcal{G}_{12} & -\mathcal{G}_{11} \end{pmatrix} \right), \quad (3.6.33)$$

where

$$d_\tau \bar{\omega} = (d_\tau \omega)^T, \quad (3.6.34)$$

and $\mathbb{0}_{N_f}$ is the $(N_f \times N_f)$ -matrix containing only zeros. In addition,

$$g = \left((\Gamma_m)_{1,N_f+1} + 1, \dots, (\Gamma_m)_{N_f,2N_f} + 1 \right)^T, \quad (3.6.35)$$

and $\text{diag}((d_\tau \omega)g)$ is a $(N_f \times N_f)$ -diagonal matrix, where the k -th diagonal entry is given by

$$\left(\text{diag}((d_\tau \omega)g) \right)_{kk} = \sum_{l=1}^{N_f} d_\tau \omega_{kl} \left((\Gamma_m)_{l,N_f+l} + 1 \right). \quad (3.6.36)$$

The matrix O_m is (just like \bar{H}_m in Eq. (3.6.28)) anti-symmetric under transposition,

$$(O_m)^T = -O_m. \quad (3.6.37)$$

We validated the expression found for Eq. (3.7.11) in Appendix 3.M numerically by comparing it with Eq. (3.J.8).

3.6.5 Time evolution of the energy

As we stated at the end of Section 3.4, having a functional of the energy depending on ω and ξ allows any optimization method to be used to find the energy minimum and thus the set of variational parameters which give an approximate ground state to the system Hamiltonian. In the following, we will compute the evolution of the energy as a function of the variational parameters and their respective time-derivatives. This will help us in deriving an analytical proof that both, the optimization method presented in Ref. [41] and the alternative method we suggest guarantee a monotonic decrease of the energy with time and builds the foundation of Section 3.7.

Since the energy depends on two sets of parameters, namely ξ (or equivalently Γ_m) and ω , the time-evolution of the energy expectation value is given by

$$d_\tau E(\xi, \omega) = \frac{1}{4} \sum_{i,j=1}^{2N_f} \left(\bar{H}_m \right)_{ij} d_\tau (\Gamma_m)_{ij} + \sum_{k,l=1}^{N_f} \frac{dE}{d\omega_{kl}} d_\tau \omega_{kl}. \quad (3.6.38)$$

As shown in Ref. [41], the energy decreases monotonically with time if the variational parameters follow an imaginary-time evolution, see Appendix 3.O.1. As detailed in Appendix 3.O.2, the above equation can be written as

$$d_\tau E(\xi, \omega) = \frac{1}{8} \text{tr} \left(\left([\bar{H}_m, \Gamma_m] - iO_m \right)^2 \right) + \frac{1}{8} \text{tr} \left(O_m^2 \right) + \sum_{k,l=1}^{N_f} \frac{dE}{d\omega_{kl}} d_\tau \omega_{kl}. \quad (3.6.39)$$

Since $iO_m, \Gamma_m, \bar{H}_m$ are real-valued anti-symmetric matrices, the first term on the right-hand side of Eq. (3.6.39) contains the trace of the square of a skew-symmetric matrix, which is a symmetric negative semi-definite matrix and thus possessing only non-positive eigenvalues. Since the trace of a square matrix is identical to the sum of its eigenvalues, we have

$$\text{tr} \left(\left([\bar{H}_m, \Gamma_m] - iO_m \right)^2 \right) \leq 0. \quad (3.6.40)$$

However, the last two terms in Eq. (3.6.39) can become positive (if one does not choose $d_\tau \omega$ according to Eq (3.6.4)), which means that the energy will in general not be monotonically decreasing in time.

When considering only FGS, i.e. the variational state is restricted to $|\Psi_{\text{GS}}\rangle$ by setting $U_{\text{FA}} = \mathbb{1}_{2^{N_f}}$, the energy is guaranteed to be monotonically decreasing within the respective time step $\Delta\tau$. In that case the VM reduces to generalized Hartree-Fock theory [63] and the equations of motions simplify to

$$d_\tau \Gamma_m = -\bar{H}_m - \Gamma_m \bar{H}_m \Gamma_m, \quad (3.6.41)$$

which is identical to the equations of motion presented in Ref. [63] and in chapter 2. In the following, we will introduce an Ansatz to reach the minimum of the energy functional which is based on a combination of imaginary-time evolution and gradient descent and presents an alternative to Eq. (3.6.4) for choosing the NGS parameters (in particular, it avoids the computation of eight-order expectation value expressions as described in Section 3.6.1), while guaranteeing a monotonic decreases of the energy during the time evolution.

Section 3.7

MINIMIZING THE ENERGY THROUGH A HYBRID IMAGINARY-TIME- GRADIENT-DESCENT ALGORITHM

As we have seen in the previous section, one can minimize the energy expectation value by letting the variational parameters $(d_\tau \Gamma_m)_{ij}$ belonging to the FGS evolve following Eq. (3.6.8) and the variational parameters describing the NGS part by computing $d_\tau \omega_{ij}$ through Eq. (3.6.4). In said approach both sets of variational parameters are chosen following an imaginary-time evolution, where the energy is guaranteed to be monotonically decreasing in time, see Appendix 3.O.1.

However, there are many ways to choose the variational parameters. As we have stated before, any algorithm that minimizes the energy function on Eq. (3.5.1) while preserving the structure of the CM (e.g. $\Gamma_m^2 = -\mathbf{1}_{2N_f}$ at all times) and ω can be employed. In what follows, we present a hybrid Imaginary-Time-Gradient-Descent (HITGD) algorithm which evolves the CM following an imaginary-time evolution (i.e. Eq. (3.6.8)), while the time evolution of the NGS variational parameters $d_\tau \omega$ is chosen based on gradient descent. Since we mix imaginary-time evolution with gradient descent, we will show that the energy will in general no longer be monotonically decreasing. We overcome this issue by carefully designing the gradient descent-motivated Ansatz for $d_\tau \omega$.

This section is structured as follows. In Section 3.7.1 we motivate the gradient descent method. Section 3.7.2 gives an explicit formula for the time evolution of the NGS parameters and Section 3.7.3 includes the gradient of the energy function w.r.t. the NGS parameters, a key ingredient for our method.

3.7.1 Gradient descent

Given a real-valued differentiable function $f(\theta)$, where θ contains a set of time-dependent parameters $\theta_1, \theta_2, \dots$, one can compute the time derivative through

$$d_t f(\theta) = \sum_j \frac{df(\theta)}{d\theta_j} \frac{d\theta_j}{dt}. \quad (3.7.1)$$

If we want the function $f(\theta)$ to be monotonically decreasing in time, we can choose

$$\frac{d\theta_j}{dt} = -\frac{df(\theta)}{d\theta_j}, \quad (3.7.2)$$

which will result in $d_t f(\theta) \leq 0$ as desired. This is known as gradient descent.

3.7.2 The HITGD Ansatz

By analogy one could assume that we simply set

$$d_\tau \omega_{ij} = -\frac{1}{\lambda_{ij}} \frac{dE}{d\omega_{ij}} \quad (3.7.3)$$

for a gradient descent based method (where $\lambda_{ij} \in \mathbb{R}$), but this rather naïve choice can cause the energy to increase during the evolution. As detailed in Appendix 3.K we will present an Ansatz for the time evolution of the NGS variational parameters that will resolve this issue.

We will make use of the block-form representation of a special case of the matrix $\mathcal{G}^{\tilde{\alpha}}$ defined in Eq. (3.4.69),

$$\mathcal{G}^0 = \Gamma_m + \Upsilon = \begin{pmatrix} \mathcal{G}_{11}^0 & \mathcal{G}_{12}^0 \\ \mathcal{G}_{21}^0 & \mathcal{G}_{22}^0 \end{pmatrix}, \quad (3.7.4)$$

where the 0 indicates that all phase factors $\tilde{\alpha}(j)$ are identical to zero and we have $(\mathcal{G}_{12}^0)^T = -\mathcal{G}_{21}^0$. We denote with M^+ the Moore-Penrose inverse of a $(m \times n)$ -matrix M , which satisfies $M^+MM^+ = M^+$ and $MM^+M = M$. We define the $(N_f \times N_f \times N_f \times N_f)$ -tensor B with elements

$$B_{jklm} = \frac{1}{4} \left(A_{kl}^{(m)} \delta_{m,j} + A_{jl}^{(k)} \delta_{m,k} + A_{km}^{(l)} \delta_{l,j} + A_{jm}^{(k)} \delta_{l,k} \right), \quad (3.7.5)$$

where

$$A_{kl}^{(j)} = \frac{1}{4} \left(gg^T + D_j \right)_{kl}. \quad (3.7.6)$$

and D_j are diagonal matrices defined as

$$D_j = \begin{pmatrix} D_{j,1} & & & \\ & D_{j,2} & & \\ & & \ddots & \\ & & & D_{j,N_f} \end{pmatrix} \quad (3.7.7)$$

with diagonal entries

$$D_{j,k} = \frac{1}{2} \left((\mathcal{G}_{11}^0)_{jk}^2 + (\mathcal{G}_{12}^0)_{jk}^2 + (\mathcal{G}_{21}^0)_{jk}^2 + (\mathcal{G}_{22}^0)_{jk}^2 \right). \quad (3.7.8)$$

Note that gg^T in Eq. (3.7.6) is a matrix—it is the dyadic product of the column vector g defined in Eq. (3.6.35). We can reshape the tensor B into a $(N_f^2 \times N_f^2)$ -matrix and similarly reshape the $(N_f \times N_f)$ -matrix ω into a column vector of length N_f^2 . This procedure of reshaping is known in applied mathematics as matricization, flattening or unfolding of a tensor. Similarly, we can reshape the symmetric $(N_f \times N_f)$ -matrix

$$\frac{dE}{d\omega} = \begin{pmatrix} \frac{dE}{d\omega_{11}} & \frac{dE}{d\omega_{12}} & \cdots & \frac{dE}{d\omega_{1N_f}} \\ \frac{dE}{d\omega_{21}} & \frac{dE}{d\omega_{22}} & \cdots & \frac{dE}{d\omega_{2N_f}} \\ \vdots & \vdots & \cdots & \vdots \\ \frac{dE}{d\omega_{N_f 1}} & \frac{dE}{d\omega_{N_f 2}} & \cdots & \frac{dE}{d\omega_{N_f N_f}} \end{pmatrix} \quad (3.7.9)$$

into a column vector of length N_f^2 . Within this representation, as explicitly shown in Appendix 3.K, the choice

$$d_\tau \omega = -4B^+ \frac{dE}{d\omega}, \quad (3.7.10)$$

will guarantee $d_\tau E \leq 0$ for the HITGD approach. The most important point of this section is to understand that the main difference between the purely imaginary-time evolution of the variational parameters as proposed in Ref. [41] and the HITGD method is the replacement of the updating rule for the NGS parameters at each time step $\Delta\tau$, i.e. Eq. (3.6.4) is replaced by Eq. (3.7.10).

At the end of Appendix 3.K we present the naïve Ansatz of Eq. (3.7.3), which avoids the computation of the Moore-Penrose inverse of the tensor B at the cost of the energy no longer being required to be monotonic decreasing in time.

3.7.3 Computing the energy gradient w.r.t. the NGS variational parameters

Central to the HITGD approach is the ability to evaluate the energy gradient in Eq. (3.7.10). This can be achieved—using the results of Section 3.6.1—by computing the commutator of the rotated Hamiltonian with the normal-ordered product of two number operators,

$$\begin{aligned} \frac{dE}{d\omega_{ij}} &= \frac{d}{d\omega_{ij}} \langle U_{\text{FA}}^\dagger H U_{\text{FA}} \rangle_{\text{GS}} \\ &= -\frac{i}{2} \langle [\bar{H}, c_i^\dagger c_j^\dagger c_i c_j] \rangle_{\text{GS}}. \end{aligned} \quad (3.7.11)$$

As detailed in Appendix 3.M and using the definition given in Eq. (3.3.19), the energy gradient in Eq. (3.7.11) can be computed through

$$\begin{aligned} \frac{dE}{d\omega_{ij}} &= \Im \left\{ \sum_{p=1}^{N_f} f_{ip}^{\text{FA}} \left\langle e^{i \sum_k \alpha_{ip}^{\text{FA}}(k) n_k^f} c_i^\dagger c_j^\dagger c_j c_p \right\rangle_{\text{GS}} \right\} + \Im \{(i \leftrightarrow j)\} \\ &\quad + 2\Im \left\{ \sum_{p=1}^{N_f} \sum_{q < r} h_{ipqr}^{\text{FA}} \left\langle e^{i \sum_k \beta_{ipqr}^{\text{FA}}(k) n_k^f} c_j^\dagger c_i^\dagger c_p^\dagger c_j c_q c_r \right\rangle_{\text{GS}} \right\} + 2\Im \{(i \leftrightarrow j)\} \\ &\quad + 2\Im \left\{ \sum_{p < q} h_{ijpq}^{\text{FA}} \left\langle e^{i \sum_k \beta_{ijpq}^{\text{FA}}(k) n_k^f} c_i^\dagger c_j^\dagger c_p c_q \right\rangle_{\text{GS}} \right\}, \end{aligned} \quad (3.7.12)$$

where $\Im\{x\} = (x - x^*)/(2i)$ denotes the imaginary part of a complex number x and the symbol $(i \leftrightarrow j)$ is a placeholder for an expression similar to the preceding one where the roles of the indices i and j are interchanged. Note that Eq. (3.7.12) always produces a real number, in accordance with the requirement that the energy is real-valued. We validated the expression found for Eq. (3.7.11) in Appendix 3.M using OPENFERMION [223] and Eq. (3.7.12) by comparing it with the finite difference of the expectation value E in Eq. (3.5.1).

Note that the added chemical potential H_μ discussed in Section 3.6.3 does not affect the time evolution of the NGS parameters when using the HITGD method, since $\langle [\bar{H}_\mu, c_i^\dagger c_j^\dagger c_i c_j] \rangle_{\text{GS}} = 0$, which implies that Eq. (3.7.9) is unaffected by an added chemical potential.

Section 3.8

EFFICIENT IMPLEMENTATION OF THE NON-GAUSSIAN STATE ANSATZ ON A QUANTUM COMPUTER

Once the classical computation has produced a final set of parameters ξ, ω , one can implement the corresponding quantum state (described by Eq. (3.3.3)) on a quantum computer, serving as an initial state for quantum algorithms. A short sketch on how this implementation can be realized is given in Fig. 3.8.1, the objective of this section is to detail how the black box circuits elements for $U_{\text{GS}}(\xi)$ and $U_{\text{FA}}(\omega)$ can be realized in experiment.

In order to generate the FGS $|\Psi_{\text{GS}}\rangle = U_{\text{GS}}|0\rangle$, one can rotate the system Hamiltonian of Eq. (3.3.1) into the eigenbasis of the CM through orbital rotations, where the initial state is a single Slater determinant

$$|\Psi_{\text{GS}}\rangle = \tilde{c}_{k_1}^\dagger \cdots \tilde{c}_{k_{N_f}}^\dagger |0\rangle, \quad (3.8.1)$$

where $\tilde{c}_{k_j}^\dagger$ are the rotated fermionic operators. Alternatively, one can rotate the state into the original basis of the system Hamiltonian through Givens rotations. The second approach is advantageous when much of the sparsity of the original system Hamiltonian would be lost and a Gaussian state preparation through $\binom{N_f}{2}$ Givens rotations would be advantageous, whereas the first approach comes at the cost of a larger number of measurements, since the number of terms in the system Hamiltonian generally increases. Construction methods to implement FGS on a quantum computer are well-known [166–169, 172] and open source software is already available to transform U_{GS} into quantum gates [223].

In order to realize the non-Gaussian unitary $U_{\text{FA}}(\omega)$ on a quantum computer, one has to first map the fermionic operators onto the spin qubit operators, for instance by means of the Jordan-Wigner transformation [18, 149]. This leads to

$$U_{\text{FA}}(\omega) = e^{\frac{i}{8} \sum_{j \neq k} \omega_{jk}} \prod_j e^{-\frac{i}{4} \sum_k \omega_{jk} \sigma_j^z} \prod_{j < k} e^{\frac{i}{4} \omega_{jk} \sigma_j^z \sigma_k^z}, \quad (3.8.2)$$

where we defined the Pauli- Z spin-operator

$$\sigma^z = \begin{pmatrix} 1 & 0 \\ 0 & -1 \end{pmatrix}. \quad (3.8.3)$$

The first two terms in Eq. (3.8.2) give rise to a global phase and are thus irrelevant when evaluating the expectation value of the energy. The other terms give rise to simple commuting single- and two qubit gates, with the latter being identical to Ising-type interactions, native to Josephson flux qubits [224, 225] or inductively coupled Josephson charge qubits [226].

In Fig. 3.8.2, we give an explicit quantum circuit that is equivalent to U_{FA} and requires N_f single-qubit σ^z -rotations and $N_f(N_f + 1)/2$ two-qubit Ising-type $\sigma_j^z \sigma_k^z$ interactions,

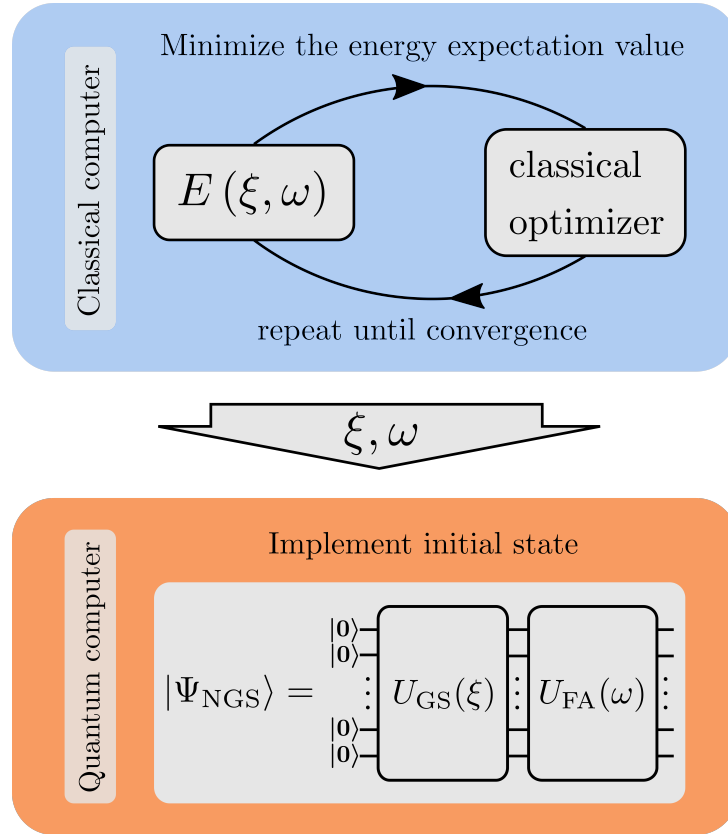


Figure 3.8.1: This sketch captures the strategy on how to use the VM method—e.g. through a purely imaginary-time evolution of the variational parameters, or using the HITGD method—to find a final set of variational parameters ξ, ω (blue box) which are then used to initialize the quantum computer in the initial state $|\Psi_{\text{init}}\rangle = |\Psi_{\text{NGS}}\rangle$ (red box).

whose rotation angles are given by the arguments in Eq. (3.8.2). The number of gates in this circuit are fixed, however, one can make use of the fact that all gates appearing in Eq. (3.8.2) commute with each other and follow various strategies to lower the circuit depth, which will scale as $O(\log(N_f))$. An additional SWAP-gate overhead appears, if the qubit layout does not allow for all-to-all connectivity, which is the case for most quantum architectures today. We will leave the task of optimizing the above quantum circuit to appropriate software [223, 227].

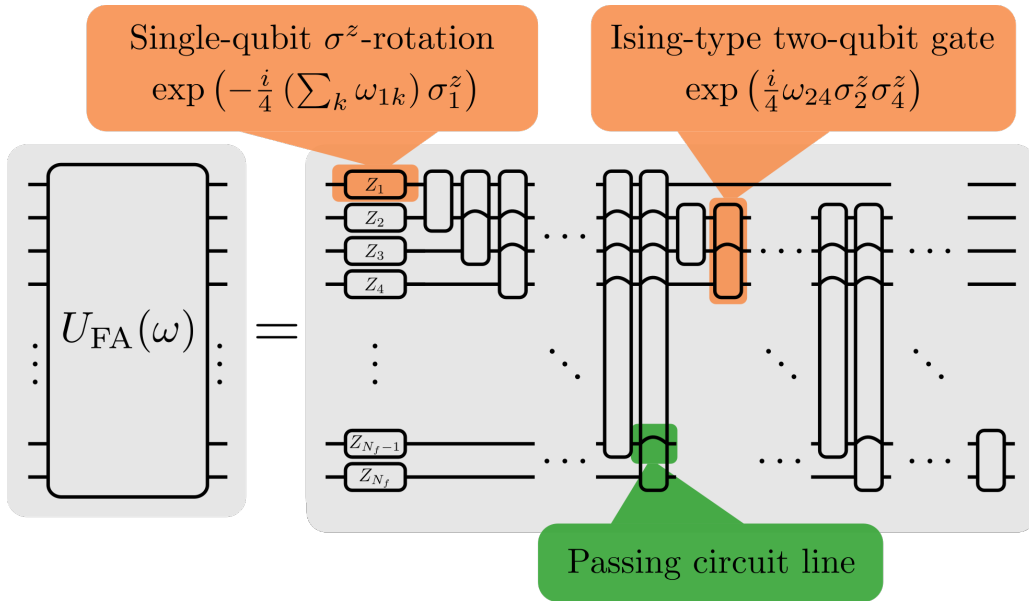


Figure 3.8.2: Explicit quantum circuit which produces the non-Gaussian unitary operator U_{GS} using N_f single-qubit σ^z -rotations and $N_f(N_f+1)/2$ Ising-type two-qubit interactions (both types of gates highlighted in red). The rotation angles are determined by the final set of variational parameters ω_{ij} obtained from a classical minimization of the energy functional in Eq. (3.3.29). Qubits which are not affected by an individual two-qubit gate are indicated by bended circuit lines (a single passing circuit line is highlighted in green). The overall phase factor present in Eq. (3.8.2) is omitted.

Section 3.9

SUMMARY AND OUTLOOK

In this chapter, we have presented an in-depth discussion of a newly proposed method to study strongly interacting fermionic systems. This method offers a way to approximate the ground state of a system of interacting fermions going beyond prior mean-field approaches by introducing non-factorizable correlations, yet remaining efficient to compute on classical processors. We contribute to the original work of Ref. [41] by adding extensive analytical formulas for all required expressions and by presenting an alternative algorithm for updating the non-Gaussian variational parameters and show analytically, that the energy monotonically decreases during the time evolution. We propose that the VM is an ideal candidate for implementing an initial/reference state on a quantum computer and provide an explicit quantum circuit. We have not assumed any prior knowledge about underlying properties of the studied fermionic systems, which makes our work applicable to all fermionic systems.

While we have numerically validated our analytical expressions—i.a. thorough OPEN-FERMION [223]—future work will require numerical experiments in order to study if and by how much the VM will improve the ground state over a generalized Hartree-Fock approach for a variety of fermionic systems. Within such studies, one should compare the minimization method which purely relies on imaginary time evolution with the HITGD method and study their respective performances. In addition, the VM’s performance should be benchmarked to other numerical methods (e.g. Monte-Carlo simulations).

A closing remark: The VM has been suggested to be an ideal candidate for studying fractional quantum Hall systems, which we encountered in Section 2. So far, all methods (not based on adiabatic state creation) used to create better initial states that go beyond mean-field require an error-corrected quantum computer due to the large coherence time requirements of the respective state-generation algorithm. The simplicity of the quantum circuit that initializes the NGS was the main motivation that lead us to study this method in the first place.

APPENDIX

Section 3.A

SYSTEM HAMILTONIAN AND CHEMICAL POTENTIAL

We consider a system containing N_{el} interacting electrons in a system of N_f fermionic modes. The system Hamiltonian is of the general form [2]

$$H = \sum_{p,q=1}^{N_f} f_{pq} c_p^\dagger c_q + \frac{1}{2} \sum_{p,q,r,s=1}^{N_f} h_{pqrs} c_p^\dagger c_q^\dagger c_r c_s, \quad (3.A.1)$$

where f_{pq} are one-body integrals and h_{pqrs} are two-body integrals, given by

$$f_{pq} = \int d\mathbf{r} \psi_p^* T(\mathbf{r}) \psi_q(\mathbf{r}) \quad (3.A.2)$$

$$h_{pqrs} = \frac{1}{4} (v_{pqrs} - v_{qprs} + v_{qpsr} - v_{pqsr}) \quad (3.A.3)$$

$$v_{pqrs} = \iint d\mathbf{r}_1 d\mathbf{r}_2 \psi_p^*(\mathbf{r}_1) \psi_q^*(\mathbf{r}_2) V(r_{12}) \psi_s(\mathbf{r}_1) \psi_r(\mathbf{r}_2), \quad (3.A.4)$$

where $T(\mathbf{r})$ describes the single-particle Hamiltonian terms (such as the kinetic energy), $V(r_{12}) = 1/(|\mathbf{r}_1 - \mathbf{r}_2|)$ describes the Coulomb interaction of two particles located at \mathbf{r}_1 and \mathbf{r}_2 and $\psi_p(\mathbf{r})$ is the p -th single particle basis. Due to the fermionic statistics, one can always bring the Hamiltonian in a form where the two-body integrals possess the symmetry $h_{pqrs} = -h_{qprs} = -h_{pqsr} = h_{qpsr}$.

One can write the Hamiltonian in the Majorana representation via the transformation $C = W_m^{-1} A$. We can represent the one-particle terms in the Majorana basis as follows,

$$\sum_{p,q=1}^{N_f} f_{pq} c_p^\dagger c_q = \sum_{p,q=1}^{N_f} \sum_{i,j=1}^{2N_f} f_{pq} (W_m^{-1})_{N+p,i} (W_m^{-1})_{q,j} A_i A_j = \sum_{i,j=1}^{2N_f} f_{ij}^{(m)} A_i A_j, \quad (3.A.5)$$

where the indices $i, j \in \{1, \dots, 2N_f\}$ and $p, q \in \{1, \dots, N_f\}$ and

$$f_{ij}^{(m)} = \sum_{p,q=1}^{N_f} (W_m^{-1})_{i,N+p}^T f_{pq} (W_m^{-1})_{q,j}. \quad (3.A.6)$$

Similarly, we obtain

$$\sum_{p,q,r,s=1}^{N_f} h_{pqrs} c_p^\dagger c_q^\dagger c_r c_s = \sum_{i,j,k,l=1}^{2N_f} h_{ijkl}^{(m)} A_i A_j A_k A_l, \quad (3.A.7)$$

where we defined

$$h_{ijkl}^{(m)} = \sum_{p,q,r,s=1}^{N_f} (W_m^{-1})_{i,N+p}^T (W_m^{-1})_{j,N+q}^T h_{pqrs} (W_m^{-1})_{r,k} (W_m^{-1})_{s,l}. \quad (3.A.8)$$

Therefore, the Hamiltonian in Eq. (3.3.1) can be written as

$$H = \sum_{i,j=1}^{2N_f} f_{ij}^{(m)} A_i A_j + \frac{1}{2} \sum_{i,j,k,l=1}^{2N_f} h_{ijkl}^{(m)} A_i A_j A_k A_l. \quad (3.A.9)$$

We define the following set of coefficients that will help us simplify the expression of the Hamiltonian when rewriting it as an ordered sum,

$$\tilde{f}_{ij}^{(m)} = f_{ij}^{(m)} - f_{ji}^{(m)}. \quad (3.A.10)$$

We can write the first sum in Eq. (3.A.9) as

$$\sum_{i,j=1}^{2N_f} f_{ij}^{(m)} A_i A_j = \sum_{i<j}^{2N_f} \tilde{f}_{ij}^{(m)} A_i A_j + \sum_{i=1}^{2N_f} f_{ii}^{(m)} \mathbb{1}_{2N_f}. \quad (3.A.11)$$

and the second sum as

$$\frac{1}{2} \sum_{i,j,k,l=1}^{2N_f} h_{ijkl}^{(m)} A_i A_j A_k A_l = \frac{1}{2} \sum_{i<j<k<l}^{2N_f} \tilde{h}_{ijkl} A_i A_j A_k A_l + \sum_{i<j} \bar{g}_{ij} A_i A_j + \bar{E}_c \mathbb{1}_{2N_f}, \quad (3.A.12)$$

where we defined

$$\tilde{h}_{ijkl} = -\bar{h}_{ikjl} + \bar{h}_{iljk} + \bar{h}_{ijkl} + \bar{h}_{jkl} - \bar{h}_{jlik} + \bar{h}_{klij}, \quad (3.A.13)$$

$$\bar{h}_{ijkl} = h_{ijkl}^{(m)} - h_{jikl}^{(m)} - h_{ijlk}^{(m)} + h_{jilk}^{(m)}, \quad (3.A.14)$$

$$\bar{g}_{ij} = \frac{1}{2} \sum_{k=1}^{2N_f} (\bar{h}_{ikkj} - \bar{h}_{jkki} + h_{ijkk}^{(m)} - h_{jikk}^{(m)}), \quad (3.A.15)$$

$$\tilde{g}_{ij} = \bar{g}_{ij} + \tilde{f}_{ij}^{(m)}, \quad (3.A.16)$$

$$\bar{E}_c = \frac{1}{2} \left(\sum_{i,j=1}^{2N_f} h_{iijj}^{(m)} + \sum_{i<j} \bar{h}_{jii} \right), \quad (3.A.17)$$

$$\tilde{E}_c = \bar{E}_c + \sum_{i=1}^{2N_f} f_{ii}^{(m)}. \quad (3.A.18)$$

Therefore, the Hamiltonian (3.A.9) can be written as an ordered sum of Majorana operators via

$$H = \tilde{E}_c \mathbb{1}_{N_f} + \sum_{i<j}^{2N_f} \tilde{g}_{ij} A_i A_j + \frac{1}{2} \sum_{i<j<k<l} \tilde{h}_{ijkl} A_i A_j A_k A_l. \quad (3.A.19)$$

Such a representation of ordered indices is needed when one wants to use Wick's theorem as for instance in Ref. [3]. For applying the VM, however, such an index ordering is not needed when evaluating the emerging expectation values.

Since Majorana operators only conserve parity, it will be crucial to include a chemical potential to enforce particle conservation in numerical studies, since small numerical errors can cause the simulation to violate particle conservation [63]. We define the number operator $\hat{N} = \sum_{j=1}^{N_f} c_j^\dagger c_j$, so that the chemical potential is given by

$$H_\mu = -\mu\hat{N} = -\mu\left(\frac{N_f}{2} + \frac{i}{2}\sum_{j<k}\Upsilon_{jk}A_jA_k\right) \quad (3.A.20)$$

in the Dirac fermion basis and Majorana basis, respectively.

Section 3.B

WICK'S THEOREM

We define normal ordering of a product of fermionic creation and annihilation operators C_j as the product where all annihilation operators are moved (anti-commuted) to the right-hand side of that expression and we will indicate a normal-ordered expression $*$ by $: * :$. The vacuum expectation value of a normal-ordered product of fermionic operators vanishes. This can be immediately understood by considering a simple example. We cast the operator $c_p c_q c_r^\dagger c_s^\dagger$ into its normal-ordered form by successively applying the canonical anti-commutation relations. For our example, one possible way to normal order said operator is given by

$$c_p c_q c_r^\dagger c_s^\dagger = c_r^\dagger c_s^\dagger c_p c_q - c_r^\dagger c_q \delta_{p,s} + c_r^\dagger c_p \delta_{q,s} + c_s^\dagger c_q \delta_{r,p} - \delta_{q,s} \delta_{r,p} - c_s^\dagger c_p \delta_{r,q} + \delta_{p,s} \delta_{r,q}, \quad (3.B.1)$$

where the terms that only contain delta-functions are called *full contractions*. They are the only terms that do not vanish when computing the vacuum expectation value. This example can be generalized to

$$\langle 0 | C_1 \cdots C_n | 0 \rangle = \sum_{\text{full contractions}} : C_1 \cdots C_n :, \quad (3.B.2)$$

where the sum runs over all full-contracted terms, so only those terms containing products of delta functions and not fermionic operators. Wick's theorem states that given a product of fermionic creation and annihilation operators $C_1 \cdots C_n$, the following identity holds [199]

$$C_1 \cdots C_n = : C_1 \cdots C_n : + \sum_{\text{all contractions}} : C_1 \cdots C_n :, \quad (3.B.3)$$

where the crucial difference in the sums of the right-hand sides of Eqs. (3.B.2) and (3.B.3) is, that the latter runs over all possible contractions, whereas the former only runs over full contractions.

We now consider the product of two Majorana operators $A_p A_q$. Normal ordering is only well defined for fermionic (and bosonic) creation and annihilation operators, but by means of linear extension, it can also be defined for Majorana operators. Let us assume that $p \in \{1, \dots, N_f\}$ and $q \in \{N_f + 1, \dots, 2N_f\}$ and we denote a contraction as

$$\overline{c_p^\# c_q^\#} = c_p^\# c_q^\# - : c_p^\# c_q^\# :, \quad (3.B.4)$$

where $\# \in \{\}, \dagger\}$, in which case we get

$$\begin{aligned} A_p A_q &= i(: c_p^\dagger c_q^\dagger : + \overline{c_p^\dagger c_q^\dagger}) - i(: c_p^\dagger c_q : + \overline{c_p^\dagger c_q}) + i(: c_p c_q^\dagger : + \overline{c_p c_q^\dagger}) - i(: c_p c_q : + \overline{c_p c_q}) \\ &= : A_p A_q : + \overline{A_p A_q}, \end{aligned} \quad (3.B.5)$$

where by linear extension we have

$$: A_p A_q : = i : c_p^\dagger c_q^\dagger : - i : c_p^\dagger c_q : + i : c_p c_q^\dagger : - i : c_p c_q :, \quad (3.B.6)$$

$$\overline{A_p A_q} = i \overline{c_p^\dagger c_q^\dagger} - i \overline{c_p^\dagger c_q} + i \overline{c_p c_q^\dagger} - i \overline{c_p c_q} = \langle 0 | A_p A_q | 0 \rangle, \quad (3.B.7)$$

where the last expression follows from Eq. (3.B.2). This approach can be extended similarly to other cases and ultimately to larger products of Majorana operators. In general, however, one should keep in mind that applying Wick's theorem is sensitive to the indices. For a generic Majorana expectation value w.r.t. some FGS $|\Psi_{\text{GS}}\rangle$, we have

$$\langle \Psi_{\text{GS}} | A_{p_1} A_{p_2} \cdots A_{p_{2k}} | \Psi_{\text{GS}} \rangle = (-i)^k \text{Pf}(\Gamma_m|_{p_1, \dots, p_{2k}}), \quad (3.B.8)$$

for $k = 1, \dots, N_f$ and $1 \leq p_1 < p_2 < \cdots < p_{2k} \leq 2N_f$, where $\Gamma_m|_{p_1, \dots, p_{2k}}$ is the $(2k \times 2k)$ submatrix of Γ_m [3]. The factor of $(-i)^k$ is a result of the definition of the CM in Eq. (3.3.8). With the definition of a normal-ordered product of Majorana operators in mind, we now apply Wick's theorem (3.B.3) to expand an operator of the form $A_p A_q A_r A_s$ to second order in the Majorana operators,

$$\begin{aligned} A_p A_q A_r A_s &= \langle 0 | A_p A_q A_r A_s | 0 \rangle + \langle 0 | A_p A_q | 0 \rangle : A_r A_s : - \langle 0 | A_p A_r | 0 \rangle : A_q A_s : \\ &+ \langle 0 | A_p A_s | 0 \rangle : A_q A_r : + \langle 0 | A_q A_r | 0 \rangle : A_p A_s : - \langle 0 | A_q A_s | 0 \rangle : A_p A_r : \\ &+ \langle 0 | A_r A_s | 0 \rangle : A_p A_q : + \mathcal{O}(A^4), \end{aligned} \quad (3.B.9)$$

where the sum over all full contractions is just the vacuum expectation value of $A_p A_q A_r A_s$ (see Eq. (3.B.2)) and the fourth-order term is in our case simply given by the normal-ordered product $: A_p A_q A_r A_s :.$ We want to express the large middle term in the above equation in terms of the derivative of the vacuum expectation value w.r.t. the CM, which is given by

$$\begin{aligned} \frac{d \langle 0 | A_p A_q A_r A_s | 0 \rangle}{d \langle 0 | A_i A_j | 0 \rangle} &= \delta_{ip} \delta_{jq} \langle 0 | A_r A_s | 0 \rangle + \delta_{ir} \delta_{js} \langle 0 | A_p A_q | 0 \rangle - \delta_{ip} \delta_{jr} \langle 0 | A_q A_s | 0 \rangle \\ &- \delta_{iq} \delta_{js} \langle 0 | A_p A_r | 0 \rangle + \delta_{ip} \delta_{js} \langle 0 | A_q A_r | 0 \rangle + \delta_{iq} \delta_{jr} \langle 0 | A_p A_s | 0 \rangle. \end{aligned} \quad (3.B.10)$$

By comparing Eq. (3.B.9) with Eq. (3.B.10), we can expand $A_p A_q A_r A_s$ in normal ordering to second order as

$$A_p A_q A_r A_s = \langle 0 | A_p A_q A_r A_s | 0 \rangle + \sum_{i,j} \frac{d \langle 0 | A_p A_q A_r A_s | 0 \rangle}{d \langle 0 | A_i A_j | 0 \rangle} : A_i A_j : + \mathcal{O}(A^4). \quad (3.B.11)$$

Since Wick's theorem holds for any even and pure FGS, we can generalize Eq. (3.B.11),

$$\begin{aligned} U_{\text{GS}}^\dagger A_p A_q A_r A_s U_{\text{GS}} &= \langle \Psi_{\text{GS}} | A_p A_q A_r A_s | \Psi_{\text{GS}} \rangle \\ &+ i \sum_{i,j,k,l=1}^{2N_f} : A_k (U_m^T)_{ki} \left(\frac{d \langle A_p A_q A_r A_s \rangle_{\text{GS}}}{d \Gamma_m} \right)_{ij} (U_m)_{jl} A_l : + \mathcal{O}(A^4), \end{aligned} \quad (3.B.12)$$

where we used Eq. (3.3.7) to move the left Gaussian unitary past the Majorana operator. This lead to a new linear expression of the form as in Eq. (3.B.11). We can generalize the result of Eq. (3.B.12) to arbitrary sums of products of Majorana operators, which constitute elements of the Clifford algebra [3].

Section 3.C

CANONICAL TRANSFORMATION THROUGH FLUX ATTACHMENT OPERATOR

For a Majorana operator A_j , we can compute the canonical transformation in Eq. (3.3.15) using the Baker-Campbell-Hausdorff-expansion for two operators X and Y ,

$$e^X Y e^{-X} = Y + [X, Y] + \frac{1}{2!} [X, [X, Y]] + \dots \quad (3.C.1)$$

By substituting $X = \frac{i}{2} \sum_{i,j} \omega_{ij} c_i^\dagger c_j^\dagger c_i c_j$ and $Y = c_k$, we compute the commutators of Eq. (3.C.1) with the help of the commutator relations

$$[n_j^f, c_p] = -\delta_{j,p} c_p \quad (3.C.2)$$

$$[n_j^f, c_p^\dagger] = \delta_{j,p} c_p^\dagger \quad (3.C.3)$$

and obtain for the first commutator

$$[X, Y] = i \sum_j \omega_{jk} n_j^f c_k. \quad (3.C.4)$$

and for the second commutator we get

$$[X, [X, Y]] = - \sum_{i,l} \omega_{ik} \omega_{lk} n_i^f n_l^f c_k. \quad (3.C.5)$$

Continuing in this manner gives the relation in Eq. (3.3.15).

Section 3.D

 DERIVING THE ROTATED HAMILTONIAN

We now study the action of the Hamiltonian of Eq. (3.3.1) on the non-Gaussian state $|\Psi_{\text{NGS}}\rangle$ of Eq. (3.3.3). First, we introduce the rotated Hamiltonian $\bar{H} = U_{\text{FA}}^\dagger H U_{\text{FA}}$ which allows us to write

$$H |\Psi_{\text{NGS}}\rangle = U_{\text{FA}} \bar{H} |\Psi_{\text{GS}}\rangle. \quad (3.D.1)$$

The transformation acts as

$$\bar{H}(C) = H(\bar{C}_{\text{FA}}), \quad (3.D.2)$$

where we define

$$\bar{C}_{\text{FA}} = U_{\text{FA}}^\dagger C U_{\text{FA}}, \quad (3.D.3)$$

which are vectors of rotated spin orbitals given by

$$\bar{C}_{\text{FA}} = (\bar{c}_{\text{FA}}, \bar{c}_{\text{FA}}^\dagger)^T, \quad (3.D.4)$$

$$\bar{c}_{\text{FA}} = \bar{c}_{1\text{FA}}, \bar{c}_{2\text{FA}}, \dots, \bar{c}_{N_{\text{fFA}}}, \quad (3.D.5)$$

with elements

$$\begin{aligned} \bar{c}_{j\text{FA}} &= e^{-\frac{i}{2} \sum_{i,j} \omega_{ij} n_i^f n_j^f} c_j e^{\frac{i}{2} \sum_{i,j} \omega_{ij} n_i^f n_j^f} \\ &= e^{i \sum_i \omega_{ij} n_i^f} c_j, \end{aligned} \quad (3.D.6)$$

where the last equations follows from Eq. (3.3.15). To compute the rotated Hamiltonian \bar{H} , we compute the commutator of the unitary rotated number operator w.r.t. the raising and lowering operators using Eq. (3.3.16). Given the second quantized system Hamiltonian of Eq. (3.3.1), by commuting all terms containing exponentials of number operators, the rotated Hamiltonian $\bar{H}_{\text{FA}} = H(\bar{C}_{\text{FA}})$ takes the form given by Eq. (3.3.19).

Section 3.E

CALCULATION OF EXPECTATION VALUES

In this section, we will provide explicit expressions for the expectation values appearing in Eq. (3.3.27). Since we are computing the above expressions from the derivative of an exponential term given by Eq. (3.4.67), we can expand the latter through

$$\exp(X) = 1 + X + X^2/(2!) + \dots \quad (3.E.1)$$

The argument $X = X(J, J^*)$ is defined in Eq. (3.E.16) and is a homogeneous second degree polynomial, which is evaluated at zero in the Grassmann variables J_1, \dots, J_{N_f} and $J_1^*, \dots, J_{N_f}^*$ after taking the derivatives via \mathcal{F}_j . We mostly ignore the superscript $\tilde{\alpha}$ in the notation throughout this section, as we are considering only a single expression.

Since \mathcal{F}_J is acting on the exponential, the only non-vanishing term in the exponential expansion of Eq. (3.E.1) that survives is the a -th power of X , where a corresponds to the number of creation operators in Eq. (3.3.27) (which is identical to the number of annihilation operators since we assume $a = b$). Therefore, we can write

$$\mathcal{A}_{j_1^\dagger \dots j_a^\dagger k_1 \dots k_a} = s_{N_f} \left(\frac{1}{2}\right)^{N_f} \text{Pf}(\Gamma_F) \frac{1}{a!} \mathcal{F}_J X^a. \quad (3.E.2)$$

We define the $(2N_f \times 2N_f)$ -matrix

$$\mathcal{M} = \Sigma_F^\dagger \text{sqrt}(1 - e^{i\tilde{\alpha}}) \Gamma_F^{-1} \text{sqrt}(1 - e^{i\tilde{\alpha}}) \Sigma_F \quad (3.E.3)$$

and we rewrite Eq. (3.4.62) as

$$\Gamma_F = \text{sqrt}(1 - e^{i\tilde{\alpha}}) \left(\Gamma_m - \frac{1 + e^{i\tilde{\alpha}}}{1 - e^{i\tilde{\alpha}}} \Upsilon \right) \text{sqrt}(1 - e^{i\tilde{\alpha}}), \quad (3.E.4)$$

where we defined the $(2N_f \times 2N_f)$ -diagonal matrix

$$\frac{1 + e^{i\tilde{\alpha}}}{1 - e^{i\tilde{\alpha}}} = \mathbf{1}_2 \otimes \begin{pmatrix} \frac{1+e^{i\tilde{\alpha}}}{1-e^{i\tilde{\alpha}}} & & & \\ & \ddots & & \\ & & & \frac{1+e^{i\tilde{\alpha}}}{1-e^{i\tilde{\alpha}}} \end{pmatrix}. \quad (3.E.5)$$

We insert Eq. (3.E.4) into Eq. (3.E.3), which leads to

$$\mathcal{M} = \Sigma_F^\dagger \Gamma_H^{-1} \Sigma_F, \quad (3.E.6)$$

where we set

$$\Gamma_H = \Gamma_m - \frac{1 + e^{i\tilde{\alpha}}}{1 - e^{i\tilde{\alpha}}} \Upsilon \quad (3.E.7)$$

and we will use the following short-hand notation for Σ_F and its adjoint Σ_F^\dagger (which we defined in Eq. (3.4.64)),

$$\Sigma_F^\dagger = \begin{pmatrix} \frac{-1}{1-e^{i\tilde{\alpha}}} & \frac{-i}{1-e^{i\tilde{\alpha}}} \\ \frac{e^{i\tilde{\alpha}}}{1-e^{i\tilde{\alpha}}} & \frac{-ie^{i\tilde{\alpha}}}{1-e^{i\tilde{\alpha}}} \end{pmatrix}, \quad \Sigma_F = \begin{pmatrix} \frac{e^{i\tilde{\alpha}}}{1-e^{i\tilde{\alpha}}} & \frac{-1}{1-e^{i\tilde{\alpha}}} \\ \frac{-ie^{i\tilde{\alpha}}}{1-e^{i\tilde{\alpha}}} & \frac{-i}{1-e^{i\tilde{\alpha}}} \end{pmatrix}. \quad (3.E.8)$$

The block entries are the $(N_f \times N_f)$ -matrices as given in Eq. (3.4.64). Note that in what follows, we will make extensive use of the notation introduced in Section 3.2.

3.E.1 Matrix elements of \mathcal{M}

In this section, we will study the properties of the matrix elements of \mathcal{M} defined in Eq. (3.E.6), as they will be extensively used when computing $\mathcal{A}_{j_1^\dagger \dots j_a^\dagger k_1 \dots k_a}$.

\mathcal{M} is a $(2N_f \times 2N_f)$ -matrix of four equal sized blocks and we are going to compute the matrix elements in each of the four blocks. Setting $p, q \in \{1, 2, \dots, N_f\}$, we first compute $\mathcal{M}_{q,p}$, which is restricted to the upper-left block of \mathcal{M} ,

$$\begin{aligned} \mathcal{M}_{q,p} &= \left(\mathbf{1}_q^T, \mathbf{0}^T \right) \Sigma_F^\dagger \Gamma_H^{-1} \Sigma_F \begin{pmatrix} \mathbf{1}_p \\ \mathbf{0} \end{pmatrix} = \mathbf{1}_q^T \left(\frac{-1}{1-e^{i\tilde{\alpha}_q}}, \frac{-i}{1-e^{i\tilde{\alpha}_q}} \right) \Gamma_H^{-1} \begin{pmatrix} \frac{e^{i\tilde{\alpha}_p}}{1-e^{i\tilde{\alpha}_p}} \\ \frac{-ie^{i\tilde{\alpha}_p}}{1-e^{i\tilde{\alpha}_p}} \end{pmatrix} \mathbf{1}_p \\ &= \frac{-1}{1-e^{i\tilde{\alpha}_q}} \frac{e^{i\tilde{\alpha}_p}}{1-e^{i\tilde{\alpha}_p}} \left(\mathbf{1}_q^T, \mathbf{i}_q^T \right) \Gamma_H^{-1} \begin{pmatrix} \mathbf{1}_p \\ -\mathbf{i}_p \end{pmatrix}. \end{aligned} \quad (3.E.9)$$

Note that the first two factors in the last line of the above equation are complex numbers. In an analogous manner, we compute the matrix element $\mathcal{M}_{p+N_f, q+N_f}$ of the lower-right block of \mathcal{M} , giving

$$\mathcal{M}_{p+N_f, q+N_f} = \left(\mathbf{0}^T, \mathbf{1}_p^T \right) \Sigma_F^\dagger \Gamma_H^{-1} \Sigma_F \begin{pmatrix} \mathbf{0} \\ \mathbf{1}_q \end{pmatrix} = \frac{e^{i\tilde{\alpha}_p}}{1-e^{i\tilde{\alpha}_p}} \frac{-1}{1-e^{i\tilde{\alpha}_q}} \left(\mathbf{1}_p^T, -\mathbf{i}_p^T \right) \Gamma_H^{-1} \begin{pmatrix} \mathbf{1}_q \\ \mathbf{i}_q \end{pmatrix}. \quad (3.E.10)$$

Since $\mathcal{M}_{p+N_f, q+N_f}$ is a number (if you want, consider it to be a (1×1) -matrix), it is equal to its transpose, $\mathcal{M}_{p+N_f, q+N_f} = \mathcal{M}_{p+N_f, q+N_f}^T$, therefore we can write Eq. (3.E.10) as

$$\begin{aligned} \mathcal{M}_{p+N_f, q+N_f} &= \frac{e^{i\tilde{\alpha}_p}}{1-e^{i\tilde{\alpha}_p}} \frac{-1}{1-e^{i\tilde{\alpha}_q}} \left(\mathbf{1}_q^T, \mathbf{i}_q^T \right) \left(\Gamma_H^{-1} \right)^T \begin{pmatrix} \mathbf{1}_p \\ -\mathbf{i}_p \end{pmatrix} \\ &= \frac{e^{i\tilde{\alpha}_p}}{1-e^{i\tilde{\alpha}_p}} \frac{1}{1-e^{i\tilde{\alpha}_q}} \left(\mathbf{1}_q^T, \mathbf{i}_q^T \right) \Gamma_H^{-1} \begin{pmatrix} \mathbf{1}_p \\ -\mathbf{i}_p \end{pmatrix} \\ &= -\mathcal{M}_{q,p}, \end{aligned} \quad (3.E.11)$$

where the second line follows from the fact that Γ_H is a skew-symmetric matrix and as such its inverse is also skew-symmetric, in other words $\left(\Gamma_H^{-1} \right)^T = -\Gamma_H^{-1}$.

We now turn to the cross terms $\mathcal{M}_{q, p+N_f}$ and $\mathcal{M}_{p+N_f, q}$. The upper right matrix elements of the block matrix \mathcal{M} are given by

$$\begin{aligned} \mathcal{M}_{q, p+N_f} &= \frac{1}{(1-e^{i\tilde{\alpha}_q})(1-e^{i\tilde{\alpha}_p})} \left(\mathbf{1}_q^T, \mathbf{i}_q^T \right) \Gamma_H^{-1} \begin{pmatrix} \mathbf{1}_p \\ \mathbf{i}_p \end{pmatrix} \\ &= -\mathcal{M}_{p, q+N_f}, \end{aligned} \quad (3.E.12)$$

while the elements of the lower left block matrix are

$$\begin{aligned} \mathcal{M}_{p+N_f, q} &= \frac{1}{(1-e^{-i\tilde{\alpha}_p})(1-e^{-i\tilde{\alpha}_q})} \left(\mathbf{1}_p^T, -\mathbf{i}_p^T \right) \Gamma_H^{-1} \begin{pmatrix} \mathbf{1}_q \\ -\mathbf{i}_q \end{pmatrix} \\ &= -\mathcal{M}_{q+N_f, p}. \end{aligned} \quad (3.E.13)$$

3.E.2 Zeroth order

Using Eqs. (3.4.67) and (3.E.2), Eq. (3.3.28) is simply given by

$$\mathcal{A} = s_{N_f} \left(\frac{1}{2} \right)^{N_f} \text{Pf}(\Gamma_F), \quad (3.E.14)$$

since only the constant survives in Eq. (3.E.1) when X is evaluated at zero. The result in Eq. (3.E.14) is identical to Eq. (D.10) in Ref. [41].

3.E.3 Second order

We evaluate

$$\mathcal{A}_{p^\dagger q} = \mathcal{A} \mathcal{F}_J X, \quad (3.E.15)$$

where

$$\begin{aligned} X = & -\frac{i}{2} \sum_{v,w=1}^{N_f} \left(J_v^* \mathcal{M}_{v,w} J_w + J_v^* \mathcal{M}_{v,w+N_f} J_w^* + J_v \mathcal{M}_{v+N_f,w} J_w + J_v \mathcal{M}_{v+N_f,w+N_f} J_w^* \right) \\ & + \sum_{k=1}^{N_f} \frac{J_k^* J_k}{1 - e^{-i\tilde{\alpha}_k}} \end{aligned} \quad (3.E.16)$$

is the argument inside the exponential in Eq. (3.4.67). We need to reorder the Grassmann variables, since we use the definition of Grassmann differentiation "from the left" [4]. The first derivative is given by $\delta/(\delta J_q^*)$, therefore all conjugate Grassmann variables need to be moved to the left. Once the term in Eq. (3.E.16) is reordered, we can take the derivative \mathcal{F}_J as defined in Eq. (3.4.25). Using the findings of Section 3.E.1—after performing the derivative in Eq. (3.E.15)—leads to

$$\mathcal{F}_J X = \left(\frac{\delta_{pq}}{1 - e^{-i\tilde{\alpha}_q}} - i \mathcal{M}_{q,p} \right). \quad (3.E.17)$$

We now insert Eqs. (3.E.9) into Eq. (3.E.17) to get an expression for the second-order expression of Eq.(3.E.15),

$$\mathcal{A}_{p^\dagger q} = \mathcal{A} \left(\frac{\delta_{pq}}{1 - e^{-i\tilde{\alpha}_q}} + \frac{i e^{i\tilde{\alpha}_p}}{(1 - e^{i\tilde{\alpha}_p})(1 - e^{i\tilde{\alpha}_q})} \left(\mathbf{1}_q^T, \mathbf{i}_q^T \right) \Gamma_H^{-1} \begin{pmatrix} \mathbf{1}_p \\ -\mathbf{i}_p \end{pmatrix} \right). \quad (3.E.18)$$

Eq. (3.E.18) should coincide with Eq. (D.11) in Ref. [41], which states that

$$\mathcal{A}_{p^\dagger q} = \mathcal{A} \frac{i}{4} e^{i\tilde{\alpha}_p} \left(\mathbf{1}_q^T, \mathbf{i}_q^T \right) \mathcal{G} \begin{pmatrix} \mathbf{1}_p \\ -\mathbf{i}_p \end{pmatrix}, \quad (3.E.19)$$

where \mathcal{G} is identical to Eq. (3.4.69) and the $(2N_f \times 2N_f)$ -matrix $(1 - e^{i\tilde{\alpha}})$ is defined in Eq. (3.4.53). In the rest of this subsection, we will show that the expressions in Eq. (3.E.18) and Eq. (3.E.19) are equivalent. To achieve this, we first rewrite the denominator of on the right-hand side of Eq. (3.4.69) as

$$\mathbb{1}_{2N_f} + \frac{1}{2} (1 - e^{i\tilde{\alpha}}) (\Upsilon \Gamma_m - \mathbb{1}_{2N_f}) = \frac{1}{2} (1 - e^{i\tilde{\alpha}}) \Upsilon \Gamma_H \quad (3.E.20)$$

where we used that

$$\left[\Upsilon, \frac{1 + e^{i\tilde{\alpha}}}{1 - e^{i\tilde{\alpha}}} \right] = 0 \quad (3.E.21)$$

and $\Upsilon^2 = -\mathbb{1}_{2N_f}$. Using the above identity, $\mathcal{G}^{\tilde{\alpha}}$ can be written as

$$\mathcal{G}^{\tilde{\alpha}} = 2(\Gamma_m + \Upsilon) \left((1 - e^{i\tilde{\alpha}}) \Upsilon \Gamma_H \right)^{-1} = 2 \left(-\Upsilon \frac{1}{1 - e^{i\tilde{\alpha}}} + \frac{2}{1 - e^{i\tilde{\alpha}}} \Upsilon \Gamma_H^{-1} \Upsilon^{-1} \frac{1}{1 - e^{i\tilde{\alpha}}} \right). \quad (3.E.22)$$

For any $\tilde{\alpha}$, $\mathcal{G}^{\tilde{\alpha}}$ will be anti-symmetric under transposition, which is proven below,

$$\begin{aligned} (\mathcal{G}^{\tilde{\alpha}})^T &= - \left[\mathbb{1}_{2N_f} + \frac{1}{2} (\Gamma_m \Upsilon - \mathbb{1}_{2N_f}) (1 - e^{i\tilde{\alpha}}) \right]^{-1} (\Gamma_m + \Upsilon) \\ &= - (\Gamma_m + \Upsilon) \left[\mathbb{1}_{2N_f} + \frac{1}{2} (1 - e^{i\tilde{\alpha}}) (\Upsilon \Gamma_m - \mathbb{1}_{2N_f}) \right]^{-1} \\ &= - \mathcal{G}^{\tilde{\alpha}}. \end{aligned} \quad (3.E.23)$$

We insert Eq.(3.E.22) into Eq. (3.E.19) and note that $(\mathbf{1}_q^T, \mathbf{i}_k^T)$ and $(\mathbf{1}_q, \mathbf{i}_k)^T$ are eigenvectors of Υ and therefore also of its inverse, resulting in

$$\begin{aligned} \mathcal{A}_{p^\dagger q} &= \mathcal{A} \frac{i}{2} e^{i\tilde{\alpha}_p} (\mathbf{1}_q^T, \mathbf{i}_q^T) \left(-\Upsilon \frac{1}{1 - e^{i\tilde{\alpha}}} + \frac{2}{1 - e^{i\tilde{\alpha}}} \Upsilon \Gamma_H^{-1} \Upsilon^{-1} \frac{1}{1 - e^{i\tilde{\alpha}}} \right) \begin{pmatrix} \mathbf{1}_p \\ -\mathbf{i}_p \end{pmatrix} \\ &= \mathcal{A} \left(\frac{\delta_{pq}}{1 - e^{-i\tilde{\alpha}_q}} + \frac{i e^{i\tilde{\alpha}_p}}{(1 - e^{i\tilde{\alpha}_p})(1 - e^{i\tilde{\alpha}_q})} (\mathbf{1}_q^T, \mathbf{i}_q^T) \Gamma_H^{-1} \begin{pmatrix} \mathbf{1}_p \\ -\mathbf{i}_p \end{pmatrix} \right), \end{aligned} \quad (3.E.24)$$

which is identical to Eq. (3.E.18), and therefore our result is in exact agreement with Eq. (D.11) in Ref. [41]. An alternative expression for $\mathcal{A}_{p^\dagger q}$ in terms of the matrix elements \mathcal{M} is given by

$$\mathcal{A}_{p^\dagger q} = \mathcal{A} \left(\frac{\delta_{pq}}{1 - e^{-i\tilde{\alpha}_q}} - i \mathcal{M}_{q,p} \right). \quad (3.E.25)$$

The representation in Eq. (3.E.25) in terms of \mathcal{M} will be useful for finding underlying symmetries, while the representation in terms of \mathcal{G} as in Eq. (3.E.19) gives a desirable form for numerical implementation which becomes critical whenever $\tilde{\alpha}(j) = 0$.

We now turn to the terms $\mathcal{A}_{p^\dagger q^\dagger}$ and \mathcal{A}_{pq} . Using Eq. (3.E.2), we compute for the first term

$$\lim_{J \rightarrow 0} \frac{d}{dJ_p} \frac{d}{dJ_q} X = i \mathcal{M}_{p+N_f, q}, \quad (3.E.26)$$

and for the second term

$$\lim_{J \rightarrow 0} \frac{d}{dJ_p^*} \frac{d}{dJ_q^*} X = i \mathcal{M}_{p, q+N_f}. \quad (3.E.27)$$

This leads to

$$\mathcal{A}_{p^\dagger q^\dagger} = -i \mathcal{A} \mathcal{M}_{q+N_f, p} = \frac{-i \mathcal{A}}{(1 - e^{-i\tilde{\alpha}_p})(1 - e^{-i\tilde{\alpha}_q})} (\mathbf{1}_q^T, -\mathbf{i}_q^T) \Gamma_H^{-1} \begin{pmatrix} \mathbf{1}_p \\ -\mathbf{i}_p \end{pmatrix}, \quad (3.E.28)$$

$$\mathcal{A}_{pq} = -i \mathcal{A} \mathcal{M}_{q, p+N_f} = \frac{-i \mathcal{A}}{(1 - e^{i\tilde{\alpha}_q})(1 - e^{i\tilde{\alpha}_p})} (\mathbf{1}_q^T, \mathbf{i}_q^T) \Gamma_H^{-1} \begin{pmatrix} \mathbf{1}_p \\ \mathbf{i}_p \end{pmatrix}. \quad (3.E.29)$$

We first turn to Eq. (3.E.28). Using Eq. (3.E.13) and Eq. (3.E.22), we compute

$$\begin{aligned} (\mathbf{1}_q^T, -\mathbf{i}_q^T) \mathcal{G} \begin{pmatrix} \mathbf{1}_p \\ -\mathbf{i}_p \end{pmatrix} &= 2 (\mathbf{1}_q^T, -\mathbf{i}_q^T) \left(-\Upsilon \frac{1}{1 - e^{i\tilde{\alpha}}} + \frac{2}{1 - e^{i\tilde{\alpha}}} \Upsilon \Gamma_H^{-1} \Upsilon^{-1} \frac{1}{1 - e^{i\tilde{\alpha}}} \right) \begin{pmatrix} \mathbf{1}_p \\ -\mathbf{i}_p \end{pmatrix} \\ &= \frac{2i}{1 - e^{i\tilde{\alpha}}} (\mathbf{1}_q^T, -\mathbf{i}_q^T) \begin{pmatrix} \mathbf{1}_p \\ -\mathbf{i}_p \end{pmatrix} \\ &\quad - \frac{4}{(1 - e^{i\tilde{\alpha}_q})(1 - e^{i\tilde{\alpha}_p})} (\mathbf{1}_q^T, -\mathbf{i}_q^T) \Gamma_H^{-1} \begin{pmatrix} \mathbf{1}_p \\ -\mathbf{i}_p \end{pmatrix}, \end{aligned} \quad (3.E.30)$$

which (since the first term gives zero for all choices p, q) leads to

$$\left(\mathbf{1}_q^T, -\mathbf{i}_q^T\right) \mathcal{G} \begin{pmatrix} \mathbf{1}_p \\ -\mathbf{i}_p \end{pmatrix} = \frac{-4e^{-i(\tilde{\alpha}_p + \tilde{\alpha}_q)}}{(1 - e^{-i\tilde{\alpha}_p})(1 - e^{-i\tilde{\alpha}_q})} \left(\mathbf{1}_q^T, -\mathbf{i}_q^T\right) \Gamma_H^{-1} \begin{pmatrix} \mathbf{1}_p \\ -\mathbf{i}_p \end{pmatrix}. \quad (3.E.31)$$

By comparing the result of Eq (3.E.31) with Eq. (3.E.28), we arrive at the following expression,

$$\mathcal{A}_{p^\dagger q^\dagger} = \frac{i}{4} \mathcal{A} e^{i(\tilde{\alpha}_p + \tilde{\alpha}_q)} \left(\mathbf{1}_q^T, -\mathbf{i}_q^T\right) \mathcal{G} \begin{pmatrix} \mathbf{1}_p \\ -\mathbf{i}_p \end{pmatrix}. \quad (3.E.32)$$

Analogously, we compute

$$\left(\mathbf{1}_q^T, \mathbf{i}_q^T\right) \mathcal{G} \begin{pmatrix} \mathbf{1}_p \\ \mathbf{i}_p \end{pmatrix} = \frac{-4}{(1 - e^{-i\tilde{\alpha}_p})(1 - e^{-i\tilde{\alpha}_q})} \left(\mathbf{1}_q^T, \mathbf{i}_q^T\right) \Gamma_H^{-1} \begin{pmatrix} \mathbf{1}_p \\ \mathbf{i}_p \end{pmatrix}, \quad (3.E.33)$$

which leads to

$$\mathcal{A}_{pq} = \frac{i}{4} \mathcal{A} \left(\mathbf{1}_q^T, \mathbf{i}_q^T\right) \mathcal{G} \begin{pmatrix} \mathbf{1}_p \\ \mathbf{i}_p \end{pmatrix}. \quad (3.E.34)$$

3.E.4 Fourth order

In the same vein as in the previous subsection, we evaluate

$$\mathcal{A}_{p^\dagger q^\dagger rs} = \mathcal{A} \frac{1}{2!} \mathcal{F}_J X^2 \quad (3.E.35)$$

by first computing X^2 , where X is given in Eq. (3.E.16). Since the derivative operator \mathcal{F}_J will only give non-zero contributions to terms where the number $b = a$ ($= 2$) of non-conjugated Grassmann variables equals the number a ($= 2$) of conjugated variables, we can drop all other terms. The derivatives are ordered in such a way that the conjugated Grassmann variables are acted on first. Therefore, we reorder the remaining terms, simplifying the relevant part of the squared expression X^2 (by using the symmetry properties of Eqs. (3.E.11)-(3.E.13) derived in Section 3.E.1) to

$$\begin{aligned} & \sum_{k,l=1}^{N_f} \frac{-J_k^* J_l^* J_k J_l}{(1 - e^{-i\tilde{\alpha}_k})(1 - e^{-i\tilde{\alpha}_l})} + i \sum_{k,v,w=1}^{N_f} \frac{J_k^* J_v^* J_k J_w \mathcal{M}_{v,w} - J_k^* J_w^* J_k J_v \mathcal{M}_{v+N_f, w+N_f}}{1 - e^{-i\tilde{\alpha}_k}} \\ & + \sum_{k,l,v,w=1}^{N_f} J_k^* J_v^* J_l J_w \mathcal{M}_{k,l} \mathcal{M}_{v,w} - \frac{1}{2} \sum_{k,l,v,w=1}^{N_f} J_k^* J_l^* J_v J_w \mathcal{M}_{k,l+N_f} \mathcal{M}_{v+N_f, w}, \end{aligned} \quad (3.E.36)$$

where only the non-vanishing terms of X^2 under the action of \mathcal{F}_J are displayed. For the fourth-order expression, the derivative operator has the form

$$\mathcal{F}_J = \lim_{J \rightarrow 0} \frac{\delta}{\delta J_p} \frac{\delta}{\delta J_q} \frac{\delta}{\delta J_r^*} \frac{\delta}{\delta J_s^*}. \quad (3.E.37)$$

The derivative \mathcal{F}_J acts "from the left" and the derivative w.r.t. a product of Grassmann variables is simply given by

$$\frac{\delta}{\delta \alpha_n} \alpha_j \alpha_k = \delta_{nj} \alpha_k - \delta_{nk} \alpha_j, \quad (3.E.38)$$

from which one can derive the following useful relations

$$\frac{\delta^2(\alpha_j \alpha_k)}{\delta \alpha_m \delta \alpha_n} = \delta_{nj} \delta_{mk} - \delta_{nk} \delta_{mj}, \quad (3.E.39)$$

$$\frac{\delta^4(\alpha_i^* \alpha_j^* \alpha_k \alpha_l)}{\delta \alpha_p \delta \alpha_q \delta \alpha_r^* \delta \alpha_s^*} = \delta_{si} \delta_{rj} \delta_{qk} \delta_{pl} - \delta_{si} \delta_{rj} \delta_{ql} \delta_{pk} - \delta_{sj} \delta_{ri} \delta_{qk} \delta_{pl} + \delta_{sj} \delta_{ri} \delta_{ql} \delta_{pk}. \quad (3.E.40)$$

Applying \mathcal{F}_J on X^2 , the only non-vanishing part that needs to be considered is displayed in Eq. (3.E.36). We will list the results for each of the four sums below. The derivative contributions from the first sum result in

$$\mathcal{F}_J \sum_{k,l=1}^{N_f} \frac{-J_k^* J_l^* J_k J_l}{(1 - e^{-i\tilde{\alpha}_k})(1 - e^{-i\tilde{\alpha}_l})} = \frac{2(\delta_{ps} \delta_{qr} - \delta_{qs} \delta_{pr})}{(1 - e^{-i\tilde{\alpha}_q})(1 - e^{-i\tilde{\alpha}_p})}, \quad (3.E.41)$$

the derivative on the second sum in Eq. (3.E.36) gives

$$\begin{aligned} & i \mathcal{F}_J \sum_{k,v,w=1}^{N_f} \frac{J_k^* J_v^* J_k J_w \mathcal{M}_{v,w} - J_k^* J_w^* J_k J_v \mathcal{M}_{v+N_f, w+N_f}}{1 - e^{-i\tilde{\alpha}_k}} \\ &= \frac{2i(\delta_{qs} \mathcal{M}_{r,p} - \delta_{qr} \mathcal{M}_{s,p})}{1 - e^{-i\tilde{\alpha}_q}} - \frac{2i(\delta_{ps} \mathcal{M}_{r,q} - \delta_{pr} \mathcal{M}_{s,q})}{1 - e^{-i\tilde{\alpha}_p}}. \end{aligned} \quad (3.E.42)$$

By applying the derivatives \mathcal{F}_J w.r.t. the Grassmann variables on the third and fourth terms of Eq. (3.E.36), we end up with the following expression,

$$\begin{aligned} & \mathcal{F}_J \sum_{k,l,v,w=1}^{N_f} J_k^* J_v^* J_l J_w \mathcal{M}_{k,l} \mathcal{M}_{v,w} - \frac{1}{2} \mathcal{F}_J \sum_{k,l,v,w=1}^{N_f} J_k^* J_l^* J_v J_w \mathcal{M}_{k,l+N_f} \mathcal{M}_{v+N_f,w} \\ &= 2(\mathcal{M}_{r,p} \mathcal{M}_{s,q} - \mathcal{M}_{r,q} \mathcal{M}_{s,p} - \mathcal{M}_{s,r+N_f} \mathcal{M}_{q+N_f,p}). \end{aligned} \quad (3.E.43)$$

Inserting the results of Eqs. (3.E.41)-(3.E.43) into Eq. (3.E.35), we get

$$\begin{aligned} \mathcal{A}_{p^\dagger q^\dagger r s} &= \mathcal{A} \left(\frac{(\delta_{ps} \delta_{qr} - \delta_{qs} \delta_{pr})}{(1 - e^{-i\tilde{\alpha}_q})(1 - e^{-i\tilde{\alpha}_p})} + \frac{i\delta_{qs}}{1 - e^{-i\tilde{\alpha}_q}} \mathcal{M}_{r,p} - \frac{i\delta_{ps}}{1 - e^{-i\tilde{\alpha}_p}} \mathcal{M}_{r,q} - \frac{i\delta_{qr}}{1 - e^{-i\tilde{\alpha}_q}} \mathcal{M}_{s,p} \right. \\ & \quad \left. + \frac{i\delta_{pr}}{1 - e^{-i\tilde{\alpha}_p}} \mathcal{M}_{s,q} + \mathcal{M}_{r,p} \mathcal{M}_{s,q} - \mathcal{M}_{r,q} \mathcal{M}_{s,p} - \mathcal{M}_{s,r+N_f} \mathcal{M}_{q+N_f,p} \right) \\ &= \mathcal{A} \left(\left(\frac{\delta_{ps}}{1 - e^{-i\tilde{\alpha}_p}} - i\mathcal{M}_{s,p} \right) \left(\frac{\delta_{qr}}{1 - e^{-i\tilde{\alpha}_q}} - i\mathcal{M}_{r,q} \right) \right. \\ & \quad \left. - \left(\frac{\delta_{pr}}{1 - e^{-i\tilde{\alpha}_p}} - i\mathcal{M}_{r,p} \right) \left(\frac{\delta_{qs}}{1 - e^{-i\tilde{\alpha}_q}} - i\mathcal{M}_{s,q} \right) + (-i)^2 \mathcal{M}_{q+N_f,p} \mathcal{M}_{s,r+N_f} \right). \end{aligned} \quad (3.E.44)$$

Eq. (3.E.44) was written in order to highlight the factorization of the fourth-order expression into second-order expressions. We know that the expectation value w.r.t. some Gaussian state (indicated by the lower script GS') of a quartic fermionic operator transforms due to Wick's theorem as

$$\langle c_p^\dagger c_q^\dagger c_r c_s \rangle_{\text{GS}'} = \langle c_p^\dagger c_s \rangle_{\text{GS}'} \langle c_q^\dagger c_r \rangle_{\text{GS}'} - \langle c_p^\dagger c_r \rangle_{\text{GS}'} \langle c_q^\dagger c_s \rangle_{\text{GS}'} + \langle c_p^\dagger c_q^\dagger \rangle_{\text{GS}'} \langle c_r c_s \rangle_{\text{GS}'}, \quad (3.E.45)$$

identical to the splitting in the second line of Eq. (3.E.44) w.r.t. the the second-order monomials as given in Eq. (3.E.25), i.e.

$$\frac{\mathcal{A}_{p^\dagger q^\dagger r s}}{\mathcal{A}} = \left(\frac{\mathcal{A}_{p^\dagger s}}{\mathcal{A}} \frac{\mathcal{A}_{q^\dagger r}}{\mathcal{A}} - \frac{\mathcal{A}_{p^\dagger r}}{\mathcal{A}} \frac{\mathcal{A}_{q^\dagger s}}{\mathcal{A}} + \frac{\mathcal{A}_{p^\dagger q^\dagger}}{\mathcal{A}} \frac{\mathcal{A}_{rs}}{\mathcal{A}} \right). \quad (3.E.46)$$

Section 3.F

SYMMETRY PROPERTIES OF THE FERMIONIC EXPECTATION VALUES

So far, we have neglected the superscript $\tilde{\alpha}$, which in our case are most commonly of the form $\alpha_{pq}^{\text{FA}}(k)$ and $\beta_{pqrs}^{\text{FA}}(k)$. We will also use the symmetry properties of f_{pq}^{FA} and h_{pqrs}^{FA} , all defined in Eqs. (3.3.22)-(3.3.25). By definition, we have

$$\alpha_{pq}^{\text{FA}}(k) = -\alpha_{qp}^{\text{FA}}(k) \quad (3.F.1)$$

and

$$\begin{aligned} \beta_{pqrs}^{\text{FA}}(k) &= \beta_{qprs}^{\text{FA}}(k) = \beta_{pqsr}^{\text{FA}}(k) = \beta_{qpsr}^{\text{FA}}(k) \\ &= -\beta_{srqp}^{\text{FA}}(k) = -\beta_{rsqp}^{\text{FA}}(k) = -\beta_{srpq}^{\text{FA}}(k) = -\beta_{rspq}^{\text{FA}}(k). \end{aligned} \quad (3.F.2)$$

Given the definition of the second-order expressions in Eq. (3.3.28),

$$\mathcal{A}_{p^\dagger q}^{\tilde{\alpha}} = \left\langle e^{i \sum_j \tilde{\alpha}(j) n_j^f} c_p^\dagger c_q \right\rangle_{\text{GS}}, \quad (3.F.3)$$

we compute its adjoint,

$$\left(\mathcal{A}_{p^\dagger q}^{\tilde{\alpha}} \right)^* = \left\langle c_q^\dagger c_p e^{-i \sum_j \tilde{\alpha}(j) n_j^f} \right\rangle_{\text{GS}}, \quad (3.F.4)$$

where we always assume that $\tilde{\alpha}(j)^* = \tilde{\alpha}(j)$, which holds for all expressions we encounter in this work, see e.g. Eqs. (3.3.24)-(3.3.25). We can move the exponential expression past the quadratic fermionic operator by means of Eq. (3.3.16), such that Eq. (3.F.4) can be written as

$$\left(\mathcal{A}_{p^\dagger q}^{\tilde{\alpha}} \right)^* = e^{i(\tilde{\alpha}(q) - \tilde{\alpha}(p))} \left\langle e^{-i \sum_j \tilde{\alpha}(j) n_j^f} c_q^\dagger c_p \right\rangle_{\text{GS}}. \quad (3.F.5)$$

Similarly, we obtain

$$\left(\mathcal{A}_{p^\dagger q^\dagger rs}^{\tilde{\alpha}} \right)^* = e^{i(\tilde{\alpha}(s) + \tilde{\alpha}(r) - \tilde{\alpha}(q) - \tilde{\alpha}(p))} \left\langle e^{-i \sum_j \tilde{\alpha}(j) n_j} c_s^\dagger c_r^\dagger c_q c_p \right\rangle. \quad (3.F.6)$$

3.F.1 Expectation values belonging to one-body terms

Here, we consider one-body terms of the form $f_{pq}^{\text{FA}} \mathcal{A}_{p^\dagger q}^{\alpha_{pq}^{\text{FA}}}$ which appear in Eq.(3.3.29). For such terms, the phase factors in Eq. (3.F.5) are given by Eq. (3.3.24) and we have

$$\mathcal{A}_{p^\dagger q}^{\alpha_{pq}^{\text{FA}}} = \left\langle e^{i \sum_j \alpha_{pq}^{\text{FA}}(j) n_j^f} c_p^\dagger c_q \right\rangle_{\text{GS}}. \quad (3.F.7)$$

Computing the adjoint of the above expression using Eqs. (3.F.1) and (3.F.5) yields

$$\left(\mathcal{A}_{p^\dagger q}^{\alpha_{pq}^{\text{FA}}} \right)^* = e^{i(\alpha_{pq}^{\text{FA}}(q) - \alpha_{pq}^{\text{FA}}(p))} \left\langle e^{-i \sum_j \tilde{\alpha}_{pq}^{\text{FA}}(j) n_j^f} c_q^\dagger c_p \right\rangle_{\text{GS}} = e^{-i(\omega_{pq} + \omega_{qp})} \left\langle e^{i \sum_j \tilde{\alpha}_{qp}^{\text{FA}}(j) n_j^f} c_q^\dagger c_p \right\rangle_{\text{GS}}. \quad (3.F.8)$$

We now compute the adjoint of the rotated fermionic one-body integrals, since $f_{pq}^* = f_{qp}$,

$$\left(f_{pq}^{\text{FA}}\right)^* = f_{pq} e^{i(\omega_{pq})}, \quad (3.F.9)$$

which leads to

$$\left(f_{pq}^{\text{FA}}\right)^* \left(\mathcal{A}_{p^\dagger q}^{\alpha_{pq}^{\text{FA}}}\right)^* = f_{qp}^{\text{FA}} \mathcal{A}_{q^\dagger p}^{\alpha_{pq}^{\text{FA}}}. \quad (3.F.10)$$

3.F.2 Expectation values belonging to quartic terms

Now, we consider quadratic terms of the form $h_{pqrs}^{\text{FA}} \mathcal{A}_{p^\dagger q^\dagger rs}^{\beta_{pqrs}^{\text{FA}}}$ appearing in Eq.(3.3.29). Using Eqs. (3.F.2) and (3.F.6), we get

$$\begin{aligned} \left(h_{pqrs}^{\text{FA}}\right)^* \left(\mathcal{A}_{p^\dagger q^\dagger rs}^{\beta_{pqrs}^{\text{FA}}}\right)^* &= e^{i(\beta_{pqrs}^{\text{FA}}(s) + \beta_{pqrs}^{\text{FA}}(r) - \beta_{pqrs}^{\text{FA}}(q) - \beta_{pqrs}^{\text{FA}}(p))} \left\langle e^{-i \sum_j \beta_{pqrs}^{\text{FA}}(j) n_j} c_s^\dagger c_r^\dagger c_q c_p \right\rangle \\ &= h_{srqp}^{\text{FA}} \mathcal{A}_{s^\dagger r^\dagger qp}^{\beta_{pqrs}^{\text{FA}}}. \end{aligned} \quad (3.F.11)$$

Section 3.G

DERIVATIVES OF \mathcal{A} AND \mathcal{G} W.R.T. THE COVARIANCE MATRIX Γ_m AND VARIATIONAL PARAMETERS ω

Throughout this section, we will again ignore the superscript $\tilde{\alpha}$, since we are only looking at a single expression. The purpose of the following subsections is to present the matrix derivatives of \mathcal{A} and \mathcal{G} w.r.t. all variational parameters, as they are useful for the VM, HITGD, and other approaches.

3.G.1 Computing the gradient of a Pfaffian

We compute the gradient of the Pfaffian of a $(2n \times 2n)$ -non-singular, skew-symmetric Matrix A and use that that the square of the Pfaffian yields the determinant of the respective matrix,

$$\text{Pf}(A)^2 = \det(A). \tag{3.G.1}$$

Since the Pfaffian is just a number, we can compute the gradient of the above equation to be

$$2\text{Pf}(A) \frac{d}{dA_{ij}} \text{Pf}(A) = \frac{d}{dA_{ij}} \det(A) = \det(A) \text{tr} \left[A^{-1} \frac{d}{dA_{ij}} A \right]. \tag{3.G.2}$$

We consider the trace expression,

$$\text{tr} \left[A^{-1} \frac{d}{dA_{ij}} A \right] = \sum_{p,q} A_{pq}^{-1} \left(\frac{d}{dA_{ij}} A \right)_{qp} = \sum_{p,q} (A^{-1})_{pq} \delta_{qi} \delta_{pj} = (A^{-1})_{ji}. \tag{3.G.3}$$

Inserting Eq. (3.G.3) into Eq. (3.G.2) and using Eq. (3.G.1) gives

$$\frac{d}{dA_{ij}} \text{Pf}(A) = \frac{1}{2} \text{Pf}(A) (A^{-1})_{ji}. \tag{3.G.4}$$

Since the matrix Γ_F as defined in Eq. (3.4.62)—which is the argument of the Pfaffian in \mathcal{A} —is anti-symmetric, we can use Eq. (3.G.4).

3.G.2 Derivative of \mathcal{A} with respect to the covariance matrix Γ_m

The matrix-derivative on the inverse of a square matrix Y is given by

$$\frac{\partial Y^{-1}}{\partial X_{ij}} = -Y^{-1} \frac{\partial Y}{\partial X_{ij}} Y^{-1}, \tag{3.G.5}$$

which follows from $YY^{-1} = \mathbf{1}$. We are interested in the derivative

$$\frac{d\mathcal{A}}{(d\Gamma_m)_{ij}} = s_{N_f} \left(\frac{1}{2}\right)^{N_f} \frac{d}{(d\Gamma_m)_{ij}} \text{Pf}(\Gamma_F), \quad (3.G.6)$$

where the derivative of the Pfaffian can be computed using the results from Appendix 3.G.1,

$$\begin{aligned} \frac{d\text{Pf}(\Gamma_F)}{d(\Gamma_m)_{ij}} &= \sum_{l,m=1}^{2N_f} \left(\frac{d\text{Pf}(\Gamma_F)}{d(\Gamma_F)_{lm}} \right) \left(\frac{d(\Gamma_F)_{lm}}{d(\Gamma_m)_{ij}} \right) \\ &= \frac{1}{2} \text{Pf}(\Gamma_F) (\Gamma_F^{-1})_{ji} \sqrt{1 - e^{i\tilde{\alpha}_{ii}}} \sqrt{1 - e^{i\tilde{\alpha}_{jj}}}, \end{aligned} \quad (3.G.7)$$

which leads to

$$\frac{d\mathcal{A}}{d(\Gamma_m)_{ij}} = \frac{1}{2} \mathcal{A} (\Gamma_F^{-1})_{ji} \sqrt{1 - e^{i\tilde{\alpha}_{ii}}} \sqrt{1 - e^{i\tilde{\alpha}_{jj}}}. \quad (3.G.8)$$

We note that since $\Gamma_F^T = -\Gamma_F$ and—using its definition in Eq. (3.6.21)—we can rewrite Eq. (3.G.8) as

$$\frac{d\mathcal{A}}{d(\Gamma_m)_{ij}} = \mathcal{F}_{i,j} \mathcal{A}, \quad (3.G.9)$$

which leads to following anti-symmetry

$$\frac{d\mathcal{A}}{d(\Gamma_m)_{ji}} = - \frac{d\mathcal{A}}{d(\Gamma_m)_{ij}}, \quad (3.G.10)$$

a property that will be of particular interest for Appendix 3.L.

3.G.3 Derivative of \mathcal{G} with respect to the covariance matrix Γ_m

In the following, we simplify notation by defining

$$\mathcal{D} = \mathbf{1}_{2N_f} + \frac{1}{2} (1 - e^{i\tilde{\alpha}}) (\Upsilon \Gamma_m - \mathbf{1}_{2N_f}), \quad (3.G.11)$$

which is the denominator \mathcal{G} . By using the product rule of differentiation, we get

$$\left(\frac{d}{d(\Gamma_m)_{ij}} \mathcal{G} \right)_{kl} = \left(\left[\frac{d}{d(\Gamma_m)_{ij}} (\Gamma_m + \Upsilon) \right] \mathcal{D}^{-1} \right)_{kl} + \left((\Gamma_m + \Upsilon) \frac{d}{d(\Gamma_m)_{ij}} \mathcal{D}^{-1} \right)_{kl}. \quad (3.G.12)$$

The first term on the right-hand side of Eq. (3.G.12) gives

$$\left(\left[\frac{d}{d(\Gamma_m)_{ij}} (\Gamma_m + \Upsilon) \right] \mathcal{D}^{-1} \right)_{kl} = \delta_{ik} (\mathcal{D}^{-1})_{jl}, \quad (3.G.13)$$

while the second term can be computed using Eq. (3.G.5),

$$\left(\frac{d\mathcal{D}^{-1}}{d(\Gamma_m)_{ij}} \right)_{pl} = - \frac{1}{2} (\mathcal{D}^{-1} (1 - e^{i\tilde{\alpha}}) \Upsilon)_{pi} (\mathcal{D}^{-1})_{jl}. \quad (3.G.14)$$

Inserting Eqs. (3.G.13) and (3.G.14) into Eq. (3.G.12) gives

$$\left(\frac{d\mathcal{G}}{d(\Gamma_m)_{ij}} \right)_{kl} = \delta_{ik} (\mathcal{D}^{-1})_{jl} - \frac{1}{2} (\mathcal{G} (1 - e^{i\tilde{\alpha}}) \Upsilon)_{ki} (\mathcal{D}^{-1})_{jl}. \quad (3.G.15)$$

By using the definition of \mathcal{L} in Eq.(3.6.20) and

$$\mathcal{R} = \mathcal{D}^{-1} = \left(\mathbb{1}_{2N_f} + \frac{1}{2} (1 - e^{i\tilde{\alpha}}) (\Upsilon \Gamma_m - \mathbb{1}_{2N_f}) \right)^{-1}, \quad (3.G.16)$$

which depends on the variational parameters ξ and ω , we can simplify Eq. (3.G.15) to

$$\left(\frac{d\mathcal{G}}{d(\Gamma_m)_{ij}} \right)_{kl} = \mathcal{L}_{ki} \mathcal{R}_{jl}. \quad (3.G.17)$$

Thus, we have

$$\frac{d\mathcal{G}}{d(\Gamma_m)_{ij}} = \mathcal{L} |i\rangle \langle j| \mathcal{R}. \quad (3.G.18)$$

Note that these results can also be obtained by using the relation $d\Gamma_m / (d\Gamma_m)_{kl} = e_k e_l^T$, where $e_k = (0, \dots, 1_k, \dots, 0_{2N_f})^T$.

Using the definitions of Eqs. (3.6.20) and (3.G.16), Eq. (3.E.20) and Eq. (3.4.69), we will derive an important identity relating \mathcal{L} with \mathcal{R} . First, we compute the transpose of the latter, resulting in

$$\mathcal{R}^T = - \left(\frac{1}{2} (1 - e^{i\tilde{\alpha}}) \right)^{-1} \Upsilon \Gamma_H^{-1}. \quad (3.G.19)$$

We then rewrite

$$\mathcal{L} = (\Gamma_H - \Gamma_m - \Upsilon) \Gamma_H^{-1} = - \left(\frac{1 + e^{i\tilde{\alpha}}}{1 - e^{i\tilde{\alpha}}} + 1 \right) \Upsilon \Gamma_H^{-1}, \quad (3.G.20)$$

which by comparison with Eq. (3.G.19) leads to

$$\mathcal{L} = \mathcal{R}^T. \quad (3.G.21)$$

The above identity is motivation for rewriting the matrix derivative of the matrix \mathcal{G} w.r.t. the CM Γ_m that was presented in Appendix 3.G.3. By using the skew-symmetry of the CM and understanding the matrix $\mathcal{G} = \mathcal{G}(\Gamma_m)$ in Eq. (3.4.69) as a function of the CM Γ_m (and of course of the parameters $\tilde{\alpha}$, which are however irrelevant in the following consideration), the following identity holds,

$$\mathcal{G}(\Gamma_m) = \frac{1}{2} \left(\mathcal{G}(\Gamma_m) + \mathcal{G}(-\Gamma_m^T) \right). \quad (3.G.22)$$

From Eq. (3.G.18) and due the property of Eq. (3.G.21), we know that the derivative of the first term on the right-hand side of Eq. (3.G.22) is given by

$$\frac{d\mathcal{G}(\Gamma_m)_{kl}}{d(\Gamma_m)_{ij}} = \mathcal{L}_{ki} \mathcal{L}_{lj}. \quad (3.G.23)$$

Similarly, one can show that the derivative of the second term on the right-hand side of Eq. (3.G.22) is given by

$$\frac{d\mathcal{G}(-\Gamma_m^T)_{kl}}{d(\Gamma_m)_{ij}} = - \mathcal{L}_{li} \mathcal{L}_{kj}. \quad (3.G.24)$$

Inserting Eqs. (3.G.23)-(3.G.24) in combination with Eq. (3.G.22) leads to

$$\frac{d\mathcal{G}(\Gamma_m)_{kl}}{d(\Gamma_m)_{ij}} = \frac{1}{2} (\mathcal{L}_{ki} \mathcal{L}_{l,j} - \mathcal{L}_{li} \mathcal{L}_{kj}), \quad (3.G.25)$$

or, in matrix representation

$$\frac{d\mathcal{G}(\Gamma_m)}{d(\Gamma_m)_{ij}} = \frac{1}{2} \mathcal{L} (|i\rangle \langle j| - |j\rangle \langle i|) \mathcal{L}^T. \quad (3.G.26)$$

From the anti-symmetry inherent to Eq. (3.G.26), it follows that

$$\frac{d\mathcal{G}(\Gamma_m)}{d(\Gamma_m)_{ji}} = - \frac{d\mathcal{G}(\Gamma_m)}{d(\Gamma_m)_{ij}}. \quad (3.G.27)$$

We note that taking the representation on the right-hand side of Eq. (3.G.22) is necessary, since we are dealing with the derivative of a structured matrix. Alternatively one could also take the left-hand side of Eq. (3.G.22), but would then have to use a differential operator which takes into account the structure of the respective matrix.

3.G.4 Derivative of \mathcal{A} with respect to the NGS variational parameters ω

We are interested in the derivative

$$\frac{d\mathcal{A}}{d\omega_{ij}} = s_{N_f} \left(\frac{1}{2}\right)^{N_f} \frac{d\text{Pf}(\Gamma_F)}{d\omega_{ij}}. \quad (3.G.28)$$

which leads to

$$\frac{d\text{Pf}(\Gamma_F)}{d\omega_{ij}} = \sum_{l,m=1}^{2N_f} \left(\frac{d\text{Pf}(\Gamma_F)}{d(\Gamma_F)_{lm}} \right) \left(\frac{d(\Gamma_F)_{lm}}{d\omega_{ij}} \right). \quad (3.G.29)$$

As mentioned in the main text, the explicit form of the parameters $\tilde{\alpha}(j)$ depends on whether the term at hand belongs to one- or two-body terms in Eq. (3.3.29). For the one-body (two-body) terms H_1 (H_2), the coefficients $\tilde{\alpha}(j)$ are given by Eq. (3.3.24) (Eq. (3.3.25)), i.e. α_{pq}^{FA} (β_{pqrs}^{FA}). We first look at the last term in Eq. (3.G.29),

$$\frac{d(\Gamma_F)_{lm}}{d\omega_{ij}} = \sum_{k=1}^{N_f} \frac{d(\Gamma_F)_{lm}}{d\tilde{\alpha}(k)} \frac{d\tilde{\alpha}(k)}{d\omega_{ij}}, \quad (3.G.30)$$

where

$$\begin{aligned} \frac{d(\Gamma_F)_{lm}}{d\tilde{\alpha}(k)} &= - \frac{ie^{i\tilde{\alpha}(k)}}{2\sqrt{1-e^{i\tilde{\alpha}(k)}}} (\Gamma_m)_{l,m} \left[(\delta_{l,k} + \delta_{l,k+N_f}) \sqrt{1-e^{i\tilde{\alpha}(m)}} \right. \\ &\quad \left. + (\delta_{m,k} + \delta_{m,k+N_f}) \sqrt{1-e^{i\tilde{\alpha}(l)}} \right] - i (\sigma \otimes e^{i\tilde{\alpha}})_{lm} \end{aligned} \quad (3.G.31)$$

and we defined

$$e^{i\tilde{\alpha}} = \begin{pmatrix} e^{i\tilde{\alpha}(1)} & & \\ & \ddots & \\ & & e^{i\tilde{\alpha}(N_f)} \end{pmatrix}. \quad (3.G.32)$$

Turning now to the second term in Eq. (3.G.30) using Eqs. (3.3.24) and (3.3.25),

$$\frac{d\tilde{\alpha}(k)}{d\omega_{ij}} = \begin{cases} \delta_{k,i}\delta_{q,j} - \delta_{k,i}\delta_{p,j}, & (H_1) \\ \delta_{k,i}\delta_{r,j} + \delta_{k,i}\delta_{s,j} - \delta_{k,i}\delta_{p,j} - \delta_{k,i}\delta_{q,j}, & (H_2) \end{cases} \quad (3.G.33)$$

where (H_1) refers to the case $\tilde{\alpha}(k) = \alpha_{pq}^{\text{FA}}(k)$ and (H_2) refers to $\tilde{\alpha}(k) = \beta_{pqrs}^{\text{FA}}(k)$. The first term on the right-hand side of Eq. (3.G.29) is given by the findings of Section 3.G.1,

$$\frac{d\text{Pf}(\Gamma_F)}{d(\Gamma_F)_{lm}} = \frac{1}{2}\text{Pf}(\Gamma_F) (\Gamma_F^{-1})_{ml}. \quad (3.G.34)$$

Eqs. (3.G.31), (3.G.33) and (3.G.34) give the derivative of \mathcal{A} w.r.t. the variational parameters ω_{ij} of Eq. (3.G.28).

3.G.5 Derivative of \mathcal{G} with respect to the NGS variational parameters ω_{ij}

In this subsection, we will compute

$$\left(\frac{d\mathcal{G}}{d\omega_{ij}}\right)_{kl} = \sum_{m=1}^{2N_f} (\Gamma_m + \Upsilon)_{km} \left(\frac{d}{d\omega_{ij}}\mathcal{D}^{-1}\right)_{ml}, \quad (3.G.35)$$

where \mathcal{D} was defined in Eq. (3.G.11) and the derivative of its inverse can be computed from Eq. (3.G.5),

$$\left(\frac{d\mathcal{D}^{-1}}{d\omega_{ij}}\right)_{ml} = - \sum_{v,w=1}^{2N_f} (\mathcal{D}^{-1})_{mv} \left(\frac{d\mathcal{D}}{d\omega_{ij}}\right)_{vw} (\mathcal{D}^{-1})_{wl}, \quad (3.G.36)$$

where

$$\left(\frac{d\mathcal{D}}{d\omega_{ij}}\right)_{vw} = \sum_{n=1}^{N_f} \left(\frac{d\mathcal{D}}{d\tilde{\alpha}(n)}\right)_{vn} \frac{d\tilde{\alpha}(n)}{d\omega_{ij}}. \quad (3.G.37)$$

We compute

$$\left(\frac{d\mathcal{D}}{d\tilde{\alpha}(n)}\right)_{vw} = -\frac{i}{2}e^{i\tilde{\alpha}(n)} (\delta_{n,v} + \delta_{n+N_f,v}) (\Upsilon\Gamma_m - \mathbb{1}_{2N_f})_{vw}. \quad (3.G.38)$$

Inserting Eqs.(3.G.33) and (3.G.38) into Eq. (3.G.37) leads to two expressions, depending on whether case (H_1) or (H_2) applies. For terms belonging to case (H_1) , we have

$$\left(\frac{d\mathcal{D}}{d\omega_{ij}}\right)_{vw} = -\frac{i}{2} \sum_{n=1}^{N_f} e^{i\tilde{\alpha}(n)} (\delta_{n,v} + \delta_{n+N_f,v}) (\Upsilon\Gamma_m - \mathbb{1}_{2N_f})_{vw} (\delta_{n,i}\delta_{q,j} - \delta_{n,i}\delta_{p,j}), \quad (3.G.39)$$

We obtain a similar expression for case (H_2) . Inserting Eqs. (3.G.37)-(3.G.39) into Eq. (3.G.36) and using Eq. (3.G.33), we obtain

$$\begin{aligned} & \left(\frac{d\mathcal{D}^{-1}}{d\omega_{ij}}\right)_{ml} \\ &= \begin{cases} \frac{i}{2}e^{i\tilde{\alpha}(i)} \left[(D^{-1})_{m,i} ((\Upsilon\Gamma_m - \mathbb{1})\mathcal{D}^{-1})_{i,l} + (D^{-1})_{m,i+N_f} ((\Upsilon\Gamma_m - \mathbb{1})\mathcal{D}^{-1})_{i+N_f,l} \right] (\delta_{q,j} - \delta_{p,j}) \\ \text{for case } (H_1), \\ \frac{i}{2}e^{i\tilde{\alpha}(i)} \left[(D^{-1})_{m,i} ((\Upsilon\Gamma_m - \mathbb{1})\mathcal{D}^{-1})_{i,l} + (D^{-1})_{m,i+N_f} ((\Upsilon\Gamma_m - \mathbb{1})\mathcal{D}^{-1})_{i+N_f,l} \right] \\ \times (\delta_{r,j} + \delta_{s,j} - \delta_{p,j} - \delta_{q,j}) \text{ for case } (H_2). \end{cases} \end{aligned} \quad (3.G.40)$$

Inserting the results of Eq. (3.G.40) into Eq. (3.G.35) leads to the following expression for the derivative of \mathcal{G} w.r.t. the variational parameters ω ,

$$\begin{aligned} \left(\frac{d\mathcal{G}}{d\omega_{ij}} \right)_{kl} &= \frac{i}{2} e^{i\tilde{\alpha}_i} \left(\mathcal{G}_{k,i} (\Upsilon\mathcal{G})_{i,l} + \mathcal{G}_{k,i+N_f} (\Upsilon\mathcal{G})_{i+N_f,l} \right) \\ &\times \begin{cases} (\delta_{q,j} - \delta_{p,j}) & \text{for case } (H_1), \\ (\delta_{r,j} + \delta_{s,j} - \delta_{p,j} - \delta_{q,j}) & \text{for case } (H_2), \end{cases} \end{aligned} \quad (3.G.41)$$

where we used the definition of \mathcal{G} and the fact that $\Upsilon^2 = -\mathbf{1}$.

Section 3.H

NORMAL-ORDER EXPANSION OF A PRODUCT OF MAJORANA OPERATORS

A sum of products of an even numbers of Majorana operators are elements of the Clifford algebra [3]. Let Ξ denote such an (otherwise arbitrary) operator. Following the convention in [41], we define the $(2N_f \times 2N_f)$ -matrix

$$(\Xi_m)_{ij} = 4 \frac{d \langle \Xi \rangle_{\text{GS}}}{d(\Gamma_m)_{ij}}. \quad (3.H.1)$$

By the arguments of Appendix 3.B and more specifically, by Eq. (3.B.12), we can perform a normal-ordered expansion of Ξ to quadratic order in the Majorana operators through

$$U_{\text{GS}}^\dagger \Xi U_{\text{GS}} = \langle \Xi \rangle_{\text{GS}} + \frac{i}{4} : A^T U_m^T \Xi_m U_m A : + \delta \Xi, \quad (3.H.2)$$

where $\delta \Xi$ denotes polynomials of Majorana operators of order four or higher. This normal-ordered expansion will be used for the derivation of the equations of motion of the CM in the following sections.

Section 3.I

COMPUTATIONS FOR EQUATIONS OF MOTION UNDER IMAGINARY-TIME EVOLUTION

This section contains calculations needed to derive the equations of motion of the CM under imaginary-time evolution and is necessary for the ensuing Section 3.N.

3.I.1 Moving the non-Gaussian operator to the left-hand side

In the following, we will pursue the strategy to move the non-Gaussian operator U_{FA} all the way to the left-hand side, more specifically, we write

$$d_\tau |\Psi_{\text{NGS}}\rangle = U_{\text{FA}} \left((d_\tau U_{\text{GS}}) U_{\text{GS}}^{-1} + U_{\text{FA}}^{-1} d_\tau U_{\text{FA}} \right) U_{\text{GS}} |0\rangle = U_{\text{FA}} (u_{\text{L}} + O) |\Psi_{\text{GS}}\rangle, \quad (3.I.1)$$

where we defined

$$u_{\text{L}} = (d_\tau U_{\text{GS}}) U_{\text{GS}}^{-1} \quad (3.I.2)$$

$$O = U_{\text{FA}}^{-1} d_\tau U_{\text{FA}}. \quad (3.I.3)$$

We will derive explicit expressions for Eqs. (3.I.2) and (3.I.3) in Sections 3.I.1 and 3.I.1.

Computing the operator u_{L}

Independently of the form of the NGS Ansatz used, one can compute the operator u_{L} as defined in Eq. (3.I.2), by using the following identity that holds for any operator $J(\tau)$ [228]

$$\begin{aligned} d_\tau e^{J(\tau)} &= \int_0^1 du e^{uJ(\tau)} (d_\tau J(\tau)) e^{(1-u)J(\tau)} \\ &= \int_0^1 du e^{(1-u)J(\tau)} (d_\tau J(\tau)) e^{uJ(\tau)}. \end{aligned} \quad (3.I.4)$$

We compute the time derivative of the Gaussian unitary operator

$$\begin{aligned} d_\tau U_{\text{GS}} &= \frac{i}{4} \int_0^1 du e^{\frac{i}{4} u A^T \xi A} A^T (d_\tau \xi) A e^{\frac{i}{4} (1-u) A^T \xi A} \\ &= \frac{i}{4} \int_0^1 du e^{\frac{i}{4} u A^T \xi A} A^T (d_\tau \xi) A e^{-\frac{i}{4} u A^T \xi A} U_{\text{GS}}. \end{aligned} \quad (3.I.5)$$

Similarly to Eq. (3.3.7), the following relation holds for Majorana operators

$$e^{\frac{i}{4} u A^T \xi A} A_j e^{-\frac{i}{4} u A^T \xi A} = \sum_{k=1}^{2N_f} \left(e^{-iu\xi} \right)_{jk} A_k = \left(e^{-iu\xi} A \right)_j. \quad (3.I.6)$$

Inserting Eq. (3.I.6) into Eq. (3.I.5) and using the anti-symmetry of the matrix ξ yields

$$d_\tau U_{\text{GS}} = \frac{1}{4} A^T (d_\tau U_m) U_m^T A U_{\text{GS}}, \quad (3.I.7)$$

as detailed in Appendix 3.P.4. We can insert the result of Eq. (3.I.7) back into the expression for u_{L} in Eq. (3.I.2), where the unitary Gaussian operators cancel to give

$$u_{\text{L}} = \frac{1}{4} A^T (d_\tau U_m) U_m^T A. \quad (3.I.8)$$

Computing the operator O

We turn to the second operator in Eq. (3.I.3), which will depend on the specific form of the non-Gaussian Ansatz chosen¹⁰. We will follow the same strategy as in Appendix 3.I.1, moving the operator to the left-hand side,

$$d_\tau U_{\text{FA}} = \frac{i}{2} \sum_{i,j=1}^{N_f} (d_\tau \omega_{ij}) : n_i^f n_j^f : U_{\text{FA}} = U_{\text{FA}} \frac{i}{2} \sum_{i,j=1}^{N_f} (d_\tau \omega_{ij}) : n_i^f n_j^f : . \quad (3.I.9)$$

By inserting Eq. (3.I.9) into Eq. (3.I.3), we observe that the two unitary operators cancel, resulting in

$$O = \frac{i}{2} \sum_{i,j=1}^{N_f} (d_\tau \omega_{ij}) : n_i^f n_j^f : . \quad (3.I.10)$$

3.I.2 Compute the Gaussian transformation of u_L and O

We start by rewriting the left-hand side of Eq. (3.I.1) to explicitly include the Gaussian unitary,

$$d_\tau |\Psi_{\text{NGS}}\rangle = U_{\text{FA}} (u_L + O) U_{\text{GS}} |0\rangle . \quad (3.I.11)$$

The objective of this section is to move the Gaussian unitary U_{GS} right next to the entangling unitary U_{FA} , such that

$$d_\tau |\Psi_{\text{NGS}}\rangle = U_{\text{FA}} U_{\text{GS}} U_L |0\rangle , \quad (3.I.12)$$

with an operator U_L that acts on the fermionic vacuum state $|0\rangle$, given by

$$U_L = U_{\text{GS}}^\dagger (u_L + O) U_{\text{GS}} . \quad (3.I.13)$$

The expressions for u_L and O for the NGS Ansatz are given by Eqs. (3.I.8) and (3.I.10). In the next two subsections, we will compute the two terms appearing in Eq. (3.I.13) to get an expression for U_L . We will make extensive use of the normal-ordered expansion of Gaussian-transformed operators of Eq. (3.H.2) in both, Appendices 3.I.2 and 3.I.2.

Computing the Gaussian-transformed u_L

We express the first term on the right-hand side of Eq. (3.I.13) in normal order using Eq. (3.H.2) as

$$U_{\text{GS}}^\dagger u_L U_{\text{GS}} = \langle u_L \rangle_{\text{GS}} + \frac{i}{4} : A^T U_m^T u_{Lm} U_m A : , \quad (3.I.14)$$

where higher order corrections δu_L do not appear since u_L only contains quadratic terms. The expression u_{Lm} can be obtained by replacing Ξ in Eq. (3.H.1) with u_L . We can compute the expectation value w.r.t. the Gaussian state using the definition of the CM in Eq. (3.3.8), giving

$$\langle u_L \rangle_{\text{GS}} = \frac{i}{4} \text{tr} \left[(d_\tau U_m) U_m^T \Gamma_m \right] , \quad (3.I.15)$$

¹⁰While in this work the non-Gaussian Ansatz is simply U_{FA} , a general Ansatz such as one including an additional Gaussian unitary operator will lead to additional terms.

see Appendix 3.P.5. We compute the i -th row and j -th column entry of the matrix u_{Lm} by first writing the trace operation as a sum and then taking the derivative of Eq. (3.I.15) w.r.t. the CM,

$$(u_{Lm})_{ij} = i \left(\frac{d}{d\Gamma_m} \sum_{p,q,r=1}^{2N_f} (d_\tau U_m)_{pq} (U_m^T)_{qr} (\Gamma_m)_{rp} \right)_{ij} = i \sum_{q=1}^{2N_f} (d_\tau U_m)_{jq} (U_m^T)_{qi}, \quad (3.I.16)$$

which, using the orthogonality property of U_m , eventually leads to

$$U_{\text{GS}}^\dagger u_L U_{\text{GS}} = \frac{i}{4} \text{tr} [d_\tau U_m U_m^T \Gamma_m] + \frac{1}{4} : A^T U_m^T (d_\tau U_m) A : . \quad (3.I.17)$$

Computing the Gaussian-transformed u_L and U_L

We express the second term on the right-hand side of Eq. (3.I.13) in normal order using Eq. (3.H.2) as

$$U_{\text{GS}}^\dagger O U_{\text{GS}} = \langle O \rangle_{\text{GS}} + \frac{i}{4} : A^T U_m^T O_m U_m A : + \delta O. \quad (3.I.18)$$

The time-evolution of the non-Gaussian variational parameters, $d_\tau \omega_{kl}$, has to be computed in each time step $\Delta\tau$ using Eqs. (3.6.4) or (3.7.10). Therefore, we can write the operator u_L appearing in Eq. (3.I.13) as

$$U_L = \frac{i}{4} \text{tr} [(d_\tau U_m) U_m^T \Gamma_m] + \langle O \rangle_{\text{GS}} + \frac{1}{4} : A^T U_m^T ((d_\tau U_m) + i O_m U_m) A : + \delta O. \quad (3.I.19)$$

The specific form of O_m is given in Appendix 3.J

Section 3.J

DERIVATION OF THE MEAN-FIELD NGS MATRIX

The explicit expression of O_m in Eq.(3.I.19) can be obtained by replacing Ξ in Eq. (3.H.1) with O , more specifically,

$$(O_m)_{ij} = -2i \sum_{k,l=1}^{N_f} (d_\tau \omega_{kl}) \frac{d \langle c_k^\dagger c_l^\dagger c_k c_l \rangle_{\text{GS}}}{d(\Gamma_m)_{ij}}, \quad (3.J.1)$$

where the last expectation value can be expressed in terms of the CM Γ_m (or, alternatively Γ_f defined in Section 3.3.2) using Eq. (3.4.73). It is important to note, that we assume $k \neq l$ in the above sum, since we set the diagonal entries $\omega_{kk} = 0$ for $k = 1, \dots, N_f$. First, we compute

$$\frac{\langle c_k^\dagger c_l^\dagger c_k c_l \rangle_{\text{GS}}}{\mathcal{A}} = \left(\frac{\mathcal{A}_{k^\dagger l}}{\mathcal{A}} \frac{\mathcal{A}_{l^\dagger k}}{\mathcal{A}} - \frac{\mathcal{A}_{k^\dagger k}}{\mathcal{A}} \frac{\mathcal{A}_{l^\dagger l}}{\mathcal{A}} + \frac{\mathcal{A}_{k^\dagger l^\dagger}}{\mathcal{A}} \frac{\mathcal{A}_{kl}}{\mathcal{A}} \right) \quad (3.J.2)$$

as a special case of Eq. (3.4.73), where we have set $\tilde{\alpha}(k) = 0$ for all entries in the following (we will omit the phase factor entirely in what follows). This simplifies the expressions, since we can expression of Eq. (3.4.69) to

$$\mathcal{G} = \Gamma_m + \Upsilon \quad (3.J.3)$$

and of Eq. (3.4.62) to

$$\Gamma_F = -2\sigma \otimes \mathbf{1}_{N_f}. \quad (3.J.4)$$

In order to compute \mathcal{A} for $\tilde{\alpha}_k = 0$, we have to compute the Pfaffian of Eq. (3.J.4),

$$\begin{aligned} \text{Pf}(\Gamma_F) &= (-2i)^{N_f} \text{Pf}(-i\sigma \otimes \mathbf{1}_{N_f}) \\ &= (-2i)^{N_f} (-i)^{N_f^2} \end{aligned} \quad (3.J.5)$$

We will now compute \mathcal{A} of Eq. (3.4.68) for the two distinct cases, N_f being an even ($N_f = 2n$) or odd ($N_f = 2n + 1$) integer,

$$\mathcal{A} = s_{N_f} \left(\frac{1}{2}\right)^{N_f} \text{Pf}(\Gamma_F) = \begin{cases} (-1)^n \left(\frac{1}{2}\right)^{2n} (-2i)^{2n} (-i)^{4n^2} \\ (-1)^{\frac{N_f+1}{2}} \left(\frac{1}{2}\right)^{N_f} (-2i)^{N_f} (-i)^{N_f^2} \end{cases} = 1. \quad (3.J.6)$$

Since $\mathcal{G}^T = -\mathcal{G}$, Eq. (3.J.2) reduces to

$$\langle c_k^\dagger c_l^\dagger c_k c_l \rangle_{\text{GS}} = -\frac{1}{16} (\mathcal{G}_{\mathbf{k}^\dagger \mathbf{l}} \mathcal{G}_{\mathbf{l}^\dagger \mathbf{k}} - \mathcal{G}_{\mathbf{k}^\dagger \mathbf{k}} \mathcal{G}_{\mathbf{l}^\dagger \mathbf{l}} + \mathcal{G}_{\mathbf{k}^\dagger \mathbf{l}^\dagger} \mathcal{G}_{\mathbf{k} \mathbf{l}}). \quad (3.J.7)$$

Using Eq. (3.G.26) and noting that here, $\mathcal{L} = \mathbf{1}_{2N_f}$ (since $\tilde{\alpha}(k) = 0$), we insert Eq. (3.J.7) into Eq. (3.J.1), resulting in

$$\begin{aligned} (O_m)_{ij} = & i \frac{1}{16} \sum_{k,l=1}^{N_f} (d_\tau \omega_{kl}) [(|i\rangle \langle j| - |j\rangle \langle i|)_{\mathbf{k}\dagger\mathbf{l}} \mathcal{G}_{\mathbf{l}\dagger\mathbf{k}} + \mathcal{G}_{\mathbf{k}\dagger\mathbf{l}} (|i\rangle \langle j| - |j\rangle \langle i|)_{\mathbf{l}\dagger\mathbf{k}} \\ & - (|i\rangle \langle j| - |j\rangle \langle i|)_{\mathbf{k}\dagger\mathbf{k}} \mathcal{G}_{\mathbf{l}\dagger\mathbf{l}} - \mathcal{G}_{\mathbf{k}\dagger\mathbf{k}} (|i\rangle \langle j| - |j\rangle \langle i|)_{\mathbf{l}\dagger\mathbf{l}} \\ & + (|i\rangle \langle j| - |j\rangle \langle i|)_{\mathbf{k}\dagger\mathbf{l}} \mathcal{G}_{\mathbf{k}\mathbf{l}} + \mathcal{G}_{\mathbf{k}\dagger\mathbf{l}} (|i\rangle \langle j| - |j\rangle \langle i|)_{\mathbf{k}\mathbf{l}}]. \end{aligned} \quad (3.J.8)$$

The following parts of this appendix are dedicated to simplifying the above expression.

3.J.1 Second line simplification

The matrix \mathcal{G} has block-form which we denote as

$$\mathcal{G} = \begin{pmatrix} \mathcal{G}_{11} & \mathcal{G}_{12} \\ \mathcal{G}_{21} & \mathcal{G}_{22} \end{pmatrix}. \quad (3.J.9)$$

Due to the specific form of Eq. (3.J.3), we have $\mathcal{G}_{\mathbf{l}\dagger\mathbf{l}} = -2i\mathcal{G}_{l,N_f+l}$. Looking at the third and fourth terms of Eq. (3.J.8), we get

$$\begin{aligned} & - \frac{i}{16} \sum_{k,l=1}^{N_f} (d_\tau \omega)_{kl} [(|i\rangle \langle j| - |j\rangle \langle i|)_{\mathbf{k}\dagger\mathbf{k}} \mathcal{G}_{\mathbf{l}\dagger\mathbf{l}} + \mathcal{G}_{\mathbf{k}\dagger\mathbf{k}} (|i\rangle \langle j| - |j\rangle \langle i|)_{\mathbf{l}\dagger\mathbf{l}}] \\ = & \frac{i}{4} \sum_{k=1}^{N_f} \left([(d_\tau \omega)_{ik} + (d_\tau \omega)_{ki}] \delta_{j,N_f+i} - [(d_\tau \omega)_{jk} + (d_\tau \omega)_{kj}] \delta_{i,N_f+j} \right) (\mathcal{G}_{12})_{kk}. \end{aligned} \quad (3.J.10)$$

For a vector v of length N_f , we define $\text{diag}(v)$ to be a $(N_f \times N_f)$ -diagonal matrix, where $(\text{diag}(v))_{ii} = v_i$ and $(\text{diag}(v))_{ij} = 0$ for $i \neq j$. This allows us to recast Eq. (3.J.10) into a matrix form,

$$\begin{aligned} & \frac{i}{4} \sum_{k=1}^{N_f} \left([(d_\tau \omega)_{ik} + (d_\tau \omega)_{ki}] \delta_{j,N_f+i} - [(d_\tau \omega)_{jk} + (d_\tau \omega)_{kj}] \delta_{i,N_f+j} \right) (\mathcal{G}_{12})_{kk} \\ = & \frac{i}{4} \begin{pmatrix} \mathbf{0} & \text{diag}((d_\tau \omega + d_\tau \bar{\omega})g) \\ -\text{diag}((d_\tau \omega + d_\tau \bar{\omega})g) & \mathbf{0} \end{pmatrix}_{ij} \end{aligned} \quad (3.J.11)$$

We defined the column vector g containing the diagonal entries of the upper-right block matrix elements of \mathcal{G} ,

$$g = (\mathcal{G}_{1,N_f+1}, \mathcal{G}_{2,N_f+2}, \dots, \mathcal{G}_{N_f,2N_f})^T = ((\mathcal{G}_{12})_{11}, (\mathcal{G}_{12})_{22}, \dots, (\mathcal{G}_{12})_{N_f N_f})^T. \quad (3.J.12)$$

3.J.2 First and third line simplifications

For two real and anti-symmetric $(2N_f \times 2N_f)$ -matrices A and B , we have the following general property (again, using our notation introduced in Section 3.2 and letting $k, l = 1, \dots, N_f$)

$$\begin{aligned} & A_{\mathbf{k}\dagger\mathbf{l}} B_{\mathbf{l}\dagger\mathbf{k}} + A_{\mathbf{k}\dagger\mathbf{l}} B_{\mathbf{k}\mathbf{l}} + A_{\mathbf{l}\dagger\mathbf{k}} B_{\mathbf{k}\dagger\mathbf{l}} + A_{\mathbf{k}\mathbf{l}} B_{\mathbf{k}\dagger\mathbf{l}} \\ = & 4 \left(A_{k,l} B_{N_f+l, N_f+k} - A_{k, N_f+l} B_{l, N_f+k} - A_{N_f+k, l} B_{N_f+l, k} + A_{N_f+k, N_f+l} B_{l, k} \right). \end{aligned} \quad (3.J.13)$$

The above identity can be proven by multiplying out the products and using the anti-symmetry $\mathcal{G}^T = -\mathcal{G}$. The first and third line of Eq. (3.J.8) is given by

$$\begin{aligned} & \sum_{k,l=1}^{N_f} \frac{i(d_\tau\omega)_{kl}}{16} [(|i\rangle\langle j| - |j\rangle\langle i|)_{\mathbf{k}\dagger\mathbf{l}} \mathcal{G}_{\mathbf{l}\dagger\mathbf{k}} + \mathcal{G}_{\mathbf{k}\dagger\mathbf{l}} (|i\rangle\langle j| - |j\rangle\langle i|)_{\mathbf{l}\dagger\mathbf{k}} \\ & + (|i\rangle\langle j| - |j\rangle\langle i|)_{\mathbf{k}\dagger\mathbf{l}\dagger} \mathcal{G}_{\mathbf{k}\mathbf{l}} + \mathcal{G}_{\mathbf{k}\dagger\mathbf{l}\dagger} (|i\rangle\langle j| - |j\rangle\langle i|)_{\mathbf{k}\mathbf{l}}]. \end{aligned} \quad (3.J.14)$$

By substituting $A = |i\rangle\langle j| - |j\rangle\langle i|$ and $B = \mathcal{G}$ (where \mathcal{G} is here given by Eq. (3.J.3)—in the main text, this special case is denoted by \mathcal{G}^0), we can make use of Eq. (3.J.13) to write

$$\begin{aligned} & \sum_{k,l=1}^{N_f} \frac{i(d_\tau\omega)_{kl}}{16} [(|i\rangle\langle j| - |j\rangle\langle i|)_{\mathbf{k}\dagger\mathbf{l}} \mathcal{G}_{\mathbf{l}\dagger\mathbf{k}} + \mathcal{G}_{\mathbf{k}\dagger\mathbf{l}} (|i\rangle\langle j| - |j\rangle\langle i|)_{\mathbf{l}\dagger\mathbf{k}} \\ & + (|i\rangle\langle j| - |j\rangle\langle i|)_{\mathbf{k}\dagger\mathbf{l}\dagger} \mathcal{G}_{\mathbf{k}\mathbf{l}} + \mathcal{G}_{\mathbf{k}\dagger\mathbf{l}\dagger} (|i\rangle\langle j| - |j\rangle\langle i|)_{\mathbf{k}\mathbf{l}}] \\ = & \sum_{k,l=1}^{N_f} \frac{i(d_\tau\omega)_{kl}}{4} \left[(\delta_{i,k}\delta_{j,l} - \delta_{j,k}\delta_{i,l}) \mathcal{G}_{N_f+l, N_f+k} - (\delta_{i,k}\delta_{j, N_f+l} - \delta_{j,k}\delta_{i, N_f+l}) \mathcal{G}_{l, N_f+k} \right. \\ & \left. - (\delta_{i, N_f+k}\delta_{j,l} - \delta_{j, N_f+k}\delta_{i,l}) \mathcal{G}_{N_f+l, k} + (\delta_{i, N_f+k}\delta_{j, N_f+l} - \delta_{j, N_f+k}\delta_{i, N_f+l}) \mathcal{G}_{l, k} \right]. \end{aligned} \quad (3.J.15)$$

To simplify the above expression, we will consider the first and last term on the right-hand side of Eq. (3.J.15),

$$\begin{aligned} & \sum_{k,l=1}^{N_f} \frac{i(d_\tau\omega)_{kl}}{4} (\delta_{i,k}\delta_{j,l} - \delta_{j,k}\delta_{i,l}) \mathcal{G}_{N_f+l, N_f+k} \\ = & -\frac{i}{4} \left((d_\tau\omega)_{i,j} + (d_\tau\omega)_{j,i} \right) \mathcal{G}_{N_f+i, N_f+j}, \quad \text{with } i, j \in \{1, \dots, N_f\} \quad (3.J.16) \\ & \sum_{k,l=1}^{N_f} \frac{i(d_\tau\omega)_{kl}}{4} (\delta_{i, N_f+k}\delta_{j, N_f+l} - \delta_{j, N_f+k}\delta_{i, N_f+l}) \mathcal{G}_{l, k} \\ = & -\frac{i}{4} \left((d_\tau\omega)_{i-N_f, j-N_f} + (d_\tau\omega)_{j-N_f, i-N_f} \right) \mathcal{G}_{i-N_f, j-N_f}, \quad \text{with } i, j \in \{N_f+1, \dots, 2N_f\}. \end{aligned} \quad (3.J.17)$$

The two middle terms of Eq. (3.J.15) can be combined to give

$$\begin{aligned} & \sum_{k,l=1}^{N_f} \frac{-i(d_\tau\omega)_{kl}}{4} \left(\delta_{i,k}\delta_{j, N_f+l} \mathcal{G}_{l, N_f+k} - \delta_{j, N_f+k}\delta_{i,l} \mathcal{G}_{N_f+l, k} \right) \\ = & \frac{i}{4} \left((d_\tau\omega)_{i, j-N_f} + (d_\tau\omega)_{j-N_f, i} \right) \mathcal{G}_{N_f+i, j-N_f}, \quad \text{with } i \in \{1, \dots, N_f\}, j \in \{N_f+1, \dots, 2N_f\} \end{aligned} \quad (3.J.18)$$

$$\begin{aligned} & \sum_{k,l=1}^{N_f} \frac{i(d_\tau\omega)_{kl}}{4} \left(\delta_{j,k}\delta_{i, N_f+l} \mathcal{G}_{l, N_f+k} - \delta_{i, N_f+k}\delta_{j,l} \mathcal{G}_{N_f+l, k} \right) \\ = & \frac{i}{4} \left((d_\tau\omega)_{j, i-N_f} + (d_\tau\omega)_{i-N_f, j} \right) \mathcal{G}_{i-N_f, N_f+j}, \quad \text{with } i \in \{N_f+1, \dots, 2N_f\}, j \in \{1, \dots, N_f\}. \end{aligned} \quad (3.J.19)$$

3.J.3 Resulting expression

We define the $(2N_f \times 2N_f)$ -matrices

$$d_\tau \Omega = \begin{pmatrix} d_\tau \omega & d_\tau \omega \\ d_\tau \omega & d_\tau \omega \end{pmatrix}, \quad d_\tau \bar{\Omega} = \begin{pmatrix} d_\tau \bar{\omega} & d_\tau \bar{\omega} \\ d_\tau \bar{\omega} & d_\tau \bar{\omega} \end{pmatrix}, \quad (3.J.20)$$

where $d_\tau \bar{\omega}$ is defined in Eq. (3.6.34). Inserting Eqs. (3.J.16)-(3.J.19) and Eq. (3.J.11) into Eq. (3.J.8) leads to

$$(O_m)_{ij} = (O_m^I)_{ij} + (O_m^{II})_{ij}, \quad (3.J.21)$$

with

$$O_m^I = \frac{i}{4} \begin{pmatrix} \mathbb{0} & \text{diag}((d_\tau \omega + d_\tau \bar{\omega})g) \\ -\text{diag}((d_\tau \omega + d_\tau \bar{\omega})g) & \mathbb{0} \end{pmatrix} \quad (3.J.22)$$

$$O_m^{II} = \frac{i}{4} (d_\tau \Omega + d_\tau \bar{\Omega}) \odot \begin{pmatrix} -\mathcal{G}_{22} & \mathcal{G}_{21} \\ \mathcal{G}_{12} & -\mathcal{G}_{11} \end{pmatrix} \quad (3.J.23)$$

and where \odot denotes the Hadamard product. Note, that O_m remains anti-symmetric under transposition. We want to stress that Eq. (3.J.21) and all preceding equations in Appendix 3.J were derived using the particular form of Eq. (3.J.3), where we used $\mathcal{G}^T = -\mathcal{G}$. Inserting Eq. (3.J.3) into Eq. (3.J.21) leads to Eq. (3.6.33).

Section 3.K

ENSURING A MONOTONIC DECREASE OF THE ENERGY DURING THE HITGD EVOLUTION

We want to choose our Ansatz for the time evolution in such a way, that the second part of Eq. (3.6.39) is also non-positive, i.o.w.

$$\frac{1}{8} \text{tr} \left(O_m^2 \right) + \sum_{k,l=1}^{N_f} \frac{dE}{d\omega_{kl}} d_\tau \omega_{kl} \stackrel{!}{\leq} 0. \quad (3.K.1)$$

Note that this is a sufficient, but not a necessary condition in order for the energy to be decreasing. Even though the following calculations are rather cumbersome, the result will lead to an expression which will guarantee us that the energy will always monotonically decrease following a Hybrid Imaginary-Time-Gradient-Descent (HITGD) Ansatz. In order to achieve this, we have to evaluate the exact expression of the trace of the square of the mean-field NGS matrix as given by Eq. (3.J.21). The square of the mean-field NGS matrix is given by

$$\left(O_m^2 \right)_{ij} = \sum_{k=1}^{2N_f} \left(\left(O_m^I \right)_{ik} \left(O_m^I \right)_{kj} + \left(O_m^I \right)_{ik} \left(O_m^{II} \right)_{kj} + \left(O_m^{II} \right)_{ik} \left(O_m^I \right)_{kj} + \left(O_m^{II} \right)_{ik} \left(O_m^{II} \right)_{kj} \right). \quad (3.K.2)$$

Since we want to compute the trace of the squared matrix, $\text{tr}(O_m^2) = \sum_i (O_m^2)_{ii}$, we are only concerned with the diagonal terms of the above expression. We will compute the four terms appearing on the right-hand side of Eq. (3.K.2) individually. We define the column vectors

$$d_\tau \omega_j = \begin{pmatrix} (d_\tau \omega)_{j,1} \\ (d_\tau \omega)_{j,2} \\ \vdots \\ (d_\tau \omega)_{j,N_f} \end{pmatrix}, \quad \omega_j = \begin{pmatrix} \omega_{j,1} \\ \omega_{j,2} \\ \vdots \\ \omega_{j,N_f} \end{pmatrix} \quad (3.K.3)$$

$$d_\tau \bar{\omega}_j = \begin{pmatrix} (d_\tau \omega)_{1,j} \\ (d_\tau \omega)_{2,j} \\ \vdots \\ (d_\tau \omega)_{N_f,j} \end{pmatrix}, \quad \bar{\omega}_j = \begin{pmatrix} \omega_{1,j} \\ \omega_{2,j} \\ \vdots \\ \omega_{N_f,j} \end{pmatrix}. \quad (3.K.4)$$

The trace over the first expression in Eq. (3.K.2) is given by

$$\text{tr} \left(\left(O_m^I \right)^2 \right) = \frac{1}{8} \sum_{j=1}^{N_f} (d_\tau \omega_j + d_\tau \bar{\omega}_j)^T g g^T (d_\tau \omega_j + d_\tau \bar{\omega}_j). \quad (3.K.5)$$

Turning to the second term on the right-hand side of Eq. (3.K.2) we first note that O_m^I has block form and contains a diagonal matrix on each of the off-diagonal blocks. On the other hand, O_m^{II} is composed of the Hadamard product of two matrices, with the first

being $d_\tau\Omega$. By definition, $d_\tau\Omega$ is composed of four identical blocks, each one being an identical copy of the matrix $d_\tau\omega$, whose diagonal contains only zeros. Using the shorthand notation $(d_\tau\omega)_S = d_\tau\omega + d_\tau\bar{\omega}$ and $\tilde{D} = \text{diag}((d_\tau\omega)_S g)$, we compute the matrix product

$$O_m^I O_m^{II} = \frac{i^2}{16} \begin{pmatrix} \tilde{D}((d_\tau\omega)_S \odot \mathcal{G}_{12}) & -\tilde{D}((d_\tau\omega)_S \odot \mathcal{G}_{11}) \\ \tilde{D}((d_\tau\omega)_S \odot \mathcal{G}_{22}) & -\tilde{D}((d_\tau\omega)_S \odot \mathcal{G}_{21}) \end{pmatrix}. \quad (3.K.6)$$

Since we are interested in computing the trace of Eq. (3.K.6), we are only considering the upper-left and lower-right blocks. Each of these blocks contains the matrix product of a diagonal matrix \tilde{D} with the matrix that contains the Hadamard product of $(d_\tau\omega)_S$ with \mathcal{G}_{12} and $-\mathcal{G}_{21}$, respectively. The matrix product of some $(N \times N)$ -diagonal matrix \tilde{D} with a (possibly unstructured) $(N \times N)$ -matrix M is given by

$$\tilde{D}M = \begin{pmatrix} \tilde{D}_{11}M_{11} & \tilde{D}_{11}M_{12} & \dots & \tilde{D}_{11}M_{1N} \\ \tilde{D}_{22}M_{21} & \tilde{D}_{22}M_{22} & \dots & \tilde{D}_{22}M_{2N} \\ \vdots & \vdots & \ddots & \vdots \\ \tilde{D}_{NN}M_{N1} & \tilde{D}_{NN}M_{N2} & \dots & \tilde{D}_{NN}M_{NN} \end{pmatrix}. \quad (3.K.7)$$

In our case (i.e. Eq. (3.K.6)) the matrix M corresponds to the Hadamard product $(d_\tau\omega)_S \odot \mathcal{G}_{12}$ in one instance and $-(d_\tau\omega)_S \odot \mathcal{G}_{21}$ in the other instance. Since the diagonal elements of $(d_\tau\omega)_S$ are all 0 and due to the relation of Eq. (3.K.7), we have

$$\text{tr}(O_m^I O_m^{II}) = 0. \quad (3.K.8)$$

The cyclic property of the trace gives $\text{tr}(O_m^{II} O_m^I) = 0$ for the trace over the third term in Eq. (3.K.2).

A similar simplification is not possible for the last term. We can however simplify its computation by rewriting some of the expressions as matrices. More precisely, we define $(d_\tau\Omega)_S = d_\tau\Omega + d_\tau\bar{\Omega}$ and write

$$\text{tr}\left(\left(O_m^{II}\right)^2\right) = \frac{1}{16} \sum_{k,l=1}^{2N_f} \left(\left((d_\tau\Omega)_S \odot \begin{pmatrix} -\mathcal{G}_{22} & \mathcal{G}_{21} \\ \mathcal{G}_{12} & -\mathcal{G}_{11} \end{pmatrix} \right)_{kl} \right)^2. \quad (3.K.9)$$

The last line in the above equation can be interpreted as a scalar product of a vector with

itself. We simply reshape the $(2N_f \times 2N_f)$ -matrix into a vector of length $(2N_f)^2$ through

$$\begin{aligned}
 & (d_\tau \Omega)_S \odot \begin{pmatrix} -\mathcal{G}_{22} & \mathcal{G}_{21} \\ \mathcal{G}_{12} & -\mathcal{G}_{11} \end{pmatrix} \\
 &= \begin{pmatrix} \text{diag}((d_\tau \omega)_S) & & & \\ & \text{diag}((d_\tau \omega)_S) & & \\ & & \text{diag}((d_\tau \omega)_S) & \\ & & & \text{diag}((d_\tau \omega)_S) \end{pmatrix} \begin{pmatrix} -(\mathcal{G}_{22})_{1,1} \\ \vdots \\ -(\mathcal{G}_{22})_{N_f, N_f} \\ (\mathcal{G}_{21})_{1,1} \\ \vdots \\ (\mathcal{G}_{21})_{N_f, N_f} \\ (\mathcal{G}_{12})_{1,1} \\ \vdots \\ (\mathcal{G}_{12})_{N_f, N_f} \\ -(\mathcal{G}_{11})_{1,1} \\ \vdots \\ -(\mathcal{G}_{11})_{N_f, N_f} \end{pmatrix}. \tag{3.K.10}
 \end{aligned}$$

The expression in Eq. (3.K.9) is then identical to the scalar product of the vector as given on the right-hand side of Eq. (3.K.10) with itself,

$$\begin{aligned}
 & \text{tr} \left((O_m^{II})^2 \right) \\
 &= \frac{1}{16} \left(-(\mathcal{G}_{22})_{1,1}, \dots, -(\mathcal{G}_{22})_{N_f, N_f}, (\mathcal{G}_{21})_{1,1}, \dots, (\mathcal{G}_{21})_{N_f, N_f}, (\mathcal{G}_{12})_{1,1}, \dots, (\mathcal{G}_{12})_{N_f, N_f}, \right. \\
 & \quad \left. -(\mathcal{G}_{11})_{1,1}, \dots, -(\mathcal{G}_{11})_{N_f, N_f} \right) \\
 & \times \begin{pmatrix} \text{diag}((d_\tau \omega)_S)^2 & & & \\ & \text{diag}((d_\tau \omega)_S)^2 & & \\ & & \text{diag}((d_\tau \omega)_S)^2 & \\ & & & \text{diag}((d_\tau \omega)_S)^2 \end{pmatrix} \begin{pmatrix} -(\mathcal{G}_{22})_{1,1} \\ \vdots \\ -(\mathcal{G}_{22})_{N_f, N_f} \\ (\mathcal{G}_{21})_{1,1} \\ \vdots \\ (\mathcal{G}_{21})_{N_f, N_f} \\ (\mathcal{G}_{12})_{1,1} \\ \vdots \\ (\mathcal{G}_{12})_{N_f, N_f} \\ -(\mathcal{G}_{11})_{1,1} \\ \vdots \\ -(\mathcal{G}_{11})_{N_f, N_f} \end{pmatrix}, \tag{3.K.11}
 \end{aligned}$$

which can be condensed to the following simple quadratic form,

$$\text{tr} \left((O_m^{II})^2 \right) = \frac{1}{8} \sum_{j=1}^{N_f} (d_\tau \omega_j + d_\tau \bar{\omega}_j)^T D_j (d_\tau \omega_j + d_\tau \bar{\omega}_j). \tag{3.K.12}$$

Here, D_j are diagonal matrices defined as

$$D_j = \begin{pmatrix} D_{j,1} & & & \\ & D_{j,2} & & \\ & & \ddots & \\ & & & D_{j,N_f} \end{pmatrix} \quad (3.K.13)$$

with diagonal entries

$$D_{jk} = \frac{1}{2} \left((\mathcal{G}_{11})_{jk}^2 + (\mathcal{G}_{12})_{jk}^2 + (\mathcal{G}_{21})_{jk}^2 + (\mathcal{G}_{22})_{jk}^2 \right). \quad (3.K.14)$$

Combining Eqs. (3.K.5),(3.K.8) and (3.K.14), we find the following expression for the trace of the square of the NGS mean-field matrix,

$$\text{tr} \left(O_m^2 \right) = \frac{1}{8} \sum_{j=1}^{N_f} (d_\tau \omega_j + d_\tau \bar{\omega}_j)^T (gg^T + D_j) (d_\tau \omega_j + d_\tau \bar{\omega}_j). \quad (3.K.15)$$

We define the vector of the energy gradient

$$\frac{dE}{d\omega_j} = \begin{pmatrix} \frac{dE}{d\omega_{j,1}} \\ \frac{dE}{d\omega_{j,2}} \\ \vdots \\ \frac{dE}{d\omega_{j,N_f}} \end{pmatrix}. \quad (3.K.16)$$

From the form of Eq. (3.7.12) we know that the matrix containing the energy gradient is symmetric, i.e. $dE/d\omega_{ij} = dE/d\omega_{ji}$. We can use this property to rewrite the second term of the left-hand side of Eq. (3.K.1) as

$$\sum_{j,k=1}^{N_f} \frac{dE}{d\omega_{jk}} (d_\tau \omega)_{jk} = \frac{1}{2} \sum_{j=1}^{N_f} \left(\frac{dE}{d\omega_j} \right)^T (d_\tau \omega_j + d_\tau \bar{\omega}_j). \quad (3.K.17)$$

Therefore, the left-hand side of Eq. (3.K.1) can be written as

$$\begin{aligned} & \frac{1}{8} \text{tr} \left(O_m^2 \right) + \sum_{j,k=1}^{N_f} \frac{dE}{d\omega_{jk}} d_\tau \omega_{jk} \\ &= \frac{1}{64} \sum_{j=1}^{N_f} (d_\tau \omega_j + d_\tau \bar{\omega}_j)^T (gg^T + D_j) (d_\tau \omega_j + d_\tau \bar{\omega}_j) + \frac{1}{2} \sum_{j=1}^{N_f} \left(\frac{dE}{d\omega_j} \right)^T (d_\tau \omega_j + d_\tau \bar{\omega}_j). \end{aligned} \quad (3.K.18)$$

Note, that gg^T is a positive semidefinite matrix since it is the dyadic product of a real-valued vector with itself, i.e. for any real non-zero vector x of length N_f we have $x^T (gg^T) x = (g^T x)^T (g^T x) = (g^T x)^2 \geq 0$, since $(g^T x)$ is just a scalar. By definition, D_j is positive semidefinite and $gg^T + D_j$ as a sum of two positive semidefinite matrices is also positive semidefinite. We define the symmetric matrix element

$$\dot{\omega}_{jk} = \frac{1}{2} \left((d_\tau \omega)_{jk} + (d_\tau \omega)_{kj} \right) \quad (3.K.19)$$

and rewrite the first part of the right-hand side of Eq. (3.K.18) as

$$\frac{1}{64} \sum_{j=1}^{N_f} (d_\tau \omega_j + d_\tau \bar{\omega}_j)^T (gg^T + D_j) (d_\tau \omega_j + d_\tau \bar{\omega}_j) = \frac{1}{4} \sum_{j,k,l=1}^{N_f} \dot{\omega}_{jk} A_{kl}^{(j)} \dot{\omega}_{jl}, \quad (3.K.20)$$

where

$$A_{kl}^{(j)} = \frac{1}{4} (gg^T + D_j)_{kl}. \quad (3.K.21)$$

The second term on the right-hand side of Eq. (3.K.18) can be written as

$$\frac{1}{2} \sum_{j=1}^{N_f} \left(\frac{dE}{d\omega_j} \right)^T (d_\tau \omega_j + d_\tau \bar{\omega}_j) = \sum_{j,k=1}^{N_f} \dot{\omega}_{jk} \frac{dE}{d\omega_{jk}}. \quad (3.K.22)$$

By defining the tensor

$$B_{jklm} = \frac{1}{4} \left(A_{kl}^{(m)} \delta_{m,j} + A_{jl}^{(k)} \delta_{m,k} + A_{km}^{(l)} \delta_{l,j} + A_{jm}^{(k)} \delta_{l,k} \right), \quad (3.K.23)$$

we can rewrite Eq.(3.K.18) as

$$\frac{1}{8} \text{tr} (O_m^2) + \sum_{j,k=1}^{N_f} \frac{dE}{d\omega_{jk}} d_\tau \omega_{jk} = \frac{1}{4} \sum_{j,k,l,m=1}^{N_f} \dot{\omega}_{jk} B_{jklm} \dot{\omega}_{lm} + \sum_{j,k=1}^{N_f} \dot{\omega}_{jk} \frac{dE}{d\omega_{jk}}. \quad (3.K.24)$$

Note, that we can choose the form of B in Eq. (3.K.23) due to the symmetry $\dot{\omega}_{jk} = \dot{\omega}_{kj}$. The above equation vanishes, if

$$\frac{1}{4} \sum_{l,m=1}^{N_f} B_{jklm} \dot{\omega}_{lm} + \frac{dE}{d\omega_{jk}} = 0 \quad (3.K.25)$$

for all $j, k \in \{1, \dots, N_f\}$. We reshape the $(N_f \times N_f \times N_f \times N_f)$ -tensor B_{jklm} into a $(N_f^2 \times N_f^2)$ -matrix $B_{(jk),(lm)}$ with tuple entries (jk) and (lm) . Similarly, we reshape the $(N_f \times N_f)$ -matrix $dE/d\omega$ into a column vector $E_{(jk)}$ of length N_f^2 . We denote the tuple indices with capital roman letters. Then, Eq. (3.K.25) reads

$$\frac{1}{4} \sum_{L=1}^{N_f^2} B_{JL} \dot{\omega}_L = -\frac{dE}{d\omega_J} \Rightarrow \frac{1}{4} B \dot{\omega} = -\frac{dE}{d\omega}. \quad (3.K.26)$$

A sufficient condition for the energy to be monotonically decreasing during the HITGD evolution is given in the reshaped picture by

$$\frac{1}{4} \dot{\omega}^T B \dot{\omega} + \dot{\omega}^T \frac{dE}{d\omega} = 0. \quad (3.K.27)$$

By choosing

$$\dot{\omega} = -4B^+ \frac{dE}{d\omega}, \quad (3.K.28)$$

we see, that the expression in Eq. (3.K.27) will be identical to zero,

$$\frac{1}{4} \dot{\omega}^T B \dot{\omega} + \dot{\omega}^T \frac{dE}{d\omega} = 4 \left(\frac{dE}{d\omega} \right)^T (B^+ B B^+ - B^+) \frac{dE}{d\omega} = 0, \quad (3.K.29)$$

where the last line follows from the first Moore-Penrose condition.

Alternatively, it might be more favorable to choose

$$\dot{\omega}_{jk} = -\frac{1}{\lambda_{jk}} \frac{dE}{d\omega_{jk}}, \quad (3.K.30)$$

where

$$\lambda_{jk} > \frac{\|B\|_{\max}}{4}, \quad (3.K.31)$$

and $\|B\|_{\max}$ denotes the largest value of the spectral norm of the flattened tensor B . The Ansatz in Eq. (3.K.30) is particularly useful if one can find a global maximum value of λ , i.e. independent of the indices and it speeds up the optimization steps as it does not require the computation of B . On the downside, however, the Ansatz in Eq. (3.K.30) no longer guarantees a monotonic decrease of the energy during the time evolution.

Section 3.L

MEAN-FIELD EXPRESSION OF ROTATED HAMILTONIAN

In this section, we want to derive the mean-field expression

$$\left(\bar{H}_m\right)_{ij} = \left(\bar{H}_{1,m}\right)_{ij} + \left(\bar{H}_{2,m}\right)_{ij}, \quad (3.L.1)$$

where we defined the rotated mean-field of the one-body terms

$$\left(\bar{H}_{1,m}\right)_{ij} = i \frac{d}{d(\Gamma_m)_{ij}} \sum_{p,q=1}^{N_f} f_{pq} \mathcal{A}_{pq}^{\alpha_{pq}^{\text{FA}}} \mathcal{G}_{\mathbf{p}^\dagger \mathbf{q}}^{\alpha_{pq}^{\text{FA}}} \quad (3.L.2)$$

and two-body terms

$$\left(\bar{H}_{2,m}\right)_{ij} = -\frac{1}{8} \frac{d}{d(\Gamma_m)_{ij}} \sum_{p,q,r,s=1}^{N_f} h_{pqrs} e^{i(\omega_{rs} - \omega_{pq})} \mathcal{A}_{pqrs}^{\beta_{pqrs}^{\text{FA}}} \left[\mathcal{G}_{\mathbf{p}^\dagger \mathbf{s}} \mathcal{G}_{\mathbf{q}^\dagger \mathbf{r}} - \mathcal{G}_{\mathbf{p}^\dagger \mathbf{r}} \mathcal{G}_{\mathbf{q}^\dagger \mathbf{s}} + \mathcal{G}_{\mathbf{p}^\dagger \mathbf{q}^\dagger} \mathcal{G}_{\mathbf{r} \mathbf{s}} \right]^{\beta_{pqrs}^{\text{FA}}}, \quad (3.L.3)$$

where the superscript on the square brackets indicates that the expressions inside depend on β_{pqrs}^{FA} (this notation was introduced in Eq. (3.2.15)).

We first turn to the one-body terms, using the results of Appendix 3.G, as well as the expectation value definitions of Eqs. (3.5.2)-(3.5.3),

$$\left(\bar{H}_{1,m}\right)_{ij} = i \sum_{p,q=1}^{N_f} f_{pq} \mathcal{A}_{pq}^{\alpha_{pq}^{\text{FA}}} \left[\mathcal{F}_{ij} \mathcal{G}_{\mathbf{p}^\dagger \mathbf{q}} + \frac{1}{2} \left(\mathcal{L}(|i\rangle \langle j| - |j\rangle \langle i|) \mathcal{L}^T \right)_{\mathbf{p}^\dagger \mathbf{q}} \right]^{\alpha_{pq}^{\text{FA}}}, \quad (3.L.4)$$

where we used the notation defined in Eqs. (3.2.11)-(3.2.14), while the two-body terms result in

$$\begin{aligned} \left(\bar{H}_{2,m}\right)_{ij} = & -\frac{1}{16} \sum_{p,q,r,s=1}^{N_f} h_{pqrs} e^{i(\omega_{rs} - \omega_{pq})} \mathcal{A}_{pqrs}^{\beta_{pqrs}^{\text{FA}}} \left[2\mathcal{F}_{ij} \left(\mathcal{G}_{\mathbf{p}^\dagger \mathbf{s}} \mathcal{G}_{\mathbf{q}^\dagger \mathbf{r}} - \mathcal{G}_{\mathbf{p}^\dagger \mathbf{r}} \mathcal{G}_{\mathbf{q}^\dagger \mathbf{s}} + \mathcal{G}_{\mathbf{p}^\dagger \mathbf{q}^\dagger} \mathcal{G}_{\mathbf{r} \mathbf{s}} \right) \right. \\ & + \left(\mathcal{L}(|i\rangle \langle j| - |j\rangle \langle i|) \mathcal{L}^T \right)_{\mathbf{p}^\dagger \mathbf{s}} \mathcal{G}_{\mathbf{q}^\dagger \mathbf{r}} - \left(\mathcal{L}(|i\rangle \langle j| - |j\rangle \langle i|) \mathcal{L}^T \right)_{\mathbf{p}^\dagger \mathbf{r}} \mathcal{G}_{\mathbf{q}^\dagger \mathbf{s}} \\ & + \left(\mathcal{L}(|i\rangle \langle j| - |j\rangle \langle i|) \mathcal{L}^T \right)_{\mathbf{p}^\dagger \mathbf{q}^\dagger} \mathcal{G}_{\mathbf{r} \mathbf{s}} + \mathcal{G}_{\mathbf{p}^\dagger \mathbf{s}} \left(\mathcal{L}(|i\rangle \langle j| - |j\rangle \langle i|) \mathcal{L}^T \right)_{\mathbf{q}^\dagger \mathbf{r}} \\ & \left. - \mathcal{G}_{\mathbf{p}^\dagger \mathbf{r}} \left(\mathcal{L}(|i\rangle \langle j| - |j\rangle \langle i|) \mathcal{L}^T \right)_{\mathbf{q}^\dagger \mathbf{s}} + \mathcal{G}_{\mathbf{p}^\dagger \mathbf{q}^\dagger} \left(\mathcal{L}(|i\rangle \langle j| - |j\rangle \langle i|) \mathcal{L}^T \right)_{\mathbf{r} \mathbf{s}} \right]^{\beta_{pqrs}^{\text{FA}}}. \end{aligned} \quad (3.L.5)$$

We can simplify this expression by realizing that some terms in the above sum are identical after relabeling the indices and using symmetries,

$$\begin{aligned} \left(\bar{H}_{2,m}\right)_{ij} = & -\frac{1}{16} \sum_{p,q,r,s=1}^{N_f} h_{pqrs} e^{i(\omega_{rs} - \omega_{pq})} \mathcal{A}_{pqrs}^{\beta_{pqrs}^{\text{FA}}} \left[4\mathcal{F}_{ij} \mathcal{G}_{\mathbf{p}^\dagger \mathbf{s}} \mathcal{G}_{\mathbf{q}^\dagger \mathbf{r}} + 2\mathcal{F}_{ij} \mathcal{G}_{\mathbf{p}^\dagger \mathbf{q}^\dagger} \mathcal{G}_{\mathbf{r} \mathbf{s}} \right. \\ & + 4 \left(\mathcal{L}(|i\rangle \langle j| - |j\rangle \langle i|) \mathcal{L}^T \right)_{\mathbf{p}^\dagger \mathbf{s}} \mathcal{G}_{\mathbf{q}^\dagger \mathbf{r}} + \left(\mathcal{L}(|i\rangle \langle j| - |j\rangle \langle i|) \mathcal{L}^T \right)_{\mathbf{p}^\dagger \mathbf{q}^\dagger} \mathcal{G}_{\mathbf{r} \mathbf{s}} \\ & \left. + \mathcal{G}_{\mathbf{p}^\dagger \mathbf{q}^\dagger} \left(\mathcal{L}(|i\rangle \langle j| - |j\rangle \langle i|) \mathcal{L}^T \right)_{\mathbf{r} \mathbf{s}} \right]^{\beta_{pqrs}^{\text{FA}}}. \end{aligned} \quad (3.L.6)$$

Inserting Eqs. (3.L.4) and (3.L.5) into Eq. (3.L.1) gives an analytical expression for the mean-field matrix of the rotated Hamiltonian, which is anti-symmetric under transposition, $(\bar{H}_m)_{ij} = -(\bar{H}_m)_{ji}$.

Section 3.M

COMPUTING THE COMMUTATOR OF THE NUMBER-NUMBER OPERATOR W.R.T. THE ROTATED HAMILTONIAN

In this appendix, we derive the terms that lead to Eq. (3.7.12).

3.M.1 Contributions from rotated one-body Hamiltonian terms

In this subsection, we will compute

$$\left\langle \left[\bar{H}_1, c_i^\dagger c_j^\dagger c_i c_j \right] \right\rangle_{\text{GS}} = \sum_{p,q=1}^{N_f} f_{pq}^{\text{FA}} \left\langle \left[e^{i \sum_k \alpha_{pq}^{\text{FA}}(k) n_k^f} c_p^\dagger c_q, c_i^\dagger c_j^\dagger c_i c_j \right] \right\rangle_{\text{GS}}. \quad (3.M.1)$$

In order to compute the commutator $[\bar{H}, c_i^\dagger c_j^\dagger c_i c_j]$ in Eq. (3.7.11), we first turn to terms that belong to the rotated one-particle Hamiltonian part of H_1 . The commutator of the rotated single-body Hamiltonian terms \bar{H}_1 with $c_i^\dagger c_j^\dagger c_i c_j$ will require the computation of expressions of the following form,

$$\begin{aligned} \left[e^{i \sum_k \alpha_{pq}^{\text{FA}}(k) n_k^f} c_p^\dagger c_q, c_i^\dagger c_j^\dagger c_i c_j \right] &= e^{i \sum_k \alpha_{pq}^{\text{FA}}(k) n_k^f} \left(-c_i^\dagger c_j^\dagger c_i c_q \delta_{p,j} + c_i^\dagger c_j^\dagger c_j c_q \delta_{p,i} - c_p^\dagger c_i^\dagger c_i c_j \delta_{j,q} \right. \\ &\quad \left. + c_p^\dagger c_j^\dagger c_i c_j \delta_{i,q} \right). \end{aligned} \quad (3.M.2)$$

In order to get the contributions due to the one-body terms, we will have to include the appropriate coefficients and sum over the fermionic modes, resulting in

$$\begin{aligned} &\sum_{p,q=1}^{N_f} f_{pq}^{\text{FA}} \left[e^{i \sum_k \alpha_{pq}^{\text{FA}}(k) n_k^f} c_p^\dagger c_q, c_i^\dagger c_j^\dagger c_i c_j \right] \\ &= - \sum_{p=1}^{N_f} f_{jp}^{\text{FA}} e^{i \sum_k \alpha_{jp}^{\text{FA}}(k) n_k^f} c_i^\dagger c_j^\dagger c_i c_p + \sum_{p=1}^{N_f} f_{ip}^{\text{FA}} e^{i \sum_k \alpha_{ip}^{\text{FA}}(k) n_k^f} c_i^\dagger c_j^\dagger c_j c_p \\ &\quad - \sum_{p=1}^{N_f} f_{pj}^{\text{FA}} e^{i \sum_k \alpha_{pj}^{\text{FA}}(k) n_k^f} c_p^\dagger c_i^\dagger c_i c_j + \sum_{p=1}^{N_f} f_{pi}^{\text{FA}} e^{i \sum_k \alpha_{pi}^{\text{FA}}(k) n_k^f} c_p^\dagger c_j^\dagger c_i c_j \\ &= \left(\sum_{p=1}^{N_f} f_{ip}^{\text{FA}} e^{i \sum_k \alpha_{ip}^{\text{FA}}(k) n_k^f} c_i^\dagger c_j^\dagger c_j c_p + (i \leftrightarrow j) \right) + \left(\sum_{p=1}^{N_f} f_{pi}^{\text{FA}} e^{i \sum_k \alpha_{pi}^{\text{FA}}(k) n_k^f} c_p^\dagger c_j^\dagger c_i c_j + (i \leftrightarrow j) \right). \end{aligned} \quad (3.M.3)$$

Before taking the expectation value of Eq. (3.M.3), we first give the following useful identities,

$$f_{ip}^{\text{FA}} e^{i(\alpha_{ip}^{\text{FA}}(i) + \alpha_{ip}^{\text{FA}}(j))} = f_{ip} e^{i(\omega_{jp} - \omega_{ji})} \quad (3.M.4)$$

$$f_{pi}^{\text{FA}} e^{i(\alpha_{pi}^{\text{FA}}(p) + \alpha_{pi}^{\text{FA}}(j))} = f_{pi} e^{i(\omega_{ji} - \omega_{jp})}. \quad (3.M.5)$$

We now take the expectation values of Eq. (3.M.2) w.r.t. the FGS. Following the strategy of Appendix 3.F, we compute

$$\left(f_{ip}^{\text{FA}} \left\langle e^{i \sum_k \alpha_{ip}^{\text{FA}}(k) n_k^f c_i^\dagger c_j^\dagger c_j c_p} \right\rangle_{\text{GS}} \right)^* = \left(f_{ip}^{\text{FA}} \right)^* \left\langle c_p^\dagger c_j^\dagger c_j c_i e^{-i \sum_k \alpha_{ip}^{\text{FA}}(k) n_k^f} \right\rangle_{\text{GS}}. \quad (3.M.6)$$

By commuting the exponential operator to the left of the string of fermionic creation using Eq. (3.3.16), we get an additional phase factor,

$$c_p^\dagger c_j^\dagger c_j c_i e^{-i \sum_k \alpha_{ip}^{\text{FA}}(k) n_k^f} = - e^{-2i\omega_{pi}} e^{i \sum_k \alpha_{pi}^{\text{FA}}(k) n_k^f} c_p^\dagger c_j^\dagger c_i c_j. \quad (3.M.7)$$

Computing the first coefficient on the right-hand side of Eq. (3.M.6) leads to $(f_{ip}^{\text{FA}})^* = f_{pi} e^{i\omega_{pi}}$, where we used the integral symmetry $f_{pq}^* = f_{qp}$ (which follows from the definition in Eq. (3.A.2)). Using this result and Eq. (3.M.7), Eq. (3.M.6) can be recast into

$$\left(f_{ip}^{\text{FA}} \left\langle e^{i \sum_k \alpha_{ip}^{\text{FA}}(k) n_k^f c_i^\dagger c_j^\dagger c_j c_p} \right\rangle_{\text{GS}} \right)^* = - f_{pi}^{\text{FA}} \left\langle e^{i \sum_k \alpha_{pi}^{\text{FA}}(k) n_k^f c_p^\dagger c_j^\dagger c_i c_j} \right\rangle_{\text{GS}}. \quad (3.M.8)$$

Through Eq. (3.M.8) we have shown that the expectation value w.r.t. the FGS of the terms in Eq. (3.M.3) may be written as

$$\left\langle \left[\bar{H}_1, c_i^\dagger c_j^\dagger c_i c_j \right] \right\rangle_{\text{GS}} = \left(\sum_{p=1}^{N_f} f_{ip}^{\text{FA}} \left\langle e^{i \sum_k \alpha_{ip}^{\text{FA}}(k) n_k^f c_i^\dagger c_j^\dagger c_j c_p} \right\rangle_{\text{GS}} - \text{H.c.} \right) + (i \leftrightarrow j), \quad (3.M.9)$$

where H.c. denotes the Hermitian conjugated of the first expression inside the big brackets on the right-hand side of Eq. (3.M.9). Again, $(i \leftrightarrow j)$ is a duplicate of the expression inside the large brackets, but with index i replaced by j (and vice versa). The result of Eq. (3.M.9) gives a purely imaginary number.

3.M.2 Contributions from rotated two-body Hamiltonian terms

In this subsection, we will compute

$$\left\langle \left[\bar{H}_2, c_i^\dagger c_j^\dagger c_i c_j \right] \right\rangle_{\text{GS}} = \frac{1}{2} \sum_{p,q,r,s=1}^{N_f} h_{pqrs}^{\text{FA}} \left\langle \left[e^{i \sum_k \beta_{pqrs}^{\text{FA}}(k) n_k^f c_p^\dagger c_q^\dagger c_r c_s}, c_i^\dagger c_j^\dagger c_i c_j \right] \right\rangle_{\text{GS}}. \quad (3.M.10)$$

The commutator of the rotated two-body Hamiltonian terms with $c_i^\dagger c_j^\dagger c_i c_j$ is given by

$$\begin{aligned} & \left[e^{i \sum_k \beta_{pqrs}^{\text{FA}}(k) n_k^f c_p^\dagger c_q^\dagger c_r c_s}, c_i^\dagger c_j^\dagger c_i c_j \right] \\ &= e^{i \sum_k \beta_{pqrs}^{\text{FA}}(k) n_k^f} \left(c_i^\dagger c_j^\dagger c_q^\dagger c_r c_s \delta_{p,j} - c_i^\dagger c_j^\dagger c_r c_s \delta_{p,j} \delta_{q,i} - c_i^\dagger c_j^\dagger c_q^\dagger c_r c_s \delta_{p,i} + c_i^\dagger c_j^\dagger c_r c_s \delta_{p,i} \delta_{j,q} \right. \\ & \quad - c_i^\dagger c_j^\dagger c_p^\dagger c_i c_r c_s \delta_{q,j} + c_i^\dagger c_j^\dagger c_p^\dagger c_j c_r c_s \delta_{q,i} - c_p^\dagger c_q^\dagger c_i^\dagger c_i c_j c_s \delta_{r,j} + c_p^\dagger c_q^\dagger c_j^\dagger c_i c_j c_s \delta_{r,i} \\ & \quad \left. + c_p^\dagger c_q^\dagger c_i^\dagger c_r c_i c_j \delta_{s,j} - c_p^\dagger c_q^\dagger c_i^\dagger c_j \delta_{s,j} \delta_{r,i} - c_p^\dagger c_q^\dagger c_j^\dagger c_r c_i c_j \delta_{i,s} + c_p^\dagger c_q^\dagger c_i c_j \delta_{i,s} \delta_{r,j} \right). \end{aligned} \quad (3.M.11)$$

In order to get the contributions due to the rotated two-body Hamiltonian \bar{H}_2 , we will have to include the appropriate coefficients and sum over the fermionic modes. After

relabeling the indices, this results in

$$\begin{aligned}
& \frac{1}{2} \sum_{p,q,r,s=1}^{N_f} h_{pqrs}^{\text{FA}} \left[e^{i \sum_k \beta_{pqrs}^{\text{FA}}(k) n_k^f c_p^\dagger c_q^\dagger c_r c_s, c_i^\dagger c_j^\dagger c_i c_j} \right] \\
&= \frac{1}{2} \sum_{p,q,r=1}^{N_f} h_{jpqr}^{\text{FA}} e^{i \sum_k \beta_{jpqr}^{\text{FA}}(k) n_k^f c_i^\dagger c_j^\dagger c_p^\dagger c_q^\dagger c_i c_j c_r c_r} + \frac{1}{2} \sum_{p,q,r=1}^{N_f} h_{ipqr}^{\text{FA}} e^{i \sum_k \beta_{ipqr}^{\text{FA}}(k) n_k^f c_j^\dagger c_i^\dagger c_p^\dagger c_q^\dagger c_j c_j c_q c_r} \\
&+ \frac{1}{2} \sum_{p,q=1}^{N_f} h_{jipq}^{\text{FA}} e^{i \sum_k \beta_{jipq}^{\text{FA}}(k) n_k^f c_j^\dagger c_i^\dagger c_p^\dagger c_q^\dagger c_i c_j c_p c_q} + \frac{1}{2} \sum_{p,q=1}^{N_f} h_{ijpq}^{\text{FA}} e^{i \sum_k \beta_{ijpq}^{\text{FA}}(k) n_k^f c_i^\dagger c_j^\dagger c_p^\dagger c_q^\dagger c_i c_j c_p c_q} \\
&+ \frac{1}{2} \sum_{p,q,r=1}^{N_f} h_{pjqr}^{\text{FA}} e^{i \sum_k \beta_{pjqr}^{\text{FA}}(k) n_k^f c_j^\dagger c_i^\dagger c_p^\dagger c_q^\dagger c_i c_j c_q c_r} + \frac{1}{2} \sum_{p,q,r=1}^{N_f} h_{piqr}^{\text{FA}} e^{i \sum_k \beta_{piqr}^{\text{FA}}(k) n_k^f c_i^\dagger c_j^\dagger c_p^\dagger c_q^\dagger c_j c_j c_q c_r} \\
&+ \frac{1}{2} \sum_{p,q,r=1}^{N_f} h_{pqjr}^{\text{FA}} e^{i \sum_k \beta_{pqjr}^{\text{FA}}(k) n_k^f c_p^\dagger c_q^\dagger c_i^\dagger c_j^\dagger c_j c_i c_r} + \frac{1}{2} \sum_{p,q,r=1}^{N_f} h_{pqir}^{\text{FA}} e^{i \sum_k \beta_{pqir}^{\text{FA}}(k) n_k^f c_p^\dagger c_q^\dagger c_i^\dagger c_j^\dagger c_i c_j c_r} \\
&+ \frac{1}{2} \sum_{p,q,r=1}^{N_f} h_{pqrj}^{\text{FA}} e^{i \sum_k \beta_{pqrj}^{\text{FA}}(k) n_k^f c_p^\dagger c_q^\dagger c_i^\dagger c_j^\dagger c_r c_i c_j} + \frac{1}{2} \sum_{p,q,r=1}^{N_f} h_{pqri}^{\text{FA}} e^{i \sum_k \beta_{pqri}^{\text{FA}}(k) n_k^f c_p^\dagger c_q^\dagger c_j^\dagger c_r c_j c_i} \\
&+ \frac{1}{2} \sum_{p,q=1}^{N_f} h_{pqij}^{\text{FA}} e^{i \sum_k \beta_{pqij}^{\text{FA}}(k) n_k^f c_p^\dagger c_q^\dagger c_j^\dagger c_i^\dagger c_j c_i} + \frac{1}{2} \sum_{p,q=1}^{N_f} h_{pqji}^{\text{FA}} e^{i \sum_k \beta_{pqji}^{\text{FA}}(k) n_k^f c_p^\dagger c_q^\dagger c_i^\dagger c_j^\dagger c_i c_j} \quad (3.M.12)
\end{aligned}$$

By use of the coefficient symmetries of h_{pqrs}^{FA} , the anti-commutation relations of the fermionic operators, and observing that the second term on each of the lines on the right-hand side of Eq. (3.M.12) is identical to its left—when interchanging the indices i and j (which we will indicate by the notation $(i \leftrightarrow j)$)— we can rewrite the above expression as

$$\begin{aligned}
& \frac{1}{2} \sum_{p,q,r,s=1}^{N_f} h_{pqrs}^{\text{FA}} \left[e^{i \sum_k \beta_{pqrs}^{\text{FA}}(k) n_k^f c_p^\dagger c_q^\dagger c_r c_s, c_i^\dagger c_j^\dagger c_i c_j} \right] \\
&= \left[\sum_{p,q,r=1}^{N_f} h_{ipqr}^{\text{FA}} e^{i \sum_k \beta_{ipqr}^{\text{FA}}(k) n_k^f c_j^\dagger c_i^\dagger c_p^\dagger c_q^\dagger c_j c_j c_q c_r} + \sum_{p,q,r=1}^{N_f} h_{pqri}^{\text{FA}} e^{i \sum_k \beta_{pqri}^{\text{FA}}(k) n_k^f c_p^\dagger c_q^\dagger c_j^\dagger c_r c_j c_i} \right. \\
&\quad \left. + \frac{1}{2} \sum_{p,q=1}^{N_f} h_{jipq}^{\text{FA}} e^{i \sum_k \beta_{jipq}^{\text{FA}}(k) n_k^f c_i^\dagger c_j^\dagger c_p^\dagger c_q^\dagger c_i c_j c_p c_q} + \frac{1}{2} \sum_{p,q=1}^{N_f} h_{pqij}^{\text{FA}} e^{i \sum_k \beta_{pqij}^{\text{FA}}(k) n_k^f c_p^\dagger c_q^\dagger c_j^\dagger c_i^\dagger c_j c_i} \right] + (i \leftrightarrow j) \quad (3.M.13)
\end{aligned}$$

In order to simplify the expectation value of the above expression w.r.t. the FGS, we take the Hermitian conjugate of the first term on the right-hand side of Eq. (3.M.13) in the first and second line, analogous to Appendix 3.M.1 and use the symmetry property

$h_{pqrs}^* = h_{srqp}$ of the Coulomb matrix elements,

$$\begin{aligned} & \left(\sum_{p,q,r=1}^{N_f} h_{ipqr}^{\text{FA}} \left\langle e^{i \sum_k \beta_{ipqr}^{\text{FA}}(k) n_k^f c_j^\dagger c_i^\dagger c_p^\dagger c_q^\dagger c_j c_q c_r} \right\rangle_{\text{GS}} \right)^* \\ &= - \sum_{p,q,r=1}^{N_f} h_{pqri}^{\text{FA}} \left\langle e^{i \sum_k \beta_{pqri}^{\text{FA}}(k) n_k^f c_p^\dagger c_q^\dagger c_j^\dagger c_r c_j c_i} \right\rangle_{\text{GS}} \end{aligned} \quad (3.M.14)$$

$$\begin{aligned} & \left(\frac{1}{2} \sum_{p,q=1}^{N_f} h_{ijpq}^{\text{FA}} \left\langle e^{i \sum_k \beta_{ijpq}^{\text{FA}}(k) n_k^f c_i^\dagger c_j^\dagger c_p c_q} \right\rangle_{\text{GS}} \right)^* \\ &= - \frac{1}{2} \sum_{p,q=1}^{N_f} h_{pqij}^{\text{FA}} \left\langle e^{i \sum_k \beta_{pqij}^{\text{FA}}(k) n_k^f c_p^\dagger c_q^\dagger c_j c_i} \right\rangle_{\text{GS}}. \end{aligned} \quad (3.M.15)$$

Inserting Eqs. (3.M.14)-(3.M.15) into Eq. (3.M.13) leads to

$$\begin{aligned} \left\langle \left[\bar{H}_2, c_i^\dagger c_j^\dagger c_i c_j \right] \right\rangle_{\text{GS}} &= \left[\left(\sum_{p,q,r=1}^{N_f} h_{ipqr}^{\text{FA}} \left\langle e^{i \sum_k \beta_{ipqr}^{\text{FA}}(k) n_k^f c_j^\dagger c_i^\dagger c_p^\dagger c_q^\dagger c_j c_q c_r} \right\rangle_{\text{GS}} - \text{H.c.} \right) \right. \\ & \quad \left. + \frac{1}{2} \left(\sum_{p,q=1}^{N_f} h_{ijpq}^{\text{FA}} \left\langle e^{i \sum_k \beta_{ijpq}^{\text{FA}}(k) n_k^f c_i^\dagger c_j^\dagger c_p c_q} \right\rangle_{\text{GS}} - \text{H.c.} \right) \right] + (i \leftrightarrow j). \end{aligned} \quad (3.M.16)$$

Note, that the resulting number in Eq. (3.M.16) is purely imaginary. We can simplify part of the above expression through

$$\frac{1}{2} \sum_{p,q=1}^{N_f} h_{ijpq}^{\text{FA}} \left\langle e^{i \sum_k \beta_{ijpq}^{\text{FA}}(k) n_k^f c_i^\dagger c_j^\dagger c_p c_q} \right\rangle_{\text{GS}} + (i \leftrightarrow j) = 2 \sum_{p < q}^{N_f} h_{ijpq}^{\text{FA}} \left\langle e^{i \sum_k \beta_{ijpq}^{\text{FA}}(k) n_k^f c_i^\dagger c_j^\dagger c_p c_q} \right\rangle_{\text{GS}}. \quad (3.M.17)$$

Section 3.N

DERIVATIONS OF THE EQUATIONS OF MOTION OF THE VARIATIONAL PARAMETERS THROUGH IMAGINARY-TIME EVOLUTION

Within the imaginary-time evolution, the variational state fulfills

$$d_\tau |\Psi_{\text{NGS}}\rangle = -(H - E) |\Psi_{\text{NGS}}\rangle \quad (3.N.1)$$

where $E = \langle \Psi_{\text{NGS}} | H | \Psi_{\text{NGS}} \rangle$ is the variational energy at time τ . The above equation is obtained by replacing $|\varphi(\tau)\rangle$ in Eq. (3.6.2) by the variational state Ansatz $|\Psi_{\text{NGS}}\rangle$.

Following Ref. [41], we treat the left- and right-hand side of Eq. (3.N.1) separately in Appendices 3.N.1 and 3.N.2. We will make extensive use of the normal-ordered expansion of an operator as introduced in Appendix 3.H, which will help construct the tangent vectors $|\Psi_j\rangle$ in such a way that they are orthogonal to each other. This allows us to derive the equation of motion for the CM Γ_m given in Eq. (3.6.8).

3.N.1 Left-hand side of the equation of motion

Following Eq. (3.I.12), the left-hand side of Eq. (3.N.1) can be written as

$$d_\tau |\Psi_{\text{NGS}}\rangle = U_{\text{FA}} U_{\text{GS}} |\Psi_L\rangle, \quad (3.N.2)$$

where we defined the state

$$|\Psi_L\rangle = U_L |0\rangle, \quad (3.N.3)$$

and the operator U_L is given by Eq. (3.I.19)¹¹. Since U_L in Eq. (3.N.2) is a normal-ordered operator acting on the fermionic vacuum state, only fermionic creation operators will survive. Applying U_L on the vacuum state will result in a linear combination of operators containing a various number of fermionic excitations. In our case,

$$U_L |0\rangle = |L_0\rangle + |L_2\rangle + \delta O |0\rangle, \quad (3.N.4)$$

where the first term contains zero fermionic excitations operators, the second terms contains two, and the last term includes all excitations larger than two. More specifically, the first two terms are explicitly given by

$$|L_0\rangle = \left(\frac{i}{4} \text{tr} \left[(d_\tau U_m) U_m^T \Gamma_m \right] + \langle O \rangle_{\text{GS}} \right) |0\rangle \quad (3.N.5)$$

$$|L_2\rangle = \frac{1}{4} : A^T U_m^T ((d_\tau U_m) + i O_m U_m) A : |0\rangle. \quad (3.N.6)$$

By way of shifting $U_{\text{FA}} U_{\text{GS}}$ to the left in Eq. (3.N.2), Shi et al. construct an orthogonal set of states $|L_0\rangle, |L_2\rangle, \delta O_{\text{FA}} |0\rangle$, which span the tangent space of the variational manifold.

¹¹The result we obtained in Eq. (3.I.19) slightly deviates from the result in Ref. [41]. All prior equations that led to this expression were in exact agreement, but when calculating the first term in Eq. (3.I.15), we got a different order. Essentially, Eq. (3.I.19) corresponds to equation (E.1) in [41], however, instead of $(d_\tau U_m) U_m^T \Gamma_m$ inside the trace of the first term on the right-hand side, they have the term $U_m^T (d_\tau U_m) \Gamma_m$. We trust our result, as its derivation can be followed in detail.

3.N.2 Right-hand side of the equation of motion

We now turn to the right-hand side of Eq. (3.N.1),

$$|R_\Psi\rangle = - (H - E) |\Psi_{\text{NGS}}\rangle. \quad (3.N.7)$$

In a procedure analogous to the one in Section 3.N.1, we move the unitary Gaussian operator U_{GS} past the Hamiltonian operator, resulting in

$$|R_\Psi\rangle = U_S U_{\text{GS}} |\Psi_{\text{R}}\rangle, \quad (3.N.8)$$

where

$$|\Psi_{\text{R}}\rangle = - (U_{\text{R}} - E) |0\rangle \quad (3.N.9)$$

is the state obtained by moving the non-Gaussian and Gaussian operators to the left and

$$U_{\text{R}} = U_{\text{GS}}^\dagger \bar{H} U_{\text{GS}} \quad (3.N.10)$$

is the Gaussian-transformed rotated Hamiltonian in Eq. (3.3.19). The expectation value of the variational energy E can be obtained by computing Eq. (3.5.1). Using the normal-ordered expansion of Eq. (3.H.2), we can write the right-hand side of Eq. (3.N.10) as

$$U_{\text{R}} = \langle \bar{H} \rangle_{\text{GS}} + \frac{i}{4} : A^T U_m^T \bar{H}_m U_m A : + \delta \bar{H}, \quad (3.N.11)$$

where \bar{H}_m is the mean-field matrix of the rotated Hamiltonian and $\delta \bar{H}$ contains higher-order terms. This simplifies the right-hand side of Eq. (3.N.9) since constant terms $E = \langle \bar{H} \rangle_{\text{GS}}$ are cancelling each other, giving

$$|\Psi_{\text{R}}\rangle = - \left(\frac{i}{4} : A^T U_m^T \bar{H}_m U_m A : + \delta \bar{H} \right) |0\rangle. \quad (3.N.12)$$

Following the strategy of Eq. (3.N.4), applying the operator U_{R} of Eq. (3.N.11) on the fermionic vacuum state results in

$$U_{\text{R}} |0\rangle = |R_0\rangle + |R_2\rangle + \delta \bar{H} |0\rangle, \quad (3.N.13)$$

where—by construction—the terms on the right-hand side are orthogonal to each other and $\delta \bar{H} |0\rangle$ contains higher-order terms. The explicit form of the linear and quadratic term is given by

$$|R_0\rangle = \langle \bar{H} \rangle_{\text{GS}} |0\rangle \quad (3.N.14)$$

$$|R_2\rangle = \frac{i}{4} : A^T U_m^T \bar{H}_m U_m A : |0\rangle. \quad (3.N.15)$$

3.N.3 Getting the equations of motion for the covariance matrix

The equations of motion for the CM are obtained by comparing Eq. (3.N.6) with Eq. (3.N.15). Due to Eq. (3.N.1), we know that the left- and right-hand side expressions $|\Psi_{\text{L}}\rangle$ and $|\Psi_{\text{R}}\rangle$ must be identical. This leads to the condition $|L_2\rangle = -|R_2\rangle$. For the quadratic polynomials this translates to

$$: A^T U_m^T ((d_\tau U_m) + i O_m U_m) A : |0\rangle = - i : A^T U_m^T \bar{H}_m U_m A : |0\rangle. \quad (3.N.16)$$

Since all terms in the above expression are normal ordered, and the normal-ordered products act on the fermionic vacuum, we know that only the fermionic creation operators will survive¹². We define $c^\dagger = (c_1^\dagger, \dots, c_{N_f}^\dagger)$, which provides us with a shorthand notation for expressions such as

$$(c^\dagger, c^\dagger) = (c_1^\dagger, \dots, c_{N_f}^\dagger, c_{N_f+1}^\dagger, \dots, c_{2N_f}^\dagger).$$

We can use the fact, that $(c^\dagger, ic^\dagger)\Upsilon = -i(c^\dagger, ic^\dagger)$ and $\Upsilon(c^\dagger, ic^\dagger)^T = i(c^\dagger, ic^\dagger)^T$ to rewrite the right-hand side of Eq. (3.N.16) in a suitable way,

$$: A^T U_m^T \bar{H}_m U_m A : |0\rangle = \frac{i}{2} : (c^\dagger, ic^\dagger) \left(\Upsilon U_m^T \bar{H}_m U_m - U_m^T \bar{H}_m U_m \Upsilon \right) \begin{pmatrix} c^\dagger \\ ic^\dagger \end{pmatrix} : |0\rangle. \quad (3.N.17)$$

We also write the left-hand side of Eq. (3.N.16) in terms of fermionic creation operators,

$$: A^T U_m^T ((d_\tau U_m) + iO_m U_m) A : |0\rangle = : (c^\dagger, ic^\dagger) U_m^T ((d_\tau U_m) + iO_m U_m) \begin{pmatrix} c^\dagger \\ ic^\dagger \end{pmatrix} : |0\rangle. \quad (3.N.18)$$

We can now use the orthogonality of U_m , as well as the identity $\Gamma_m = -U_m \Upsilon U_m^T$ in order to derive an expression for $d_\tau U_m$ —simply by inserting Eqs. (3.N.18) and (3.N.17) into Eq. (3.N.16)— which leads to

$$d_\tau U_m = -\frac{1}{2} \Gamma_m \bar{H}_m U_m - \frac{1}{2} \bar{H}_m U_m \Upsilon - iO_m U_m. \quad (3.N.19)$$

Under the assumption, that U_m is an orthogonal matrix, we can derive an expression for the imaginary-time derivative of the transposed expression,

$$d_\tau U_m^T = -U_m^T (d_\tau U_m) U_m^T,$$

which we can use to derive an equation of motion for the parameters ξ belonging to the CM. More specifically, we can use the relation between those variational parameters with and CM to derive the equation of motion displayed in Eq. (3.6.8) with the help of Eq. (3.N.19). This is described in detail below.

Since ξ is an anti-symmetric and hermitian matrix, U_m is guaranteed to be orthogonal, i.e. $U_m U_m^T = \mathbf{1}$. Taking the time-derivative results in

$$d_\tau U_m^T = -U_m^T (d_\tau U_m) U_m^T. \quad (3.N.20)$$

We insert Eq. (3.N.19) into Eq. (3.N.20) and use the relation $\Gamma_m = -U_m \Upsilon U_m^T$, which helps us to express the time derivative of the transpose in terms of known quantities,

$$d_\tau U_m^T = \frac{1}{2} U_m^T \Gamma_m \bar{H}_m - \frac{1}{2} U_m^T \bar{H}_m \Gamma_m + iU_m^T O_m. \quad (3.N.21)$$

To get a differential equation for the CM, we insert Eqs. (3.N.21) and (3.N.19) into

$$d_\tau \Gamma_m = - (d_\tau U_m) \Upsilon U_m^T - U_m \Upsilon (d_\tau U_m^T) \quad (3.N.22)$$

and using $\Upsilon^2 = -\mathbf{1}$ and $\Gamma_m^2 = -\mathbf{1}$, we reproduce the equation of motion describing the variational parameters belonging to the FGS, i.e. Eq. (3.6.8).

One can further derive from Eq. (3.N.22) the commutation relation for the symmetrized part of the mean energy term \bar{H}_m with the CM,

$$\left[\bar{H}_m^T + \bar{H}_m, \Gamma_m \right] = 2i (O_m^T + O_m). \quad (3.N.23)$$

The above relation follows from the orthogonality of U_m .

¹²Since $c_j |0\rangle = 0 \forall j$ by definition of the fermionic vacuum.

Section 3.0

BEHAVIOR OF THE ENERGY IN TIME

Section 3.0.1 shows how the energy behaves in time when both ω and ξ follow an imaginary-time evolution. This section uses differential geometry arguments and is taken from Ref. [41] and is included solely for completeness. In Section 3.0.2 we summarize the steps to get from Eq. (3.6.38) to Eq. (3.6.39).

3.0.1 Differential geometry perspective on the imaginary-time evolution

In this section we follow Ref. [41] where it is shown that the energy expectation value $E(\xi, \omega)$ in Eq. (3.3.26) is monotonically decreasing in time when following an imaginary-time evolution of the variational parameters. Unlike Section 3.6.5, the arguments here are based on a differential geometry perspective.

We consider a general variational Ansatz $|\Psi_{\text{NGS}}\rangle = |\Psi(\theta)\rangle$, where θ describes a set of variational parameters (in our case θ would contain all variational parameters ξ and ω). Following an imaginary-time evolution as in Eq. (3.6.2), the time derivative of the variational state follows

$$d_\tau |\Psi(\theta)\rangle = \sum_{j=1}^{r_G} (d_\tau \theta_j) |\Psi_j\rangle, \quad (3.0.1)$$

where r_G is the rank of the Gram matrix. The vectors $|\Psi_j\rangle$ span the so-called *tangent space*. As stated before, in general r_G can be smaller than the total number of variational parameters if the tangent vectors are not all linearly independent. In that case, some of the variational parameters have to be held fixed. The right-hand side of Eq. (3.6.2) can be written as

$$|R_\Psi\rangle = -(H - E) |\Psi(\theta)\rangle \quad (3.0.2)$$

as in Eq. (3.N.7). The above state can be decomposed into state vectors $|\Psi_\parallel\rangle = P_\theta |\Psi(\theta)\rangle$ which lie in the tangent space, i.o.w.

$$|\Psi_\parallel\rangle = \sum_{j=1}^{r_G} (d_\tau \theta_j) |\Psi_j\rangle \quad (3.0.3)$$

and state vectors $|\Psi_\perp\rangle = (\mathbb{1} - P_\theta) |\Psi(\theta)\rangle$ which are orthogonal to the tangent space and P_θ is a projection operator onto the tangent space. The variational state is supposed to be normalized, therefore

$$d_\tau \langle \Psi(\theta) | \Psi(\theta) \rangle = 2\Re\{\langle \Psi(\theta) | \Psi_\parallel \rangle\} = 0, \quad (3.0.4)$$

where $\Re\{x\} = (x + x^*)/2$ denotes the real part of a complex number x . The time evolution of the energy can thus be computed as follows

$$d_\tau E(\theta) = d_\tau \langle \Psi(\theta) | H | \Psi(\theta) \rangle = 2\Re\{\langle \Psi(\theta) | H (d_\tau |\Psi(\theta)\rangle)\} = 2\Re\{\langle \Psi(\theta) | H | \Psi_\parallel \rangle\}. \quad (3.0.5)$$

Inserting the adjoint of Eq. (3.O.2) and using Eq. (3.O.4), we can rewrite Eq. (3.O.5) as

$$d_\tau E(\theta) = 2\Re\{E \langle \Psi(\theta) | \Psi_\parallel \rangle - \langle R_\Psi | P_\theta | R_\Psi \rangle\} = -2\Re\{\langle R_\Psi | P_\theta | R_\Psi \rangle\} \leq 0, \quad (3.O.6)$$

where the last equation follows from the fact, that the expectation value of a projection operator to any given normalized state vector is a non-negative number in the range $[0, 1]$. Therefore within the VM the energy is monotonically decreasing in time when undergoing an imaginary-time evolution.

3.O.2 Time-dependence of the energy as a function of the CM and NGS parameters

We now turn to the first term on the right-hand side of Eq. (3.6.38). We rewrite Eq. (3.6.8) as

$$d_\tau \Gamma_m = \frac{1}{2} [\Gamma_m, [\Gamma_m, \bar{H}_m]] + i [\Gamma_m, O_m]. \quad (3.O.7)$$

Since \bar{H}_m is anti-symmetric under transposition, we have

$$\sum_{i,j=1}^{2N_f} (\bar{H}_m)_{ij} d_\tau (\Gamma_m)_{ij} = -\text{tr}(\bar{H}_m (d_\tau \Gamma_m)). \quad (3.O.8)$$

Using the linearity and cyclic property of the trace, we can write

$$\text{tr}(\bar{H}_m [\Gamma_m, [\Gamma_m, \bar{H}_m]]) = \text{tr}([\bar{H}_m, \Gamma_m] [\Gamma_m, \bar{H}_m]). \quad (3.O.9)$$

Inserting Eqs. (3.O.7) and (3.O.9) into Eq. (3.O.8) leads to

$$\sum_{i,j=1}^{2N_f} (\bar{H}_m)_{ij} d_\tau (\Gamma_m)_{ij} = \frac{1}{2} \text{tr} \left(([\bar{H}_m, \Gamma_m] - iO_m)^2 \right) + \frac{1}{2} \text{tr}(O_m^2). \quad (3.O.10)$$

Inserting these results into Eq. (3.6.38) leads to Eq. (3.6.39).

Section 3.P

MISCELLANEOUS COMPUTATIONS

This Appendix contains small identities used in the main text.

3.P.1

$$\begin{aligned}
 & \langle \boldsymbol{\alpha} | e^{i \sum_{j=1}^{N_f} \tilde{\alpha}(j) n_j} c_{j_1}^\dagger \cdots c_{j_a}^\dagger c_{k_1} \cdots c_{k_a} | \boldsymbol{\alpha} \rangle \\
 &= \langle \boldsymbol{\alpha} | e^{i \tilde{\alpha}(j_1)} c_{j_1}^\dagger e^{i \sum_{j=1}^{N_f} \tilde{\alpha}(j) n_j} c_{j_2}^\dagger \cdots c_{j_a}^\dagger c_{k_1} \cdots c_{k_a} | \boldsymbol{\alpha} \rangle \\
 &= e^{i \sum_{l=1}^a \tilde{\alpha}(j_l)} \langle \boldsymbol{\alpha} | c_{j_1}^\dagger \cdots c_{j_a}^\dagger e^{i \sum_{j=1}^{N_f} \tilde{\alpha}(j) n_j} c_{k_1} \cdots c_{k_a} | \boldsymbol{\alpha} \rangle \\
 &= e^{i \sum_{l=1}^a \tilde{\alpha}(j_l)} \langle \boldsymbol{\alpha} | \alpha_{j_1}^* \cdots \alpha_{j_a}^* e^{i \sum_{j=1}^{N_f} \tilde{\alpha}(j) n_j} \alpha_{k_1} \cdots \alpha_{k_a} | \boldsymbol{\alpha} \rangle \\
 &= \langle \boldsymbol{\alpha} | \left(\prod_{l=1}^a e^{i \tilde{\alpha}(j_l)} \alpha_{j_l}^* \right) \left(\prod_{m=1}^a \alpha_{k_m} \right) e^{i \sum_{j=1}^{N_f} \tilde{\alpha}(j) n_j} | \boldsymbol{\alpha} \rangle. \tag{3.P.1}
 \end{aligned}$$

3.P.2

$$\begin{aligned}
 \mathcal{A}_{j_1^\dagger \dots j_a^\dagger k_1 \dots k_a} &= \int d^2 \boldsymbol{\alpha} \int d^2 \boldsymbol{\xi} e^{\sum_n (\alpha_n \xi_n^* - \xi_n \alpha_n^*)} \chi(\boldsymbol{\xi}) e^{\frac{1}{2} \sum_n \xi_n^* \xi_n} \\
 &\quad \times \langle \boldsymbol{\alpha} | \left(\prod_{l=1}^a e^{i \tilde{\alpha}(j_l)} \alpha_{j_l}^* \right) \left(\prod_{m=1}^a \alpha_{k_m} \right) e^{i \sum_{j=1}^{N_f} \tilde{\alpha}(j) n_j} | \boldsymbol{\alpha} \rangle \\
 &= \int d^2 \boldsymbol{\xi} \chi(\boldsymbol{\xi}) e^{\frac{1}{2} \sum_n \xi_n^* \xi_n} \int d^2 \boldsymbol{\alpha} e^{\sum_n (-\xi_n^* \alpha_n - \xi_n \alpha_n^*)} \\
 &\quad \times \left(\prod_{l=1}^a e^{i \tilde{\alpha}(j_l)} \alpha_{j_l}^* \right) \left(\prod_{m=1}^a \alpha_{k_m} \right) \left(\prod_{j=1}^{N_f} e^{(e^{i \tilde{\alpha}(j)} - 1) \alpha_j^* \alpha_j} \right) \\
 &= \int d^2 \boldsymbol{\xi} \chi(\boldsymbol{\xi}) e^{\frac{1}{2} \sum_n \xi_n^* \xi_n} \int d^2 \boldsymbol{\alpha} \left(\prod_{l=1}^a e^{i \tilde{\alpha}(j_l)} \alpha_{j_l}^* \right) \left(\prod_{m=1}^a \alpha_{k_m} \right) \\
 &\quad \times e^{\sum_n (-\xi_n^* \alpha_n - \xi_n \alpha_n^*)} e^{\sum_n (e^{i \tilde{\alpha}(n)} - 1) \alpha_n^* \alpha_n} \\
 &= \mathcal{F}_J \int d^2 \boldsymbol{\xi} \chi(\boldsymbol{\xi}) e^{\frac{1}{2} \sum_n \xi_n^* \xi_n} \\
 &\quad \times \int d^2 \boldsymbol{\alpha} \left(\prod_{n=1}^{N_f} e^{(J_n^* - \xi_n^*) \alpha_n + (J_n e^{i \tilde{\alpha}(n)} - \xi_n) \alpha_n^* + (e^{i \tilde{\alpha}(n)} - 1) \alpha_n^* \alpha_n} \right) \\
 &= \mathcal{F}_J \int d^2 \boldsymbol{\xi} \chi(\boldsymbol{\xi}) e^{\frac{1}{2} \sum_n \xi_n^* \xi_n} \\
 &\quad \times \int d^2 \boldsymbol{\alpha} \left(\prod_{n=1}^{N_f} e^{-(1 - e^{i \tilde{\alpha}(n)}) \alpha_n^* \alpha_n + (J_n^* - \xi_n^*) \alpha_n + (J_n e^{i \tilde{\alpha}(n)} - \xi_n) \alpha_n^*} \right). \tag{3.P.2}
 \end{aligned}$$

3.P.3

$$\begin{aligned}
 \mathcal{A}_{j_1^\dagger \dots j_a^\dagger k_1 \dots k_a} &= s_{N_f} \mathcal{F}_J e^{\sum_{k=1}^{N_f} \frac{J_k J_k^* e^{i\bar{\alpha}(k)}}{1 - e^{i\bar{\alpha}(k)}}} (2i)^{N_f} \int d\vec{\eta} e^{-\frac{1}{2} \vec{\eta}^T B' \vec{\eta} + \vec{K}^T \vec{\eta}} \\
 &= s_{N_f} \mathcal{F}_J e^{\sum_{k=1}^{N_f} \frac{J_k J_k^* e^{i\bar{\alpha}(k)}}{1 - e^{i\bar{\alpha}(k)}}} (2i)^{N_f} \text{Pf}(B') e^{-\frac{1}{2} \vec{K}^T B'^{-1} \vec{K}} \\
 &= \frac{s_{N_f}}{2^{N_f}} \mathcal{F}_J e^{\sum_{k=1}^{N_f} \frac{J_k J_k^* e^{i\bar{\alpha}(k)}}{1 - e^{i\bar{\alpha}(k)}}} \text{Pf}(\Gamma_F) e^{2i \vec{K}^T \Gamma_F^{-1} \vec{K}}. \tag{3.P.3}
 \end{aligned}$$

3.P.4

$$\begin{aligned}
 d_\tau U_{\text{GS}} &= \frac{i}{4} \int_0^1 du e^{\frac{i}{4} u A^T \xi A} \sum_{k,l=1}^{2N_f} A_k(d_\tau \xi)_{kl} A_l e^{-\frac{i}{4} u A^T \xi A} U_{\text{GS}} \\
 &= \frac{i}{4} \sum_{k,l,p,q=1}^{2N_f} \int_0^1 du (e^{-iu\xi})_{kp} A_p e^{\frac{i}{4} u A^T \xi A} (d_\tau \xi)_{kl} e^{-\frac{i}{4} u A^T \xi A} (e^{-iu\xi})_{lq} A_q U_{\text{GS}} \\
 &= \frac{i}{4} \sum_{k,l,p,q=1}^{2N_f} \int_0^1 du A_p (e^{iu\xi})_{pk} (d_\tau \xi)_{kl} (e^{-iu\xi})_{lq} A_q U_{\text{GS}} \\
 &= \frac{1}{4} A^T (d_\tau U_m) U_m^T A U_{\text{GS}}, \tag{3.P.4}
 \end{aligned}$$

where the last line follows again from applying the operator identity of Eq. (3.I.4), which necessitates the insertion of an the matrix identity $\mathbf{1}_{2N_f} = U_m U_m^T$.

3.P.5

$$\begin{aligned}
 \langle u_L \rangle_{\text{GS}} &= \frac{1}{4} \langle \Psi_{\text{GS}} | A^T (d_\tau U_m) U_m^T A | \Psi_{\text{GS}} \rangle = \frac{1}{4} \sum_{p,q,r=1}^{2N_f} \langle \Psi_{\text{GS}} | A_p (d_\tau U_m)_{pq} (U_m^T)_{qr} A_r | \Psi_{\text{GS}} \rangle \\
 &= \frac{i}{4} \sum_{p,q,r=1}^{2N_f} (d_\tau U_m)_{pq} (U_m^T)_{qr} (\Gamma_m)_{rp} = \frac{i}{4} \text{tr} [(d_\tau U_m) U_m^T \Gamma_m]. \tag{3.P.5}
 \end{aligned}$$

CHAPTER 4

**LINEAR AND LOGARITHMIC TIME
COMPOSITIONS OF QUANTUM
MANY-BODY OPERATORS**

In this chapter, we develop a generalized framework for constructing many-body-interaction operations either in linear time, or in logarithmic time with a linear number of ancilla qubits. Exact gate decompositions are given for Pauli strings, many-control Toffoli gates, number- and parity-conserving interactions, Unitary Coupled Cluster operations, and sparse matrix generators. We provide a linear time protocol, the *decoupling protocol*, that works by creating a superposition of exponentially many different possible operator strings and then uses dynamical decoupling methodology to undo all the unwanted terms. A logarithmic time protocol, the *selection protocol*, overcomes the speed limit of the first by using ancilla registers to condition evolution to the support of the desired many-body interaction before using parallel chaining operations to expand the string length. The two techniques improve substantially on current strategies (reductions in time and space ranging from linear to exponential), are applicable to different physical interaction mechanisms such as CNOT, XX , and $XX + YY$, and generalize to a wide range of many-body operators.

This chapter represents an altered form of work carried out in collaboration with Felix Motzoi (FM) and Frank Wilhelm-Mauch (FWM) in the years 2016-17, published in Physical Review Letters [176]. FM and the author contributed equally to the contents of Section 4.2-4.5, while all derivations in Sections 4.A-4.C were carried out by the author. FWM supervised the project. The reproduction seen in this chapter were approved by FM and FWM.

Section 4.1

INTRODUCTION

Generating multi-body entanglement is the hallmark of most quantum information technologies. Such technologies promise to harness entanglement across multiple quantum registers (qubits) to enable potentially significant improvements in speed or precision compared to their classical counterparts. Yet much of the difficulty in the control of quantum systems lies in the constraint that entanglement naturally arises on a local scale while scaling improvements occur as a result of wave functions spread over much larger spaces. More specifically, interactions used to create entangling operations on a quantum computer, are two-body interactions¹, while the operations which naturally appear in digital quantum simulations can contain operators which potentially create entanglement across the whole quantum register [18, 149–157].

Quantum circuits generating entanglement across n qubits in linear or sub-linear time (i.e. circuit depth) in n have been the subject of many studies, with direct use as sub-routines in quantum algorithms for factoring [30, 229], simulation [50, 69, 79, 102, 105, 169, 230–238], unstructured search [239, 240], error-correction [87, 241], and solutions to systems of differential equations [38, 230, 236, 242]. Much progress has been made for constructing many-body operations, either by finding architectures where commuting two-qubit interactions could be executed simultaneously on overlapping Hilbert spaces, or via cases where a particular many-body gate with known or suspected sublinear implementation can be used to synthesize other many-body circuits. The former has been used for so-called *collective* Pauli operations on qubits in ion chains [243–245], while the workhorse for the latter has been the *fanout* operation [246, 247], which has successfully lead to $O(\log(n))$ depth quantum circuits for various flavours of quantum adders [248–250], with related arithmetic operations [229, 251].

Other many-body interactions have also been synthesized to mixed success. The ubiquitous multi-control NOT has found general linear-depth implementations, though with a relatively large prefactor [252–254]. Likewise, operations have been sought for rotating between two arbitrary multi-qubit states, for use in sparse matrix generation [38, 230, 236], or, equivalently, pairwise inversion of opposing spins in Unitary Couple Cluster (UCC) theory [135, 255–258]. Here, proposals have typically involved (based on intended application) either multi-control CNOT and arithmetic gates with linear ancillary memory [38, 236], or Trotter decomposition of the dynamics into Pauli-string factors but with no ancillas, which however results in an exponentially growing numbers of Pauli-string terms with system size.

In this chapter, we provide a formalism on how to directly compose a wide class of such many-body entangling operations, each generated by an equivalent many-body operator \bar{H} , using one- and two-local interactions and for which the above discussed protocols and algorithms form important examples of its application. In some cases, we will use operations which are more than two-local, in which case they however can be easily decomposed into the latter.

We aim to minimize two standard figures of merit of the generic circuit, namely its

¹We will use the terms n -local and n -body equivalently.

depth, defined as the number of layers of gates acting simultaneously on disjoint sets of qubits, and its *width*, defined as the total number of qubits acted on by the circuit [259]. We find a width-optimized general algorithm, which we label the *decoupling protocol*, to compose \bar{H} with zero or constant memory overhead and depth limited to linear n scaling. Moreover, we demonstrate a depth-optimized algorithm for simulating \bar{H} , the *selection protocol*, which has logarithmic depth and requires at most linear memory overhead. We demonstrate the formalism towards the linear or exponential speed-up of the aforementioned examples, given in our notation by

$$\bar{H} = \begin{cases} \prod_{i=1}^n X_i, & \text{n-qubit Pauli strings} \\ (\prod_{i=1}^n P_i) X_{n+1}, & \text{an n-controlled-NOT gate} \\ \prod_{i=1}^{n/2} (\sigma_{2i-1}^+ \sigma_{2i}^- + h.c.), & \text{number (or parity) conserving string} \\ (\prod_{i=1}^n \sigma_i^+) + h.c. & \text{argument of UCC- and sparse matrix-type generators,} \end{cases}$$

where X_i, Y_i and Z_i are Pauli operators acting on qubit i , $\sigma_j^\pm = (X_j \mp iY_j)/2$ are the spin-raising and -lowering operators and $P_i = |1\rangle\langle 1|_i$ is the projector onto the $|1\rangle$ -state of qubit i . The common notation to write tensor products between disjoint Hilbert subspaces as regular products is used. The operator generated by the many-body composed operator \bar{H} can then be written succinctly via the notation

$$[\bar{H}]^\alpha \equiv \exp(-i\alpha\bar{H}), \tag{4.1.1}$$

with real-valued rotation parameter α .

In what follows, we will introduce the decoupling and selection protocols and show how these can construct operators as given in Eq. (4.1.1) from two-body interactions and improve upon prior construction schemes.

Section 4.2

NOTATION

Let \mathcal{H} denote an Hilbert space spanned by n subspaces \mathcal{H}_j (where each subspace could be a single, or multi-qubit Hilbert space), $\mathcal{H} = \mathcal{H}_1 \otimes \mathcal{H}_2 \cdots \mathcal{H}_n$. We will introduce a symbolic notation, that aims at making a somewhat simplified effort to illustrate the two protocols. We denote with $R_i = \mathbb{1}^{\otimes(i-1)} \otimes R \otimes \mathbb{1}^{\otimes(n-i)}$, and similarly $S_j = \mathbb{1}^{\otimes(j-1)} \otimes S \otimes \mathbb{1}^{\otimes(n-j)}$, two sets of operators, where each set only acts non-trivially (meaning, not just as the identity) within the subspaces \mathcal{H}_i and \mathcal{H}_j . For us, the rank of the operators contained in R are always greater or equal than the rank of the operators in S . The rank of an operator is defined as the dimension of the image of the linear transformation that is given by multiplication of a quantum state within that subsystem with that operator.

As a simple example, we could have $R_i = \{X_k X_l, Y_k Z_l\}$, since the support of both operators is identical (the Pauli-strings have full rank, thus R_i would have rank four). The indices k, l of the Pauli-strings correspond to qubit k and qubit l , while the index i refers to the Hilbert subspace \mathcal{H}_i which is spanned by the basis vectors $\{|0\rangle_k |0\rangle_l, |0\rangle_k |1\rangle_l, |1\rangle_k |0\rangle_l, |1\rangle_k |1\rangle_l\}$. An example of an operator S_i which would live in the same Hilbert subspace as R_i , but have a smaller support (rank), is given by $S_i = \{\sigma_k^+ \sigma_l^- + \sigma_k^- \sigma_l^+, -i(\sigma_k^+ \sigma_l^- - \sigma_k^- \sigma_l^+)\}$, where the two elements can be transformed via single qubit unitary transformations and S has rank two, as its operators only have support on $\{|0\rangle_k |1\rangle_l, |1\rangle_k |0\rangle_l\}$.

If the rank of S_i is smaller than the rank of R_i , we can define a third set of operators which lie in the support of R_i but in the kernel of S_i (we will indicate the kernel with \perp) and symbolically write $R_i = S_i + S_i^\perp$.

The hardest part in understanding the protocols is trying to disentangle the symbolic notation, we therefore include some explicit examples at the end of this chapter. It is by chaining together the lower-rank S_i operators that we will be able to construct our many-body dynamics \bar{H} . We will see that besides creating the desired operator \bar{H} , the chaining operations will result in a plethora of unwanted additional terms and the elimination of the latter is the objective of the decoupling and selection protocols.

Section 4.3

THE DECOUPLING PROTOCOL

Our main tool is a unitary operator $U_{i,j}$, which will be used to iteratively increase in length a string of operators $S_1 S_2 \cdots S_j$. This unitary operator is at least a two-body operator and can be viewed as an entangling operation. Recall that, generally, S_j must not be of full rank, and so $U_{i,j}$ will invariably have to also act outside the support of S_j . Thus, our protocol will have to execute the desired system dynamics (given by \bar{H}) while leaving the rest of the Hilbert space (namely the kernels of S_j) intact. Though the decoupling protocol could be applicable to a wide range of entangling operations, we will be looking at the specific case, whereby a unitary operator $U_{i,j}$ acts on some operator R_i via

$$U_{i,j}^\dagger R_i U_{i,j} = S_i R_j + S_i^\perp + S_i R_j^\perp, \quad (4.3.1)$$

stressing that the entangler's action is chosen such that it covers the example cases as well as the prerequisites of the selection protocol. A unitary transformation via $U_{i,j}$ described by Eq. (4.3.1) therefore increments the length of a string of non-identity operators by one when acting on R_i . The kernel of R_j , here denoted by R_j^\perp , is not necessarily empty, since R_j may have smaller rank than a full rank operator living in \mathcal{H}_j . If, instead of just acting on the operator R_i , we consider the action of the unitary transformation in Eq. (4.3.1) on the exponential $[R_i]^\alpha$, we know that from expansion of the exponential it follows, that

$$U_{i,j}^\dagger [R_i]^\alpha U_{i,j} = [U_{i,j}^\dagger R_i U_{i,j}]^\alpha. \quad (4.3.2)$$

By building staircase circuits of the entangling operation as in Fig.4.3.1, we obtain a new operator \hat{H} , which contains a large sum of terms, which have support on various parts on the Hilbert space. More specifically, it results in

$$\begin{aligned} \hat{H} &= \left(\prod_{i=1}^n S_i \right) R_{n+1} + \sum_{m=1}^n \left(\prod_{i=1}^{m-1} S_i \right) (S_m R_{m+1}^\perp + S_m^\perp) \\ &\equiv \bar{H} + \Sigma_{\text{res}}, \end{aligned} \quad (4.3.3)$$

where $\bar{H} = (\prod_{i=1}^n S_i) R_{n+1}$ is the desired many-body operator, we wish to isolate and Σ_{res} contains the exponentially growing number of residue terms we wish to eliminate. We give a proof of Eq. (4.3.3) in Appendix 4.A.

The residue terms Σ_{res} commute with \bar{H} , while also acting as the identity on the support of R_{n+1} . Thus, if we can find a unitary operator M_{n+1} such that it imparts an opposite phase to R_{n+1} (and thus \bar{H}), but does not change Σ_{res} ², we can remove Σ_{res} via the decoupling protocol,

$$[\bar{H}]^{2\alpha} = [\hat{H}]^\alpha M_{n+1}^\dagger [\hat{H}]^{-\alpha} M_{n+1}, \quad (4.3.4)$$

²A sufficient condition that such a transformation exists is that the eigenvalues $\{\lambda_{n+1}^{(i)}\}$ of R_{n+1} are symmetric, i.e. for each eigenvalue $\lambda_{n+1}^{(i)}$ there exists a corresponding eigenvalue $\lambda_{n+1}^{(j)}$ ($i \neq j$) such that $\lambda_{n+1}^{(i)} = -\lambda_{n+1}^{(j)}$.

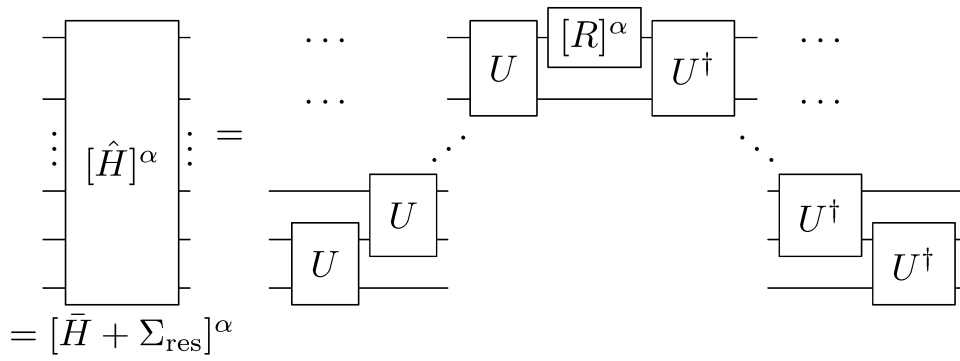


Figure 4.3.1: Staircase circuit for generating the unitary dynamics $[\hat{H}]^\alpha = [\bar{H} + \Sigma_{\text{res}}]^\alpha$ which follow from Eqs. (4.3.1) and (4.3.3) using one single-body operator which creates a rotation of R about an angle α and $2n$ (nearest-neighbor) entanglers $U_{j-1,j}$, when the total number of quantum registers in the staircase circuit is $n + 1$.

as presented in Fig. 4.3.2. The exact form of the functional form of the decoupling protocol is not set in stone, any gate construction which will introduce a sign flip exclusively in front of the desired term \bar{H} while leaving the remainder unaffected will do. To construct an effective Hamiltonian \bar{H} of string-length $n + 1$, a total of $4n$ unitary operators $U_{i,j}$ are needed. Note that if R is full rank, Eq. (4.3.1) reduces to $U_{i,j}^\dagger R_i U_{i,j} = R_i R_j$ and only half as many operators are used, since decoupling is not required.

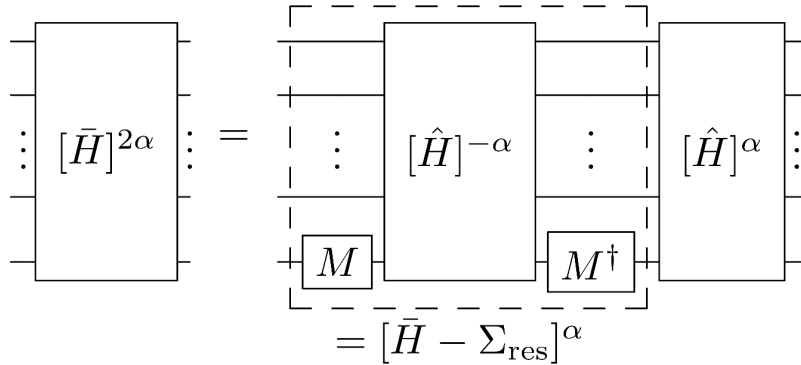


Figure 4.3.2: Gate sequence for realizing the decoupling protocol given by Eq. (4.3.4) using a total of $4n$ entangling operators $U_{j-1,j}$ and two single qubit operators M and M^\dagger , which are applied to the last index of \bar{H} (at angle $-\alpha$) in order to introduce a minus sign. The succeeding application of the un-rotated staircase circuit with an opposite phase α will cancel all residue terms, since by construction, Σ_{res} and \bar{H} commute, see the first line of Eq. (4.3.3).

Section 4.4

THE SELECTION PROTOCOL

We now present a composition scheme that can further decrease the required circuit depth for n -body operators from a $\mathcal{O}(n)$ to a $\mathcal{O}(\log(n))$ scaling, at the cost of $n - 1$ ancillary qubits. Without loss of generality we set $n = 2^m$ where $m \in \mathbb{N}$ and introduce two sets of qubit indices, namely register qubits $q_{\text{reg}} = \{1, 2, \dots, n + 1\}$ containing the qubits of the desired string and ancilla qubits $q_{\text{anc}} = \{n + 2, n + 3, \dots, 2n\}$, where the latter are all initialized to the $|0\rangle$ state.

A NOT operation corresponds to the Pauli-gate X , as it negates the qubit register, $X|0\rangle = |1\rangle$ and $X|1\rangle = |0\rangle$. A Toffoli gate is a conditional NOT gate, depending on the control registers being in the $|1\rangle$ -state, see Table 4.4.1.

We introduce Toffoli-*type*³ operators $C_{i,j}X_k$, whereby a NOT operation is applied to qubit k conditioned on the state of qubits i, j both being in the $|1\rangle$ -state. Here, the indices i and j can either belong to register or ancillary qubits. The operation can be written mathematically as e.g. $C_{i,j}X_k = [S_i S_j X_k]^{\pi/2}$. For $i, j \in q_{\text{reg}}$, $C_{i,j}$ conditions the application of X_k on registers i and j being in the support of both S_i and S_j . As previously stated, if $i, j \in q_{\text{anc}}$, the operation is a standard Toffoli (thus $S_i = |1\rangle\langle 1|_i$). Note that if n is not a power of two, some $C_{i,j}X_k$ operations can have both register and ancilla qubits as controls. We define compound operations

$$C_{\text{tot}} \equiv \prod_{k=1}^{\log n} \left(\prod_{l=n-2^k+1}^{n-2^{k-1}} C_{2l+1, 2l} X_{n+l+1} \right), \quad (4.4.1)$$

$$U_{\text{tot}} \equiv \prod_{k=\log n}^1 \left(\prod_{l=1}^{n/2^k} U_{l2^k+1, (l-\frac{1}{2})2^k+1} \right) U_{1, n+1}, \quad (4.4.2)$$

as displayed in Fig. 4.4.1. Eq. (4.4.2) is a logarithmically compressed staircase circuit, similar to Eq. (4.3.3) and Eq. (4.4.1) is the selection operation which applies the operator X only on the part of the ancillary register, if the register qubits are in the desired support of the operator \bar{H} . All operations in Eq. (4.4.1) and (4.4.2) which are contained within the same bracket, may be executed in parallel. Successive operators within the same bracket are highlighted in Fig. 4.4.1 by shaded and un-shaded regions.

Table 4.4.1: A truth table for the Toffoli gate $C_{i,j}X_x$, where \bar{y} is the negation of y , i.e. $\bar{y} = 1 - y$. This can be similarly extended to an n -controlled NOT $C_{i_1, \dots, i_n} X_{n+1}$.

y of $ y\rangle_i$	0	1	0	1
y of $ y\rangle_j$	0	0	1	1
$C_{i,j}X_k$ $ y\rangle_i y\rangle_j y\rangle_k$	$ y\rangle_i y\rangle_j y\rangle_k$	$ y\rangle_i y\rangle_j y\rangle_k$	$ y\rangle_i y\rangle_j y\rangle_k$	$ y\rangle_i y\rangle_j \bar{y}\rangle_k$

³The operator $C_{i,j}X_k$ is identical to a regular Toffoli, if both control registers i and j are elements of the Toffoli gate, as indicated in Fig.4.4.1 with two full circles.

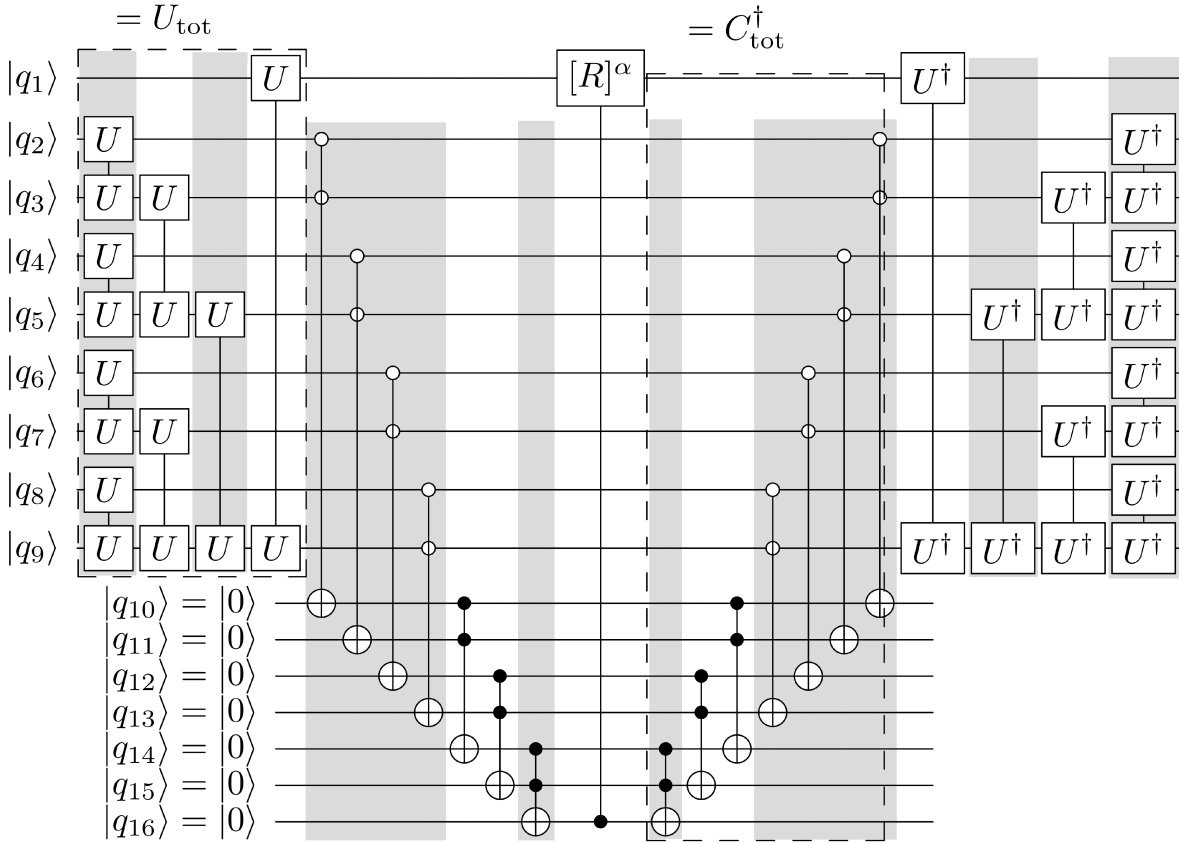


Figure 4.4.1: Gate sequence for realizing the selection protocol given by Eq. (4.4.3) for a total of 16 qubits, with nine qubits belonging to the register qubits and seven to the ancillary qubits. dashed boxed on the left-hand side marks the entangling operation U_{tot} defined in Eq. (4.4.2), while the right dashed box marks the adjoint of the operation C_{tot} as defined in Eq. (4.4.1). The individual three qubit gates of the latter are Toffoly-*type* gates, which apply a NOT operation on the ancillary qubits by projecting onto the support of S (onto $|1\rangle\langle 1|$), if the control qubits belong to the register (ancillary) qubits. The middle operation is a controlled operation, which applies $[|1\rangle\langle 1|_{16} R_1]^\alpha$. Instead of generating \bar{H} of Eq. (4.3.3), the combination of C_{tot} and the middle conditional operation guarantee, that only the desired operator \bar{H} is expanded, while no rotation $[|1\rangle\langle 1|_{16} R_1]^\alpha$ is applied to states that are outside of the support of \bar{H} , thus no Σ_{res} are generated. The barcode-like shading indicates segments which can be run in parallel on quantum architectures with all-to-all qubit connectivity.

If $\text{rank}(S) = 1$, we set $U_{\text{tot}} = \mathbb{1}$, while if S has full rank, we set $C_{\text{tot}} = \mathbb{1}$. The full selection protocol is given by the sequence

$$[\bar{H}]^\alpha = U_{\text{tot}}^\dagger C_{\text{tot}}^\dagger (C[R_1]^\alpha) C_{\text{tot}} U_{\text{tot}}, \quad (4.4.3)$$

and is summarized in Fig. (4.4.1), as well as in a more condensed, but equivalent Fig. 4.4.2.

The middle operator in Eq. (4.4.3) is defined as $C[R_1]^\alpha = [R_1]^\alpha$, if S full rank and as $[|1\rangle\langle 1|_{2n} R_1]^\alpha$ otherwise. The result of the sequence is that the many-body rotation is applied only on states that are supported by \bar{H} , while the identity is applied otherwise.

The selection protocol improves on previous generic algorithms by quadratically reducing the space requirements [246].

While our construction schemes are not general, in the sense that they allow us to construct any many-body operator we desire, they can be applied to a fairly large family of many-body operators, among which we find the expressions stated at the end of Section 4.1. The following sections demonstrate how to apply the *decoupling* and *selection* protocols to existing problems, to either take advantage of specific two-body interaction mechanisms, or to reduce the time and gate complexity of known implementations.

As a remark, we want to mention that a recent paper also found logarithmic depth construction for multi-control Toffoli gates with a linear number of ancillary qubits (which we present in Section 4.5.3), but failed to reference our work [260]. Also, we were made aware of the fact that the existence of a fermionic SWAP gate allows one to swap register qubits in layers, such that after n such layers, each spin-orbital (which is indexed by the subscript) was adjacent to each other spin-orbital at some point in those layers [169]. One can thus devise a procedure where the UCC-type terms we generate in Section 4.5.4 are identical to the unitary coupled cluster operators, as the requirement of additional Pauli- Z strings due to the Jordan-Wigner mapping vanishes when all spin-orbitals are nearest neighbor.

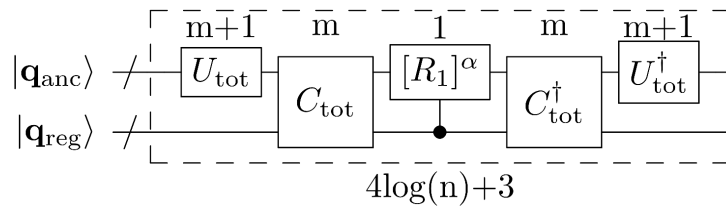


Figure 4.4.2: Generalized version of Fig. (4.4.1), where all register and ancillary qubits \mathbf{q}_{reg} and \mathbf{q}_{anc} are compounded to one circuit line. All operations that can be run in parallel are synthesized to one box, with the number above it indicating how many non-commutative time steps are necessary for each box. For $n = 2^m$, a string of length $n + 1$ is composed in $\mathcal{O}(\log(n))$ depth, provided an all-to all connectivity.

Section 4.5

EXAMPLES

4.5.1 Strings of Pauli operators

Well known formulas exist for forming strings of Pauli operators in linear time (e.g. [6, 261]), which we first reproduce using our formalism, as a sanity check. Since Pauli operators are full rank, $\hat{H} = R_1 R_2 \cdots R_j = \bar{H}$ and there is no need for a decoupling sequence, since nor residue terms Σ_{res} appear. The optimal form of $U_{i,j}$ will depend on the architecture of the quantum computing platform and its native, i.e. most simple-to-realize entangling gate.

A standard entangling operation is via the CNOT gate $U_{i,j} = C_i X_j$ [6]. This gate, if applied as in Fig. 4.3.1, produces a many-body operator \bar{H} , without any residue terms Σ_{res} . However, the CNOT gate is neither natural for superconducting, nor for trapped ion qubits. A native gate for ion-trap designs is the Mølmer-Sørensen gate (MSG), where commuting $U_{i,j} = [X_i X_j]^{\frac{\pi}{4}}$ interactions can be applied simultaneously [243, 262]. We present a third composition with the same size as the CNOT and MSG circuits, designed for architectures with exchange gate interactions, $G_{i,j} = \frac{1}{2}(X_i X_j + Y_i Y_j)$, which is the fastest perfect entangler for most circuit-QED quantum processors [263, 264], quantum dot spins coupled by a cavity [265] and for nuclear spins interacting via a two-dimensional electron gas [266].

Here, $U_{i,j} = [G_{i,j}]^{-\frac{\pi}{2}}$ produces the iSWAP gate. Eq. (4.3.1) and $U_{i,j}^\dagger Y_i U_{i,j} = Z_i X_i$ allow us to construct a Pauli-string of length n using $2(n-1)$ iSWAP gates. In our symbolic notation, $R_i = \{X_i, Y_i, Z_i\}$ and $S_i = R_i$. Since the Pauli operators are full rank, one can reduce the depth of the circuit from $\mathcal{O}(n)$ to $\mathcal{O}(\log(n))$ by using the selection protocol, more specifically, Eq.(4.4.2) with $\prod_{i=1}^n R_i = U_{\text{tot}}^\dagger R_1 U_{\text{tot}}$, without needing any ancillary qubits. Note further that instead of increasing the length of the string using $U_{i,j}$, one can also use the inverse operation to remove a qubit from the string, e.g. to form a disconnected string from a nearest neighbour architecture.

4.5.2 Number- and parity-conserving strings

We now turn to generating many-body operators that act conditionally only within the fixed excitation-number subspace. These are natural fit for a two-body, exchange gate interaction, $G_{i,j}^+ = \sigma_i^+ \sigma_j^- + \sigma_i^- \sigma_j^+$, which has the same symmetry, noting also its rank is smaller than $\dim(\mathcal{H}_i \otimes \mathcal{H}_j)$. Defining $F_{i,j} = -i(\sigma_i^+ \sigma_j^- - \sigma_i^- \sigma_j^+)$, we desire strings of $R_{i,j} = \{G_{i,j}, F_{i,j}\}$, where the indices of the operator R now explicitly refers to the spin-orbitals. We choose $U_{i,k,l} = [Z_i G_{k,l}]^{\frac{\pi}{4}}$ as the (now three-qubit) entangling operation, giving

$$U_{i,k,l}^\dagger G_{i,j} U_{i,k,l} = F_{i,j} G_{k,l} + G_{i,j} P_{\ker\{G_{k,l}\}}, \quad (4.5.1)$$

where $P_{\ker_{k,l}} = \frac{1}{2}(\mathbb{1} + Z_k Z_l)$ is a projector onto the kernel of $R_{k,l}$, which we denote with $R_{k,l}^\perp$. Following the steps in Eq. (4.3.3), one obtains

$$\hat{H} = \prod_{i=1}^{n/2} R_{2i-1,2i} + \sum_{i=1}^{n/2-1} \left(\prod_{j=1}^i R_{2j-1,2j} \right) R_{2i+1,2i+2}^\perp, \quad (4.5.2)$$

a many-body Hamiltonian that collectively excites and de-excites n qubits in a number-conserving way. We provide more details for the derivation of Eq. (4.5.2) in Appendix 4.B. By choosing $M_n = [Z_n]^\frac{\pi}{2}$ one can apply the decoupling sequence of Eq. (4.3.4) to pick out one particular number-conserving string (namely, the longest),

$$[\bar{H}]^\alpha = \left[\prod_{i=1}^{n/2} R_{2i-1,2i} \right]^\alpha. \quad (4.5.3)$$

More generally, parity can be conserved without conserving particle number, by applying local operations to transform operators in the string from $G_{i,j}$ to such as $\sigma_i^+ \sigma_j^+ + \sigma_i^- \sigma_j^-$, where only the parity (*even* or *odd* number of excitations) is conserved. As we detail in Appendix 4.B, the whole sequence takes $(2n - 4)$ $U_{i,j,k}$ gates, or equivalently $(6n - 10)$ iSWAPs. Half as many are required if \hat{H} is used instead. Alternatively, the many-body dynamics can be generated with the selection protocol at the cost of $(n - 1)$ ancillary qubits. For this, we can reuse $U_{i,j,k} = C_{i,j} X_k$ for $i, j \in q_{\text{reg}}$. That sequence uses a total of $(2n - 4)$ entanglers $U_{i,k,l}$ and $(n + 4)$ Toffolis in a circuit depth of $4 \log(n) + 3$.

4.5.3 Multi-control NOT gates

The extension of regular CNOT gates to such with multiple control registers, $C_{1,\dots,n} X_{n+1}$, have widespread use in quantum and reversible computation, including for circuit distillation [267], unstructured search [239], factorization [229], error-correction [241], and linear equations system solvers [38]. For our construction, $U_{i,j,k} = [P_i X_j P_k]^\frac{\pi}{2}$ (a Toffoli gate with a relative phase [254], though a regular Toffoli can also be used) acts on three qubits, recalling $P_i = |1\rangle\langle 1|_i$. The chaining operation is given by $U_{i,j,k} Z_i Z_j U_{i,j,k}^\dagger = -P_i Z_j Z_k + P_i^\perp Z_j$, with $P_i^\perp = |0\rangle\langle 0|_i$. Repeated application of the chaining operation on $R_{1,2} = Z_1 Z_2$ following Eq. (4.3.3) gives

$$\hat{H}_n = (-1)^n \prod_{i=1}^n P_i Z_{n+1} Z_{n+2} - \sum_{j=1}^n (-1)^j \prod_{k=1}^{j-1} P_k P_j^\perp Z_{j+1}, \quad (4.5.4)$$

see Appendix 4.C. Choosing $M_{n+2} = [X_{n+2}]^\frac{\pi}{2}$, one can use the decoupling protocol of Eq. (4.3.4) to obtain a multi-qubit-controlled rotation around an arbitrary angle. For a phase-less multi-control NOT gate (which is the standard multi-controlled Toffoli gate, whose matrix representation consists only of ones and zeros), one can compose instead the sequence

$$C_{1,\dots,n} Z_{n+2} = [\hat{H}_n]^\frac{\pi}{2} M_{n+1}^\dagger [\hat{H}_{n-1}]^{-\pi/2} M_{n+1}, \quad (4.5.5)$$

using $4n - 2$ Toffolis, cutting down in half the size and depth of the longstanding construction [252–254]. Using single qubit rotations, the Pauli- Z term in Eq. (4.5.5) can be rotated into the native NOT representation.

A more drastic reduction results with the selection protocol, where $S_i = P_i$ and thus $\text{rank}(S) = 1$. Eq. (4.4.3) simplifies to

$$C_{2,\dots,n+1}X_1 = C_{\text{tot}}^\dagger(C_{n+1}X_1)C_{\text{tot}}, \quad (4.5.6)$$

resulting in an n -control NOT gate using $2(n - 1)$ Toffoli gates and $2 \log(n)$ depth (cf. Fig. 4.5.1). This gives exponential speedup in circuit depth compared to the $O(n)$ ancilla solution found in [254].

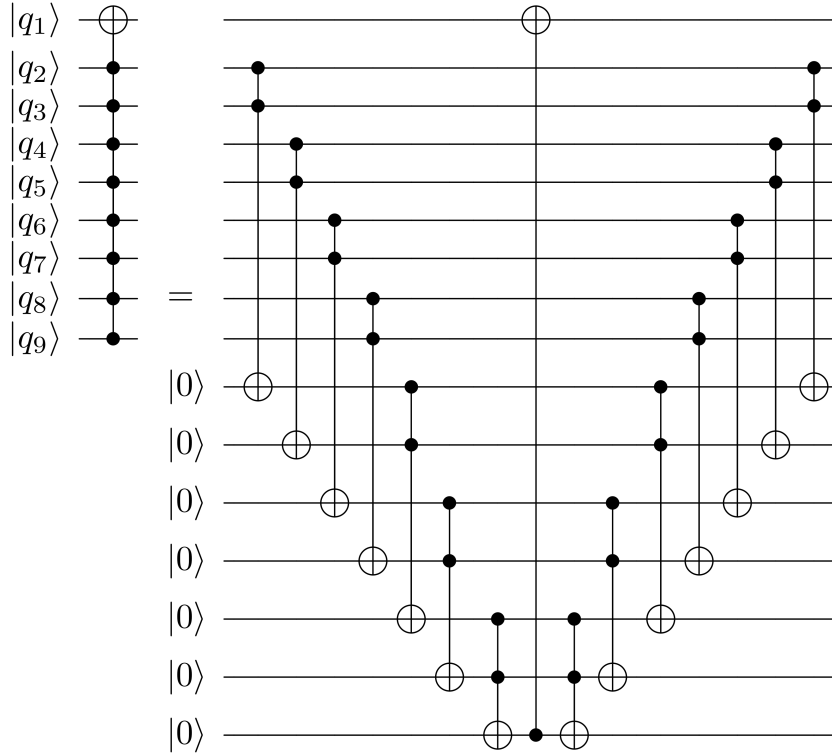


Figure 4.5.1: Explicit construction of a multi-controlled NOT gate using the selection protocol as given in Section 4.4. Since the rank of $S = |1\rangle\langle 1|$ is equal to one, the staircase part of Eq. (4.4.2) is lacking in Eq. (4.4.3). In the above circuit, the qubit 1 is the target, while qubits 2-9 are the controls.

4.5.4 Unitary coupled cluster

Another many-body operator which is frequently used (e.g. in quantum chemistry algorithms for computing energy landscapes [80, 135, 257]), is an operator which transfers population between electronic spin-orbitals (encoded in the qubits) while conserving electron number and spin. More generally, when the operator couples arbitrary many-qubit states, it corresponds to a sparse matrix off-diagonal element [38]. It takes the form

$$\text{UCC}(m, n) \equiv \prod_{i=1}^m \sigma_i^+ \prod_{j=m+1}^{m+n} \sigma_j^- + h.c., \quad (4.5.7)$$

where we typically have $m = n$, in which case the above operator times a phase factor corresponds to a single unitary coupled cluster (UCC)-*type* operator⁴. Due to the missing

⁴The case $m \neq n$ does not conserve particle number and parity.

Pauli- Z strings which are a result of the Jordan-Wigner transformation, Eq. (4.5.7) only corresponds to a UCC operator, provided that the appearing indices are nearest neighbor spin-orbitals. This can however be achieved using layers of a linear number of fermionic SWAP gates as outlined shortly in Section 4.4 and in more detail in [169]. Alternatively, one can create the necessary Pauli- Z operators by sandwiching the construction between additional staircase circuits of iSWAP gates as presented in Section 4.5.1.

We have seen in the section on Pauli strings how to construct $R = \prod_{i=1}^{m+n} X_i$ using $2(m+n-1)$ iSWAP gates. Since $\prod_{i=1}^{m+n} X_i$ contains all 2^{m+n} combinations of products of σ^+ and σ^- , we use $U = [(\prod_{i=1}^m P_i)(\prod_{j=m+1}^n P_j^\perp)X_{n+m+1}]^{\frac{\pi}{2}}$, which we know how to construct from Section 4.5.3, computing $\hat{H} = U^\dagger R U$ to get

$$\begin{aligned} \hat{H} = & \prod_{i=1}^{m+n} X_i - \left(\prod_{i=1}^m \sigma_i^+ \prod_{j=m+1}^{m+n} \sigma_j^- + h.c. \right) \\ & + i \left(\prod_{i=1}^m \sigma_i^+ \prod_{j=m+1}^{m+n} \sigma_j^- - h.c. \right) X_{m+n+1}. \end{aligned} \quad (4.5.8)$$

By setting $M = [Z_{m+n+1}]^{\frac{\pi}{2}}$ we can apply the decoupling protocol to construct the unitary dynamics of the UCC(m, n) operator in a circuit with $m+n+1$ qubits (q_{m+n+1} is an ancillary qubit) using $4(m+n-1)$ iSWAPs and $4(m+n)$ (relative phase) Toffolis,

$$[\bar{H}]^{2\alpha} = [\hat{H}]^\alpha M^\dagger [\hat{H}]^{-\alpha} M = [\text{UCC}(m, n)]^{2\alpha}. \quad (4.5.9)$$

Conventional factorization of the UCC(m, n) terms into Pauli strings scales exponentially as $O(2^{m+n-1})$ in the number of two-qubit gates cost whereas only $36(m+n)+O(1)$ iSWAPs are required when using our decoupling protocol. Even further decrease in composition time is once again achieved if the Pauli string and multi-CNOT gates are produced using the selection protocol, down to a depth of $O(\log n)$.

Section 4.6

ARCHITECTURAL CONSIDERATIONS

Clearly, any time and space complexity advantages will be subject to limitations set by the architecture, which is essentially a comprising characterization of the qubit chip details, e.g. which qubits a single qubit can couple to directly. One of the major bottlenecks of our logarithmic scaling is that it assumes that we have a fully connected qubit architecture, i.o.w. a chip where each qubit can interact with each other qubit on the chip, without crosstalk. Even more so, we neglect quantum error-correction. All our qubits are assumed to be perfect, i.e. logical qubits and the gates are assumed to be error-less as well. While these are common assumptions in most theoretical works, we stress that reality does not grant us such freedom just yet. Current-day quantum chips are usually containing qubits in a linear or two-dimensional grid, where the number of qubits each qubit can couple to is at most two and four.

The presence of $O(n)$ ancillary qubits (needed for exponential speedup) is actually fairly easy to include, as most architectures have ancillary electronic, motional or photon bus degrees of freedom. Despite much worse lifetimes typically found in these states, the (linear) tradeoff in error rate is more than made up by an exponential speedup in time and justifies their use for many-body gates. Note previous generic $O(\log(n))$ circuit constructions require $O(n^2)$ space [246], which may be practically infeasible. As mentioned before, a recently published work shows how to generate multi-controlled Toffolis with $O(n)$ ancillary qubits and $O(\log(n))$ circuit depth [260]. The adjacency graph of qubits that couple to each other in the architecture will also greatly impact composition time. For many-body operators spanning much of the graph, the circuit depth can range from $O(\log(n))$ when the depth of the spanning tree is $O(\log(n))$, as in [268–274], to $O(\sqrt[d]{n})$ for d -dimensional, nearest-neighbor architectures being expected [275, 276].

Section 4.7

CONCLUSION

We have developed two protocols, the decoupling and selection protocols, to generate many-body operators in $O(n)$ time for zero or constant memory overhead, and $O(\log(n))$ time for $O(n)$ ancillas, respectively. The former enhances previous constant-overhead approaches, with improvement ranging for prominent examples from linear (multi-control CNOT) to exponential (UCC). Our construction to bring down further the runtime to $O(\log(n))$ depth also improves quadratically on the space requirements of previous generic methods. As one of our main results, we exponentially reduce the circuit depth needed to construct an n -control NOT gate upon previous compositions, that also used an $O(n)$ ancillary qubit overhead. Our approach is generated directly from Hamiltonian dynamics, allowing straightforward incorporation of different coupling mechanisms and architectures.

APPENDIX

Section 4.A

ACTION OF THE STAIRCASE CIRCUIT

Though the decoupling protocol can be applied to a wide range of entangling operations, we will be looking at the specific case whereby a unitary operator $U_{i,j}$ acts on some operator R_i via Eq. (4.3.1), stressing that the entangler's action is chosen so that it covers the example cases as well as the prerequisite of the selection protocol. We will now prove the result of Eq. (4.3.3) of the staircase circuit of Fig. 4.3.1 from the main text,

$$\begin{aligned}\hat{H}_n &= \left(\prod_{j=1}^n U_{j,j+1} \right)^\dagger R_1 \left(\prod_{j=1}^n U_{j,j+1} \right) \\ &= \left(\prod_{i=1}^n S_i \right) R_{n+1} + \sum_{m=1}^n \left(\prod_{i=1}^{m-1} S_i \right) (S_m R_{m+1}^\perp + S_m^\perp),\end{aligned}\tag{4.A.1}$$

where we set $\hat{H} = \hat{H}_n$ to clearly indicate that Eq. (4.A.1) is the result after n entanglers in the staircase. We assume Eq. (4.A.1) holds for some fixed $n \in \mathbb{N}$. Then, using Eq. (4.3.1) gives

$$\begin{aligned}\hat{H}_{n+1} &= U_{n+1,n+2}^\dagger \hat{H}_n U_{n+1,n+2} \\ &= U_{n+1,n+2}^\dagger \left(\prod_{i=1}^n S_i \right) R_{n+1} U_{n+1,n+2} + \sum_{m=1}^n \left(\prod_{i=1}^{m-1} S_i \right) (S_m R_{m+1}^\perp + S_m^\perp) \\ &= \left(\prod_{i=1}^n S_i \right) (S_{n+1} R_{n+2} + S_{n+1}^\perp + R_{n+1} R_{n+2}^\perp) + \sum_{m=1}^n \left(\prod_{i=1}^{m-1} S_i \right) (S_m R_{m+1}^\perp + S_m^\perp) \\ &= \left(\prod_{i=1}^{n+1} S_i \right) R_{n+2} + \sum_{m=1}^{n+1} \left(\prod_{i=1}^{m-1} S_i \right) (S_m^\perp + S_m R_{m+1}^\perp).\end{aligned}\tag{4.A.2}$$

Therefore, Eq. (4.A.1) must hold for all $n \in \mathbb{N}$.

Section 4.B

DERIVATIONS FOR THE NUMBER AND PARITY CONSERVING TERMS VIA ISWAP GATES

For composing number-conserving strings, we have $U_{i,k,l} = [Z_i G_{kl}]^{\frac{\pi}{4}}$, with $G_{i,j} = \sigma_i^+ \sigma_j^- + \sigma_i^- \sigma_j^+$, $F_{i,j} = -i(\sigma_i^+ \sigma_j^- - \sigma_i^- \sigma_j^+)$, $P_{\ker\{G_{k,l}\}} = \frac{1}{2}(\mathbb{1} + Z_k Z_l)$ and $P_{\text{supp}\{G_{k,l}\}} = \frac{1}{2}(\mathbb{1} - Z_k Z_l)$. We use the result of Eq. (4.5.1), which can be verified by e.g. a simple matrix multiplication. From the relation $Z_i G_{i,j} = -i F_{i,j}$, it follows that

$$U_{i,k,l} F_{i,j} U_{i,k,l}^\dagger = F_{i,j} P_{G_{k,l}^\perp} + G_{i,j} G_{k,l}. \quad (4.B.1)$$

The generator $Z_i G_{k,l}$ can be obtained from iSWAPs through

$$[F_{i,l}]^{-\pi/2} [G_{i,k}]^\alpha [F_{i,l}]^{\pi/2} = [Z_i G_{k,l}]^\alpha. \quad (4.B.2)$$

For simplicity, we let n be even. We will start with entanglers which connect the first two qubits with the qubits three and four,

$$U_{1,3,4}^\dagger G_{1,2} U_{1,3,4} = F_{1,2} G_{3,4} + G_{1,2} P_{\ker\{G_{3,4}\}}. \quad (4.B.3)$$

Note that $P_{\ker\{G_{3,4}\}}$ commutes with the next entangler of the staircase circuit of Fig.4.3.1, yielding

$$U_{3,5,6}^\dagger U_{1,3,4}^\dagger G_{1,2} U_{1,3,4} U_{3,5,6} = F_{1,2} F_{3,4} G_{5,6} + G_{1,2} P_{\ker\{G_{3,4}\}} + F_{1,2} G_{3,4} P_{\ker\{G_{5,6}\}}. \quad (4.B.4)$$

Moving on in identical fashion, after the $n/2$ -th entangler we get

$$\begin{aligned} \hat{H}_{\frac{n}{2}} &= \left(\prod_{i=1}^{n/2} U_{2i,2i+1,2i+2} \right)^\dagger G_{1,2} \left(\prod_{i=1}^{n/2} U_{2i,2i+1,2i+2} \right) \\ &= G_{1,2} P_{\ker\{G_{3,4}\}} + \left(\prod_{m=1}^{n/2-1} F_{2m-1,2m} \right) G_{n-1,n} \\ &\quad + \sum_{m=1}^{n/2-2} \left(\prod_{k=1}^m F_{2k-1,2k} \right) G_{2m+1,2m+2} P_{\ker\{G_{2m+3,2m+4}\}} \end{aligned} \quad (4.B.5)$$

In terms of our symbolic notation, Eq. (4.B.3) is equivalent to

$$U_{i,k,l} R_{i,j} U_{i,k,l}^\dagger = R_{i,j} R_{k,l} + R_{i,j} R_{k,l}^\perp \quad (4.B.6)$$

while Eq. (4.B.5) may be written as

$$\hat{H}_{\frac{n}{2}} = \prod_{i=1}^{n/2} R_{2i-1,2i} + \sum_{i=1}^{n/2-1} \left(\prod_{j=1}^i R_{2j-1,2j} \right) R_{2i+1,2i+2}^\perp. \quad (4.B.7)$$

Section 4.C

DERIVATIONS FOR MULTI-CNOT GATES

For generating multi-control NOT gates, we have $U_{i,j,k} = [P_i X_j P_k]^{\frac{\pi}{2}}$ with $P_i = |1\rangle\langle 1|_i$. Note that one can just as well use a regular Toffoli for $U_{i,j,k}$ though typically the construction uses more gates than its relative phase version [254]. The following identity is central to the construction,

$$\begin{aligned} U_{i,j,k}^\dagger Z_i Z_j U_{i,j,k} &= Z_i Z_j P_k^\perp + P_i^\perp Z_j P_k + P_i Z_j P_k \\ &= Z_i Z_j P_k^\perp + Z_j P_k \\ &= -P_i Z_j Z_k + P_i^\perp Z_j, \end{aligned} \tag{4.C.1}$$

and can be verified most easily by comparing the left and right-hand side matrices. From Eq.(4.B.3), we have $U_{1,2,3}^\dagger Z_1 Z_2 U_{1,2,3} = -P_1 Z_2 Z_3 + P_1^\perp Z_2$. Since $U_{2,3,4}$ commutes with $P_1^\perp Z_2$, we get for the second step on the staircase circuit

$$\begin{aligned} U_{2,3,4}^\dagger U_{1,2,3}^\dagger Z_1 Z_2 U_{1,2,3} U_{2,3,4} &= P_1^\perp Z_2 - U_{2,3,4}^\dagger P_1 Z_2 Z_3 U_{2,3,4} \\ &= P_1 P_2 Z_3 Z_4 - P_1 P_2^\perp Z_3 + P_1^\perp Z_2. \end{aligned} \tag{4.C.2}$$

We assume now, that for a fixed $n \in \mathbb{N}$, the staircase circuit of Fig. 4.3.1 yields

$$\begin{aligned} \hat{H}_n &= \left(\prod_{j=1}^n U_{j,j+1,j+2} \right)^\dagger Z_1 Z_2 \left(\prod_{j=1}^n U_{j,j+1,j+2} \right) \\ &= (-1)^n \left(\prod_{i=1}^n P_i \right) Z_{n+1} Z_{n+2} + \sum_{j=1}^n (-1)^{j+1} \left(\prod_{k=1}^{j-1} P_k \right) P_j^\perp Z_{j+1}. \end{aligned} \tag{4.C.3}$$

Then, we use Eq. (4.C.1) to compute the $n + 1$ -th step,

$$\begin{aligned} \hat{H}_{n+1} &= U_{n+1,n+2,n+3}^\dagger \hat{H}_n U_{n+1,n+2,n+3} \\ &= (-1)^n \left(\prod_{i=1}^n P_i \right) (-P_{n+1} Z_{n+2} Z_{n+3} + P_{n+1}^\perp Z_{n+2}) + \sum_{j=1}^n (-1)^{j+1} \left(\prod_{k=1}^{j-1} P_k \right) P_j^\perp Z_{j+1} \\ &= (-1)^{n+1} \left(\prod_{i=1}^{n+1} P_i \right) Z_{n+2} Z_{n+3} + \sum_{j=1}^{n+1} (-1)^{j+1} \left(\prod_{k=1}^{j-1} P_k \right) P_j^\perp Z_{j+1}, \end{aligned} \tag{4.C.4}$$

therefore Eq. (4.C.3) holds for all $n \in \mathbb{N}$. At this point either the usual mirroring pulse can be used to decouple unwanted terms, or, as mentioned in the main text, a full rotation on the last qubit (here, the qubit with the highest index) can be used to remove the unwanted phase and generate a phase-less multi-control Toffoli [253].

CONCLUDING REMARKS

In this work, we have presented the first *ab-initio* roadmap for studying a system of interacting electrons, whose movement is restricted to a two-dimensional plane which is pierced by a strong transversal magnetic field, on a quantum computer. We derived a closed form for the two-body Coulomb matrix elements of the Hamiltonian that incorporates Landau level mixing in symmetric Gauge and presented approximate T gate counts for the simulation of the spectrum of the underlying Hamiltonian (in the lowest Landau level) using the currently most efficient quantum algorithm for studying the spectrum of electronic structure Hamiltonians [143]. We have investigated the performance of classical methods [23, 63] to find an approximate ground state that are efficiently implementable on noisy, as well as quantum error-corrected quantum computers. These numerical simulations should be extended to larger system sizes to see how the sub-exponential trend of the required Slater determinants (to get high-fidelity initial states through the ASCI algorithm) continues and compare it to other approaches. Future studies should also incorporate scaling analysis for the T gate complexity when Landau level mixing is included and focus on studying efficient algorithms that implement the Laughlin wave function as an initial state. Furthermore, a more thorough study of quantum algorithms that can exploit the fact that we have a closed form for the Hamiltonian coefficients should be performed.

A novel method to study approximate ground states, determine collective excitations, and to describe out of equilibrium dynamics of strongly interacting bosonic and fermionic systems has been recently introduced by Shi et al. [41]. While a variety of physical models have been successfully studied using this variational method, the simulation of long-range interacting fermionic systems has yet to be realized. This is mainly due to the fact that the presence of the four-body terms in the second quantized Hamiltonian greatly complicates the equations of motion for the variational parameters. Our work closed this gap by presenting analytical formulas for all required expressions in a form that does not assume any prior knowledge about the system at hand and providing an alternative algorithm for the time evolution of the non-Gaussian part of the many-body wave function. Our current efforts are directed at finding numerical validation that one can get below the mean-field energy and improve upon the ground state overlap using the VM and the here presented approach.

We have presented two novel algorithms for creating many-body operators from two-body interactions. While these algorithms improved upon longstanding construction schemes for important operations such as the multi-Toffoli gate, the recently developed fermionic-SWAP gate [70] should be taken into account in future studies, as it potentially could lower the overhead required for implementing the Jordan-Wigner transformation.

BIBLIOGRAPHY

- [1] S. Aaronson, *Quantum computing since Democritus* (Cambridge University Press, 2013).
- [2] T. Helgaker, P. Jorgensen, and J. Olsen, *Molecular electronic-structure theory* (John Wiley & Sons, 2014).
- [3] S. Bravyi, arXiv preprint quant-ph/0404180 (2004).
- [4] K. E. Cahill and R. J. Glauber, *Physical Review A* **59**, 1538 (1999).
- [5] P. Bader, S. Blanes, and F. Casas, *The Journal of chemical physics* **139**, 124117 (2013).
- [6] M. A. Nielsen and I. Chuang, “Quantum computation and quantum information,” (2002).
- [7] R. Kothari, *Efficient algorithms in quantum query complexity*, Ph.D. thesis (2014).
- [8] D. Tong, arXiv preprint arXiv:1606.06687 (2016).
- [9] J. K. Jain, *Composite fermions* (Cambridge University Press, 2007).
- [10] Y. Cao, J. Romero, and A. Aspuru-Guzik, *IBM Journal of Research and Development* **62**, 6 (2018).
- [11] A. Aspuru-Guzik, R. Lindh, and M. Reiher, *ACS central science* **4**, 144 (2018).
- [12] N. Wiebe, D. Braun, and S. Lloyd, *Physical review letters* **109**, 050505 (2012).
- [13] E. Farhi, J. Goldstone, and S. Gutmann, arXiv preprint arXiv:1411.4028 (2014).
- [14] S. Woerner and D. J. Egger, *npj Quantum Information* **5**, 1 (2019).
- [15] R. Orus, S. Mugel, and E. Lizaso, *Reviews in Physics* , 100028 (2019).
- [16] C. Gidney and M. Ekerå, arXiv preprint arXiv:1905.09749 (2019).
- [17] S. Rahimi-Keshari, T. C. Ralph, and C. M. Caves, *Physical Review X* **6**, 021039 (2016).
- [18] S. B. Bravyi and A. Y. Kitaev, *Annals of Physics* **298**, 210 (2002).
- [19] W. Kohn, *Reviews of Modern Physics* **71**, 1253 (1999).

- [20] M. Troyer and U.-J. Wiese, *Physical review letters* **94**, 170201 (2005).
- [21] U. Schollwöck, *Reviews of modern physics* **77**, 259 (2005).
- [22] G. H. Booth, A. J. Thom, and A. Alavi, *The Journal of chemical physics* **131**, 054106 (2009).
- [23] N. M. Tubman, J. Lee, T. Y. Takeshita, M. Head-Gordon, and K. B. Whaley, *The Journal of chemical physics* **145**, 044112 (2016).
- [24] V. Bach, E. H. Lieb, and J. P. Solovej, *Journal of statistical physics* **76**, 3 (1994).
- [25] E. Loh Jr, J. Gubernatis, R. Scalettar, S. White, D. Scalapino, and R. Sugar, *Physical Review B* **41**, 9301 (1990).
- [26] E. Bernstein and U. Vazirani, *SIAM Journal on computing* **26**, 1411 (1997).
- [27] L. K. Grover, in *Proceedings of the twenty-eighth annual ACM symposium on Theory of computing* (1996) pp. 212–219.
- [28] C. H. Bennett, E. Bernstein, G. Brassard, and U. Vazirani, *SIAM journal on Computing* **26**, 1510 (1997).
- [29] D. R. Simon, *SIAM journal on computing* **26**, 1474 (1997).
- [30] P. W. Shor, *SIAM review* **41**, 303 (1999).
- [31] F. Arute, K. Arya, R. Babbush, D. Bacon, J. C. Bardin, R. Barends, R. Biswas, S. Boixo, F. G. Brandao, D. A. Buell, *et al.*, *Nature* **574**, 505 (2019).
- [32] J. Preskill, arXiv preprint arXiv:1203.5813 (2012).
- [33] J. L. Park, *Foundations of Physics* **1**, 23 (1970).
- [34] A. Y. Kitaev, *Annals of Physics* **303**, 2 (2003).
- [35] A. G. Fowler, M. Mariantoni, J. M. Martinis, and A. N. Cleland, *Physical Review A* **86**, 032324 (2012).
- [36] M. Freedman, A. Kitaev, M. Larsen, and Z. Wang, *Bulletin of the American Mathematical Society* **40**, 31 (2003).
- [37] E. Farhi, J. Goldstone, S. Gutmann, and M. Sipser, arXiv preprint quant-ph/0001106 (2000).
- [38] D. Aharonov and A. Ta-Shma, in *Proceedings of the thirty-fifth annual ACM symposium on Theory of computing* (2003) pp. 20–29.
- [39] D. Aharonov, W. Van Dam, J. Kempe, Z. Landau, S. Lloyd, and O. Regev, *SIAM review* **50**, 755 (2008).
- [40] N. M. Tubman, C. Mejuto-Zaera, J. M. Epstein, D. Hait, D. S. Levine, W. Huggins, Z. Jiang, J. R. McClean, R. Babbush, M. Head-Gordon, *et al.*, arXiv preprint arXiv:1809.05523 (2018).

- [41] T. Shi, E. Demler, and J. I. Cirac, *Annals of Physics* **390**, 245 (2018).
- [42] J. C. Slater, *Physical Review* **34**, 1293 (1929).
- [43] E. Condon, *Physical Review* **36**, 1121 (1930).
- [44] A. Szabo and N. S. Ostlund, *Modern quantum chemistry: introduction to advanced electronic structure theory* (Courier Corporation, 2012).
- [45] M. Born and R. Oppenheimer, *Annalen der physik* **389**, 457 (1927).
- [46] B. Sutcliffe, in *Computational Techniques in Quantum Chemistry and Molecular Physics* (Springer, 1975) pp. 1–105.
- [47] I. Kassal, S. P. Jordan, P. J. Love, M. Mohseni, and A. Aspuru-Guzik, *Proceedings of the National Academy of Sciences* **105**, 18681 (2008).
- [48] I. D. Kivlichan, N. Wiebe, R. Babbush, and A. Aspuru-Guzik, *Journal of Physics A: Mathematical and Theoretical* **50**, 305301 (2017).
- [49] B. Toloui and P. J. Love, arXiv preprint arXiv:1312.2579 (2013).
- [50] R. Babbush, D. W. Berry, Y. R. Sanders, I. D. Kivlichan, A. Scherer, A. Y. Wei, P. J. Love, and A. Aspuru-Guzik, *Quantum Science and Technology* **3**, 015006 (2017).
- [51] E. Knill, arXiv preprint quant-ph/0108033 (2001).
- [52] B. M. Terhal and D. P. DiVincenzo, *Physical Review A* **65**, 032325 (2002).
- [53] E. Knill, R. Laflamme, and G. Milburn, arXiv preprint quant-ph/0006088 (2000).
- [54] M. C. Tichy, P. A. Bouvrie, and K. Mølmer, *Applied Physics B* **117**, 785 (2014).
- [55] A. Y. Kitaev, *Physics-Uspekhi* **44**, 131 (2001).
- [56] J. Valatin, *Il Nuovo Cimento (1955-1965)* **7**, 843 (1958).
- [57] N. Bogoljubov, V. V. Tolmachov, and D. Širkov, *Fortschritte der physik* **6**, 605 (1958).
- [58] J. Schwinger, *Physical Review* **93**, 615 (1954).
- [59] F. Berazin, *The method of second quantization*, Vol. 24 (Elsevier, 2012).
- [60] “List of quantum-mechanical systems with analytical solutions,” https://en.wikipedia.org/wiki/List_of_quantum-mechanical_systems_with_analytical_solutions.
- [61] G. C. Wick, *Physical Review* **96**, 1124 (1954).
- [62] T. Albash and D. A. Lidar, *Reviews of Modern Physics* **90**, 015002 (2018).
- [63] C. V. Kraus and J. I. Cirac, *New Journal of Physics* **12**, 113004 (2010).

- [64] S. McArdle, T. Jones, S. Endo, Y. Li, S. C. Benjamin, and X. Yuan, *npj Quantum Information* **5**, 1 (2019).
- [65] M. Motta, C. Sun, A. T. K. Tan, M. J. Rourke, E. Ye, A. J. Minnich, F. G. Brandao, and G. K. Chan, arXiv preprint arXiv:1901.07653 (2019).
- [66] Y. Manin, *Sovetskoye Radio, Moscow* **128** (1980).
- [67] R. P. Feynman, *International journal of theoretical physics* **21**, 467 (1982).
- [68] E. Knill, arXiv preprint quant-ph/9508006 (1995).
- [69] A. Aspuru-Guzik, A. D. Dutoi, P. J. Love, and M. Head-Gordon, *Science* **309**, 1704 (2005).
- [70] R. Babbush, N. Wiebe, J. McClean, J. McClain, H. Neven, and G. K.-L. Chan, *Physical Review X* **8**, 011044 (2018).
- [71] P. W. Shor, in *Proceedings 35th annual symposium on foundations of computer science* (Ieee, 1994) pp. 124–134.
- [72] S. Lloyd, M. Mohseni, and P. Rebentrost, arXiv preprint arXiv:1307.0411 (2013).
- [73] S. Aaronson, “What quantum computint isn’t,” https://www.youtube.com/watch?v=JvIbrDR1G_c (2015).
- [74] J. A. Pople, *Reviews of Modern Physics* **71**, 1267 (1999).
- [75] É. Borel, *J. Phys. Theor. Appl.* **3**, 189 (1913).
- [76] E. National Academies of Sciences, Medicine, *et al.*, *Quantum computing: progress and prospects* (National Academies Press, 2019).
- [77] E. Pednault, J. A. Gunnels, G. Nannicini, L. Horesh, and R. Wisnieff, arXiv preprint arXiv:1910.09534 (2019).
- [78] J. Preskill, *Quantum* **2**, 79 (2018).
- [79] A. Peruzzo, J. McClean, P. Shadbolt, M.-H. Yung, X.-Q. Zhou, P. J. Love, A. Aspuru-Guzik, and J. L. O’Brien, *Nature communications* **5**, 4213 (2014).
- [80] J. R. McClean, J. Romero, R. Babbush, and A. Aspuru-Guzik, *New Journal of Physics* **18**, 023023 (2016).
- [81] L. K. Grover and J. Radhakrishnan, in *Proceedings of the seventeenth annual ACM symposium on Parallelism in algorithms and architectures* (ACM, 2005) pp. 186–194.
- [82] H. K. Büning and T. Lettmann, *Propositional logic: deduction and algorithms*, Vol. 48 (Cambridge University Press, 1999).
- [83] W. Pauli, in *Wolfgang Pauli* (Springer, 1988) pp. 282–305.
- [84] V. Braginsky and F. Y. Khalili, *Quantum measurement* (Cambridge University Press, 1995).

- [85] “Scott aaronson phys771 lecture 14: Skepticism of quantum computing,” <https://www.scottaaronson.com/democritus/lec14.html>, accessed: 16.01.2020.
- [86] D. P. DiVincenzo, *Fortschritte der Physik: Progress of Physics* **48**, 771 (2000).
- [87] A. Y. Kitaev, *Uspekhi Matematicheskikh Nauk* **52**, 53 (1997).
- [88] J. I. Cirac and P. Zoller, *Physical review letters* **74**, 4091 (1995).
- [89] I. Bloch, J. Dalibard, and S. Nascimbene, *Nature Physics* **8**, 267 (2012).
- [90] A. Aspuru-Guzik and P. Walther, *Nature physics* **8**, 285 (2012).
- [91] D. Loss and D. P. DiVincenzo, *Physical Review A* **57**, 120 (1998).
- [92] J. Clarke and F. K. Wilhelm, *Nature* **453**, 1031 (2008).
- [93] I. Buluta and F. Nori, *Science* **326**, 108 (2009).
- [94] I. Arrazola, J. S. Pedernales, L. Lamata, and E. Solano, *Scientific reports* **6**, 30534 (2016).
- [95] M. Reiher, N. Wiebe, K. M. Svore, D. Wecker, and M. Troyer, *Proceedings of the National Academy of Sciences* **114**, 7555 (2017).
- [96] R. Babbush, D. W. Berry, J. R. McClean, and H. Neven, *npj Quantum Information* **5**, 1 (2019).
- [97] R. Cleve, A. Ekert, C. Macchiavello, and M. Mosca, *Proceedings of the Royal Society of London. Series A: Mathematical, Physical and Engineering Sciences* **454**, 339 (1998).
- [98] R. B. Griffiths and C.-S. Niu, *Physical Review Letters* **76**, 3228 (1996).
- [99] A. G. Fowler and S. J. Devitt, *arXiv preprint arXiv:1209.0510* (2012).
- [100] D. W. Berry, M. Kieferová, A. Scherer, Y. R. Sanders, G. H. Low, N. Wiebe, C. Gidney, and R. Babbush, *npj Quantum Information* **4**, 22 (2018).
- [101] D. W. Berry, A. M. Childs, R. Cleve, R. Kothari, and R. D. Somma, in *Forum of Mathematics, Sigma*, Vol. 5 (Cambridge University Press, 2017).
- [102] R. Babbush, C. Gidney, D. W. Berry, N. Wiebe, J. McClean, A. Paler, A. Fowler, and H. Neven, *Physical Review X* **8**, 041015 (2018).
- [103] A. M. Childs, *Communications in Mathematical Physics* **294**, 581 (2010).
- [104] D. W. Berry and A. M. Childs, *arXiv preprint arXiv:0910.4157* (2009).
- [105] S. Lloyd, *Science* , 1073 (1996).
- [106] J. E. Cohen, S. Friedland, T. Kato, and F. P. Kelly, *Linear Algebra and its Applications* **45**, 55 (1982).
- [107] G. Brassard and P. Hoyer, in *Proceedings of the Fifth Israeli Symposium on Theory of Computing and Systems* (IEEE, 1997) pp. 12–23.

- [108] G. H. Low and I. L. Chuang, Physical review letters **118**, 010501 (2017).
- [109] G. H. Low and I. L. Chuang, Quantum **3**, 163 (2019).
- [110] D. Poulin, A. Kitaev, D. S. Steiger, M. B. Hastings, and M. Troyer, Physical review letters **121**, 010501 (2018).
- [111] E. H. Hall *et al.*, American Journal of Mathematics **2**, 287 (1879).
- [112] P. Drude, Annalen der physik **306**, 566 (1900).
- [113] H. A. Lorentz, *Versuch einer Theorie der electrischen und optischen Erscheinungen in bewegten Körpern* (Cambridge University Press, 2013).
- [114] G. S. Ohm, *Die galvanische Kette, mathematisch bearbeitet* (TH Riemann, 1827).
- [115] K. v. Klitzing, G. Dorda, and M. Pepper, Physical Review Letters **45**, 494 (1980).
- [116] H. Stormer and D. Tsui, Science **220**, 1241 (1983).
- [117] D. C. Tsui, H. L. Stormer, and A. C. Gossard, Physical Review Letters **48**, 1559 (1982).
- [118] W. Pan, H. Stormer, D. Tsui, L. Pfeiffer, K. Baldwin, and K. West, Physical review letters **88**, 176802 (2002).
- [119] R. Willett, J. Eisenstein, H. Störmer, D. Tsui, A. Gossard, and J. English, Physical review letters **59**, 1776 (1987).
- [120] R. B. Laughlin, Physical Review Letters **50**, 1395 (1983).
- [121] G. Moore and N. Read, Nuclear Physics B **360**, 362 (1991).
- [122] E. Kapit and E. Mueller, Physical review letters **105**, 215303 (2010).
- [123] I. Glasser, J. I. Cirac, G. Sierra, and A. E. Nielsen, New Journal of Physics **17**, 082001 (2015).
- [124] M. Greiter, V. Schnells, and R. Thomale, Physical Review B **98**, 081113 (2018).
- [125] F. D. M. Haldane, Reviews of Modern Physics **89**, 040502 (2017).
- [126] L. Landau, Sov. Phys. JETP **8**, 70 (1959).
- [127] B. I. Halperin, Physical Review Letters **52**, 1583 (1984).
- [128] V. Goldman and B. Su, Science **267**, 1010 (1995).
- [129] N. Read and B. Chakraborty, Physical Review B **40**, 7133 (1989).
- [130] X.-G. Wen, Physical Review B **44**, 2664 (1991).
- [131] X. Chen, Z.-C. Gu, Z.-X. Liu, and X.-G. Wen, Physical Review B **87**, 155114 (2013).
- [132] P. W. Anderson, Science **177**, 393 (1972).

- [133] D. S. Abrams and S. Lloyd, *Physical Review Letters* **79**, 2586 (1997).
- [134] P. Benioff, *Journal of statistical physics* **22**, 563 (1980).
- [135] Y. Cao, J. Romero, J. P. Olson, M. Degroote, P. D. Johnson, M. Kieferová, I. D. Kivlichan, T. Menke, B. Peropadre, N. P. Sawaya, *et al.*, arXiv preprint arXiv:1812.09976 (2018).
- [136] S. Johri, D. S. Steiger, and M. Troyer, *Physical Review B* **96**, 195136 (2017).
- [137] A. M. Childs and N. Wiebe, arXiv preprint arXiv:1202.5822 (2012).
- [138] D. S. Abrams and S. Lloyd, *Physical Review Letters* **83**, 5162 (1999).
- [139] A. Y. Kitaev, arXiv:9511026 (1995).
- [140] J. Napp, R. L. La Placa, A. M. Dalzell, F. G. Brandao, and A. W. Harrow, arXiv preprint arXiv:2001.00021 (2019).
- [141] J. R. McClean, S. Boixo, V. N. Smelyanskiy, R. Babbush, and H. Neven, *Nature communications* **9**, 4812 (2018).
- [142] Y. Ge, J. Tura, and J. I. Cirac, *Journal of Mathematical Physics* **60**, 022202 (2019).
- [143] D. Berry, C. Gidney, M. Motta, J. McClean, and R. Babbush, arXiv:1902.02134 (2019).
- [144] D. W. Berry, A. M. Childs, R. Cleve, R. Kothari, and R. D. Somma, in *STOC '14 Proceedings of the 46th Annual ACM Symposium on Theory of Computing* (2014) pp. 283–292.
- [145] R. Wooten and J. Macek, arXiv preprint arXiv:1408.5379 (2014).
- [146] E. Tsiper, *Journal of Mathematical Physics* **43**, 1664 (2002).
- [147] E. De Micheli, arXiv preprint arXiv:1708.09715 (2017).
- [148] P. Padmanabham and H. Srivastava, *Applied Mathematics Letters* **13**, 65 (2000).
- [149] P. Jordan and E. P. Wigner, in *The Collected Works of Eugene Paul Wigner* (Springer, 1993) pp. 109–129.
- [150] S. Bravyi, J. M. Gambetta, A. Mezzacapo, and K. Temme, arXiv preprint arXiv:1701.08213 (2017).
- [151] F. Verstraete and J. I. Cirac, *Journal of Statistical Mechanics: Theory and Experiment* **2005**, P09012 (2005).
- [152] J. T. Seeley, M. J. Richard, and P. J. Love, *The Journal of chemical physics* **137**, 224109 (2012).
- [153] A. Tranter, S. Sofia, J. Seeley, M. Kaicher, J. McClean, R. Babbush, P. V. Coveney, F. Mintert, F. Wilhelm, and P. J. Love, *International Journal of Quantum Chemistry* **115**, 1431 (2015).

- [154] N. Moll, A. Fuhrer, P. Staar, and I. Tavernelli, *Journal of Physics A: Mathematical and Theoretical* **49**, 295301 (2016).
- [155] J. D. Whitfield, V. Havlíček, and M. Troyer, *Physical Review A* **94**, 030301 (2016).
- [156] A. Tranter, P. J. Love, F. Mintert, and P. V. Coveney, *Journal of chemical theory and computation* **14**, 5617 (2018).
- [157] M. Steudtner and S. Wehner, arXiv preprint arXiv:1712.07067 (2017).
- [158] D. Wecker, M. B. Hastings, and M. Troyer, *Physical Review A* **92**, 042303 (2015).
- [159] A. G. Fowler, M. Mariani, J. M. Martinis, and A. N. Cleland, *Physical Review A* **86**, 32324 (2012).
- [160] D. W. Berry, A. M. Childs, R. Cleve, R. Kothari, and R. D. Somma, *Physical Review Letters* **114**, 90502 (2015).
- [161] G. H. Low and I. L. Chuang, arXiv preprint arXiv:1610.06546 (2016).
- [162] G. H. Low and N. Wiebe, arXiv:1805.00675 (2018).
- [163] R. Babbush, D. W. Berry, I. D. Kivlichan, A. Y. Wei, P. J. Love, and A. Aspuru-Guzik, *New Journal of Physics* **18**, 33032 (2016).
- [164] R. Babbush, D. W. Berry, J. R. McClean, and H. Neven, arXiv preprint arXiv:1807.09802 (2018).
- [165] Y. Nam, Y. Su, and D. Maslov, arXiv preprint arXiv:1803.04933 (2018).
- [166] G. Ortiz, J. Gubernatis, E. Knill, and R. Laflamme, *Physical Review A* **64**, 022319 (2001).
- [167] D. Wecker, M. B. Hastings, N. Wiebe, B. K. Clark, C. Nayak, and M. Troyer, *Physical Review A* **92**, 062318 (2015).
- [168] Z. Jiang, K. J. Sung, K. Kechedzhi, V. N. Smelyanskiy, and S. Boixo, *Physical Review Applied* **9**, 044036 (2018).
- [169] I. D. Kivlichan, J. McClean, N. Wiebe, C. Gidney, A. Aspuru-Guzik, G. K.-L. Chan, and R. Babbush, *Physical review letters* **120**, 110501 (2018).
- [170] J. Eisert, V. Eisler, and Z. Zimborás, *Physical Review B* **97**, 165123 (2018).
- [171] L. Lehtovaara, J. Toivanen, and J. Eloranta, *Journal of Computational Physics* **221**, 148 (2007).
- [172] R. Somma, G. Ortiz, J. E. Gubernatis, E. Knill, and R. Laflamme, *Physical Review A* **65**, 042323 (2002).
- [173] J. B. Schriber and F. A. Evangelista, “Communication: An adaptive configuration interaction approach for strongly correlated electrons with tunable accuracy,” (2016).

- [174] N. M. Tubman, C. D. Freeman, D. S. Levine, D. Hait, M. Head-Gordon, and K. B. Whaley, arXiv preprint arXiv:1807.00821 (2018).
- [175] J. B. Schriber and F. A. Evangelista, *Journal of chemical theory and computation* **13**, 5354 (2017).
- [176] F. Motzoi, M. P. Kaicher, and F. K. Wilhelm, *Physical review letters* **119**, 160503 (2017).
- [177] M. Fremling, *Journal of Physics A: Mathematical and Theoretical* **46**, 275302 (2013).
- [178] S. Trugman and S. Kivelson, *Physical Review B* **31**, 5280 (1985).
- [179] C. H. Lee, Z. Papić, and R. Thomale, *Physical Review X* **5**, 041003 (2015).
- [180] I. Glasser, J. I. Cirac, G. Sierra, and A. E. Nielsen, *Physical Review B* **94**, 245104 (2016).
- [181] J. I. Latorre, V. Picó, and A. Riera, *Physical Review A* **81**, 060309 (2010).
- [182] A. Di Gioacchino, L. G. Molinari, V. Erba, and P. Rotondo, *Physical Review B* **95**, 245123 (2017).
- [183] S. Girvin and A. H. MacDonald, *Physical review letters* **58**, 1252 (1987).
- [184] R. Kamilla, J. K. Jain, and S. Girvin, *Physical Review B* **56**, 12411 (1997).
- [185] K. S. Kölbig and H. Scherb, *Journal of computational and applied mathematics* **71**, 357 (1996).
- [186] A. K. Shukla and I. A. Salehbbhai, *Communications of the Korean Mathematical Society* **27**, 721 (2012).
- [187] I. S. Gradshteyn and I. M. Ryzhik, *Table of integrals, series, and products* (Academic press, 2014).
- [188] A. Erdelyi, W. Magnus, F. Oberhettinger, and F. G. Tricomi, *Tables of Integral Transforms: Vol.: 2* (McGraw-Hill Book Company, Incorporated, 1954).
- [189] A. Niukkanen, *Journal of Physics A: Mathematical and General* **16**, 1813 (1983).
- [190] G. Lauricella, *Rendiconti del Circolo Matematico di Palermo (1884-1940)* **7**, 111 (1893).
- [191] L. Poh-aun, S.-h. Ong, and H. Srivastava, *International Journal of Computer Mathematics* **78**, 303 (2001).
- [192] A. Erdélyi, *Mathematische Zeitschrift* **40**, 693 (1936).
- [193] K. Mayr, *Mathematische Zeitschrift* **39**, 597 (1935).
- [194] H. Srivastava and H. Manocha, JOHN WILEY & SONS, INC., 605 THIRD AVE., NEW YORK, NY 10158, USA, 1984, 500 (1984).

- [195] B. Haible and T. Papanikolaou, in *International Algorithmic Number Theory Symposium* (Springer, 1998) pp. 338–350.
- [196] E. A. Karatsuba, *Problemy Peredachi Informatsii* **27**, 76 (1991).
- [197] D. Yoshioka, *The quantum Hall effect*, Vol. 133 (Springer Science & Business Media, 2013).
- [198] S. R. White, *Physical review letters* **69**, 2863 (1992).
- [199] G.-C. Wick, *Physical review* **80**, 268 (1950).
- [200] J. Haegeman, J. I. Cirac, T. J. Osborne, I. Pižorn, H. Verschelde, and F. Verstraete, *Physical review letters* **107**, 070601 (2011).
- [201] Y. Ashida, T. Shi, M. C. Bañuls, J. I. Cirac, and E. Demler, *Physical review letters* **121**, 026805 (2018).
- [202] Y. Ashida, T. Shi, M. C. Bañuls, J. I. Cirac, and E. Demler, *Physical Review B* **98**, 024103 (2018).
- [203] P. Sala, T. Shi, S. Kühn, M. C. Bañuls, E. Demler, and J. I. Cirac, *Physical Review D* **98**, 034505 (2018).
- [204] T. Shi, J. I. Cirac, and E. Demler, arXiv preprint arXiv:1904.00932 (2019).
- [205] Y. Ashida, T. Shi, R. Schmidt, H. Sadeghpour, J. I. Cirac, and E. Demler, arXiv preprint arXiv:1905.08523 (2019).
- [206] Y. Ashida, T. Shi, R. Schmidt, H. Sadeghpour, J. I. Cirac, and E. Demler, arXiv preprint arXiv:1905.09615 (2019).
- [207] T. Guaita, L. Hackl, T. Shi, C. Hubig, E. Demler, and J. I. Cirac, arXiv preprint arXiv:1907.04837 (2019).
- [208] T. Shi, J. Pan, and S. Yi, arXiv preprint arXiv:1909.02432 (2019).
- [209] R. Liu, Y.-R. Shi, and W. Zhang, arXiv preprint arXiv:2002.09158 (2020).
- [210] Y. Wang, L. Guo, S. Yi, and T. Shi, arXiv preprint arXiv:2002.11298 (2020).
- [211] Y. Wang, I. Esterlis, T. Shi, J. I. Cirac, and E. Demler, arXiv preprint arXiv:1910.01792 (2019).
- [212] E. H. Moore, *Bull. Am. Math. Soc.* **26**, 394 (1920).
- [213] R. Penrose, in *Mathematical proceedings of the Cambridge philosophical society*, Vol. 51 (Cambridge University Press, 1955) pp. 406–413.
- [214] F. Haake and H. Haken, *Quantum signatures of chaos*, Vol. 2 (Springer, 2010).
- [215] E. Artin, *Geometric algebra* (Courier Dover Publications, 2016).
- [216] W. Magnus, *Communications on pure and applied mathematics* **7**, 649 (1954).

- [217] A. McLachlan, *Molecular Physics* **8**, 39 (1964).
- [218] L. Hackl, T. Guaita, T. Shi, J. Haegeman, E. Demler, and I. Cirac, arXiv preprint arXiv:2004.01015 (2020).
- [219] R. Martinazzo and I. Burghardt, arXiv preprint arXiv:1907.00841 (2019).
- [220] H. Tsai, K. Chan, *et al.*, *Bernoulli* **9**, 895 (2003).
- [221] I. Najfeld and T. F. Havel, *Advances in applied mathematics* **16**, 321 (1995).
- [222] K. B. Petersen, M. S. Pedersen, *et al.*, *Technical University of Denmark* **7**, 510 (2008).
- [223] J. R. McClean, K. J. Sung, I. D. Kivlichan, Y. Cao, C. Dai, E. S. Fried, C. Gidney, B. Gimby, P. Gokhale, T. Häner, *et al.*, arXiv preprint arXiv:1710.07629 (2017).
- [224] T. Orlando, J. Mooij, L. Tian, C. H. Van Der Wal, L. Levitov, S. Lloyd, and J. Mazo, *Physical Review B* **60**, 15398 (1999).
- [225] N. Schuch and J. Siewert, *Physical Review A* **67**, 032301 (2003).
- [226] Y. Makhlin, G. Schön, and A. Shnirman, *nature* **398**, 305 (1999).
- [227] H. Abraham, I. Y. Akhalwaya, G. Aleksandrowicz, T. Alexander, E. Arbel, A. Asfaw, C. Azaustre, P. Barkoutsos, G. Barron, L. Bello, *et al.*, “Qiskit: an open-source framework for quantum computing,” (2019).
- [228] R. Wilcox, *Journal of Mathematical Physics* **8**, 962 (1967).
- [229] R. Cleve and J. Watrous, in *Proceedings 41st Annual Symposium on Foundations of Computer Science (IEEE, 2000)* pp. 526–536.
- [230] A. W. Harrow, A. Hassidim, and S. Lloyd, *Physical review letters* **103**, 150502 (2009).
- [231] R. Barends, L. Lamata, J. Kelly, L. García-Álvarez, A. Fowler, A. Megrant, E. Jeffrey, T. White, D. Sank, J. Mutus, *et al.*, *Nature communications* **6**, 7654 (2015).
- [232] J. D. Whitfield, J. Biamonte, and A. Aspuru-Guzik, *Molecular Physics* **109**, 735 (2011).
- [233] I. Kassal, J. D. Whitfield, A. Perdomo-Ortiz, M.-H. Yung, and A. Aspuru-Guzik, *Annual review of physical chemistry* **62**, 185 (2011).
- [234] D. Wecker, B. Bauer, B. K. Clark, M. B. Hastings, and M. Troyer, *Physical Review A* **90**, 022305 (2014).
- [235] R. Babbush, P. J. Love, and A. Aspuru-Guzik, *Scientific reports* **4**, 6603 (2014).
- [236] N. Wiebe, D. W. Berry, P. Høyer, and B. C. Sanders, *Journal of Physics A: Mathematical and Theoretical* **44**, 445308 (2011).
- [237] P.-L. Dallaire-Demers and F. K. Wilhelm, *Physical Review A* **93**, 032303 (2016).

- [238] R. Babbush, N. Wiebe, J. McClean, J. McClain, H. Neven, and G. K. Chan, arXiv preprint arXiv:1706.00023 (2017).
- [239] K. Mølmer, L. Isenhower, and M. Saffman, *Journal of Physics B: Atomic, Molecular and Optical Physics* **44**, 184016 (2011).
- [240] L. Page, S. Brin, R. Motwani, and T. Winograd, *World Wide Web Internet And Web Information Systems* **54**, 1 (1999).
- [241] D. Crow, R. Joynt, and M. Saffman, *Physical review letters* **117**, 130503 (2016).
- [242] D. W. Berry, *Journal of Physics A: Mathematical and Theoretical* **47**, 105301 (2014).
- [243] A. Sørensen and K. Mølmer, *Physical review letters* **82**, 1971 (1999).
- [244] R. Blatt, *Nature* **404**, 231 (2000).
- [245] T. Monz, P. Schindler, J. T. Barreiro, M. Chwalla, D. Nigg, W. A. Coish, M. Harlander, W. Hänsel, M. Hennrich, and R. Blatt, *Physical Review Letters* **106**, 130506 (2011).
- [246] C. Moore and M. Nilsson, *SIAM Journal on Computing* **31**, 799 (2001).
- [247] S. Fenner, F. Green, S. Homer, and Y. Zhang, in *International Symposium on Fundamentals of Computation Theory* (Springer, 2005) pp. 44–55.
- [248] Y. Takahashi and N. Kunihiro, *Quantum Information & Computation* **8**, 636 (2008).
- [249] P. Gossett, arXiv preprint quant-ph/9808061 (1998).
- [250] T. G. Draper, S. A. Kutin, E. M. Rains, and K. M. Svore, arXiv preprint quant-ph/0406142 (2004).
- [251] R. Van Meter and K. M. Itoh, *Physical Review A* **71**, 052320 (2005).
- [252] D. Maslov and G. W. Dueck, *Electronics Letters* **39**, 1790 (2003).
- [253] A. Barenco, C. H. Bennett, R. Cleve, D. P. DiVincenzo, N. Margolus, P. Shor, T. Sleator, J. A. Smolin, and H. Weinfurter, *Physical review A* **52**, 3457 (1995).
- [254] D. Maslov, *Physical Review A* **93**, 022311 (2016).
- [255] R. J. Bartlett and M. Musiał, *Reviews of Modern Physics* **79**, 291 (2007).
- [256] A. G. Taube and R. J. Bartlett, *International journal of quantum chemistry* **106**, 3393 (2006).
- [257] J. Romero, R. Babbush, J. R. McClean, C. Hempel, P. J. Love, and A. Aspuru-Guzik, *Quantum Science and Technology* **4**, 014008 (2018).
- [258] M. R. Hoffmann and J. Simons, *The Journal of chemical physics* **88**, 993 (1988).
- [259] D. Bera, S. Fenner, F. Green, and S. Homer, *Quantum Information & Computation* **10**, 16 (2010).

- [260] P. Gokhale, J. M. Baker, C. Duckering, N. C. Brown, K. R. Brown, and F. T. Chong, arXiv preprint arXiv:1905.10481 (2019).
- [261] J. Casanova, A. Mezzacapo, L. Lamata, and E. Solano, Physical review letters **108**, 190502 (2012).
- [262] A. Sørensen and K. Mølmer, Physical Review A **62**, 022311 (2000).
- [263] M. H. Goerz, F. Motzoi, K. B. Whaley, and C. P. Koch, npj Quantum Information **3**, 37 (2017).
- [264] J. Siewert, R. Fazio, G. M. Palma, and E. Sciacca, Journal of Low Temperature Physics **118**, 795 (2000).
- [265] A. Imamog, D. D. Awschalom, G. Burkard, D. P. DiVincenzo, D. Loss, M. Sherwin, A. Small, *et al.*, Physical review letters **83**, 4204 (1999).
- [266] D. Mozyrsky, V. Privman, and M. L. Glasser, Physical review letters **86**, 5112 (2001).
- [267] D. Maslov, G. W. Dueck, and D. M. Miller, ACM Transactions on Design Automation of Electronic Systems (TODAES) **12**, 42 (2007).
- [268] G. K. Brennen, D. Song, and C. J. Williams, Physical Review A **67**, 050302 (2003).
- [269] R. v. Meter and M. Oskin, ACM Journal on Emerging Technologies in Computing Systems (JETC) **2**, 31 (2006).
- [270] H. J. Kimble, Nature **453**, 1023 (2008).
- [271] J. Hall, M. Novotny, T. Neuhaus, and K. Michielsen, Physics procedia **68**, 56 (2015).
- [272] R. Van Meter, K. Nemoto, W. Munro, and K. M. Itoh, in *33rd International Symposium on Computer Architecture (ISCA'06)* (IEEE, 2006) pp. 354–365.
- [273] S. Beigi, I. Chuang, M. Grassl, P. Shor, and B. Zeng, Journal of Mathematical Physics **52**, 022201 (2011).
- [274] D. D. Thaker, T. S. Metodi, A. W. Cross, I. L. Chuang, and F. T. Chong, ACM SIGARCH Computer Architecture News **34**, 378 (2006).
- [275] B.-S. Choi and R. Van Meter, ACM Journal on Emerging Technologies in Computing Systems (JETC) **7**, 11 (2011).
- [276] B.-S. Choi and R. Van Meter, ACM Journal on Emerging Technologies in Computing Systems (JETC) **8**, 24 (2012).

DANKSAGUNG

Zunächst möchte ich meinem Betreuer Frank Wilhelm-Mauch danken, auf dessen Unterstützung ich seit meinem zweiten Studienjahr immer zählen konnte. Er hat mir nicht nur ermöglicht meine wissenschaftlichen Schwerpunkte selbst zu bestimmen, sondern hat auch sicher gestellt, dass ich mit den richtigen Personen auf den entsprechenden Gebieten in Kontakt komme und hat immer ein offenes Ohr für meine Probleme trotz voll gepacktem Terminkalender gefunden.

Desweiteren bin ich all jenen zu Dank verpflichtet, die mit mir in den letzten Jahren zusammengearbeitet haben, vor allem Felix Motzoi, Simon Jäger, Marius Schöndorf, Pierre-Luc Dallaire-Demers, Ryan Babbush, und Andrew Tranter.

Insbesondere bin ich Giovanna Morigi und Jürgen Eschner dankbar, mir über all die Jahre immer verständnisvoll hilfreiche Anregungen und konstruktive Kritik bei jeglichen (auch den vielen dummen) Fragen entgegengebracht zu haben und mir ihre Türen immer offen standen.

Ich möchte Frank dafür danken, es mir ermöglicht zu haben Peter Love und Alán Aspuru-Guzik kennen zu lernen, dessen Gruppe ich für drei Monate an der Harvard University besuchen durfte. Dort lernte ich auch dessen Doktoranden Ryan Babbush und Jarrod McClean kennen, die mir bis heute bei jeglichen Fragen stets hilfreich zur Seite stehen. Ich danke auch dem gesamten Team von ZAPATA und Daniela Pfannkuche für ihre Gastfreundschaft.

Ich bin Andrea Dumont und Monika Francois sehr zu Dank verpflichtet, da ohne ihren Überblick, ihre Geduld und Organisation unsere Gruppe nicht funktionieren könnte.

Ich möchte mich bei allen Mitgliedern der AG Wilhelm und AG Morigi bedanken, insbesondere bei Federico Roy, Rafi Schmit, Simon Jäger, David Headley und Marius Schöndorf, die dabei geholfen haben Teile der Dissertation Korrektur zu lesen. Besonders hervorheben möchte ich Rebecca Kraus, Shraddha Sharma, Ralf Betzholz, Steffi Stalter, Luke Govia, Kevin Pack, Lukas Theis, Tobias Chasseur, Nahuel Freitas, Luigi Gianelli, Daniel Egger, Peter Groszkowski, Stefan Schütz, José Brito und allen Teilnehmern des Ctrl-Q Seminars.

Ein besonderer Dank geht an dieser Stelle an meinen ehemaligen Physiklehrer Andreas Hoffmann und Prof. Christoph Becher, ohne deren frühe Unterstützung ich mich wohl nicht für ein Studium der Physik an der Universität des Saarlandes entschieden hätte.

Ich danke der gesamten Basketball-Abteilung des TUS Herrensöhr, bei denen ich seit über zwanzig Jahren spielen durfte und die für mich Teil meiner Familie geworden sind. Insbesondere danke ich meinen beiden Freunden Jan Brinkmann und Philipp Leupoldt, die mir von Kindheitstagen an immer zur Seite gestanden haben.

Ich danke Marius Schöndorf und Simon Jäger, meinen treuen Begleitern durch die Vorlesungssäle und Kneipen Saarbrückens. Nun ist es mir von besondere Freude Simons Rolle in meinem Werdegang zu erwähnen, dem Capi zu meinem Samra. Er hat in den fast zehn Jahren jede meiner Fragen und Problemen mit einer bedingungslosen Selbstverständlichkeit und Geduld beantwortet und mich dadurch zu einem hoffentlich halbwegs passablen Wissenschaftler gemacht. Dafür und für seine tiefe Freundschaft in all den bisher sowohl besten als auch schwierigsten Jahren meines Lebens werde ich immer dankbar sein.

An dieser Stelle möchte ich meiner Familie von Herzen danken. Zuallererst gilt der

Dank meiner Freundin Hannah Ennassih, deren Unterstützung und Liebe mir immer wieder geholfen haben den Mut nicht zu verlieren, wenn es mal nicht geklappt hat. Tief verbunden und dankbar bin ich meinen Geschwistern Franzi und Christian, meinen Omas Gertrud und Else, meinen Tanten und Onkeln, Petra und Martin Kaicher, Angelika und Theo Schilling und meinem Schwager Jonas sowie meinem (noch) kleinen Neffen Anton. Meinen Eltern Elisabeth und Helmuth habe ich alles zu verdanken. Diese Arbeit ist meiner Familie und dem Andenken meines Vaters gewidmet.