



Quantum Computation for the Understanding of Mass

Simulating Quantum Field Theories

Pedro Rivero

Illinois Institute of Technology
Argonne National Laboratory

priveroramirez@hawk.iit.edu

Introduction

Introduction

Background in
Quantum
Computation

Quantum logic
Quantum information
Quantum computing
models and algorithms
Applications

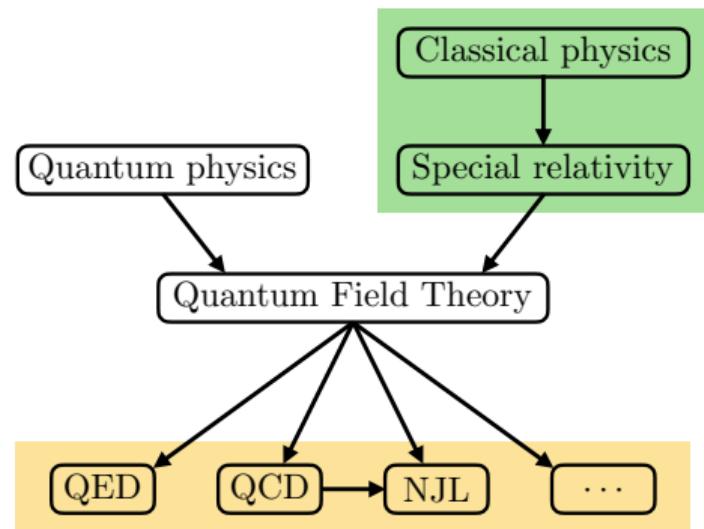
Solving the NJL
model in 1 + 1
dimensions

Analytical solution
Lattice formulation
Fermion-qubit mapping
Space parametrization
and state preparation
Ground state energy

Conclusions and
prospective work

Nature is described through the mathematical framework provided by **Quantum Field Theory**.

- The different implementations of Quantum Field Theory are referred to as quantum field theories themselves.
- Quantum Chromodynamics (QCD) is the theory of the strong nuclear force.
- QCD holds many mysteries (e.g. **mass generation** phenomena).
- QCD is currently studied using brute-force numerics on the world's largest supercomputers.
- Many aspects of quantum field theories cannot be studied using classical computers.



Objectives

Introduction

Background in Quantum Computation

Quantum logic
Quantum information
Quantum computing models and algorithms
Applications

Solving the NJL model in $1 + 1$ dimensions

Analytical solution
Lattice formulation
Fermion-qubit mapping
Space parametrization and state preparation
Ground state energy

Conclusions and prospective work

An important path forward is to develop methods to simulate these models on quantum computers. This effort is only just beginning, however, performing calculations on a quantum computer is far from being a straightforward task; and it has only been achieved for relatively simple problems. My goal will be to develop some of the techniques necessary to use quantum computers for this endeavor. We will do so by analyzing the following:

- **Nambu–Jona-Lasino model** (NJL) in $1 + 1$ dimensions: an effective field theory, regarded as a low-energy approximation to QCD.
- It retains certain key features of QCD, such as the so called Goldstone modes and **dynamical chiral symmetry breaking**; which in turn is responsible for the creation of dressed mass.
- This model can be solved nonperturbatively through the standard leading order truncation; an important characteristic since verifying the solutions returned by any quantum computation is currently a major challenge.



Contents

Introduction

Background in

Quantum

Computation

Quantum logic

Quantum information

Quantum computing

models and algorithms

Applications

Solving the NJL

model in $1 + 1$

dimensions

Analytical solution

Lattice formulation

Fermion-qubit mapping

Space parametrization

and state preparation

Ground state energy

Conclusions and

prospective work



1 Introduction

2 Background in Quantum Computation

- Quantum logic
- Quantum information
- Quantum computing models and algorithms
- Applications

3 Solving the NJL model in $1 + 1$ dimensions

- Analytical solution
- Lattice formulation
- Fermion-qubit mapping
- Space parametrization and state preparation
- Ground state energy

4 Conclusions and prospective work

Background in Quantum Computation

Introduction

Background in
Quantum
Computation

Quantum logic
Quantum information
Quantum computing
models and algorithms
Applications

Solving the NJL
model in 1 + 1
dimensions

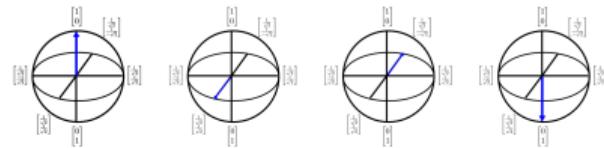
Analytical solution
Lattice formulation
Fermion-qubit mapping
Space parametrization
and state preparation
Ground state energy

Conclusions and
prospective work

Before the abacus was invented, the only way of counting was through a "thermometer-like" scale. This device introduced the concept of **digits**, which was later adopted into our own language in the form of **numerals**. This meant that, instead of counting up to N using N elements, we could count using only $\log_b(N)$ abacus elements —where b is the base of our number system (i.e. the amount of beads per row in the abacus).

Therefore, for $b \geq 2$, the abacus introduced an **exponential decrease in resource requirements**.

Quantum computers do something analogous. To represent the quantum state of a system made out of N subsystems —with b degrees of freedom each— we would usually need b^N classical elements (i.e. numbers). By using quantum systems for this representation, we would only need N quantum elements (i.e. each subsystem).



Quantum logic

Introduction

Background in
Quantum
Computation

Quantum logic

Quantum information
Quantum computing
models and algorithms
Applications

Solving the NJL
model in 1 + 1
dimensions

Analytical solution
Lattice formulation
Fermion-qubit mapping
Space parametrization
and state preparation
Ground state energy

Conclusions and
prospective work

Formally speaking, the states of a quantum system will be represented as a vector in a Hilbert space; whereas the state of a classical system is simply an element of a set. This distinction, along with the probabilistic nature of quantum mechanics, makes the entire logical system describing quantum computers fundamentally different than **classical propositional logic** (i.e. Boolean algebra).

CLASSICAL LOGIC

Set Theory (Boolean algebra)

- **AND** $\Rightarrow A \cup B$
- **OR** $\Rightarrow A \cap B$
- **NOT** $\Rightarrow \bar{A}$
- **XOR** $\Rightarrow A \cap B - A \cup B$

QUANTUM LOGIC

Quantum Theory (Non-abelian)

- **Probabilistic measurement**
- **Measurement causes disturbance**
- **Superposition**
- **Entanglement**
- **Uncertainty principle**



Qubits and the Bloch sphere

Introduction

Background in
Quantum
Computation

Quantum logic

Quantum information

Quantum computing
models and algorithms

Applications

Solving the NJL
model in 1 + 1
dimensions

Analytical solution

Lattice formulation

Fermion-qubit mapping

Space parametrization
and state preparation

Ground state energy

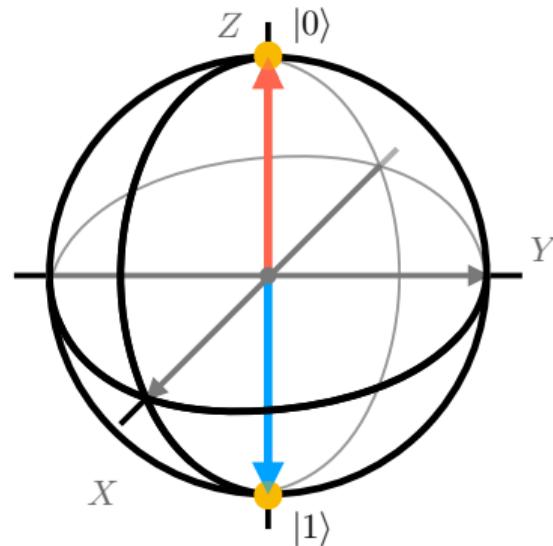
Conclusions and
prospective work



The basic unit of information in quantum mechanics is the **qubit**; a quantum system with only two distinguishable states: $|0\rangle$ and $|1\rangle$.

The main difference between bit and qubit lies in the fact that, when they are not being measured, quantum systems do not need to be in one of their so called, mutually incompatible, basis states; instead, they can be in any (normalized) **linear superposition** of them.

A simple example of a qubit is a spin- $\frac{1}{2}$ particle; whose space of states is spanned by the spin up and spin down quantum basis. Inspired by this particular system, the qubit can be pictorially represented through the **Bloch sphere**.



Quantum information

Introduction

Background in
Quantum
Computation

Quantum logic

Quantum information

Quantum computing
models and algorithms

Applications

Solving the NJL
model in 1 + 1
dimensions

Analytical solution

Lattice formulation

Fermion-qubit mapping

Space parametrization
and state preparation

Ground state energy

Conclusions and
prospective work



In the same way that the device known as an abacus had its theoretical counterpart in positional number systems, quantum computers have Quantum Information Science (QIS); which help us understand the full capabilities of these new machines. Two very important insights extracted from QIS, and with direct applicability to quantum apparatuses are:

■ Unitary quantum state transformations (i.e. reversible):

$$\langle \psi_0 | \phi_0 \rangle \equiv \langle \psi(t) | \phi(t) \rangle = \langle \psi_0 | U^\dagger U | \phi_0 \rangle \Rightarrow U^\dagger U \equiv \mathbb{1}$$

■ No-cloning theorem:

$$U_{\text{cloning}}(|\psi\rangle_A \otimes |e\rangle_B) = |\psi\rangle_A \otimes |\psi\rangle_B, \quad U_{\text{cloning}}(|\phi\rangle_A \otimes |e\rangle_B) = |\phi\rangle_A \otimes |\phi\rangle_B$$

$$\begin{aligned} \langle \psi | \phi \rangle &= (\langle \psi | \phi \rangle)_A (\langle e | e \rangle)_B = (\langle \psi |_A \otimes \langle e |_B) (|\phi\rangle_A \otimes |e\rangle_B) \\ &\equiv (\langle \psi |_A \otimes \langle e |_B) U_{\text{cloning}}^\dagger U_{\text{cloning}} (|\phi\rangle_A \otimes |e\rangle_B) = (\langle \psi |_A \otimes \langle \psi |_B) (|\phi\rangle_A \otimes |\phi\rangle_B) \\ &= (\langle \psi | \phi \rangle)_A (\langle \psi | \phi \rangle)_B = (\langle \psi | \phi \rangle)^2 \end{aligned}$$

Implications for quantum information technologies

Introduction

Background in
Quantum
Computation

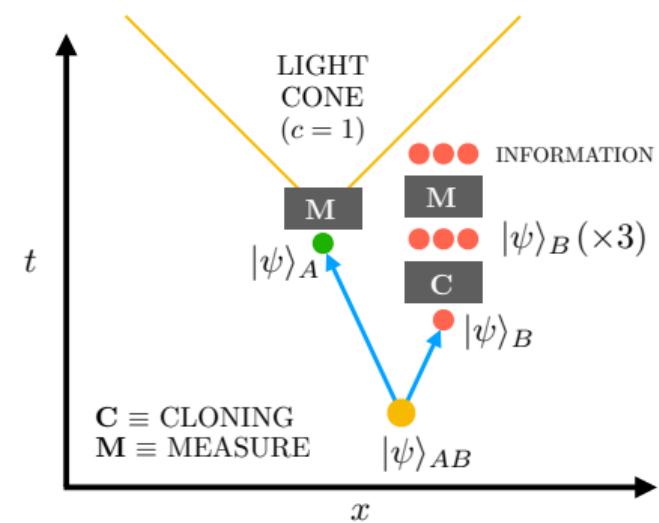
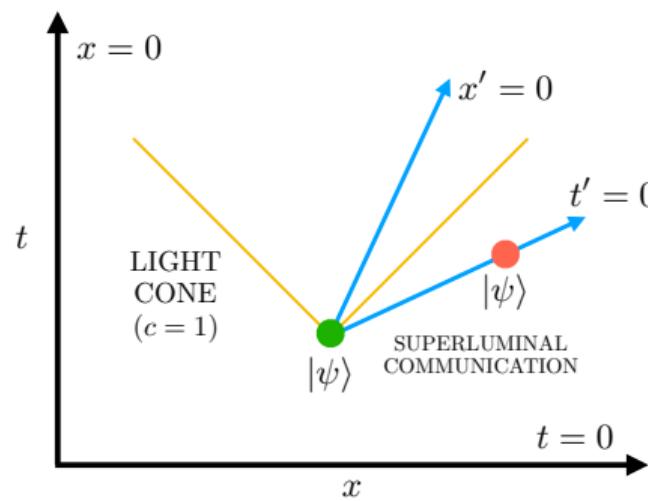
Quantum logic
Quantum information
Quantum computing
models and algorithms
Applications

Solving the NJL
model in 1 + 1
dimensions

Analytical solution
Lattice formulation
Fermion-qubit mapping
Space parametrization
and state preparation
Ground state energy

Conclusions and
prospective work

The fact that all transformations have to preserve information (i.e. entropy), imply that they are unitary. This is the only kind of transformations that a quantum computer will be able to reproduce ideally. On the other hand, the no-cloning theorem can be used to construct inherently secure communication and data storage systems. Furthermore, it means that **superluminal communication** is impossible:



Quantum computing models and algorithms

Introduction

Background in
Quantum
Computation

Quantum logic

Quantum information

Quantum computing
models and algorithms

Applications

Solving the NJL
model in 1 + 1
dimensions

Analytical solution

Lattice formulation

Fermion-qubit mapping

Space parametrization
and state preparation

Ground state energy

Conclusions and
prospective work



In order to work with quantum computers we need to be able to formulate how they operate in an abstract, mathematical way. There are a number of formulations that go by the name of **quantum computing models**; the main four of these being:

- Topological
- Adiabatic
- One-way
- Gate arrar or circuit based

They all differ in the basic elements in which any computation is divided, and therefore have a decisive role in the **hardware implementation** of these machines. Nonetheless, all of them have been proven theoretically equivalent to each other as well as to an universal quantum Turing machine.

- Josephson junctions
- Ion traps
- Nuclear magnetic resonators
- Optical cavities

Quantum circuit formalism

Introduction

Background in
Quantum
Computation

Quantum logic

Quantum information

Quantum computing
models and algorithms

Applications

Solving the NJL
model in 1 + 1
dimensions

Analytical solution

Lattice formulation

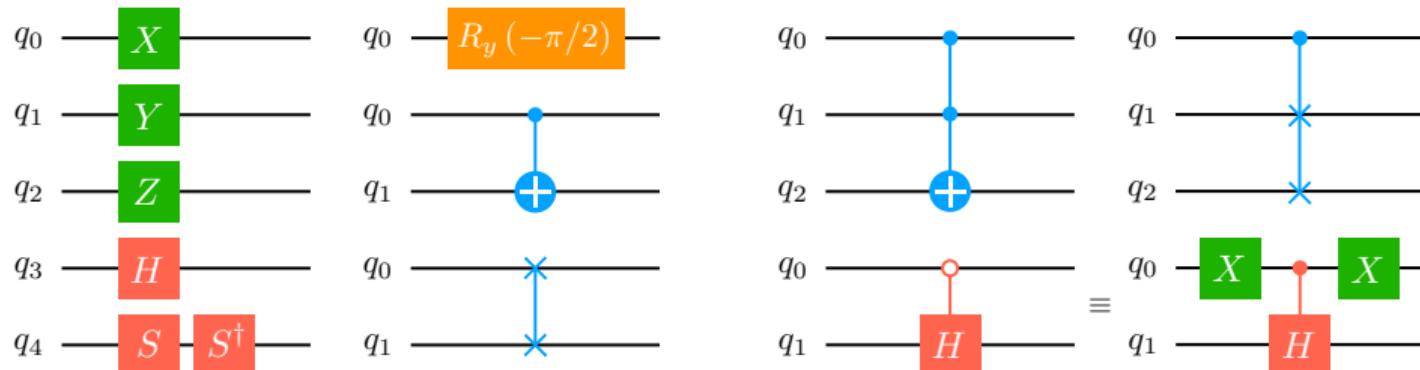
Fermion-qubit mapping

Space parametrization
and state preparation

Ground state energy

Conclusions and
prospective work

Due to its simplicity and similarity with classical models of computation, the most widespread of all is, by far, the **quantum circuit** or quantum gate array model. This model is made up of two kind of components: *registers* and *gates*.



Pauli measurements and the singlet state

Introduction

Background in
Quantum
Computation

Quantum logic

Quantum information

Quantum computing
models and algorithms

Applications

Solving the NJL
model in 1 + 1
dimensions

Analytical solution

Lattice formulation

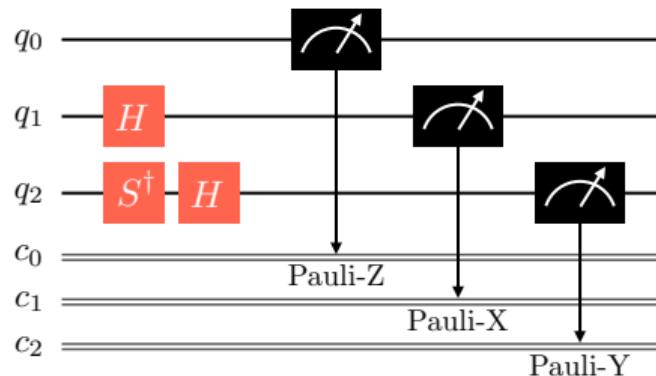
Fermion-qubit mapping

Space parametrization
and state preparation

Ground state energy

Conclusions and
prospective work

The final thing that we need to know about are measurements. **Pauli measurements** are made along the three primary axes of the Bloch sphere, allowing us to calculate the expectation value of *Pauli observables* (i.e. tensor product combinations of the Pauli hermitian operators):



$$|\text{singlet}\rangle \triangleq \frac{|01\rangle - |10\rangle}{\sqrt{2}}$$

Deutsch-Jozsa algorithm I

Introduction

Background in
Quantum
Computation

Quantum logic

Quantum information

Quantum computing
models and algorithms

Applications

Solving the NJL
model in 1 + 1
dimensions

Analytical solution

Lattice formulation

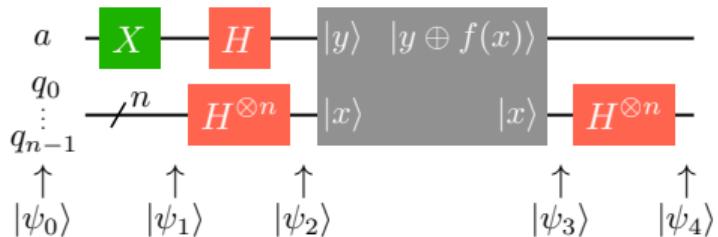
Fermion-qubit mapping

Space parametrization
and state preparation

Ground state energy

Conclusions and
prospective work

Deutsch's problem says that we are given a function $f : \{0, 1\}^n \rightarrow \{0, 1\}$, and are told that the function is either constant or balanced (i.e. half of the inputs return 0 and the other half 1). The goal is to determine, with the smallest number of evaluations possible, whether the given function is one kind or the other.



Using classical resources the only way to solve this problem is to repeatedly evaluate the function for different inputs. In the worst case scenario, we will need to perform $2^n/2 + 1$ evaluations, since it is always possible to obtain $2^n/2$ times the same number even if the function is balanced. However, making use of quantum superposition, we can find a way to solve this problem with just one evaluation. The key as to why this is possible is quantum superposition. By evaluating the target function for a superposition state, we are evaluating it simultaneously for all possible inputs. This is sometimes referred to as **quantum parallelism**.



Deutsch-Jozsa algorithm II

Introduction

Background in
Quantum
Computation

Quantum logic
Quantum information
Quantum computing
models and algorithms
Applications

Solving the NJL
model in 1 + 1
dimensions

Analytical solution
Lattice formulation
Fermion-qubit mapping
Space parametrization
and state preparation
Ground state energy

Conclusions and
prospective work



The solution to this problem as expressed by the quantum circuit in the previous slide is known as the **Deutsch-Jozsa algorithm**:

$$|\psi_0\rangle = |0\rangle^{\otimes(n+1)}$$

$$|\psi_1\rangle = |0\rangle^{\otimes n} |1\rangle$$

$$H^{\otimes n} |x\rangle = \sum_{z \in \{0,1\}^n} (-1)^{x \cdot z} \frac{|z\rangle}{\sqrt{2^n}}$$

$$|\psi_2\rangle = \sum_{x \in \{0,1\}^n} \frac{|x\rangle}{\sqrt{2^n}} \left[\frac{|0\rangle - |1\rangle}{\sqrt{2}} \right]$$

$$\frac{|0 \oplus f(x)\rangle - |1 \oplus f(x)\rangle}{\sqrt{2}} = (-1)^{f(x)} \frac{|0\rangle - |1\rangle}{\sqrt{2}}$$

$$|\psi_3\rangle = \sum_{x \in \{0,1\}^n} \frac{(-1)^{f(x)}}{\sqrt{2^n}} \left[\frac{|0\rangle - |1\rangle}{\sqrt{2}} \right]$$

$$|\psi_4\rangle = \sum_{\substack{x \in \{0,1\}^n \\ z \in \{0,1\}^n}} \frac{(-1)^{x \cdot z + f(x)}}{2^n} \left[\frac{|0\rangle - |1\rangle}{\sqrt{2}} \right]$$

Quantum Fourier transform I

Introduction

Background in
Quantum
Computation

Quantum logic
Quantum information
Quantum computing
models and algorithms
Applications

Solving the NJL
model in 1 + 1
dimensions

Analytical solution
Lattice formulation
Fermion-qubit mapping
Space parametrization
and state preparation
Ground state energy

Conclusions and
prospective work



The usual **Discrete Fourier Transform** (DFT) can be expressed as:

$$\begin{aligned} |j\rangle &\rightarrow \frac{1}{\sqrt{N}} \sum_{k=0}^{N-1} \exp\left[2\pi i \frac{jk}{N}\right] |k\rangle \\ &= \frac{1}{2^{n/2}} \sum_{k_{n-1}=0}^1 \cdots \sum_{k_0=0}^1 \exp\left[2\pi ij \left(\sum_{l=0}^{n-1} k_l 2^{l-n}\right)\right] |0b k_{n-1} \dots k_0\rangle \\ &= \frac{1}{2^{n/2}} \sum_{k_{n-1}=0}^1 \cdots \sum_{k_0=0}^1 \left\{ \bigotimes_{l=1}^n \exp\left[2\pi i j k_{n-l} 2^{-l}\right] |k_{n-l}\rangle \right\} \\ &= \frac{1}{2^{n/2}} \bigotimes_{l=1}^n \left\{ \sum_{k_{n-l}=0}^1 \exp\left[2\pi i j k_{n-l} 2^{-l}\right] |k_{n-l}\rangle \right\} = \frac{1}{2^{n/2}} \bigotimes_{l=1}^n \left\{ |0\rangle + \exp\left[2\pi i j 2^{-l}\right] |1\rangle \right\} \end{aligned}$$

Quantum Fourier transform II

Introduction

Background in
Quantum
Computation

Quantum logic

Quantum information

Quantum computing
models and algorithms

Applications

Solving the NJL
model in 1 + 1
dimensions

Analytical solution

Lattice formulation

Fermion-qubit mapping

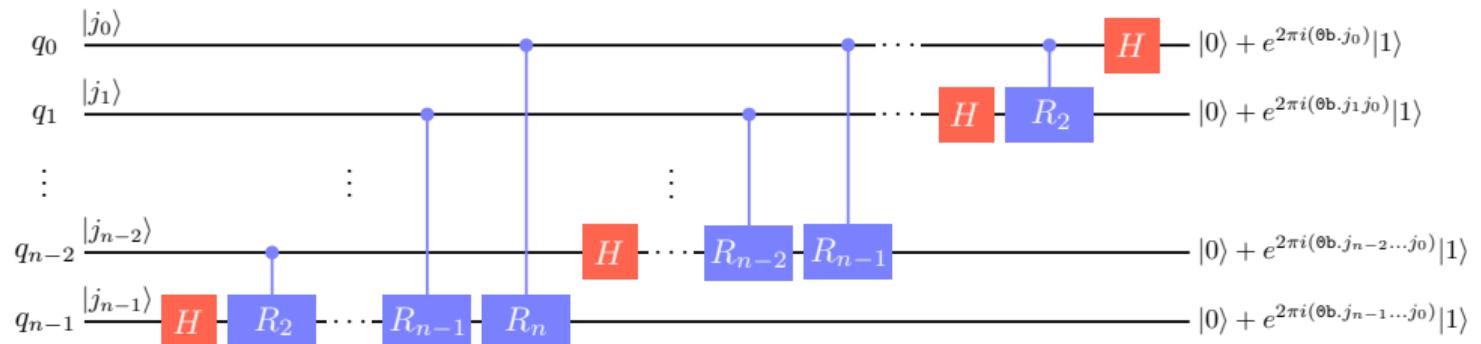
Space parametrization
and state preparation

Ground state energy

Conclusions and
prospective work

$$|j\rangle \rightarrow \frac{\{ |0\rangle + e^{2\pi i(\theta b.j_0)} |1\rangle \} \otimes \{ |0\rangle + e^{2\pi i(\theta b.j_1 j_0)} |1\rangle \} \otimes \cdots \otimes \{ |0\rangle + e^{2\pi i(\theta b.j_{n-1} j_{n-2} \dots j_0)} |1\rangle \}}{2^{n/2}}$$

$$R_k \triangleq \begin{bmatrix} 1 & 0 \\ 0 & e^{2\pi i/2^k} \end{bmatrix}$$



Quantum Fourier transform III

Introduction

Background in
Quantum
Computation

Quantum logic

Quantum information

Quantum computing
models and algorithms

Applications

Solving the NJL
model in 1 + 1
dimensions

Analytical solution

Lattice formulation

Fermion-qubit mapping

Space parametrization
and state preparation

Ground state energy

Conclusions and
prospective work

We can now compare the complexity of this algorithm with its best classical counterparts. This is done by counting the number of gates in the circuit, where we have to remember the final SWAP gates omitted in the representation.

FAST FOURIER TRANSFORM

$$N \log_2 N \equiv \Theta(n2^n)$$

QUANTUM FOURIER TRANSFORM

$$\sum_{k=1}^n k + \frac{n}{2} = \frac{n(n+1)}{2} + \frac{n}{2} \equiv \Theta(n^2)$$

Nonetheless, this technique cannot be used directly for computing the target transform; since we do not know how to recover the individual amplitudes from the quantum states. On top of that, there is no efficient general method for preparing the states to be transformed. This is a great example of an algorithm that presents huge savings compared to its classical analogs, but which cannot be generally used as much as we would like to due to our inability to extract the desired information. In some instances however, we can profit from this method to great deeds —mainly as part of bigger algorithms. An important application is **Shor's factoring algorithm**, which can be used for efficiently finding the prime decomposition of any given number.



Applications

Introduction

Background in
Quantum
Computation

Quantum logic
Quantum information
Quantum computing
models and algorithms
Applications

Solving the NJL
model in 1 + 1
dimensions

Analytical solution
Lattice formulation
Fermion-qubit mapping
Space parametrization
and state preparation
Ground state energy

Conclusions and
prospective work

Generally speaking, quantum computing is thought to be able to outperform classical processors, but not for all tasks. This raises the important question of determining which problems are easy to solve using quantum logic while remaining hard for classical approaches. As a matter of fact, we do not fully understand the extent to which there are problems of this kind; and many of the ones we suspect to be good candidates have no formal proof to back them up. So far, some of the applications that have been found are:

- Cryptography
- Secure communications
- Quantum randomness
- Quantum sensing
- Optimization
- Algebra
- Machine learning
- Quantum games
- Quantum simulation

Quantum physics —especially Quantum Field Theory— is the ideal place to look for (relevant) problems which can be solved on a quantum computer while remaining intractable for classical machines; that also makes it the best choice for analyzing the computational power behind this new paradigm.



Solving the NJL model in $1 + 1$ dimensions

Introduction

Background in
Quantum
Computation

Quantum logic
Quantum information
Quantum computing
models and algorithms
Applications

**Solving the NJL
model in $1 + 1$
dimensions**

Analytical solution
Lattice formulation
Fermion-qubit mapping
Space parametrization
and state preparation
Ground state energy

Conclusions and
prospective work



The NJL **Lagrangian density** that we will make use of looks like:

$$\mathcal{L}(x) = \bar{\psi}(x)(i\partial - m)\psi(x) + \mathcal{L}_I(x)$$

$$\mathcal{L}_I(x) = \frac{1}{2}G_\pi [\bar{\psi}(x)\psi(x)]^2$$

From this expression for the Lagrangian density we can obtain the corresponding **Hamiltonian density** through the Legendre transform. In special, for $1 + 1$ dimensions:

$$\begin{aligned}\mathcal{H} &\triangleq \frac{\delta\mathcal{L}}{\delta(\partial_0\psi)}(\partial_0\psi) + (\partial_0\bar{\psi})\frac{\delta\mathcal{L}}{\delta(\partial_0\bar{\psi})} - \mathcal{L} \\ &= \bar{\psi}(x)\left(m - i\gamma^1\partial_1\right)\psi(x) - \frac{1}{2}G_\pi [\bar{\psi}(x)\psi(x)]^2\end{aligned}$$

Dressed mass and the gap equation

Introduction

Background in
Quantum
Computation

Quantum logic
Quantum information
Quantum computing
models and algorithms
Applications

Solving the NJL
model in 1 + 1
dimensions

Analytical solution
Lattice formulation
Fermion-qubit mapping
Space parametrization
and state preparation
Ground state energy

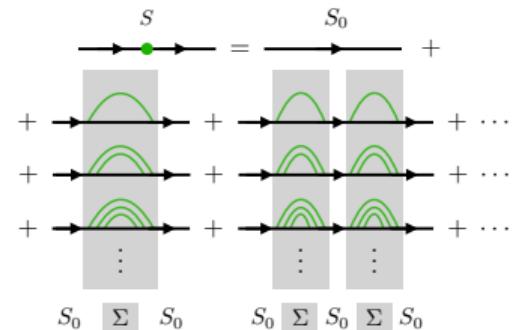
Conclusions and
prospective work



The **bare and dressed masses** appear on the bare quark propagator S_0 , and the NJL dressed quark propagator S respectively:

$$S_0^{-1} \triangleq p - m + i\varepsilon$$

$$S^{-1} \equiv S_{\text{NJL}}^{-1} \triangleq p - M + i\varepsilon$$



We can find a relationship between these two by solving the **gap equation**:

$$S^{-1} = S_0^{-1} - 2iG_\pi \int \frac{d^2 p}{(2\pi)^2} N_{\text{color}} N_{\text{flavor}} \text{Tr}_D[S]$$

$$M \simeq m + 4iG_\pi N_{\text{color}} N_{\text{flavor}} \int \frac{d^2 p}{(2\pi)^2} \frac{M}{p^2 - M^2}$$

Mass generation I

Introduction

Background in
Quantum
Computation

Quantum logic

Quantum information

Quantum computing
models and algorithms

Applications

Solving the NJL
model in 1 + 1
dimensions

Analytical solution

Lattice formulation

Fermion-qubit mapping

Space parametrization
and state preparation

Ground state energy

Conclusions and
prospective work

To solve this last expression we first perform a **Wick rotation** $p_0 \rightarrow ip_2 \Rightarrow (p_0)^2 \rightarrow -(p_2)^2$, followed by a transformation to **polar coordinates**:

$$M \simeq m + N_{\text{color}} N_{\text{flavor}} \frac{G_\pi}{\pi} \int_0^\infty \frac{M}{p_E^2 + M^2} dp_E^2$$

Finally, to make the integral converge, we introduce **proper time regularization**:

$$\frac{1}{x^n} = \frac{1}{(n-1)!} \int_0^\infty d\tau \tau^{n-1} \exp[-\tau x] \rightarrow \frac{1}{(n-1)!} \int_{1/\Lambda_{UV}^2}^{1/\Lambda_{IR}^2} d\tau \tau^{n-1} \exp[-\tau x]$$

$$M \simeq m + MN_{\text{color}} N_{\text{flavor}} \frac{G_\pi}{\pi} \int_{1/\Lambda_{UV}^2}^{1/\Lambda_{IR}^2} \frac{d\tau}{\tau} \exp[-\tau M^2]$$

Where Λ_{IR} and Λ_{UV} are the **infrared and ultraviolet cutoffs** respectively.



Mass generation II

Introduction

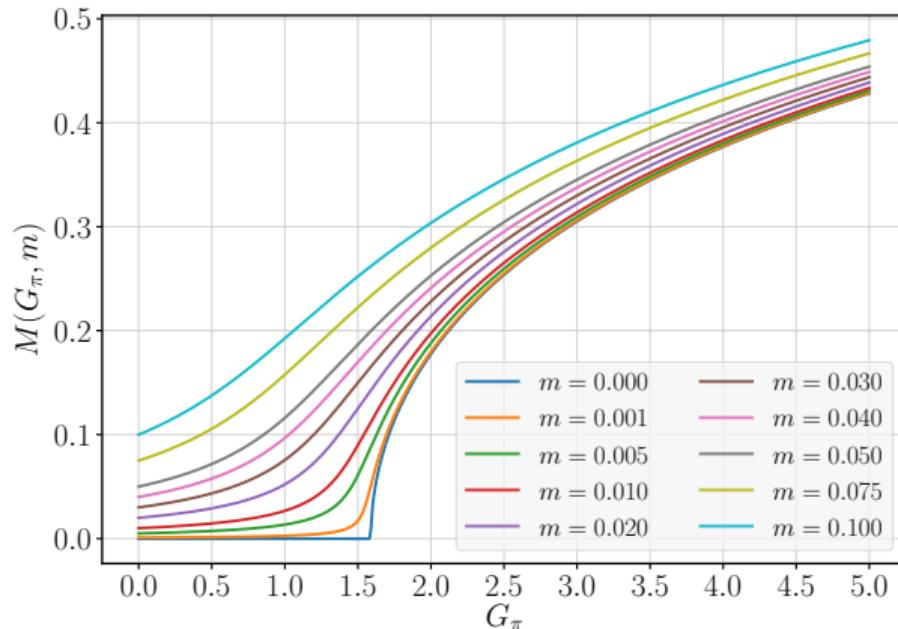
Background in
Quantum
Computation

Quantum logic
Quantum information
Quantum computing
models and algorithms
Applications

Solving the NJL
model in 1 + 1
dimensions

Analytical solution
Lattice formulation
Fermion-qubit mapping
Space parametrization
and state preparation
Ground state energy

Conclusions and
prospective work



N_{Dirac}	N_{color}	N_{flavor}	Λ_{IR}	Λ_{UV}
$1 + 1 \rightarrow 2$	1	1	0.240 GeV	0.645 GeV

Lattice formulation I

Introduction

Background in
Quantum
Computation

Quantum logic

Quantum information

Quantum computing
models and algorithms

Applications

Solving the NJL
model in 1 + 1
dimensions

Analytical solution

Lattice formulation

Fermion-qubit mapping

Space parametrization
and state preparation

Ground state energy

Conclusions and
prospective work

We can define the Hamiltonian of the system as the integral over space of the Hamiltonian density:

$$H = \int \mathcal{H}(x) dx = \int \left\{ \bar{\psi}(x) \left(m - i\gamma^1 \partial_1 \right) \psi(x) - \frac{1}{2} G_\pi [\bar{\psi}(x) \psi(x)]^2 \right\} dx$$

For a basis where:

$$\psi = \begin{bmatrix} \psi_+ \\ \psi_- \end{bmatrix}, \quad \bar{\psi} \triangleq \psi^\dagger \gamma^0, \quad \gamma^0 = \begin{bmatrix} 1 & 0 \\ 0 & -1 \end{bmatrix}, \quad \gamma^1 = \begin{bmatrix} 0 & -1 \\ 1 & 0 \end{bmatrix}$$

We can write the kinetic term as:

$$\bar{\psi} \left(-i\gamma^1 \partial_1 \right) \psi = \frac{i}{2} \left\{ \left[\psi_+^\dagger (\partial_1 \psi_-) - (\partial_1 \psi_+^\dagger) \psi_- \right] + \left[\psi_-^\dagger (\partial_1 \psi_+) - (\partial_1 \psi_-^\dagger) \psi_+ \right] \right\}$$



Lattice formulation II

Introduction

Background in
Quantum
Computation

Quantum logic
Quantum information
Quantum computing
models and algorithms
Applications

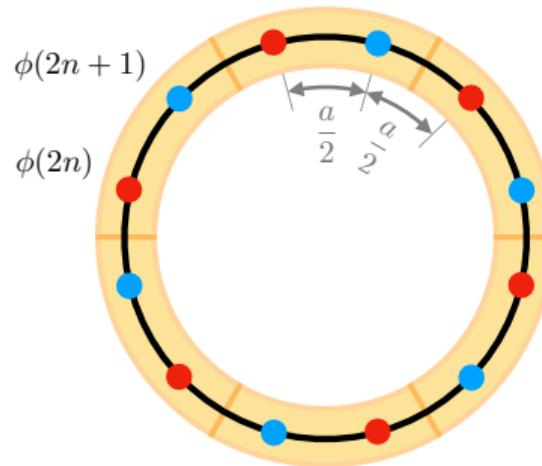
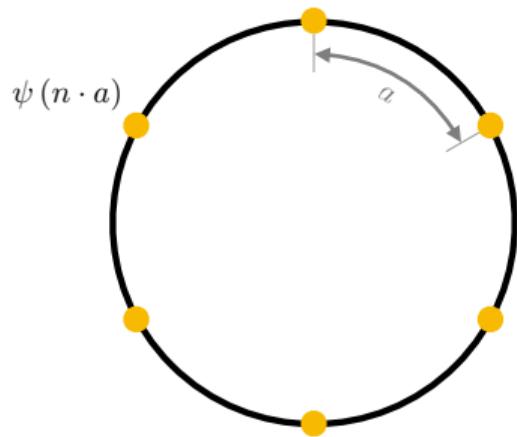
Solving the NJL
model in 1 + 1
dimensions

Analytical solution
Lattice formulation
Fermion-qubit mapping
Space parametrization
and state preparation
Ground state energy

Conclusions and
prospective work



The two groups in brackets are essentially equivalent to one another by virtue of exchanging positive and negative energy components. This is the motivation behind **staggered fermion lattices**, which use two computational lattice sites for each theoretical value of ψ .



Lattice formulation III

Introduction

Background in
Quantum
Computation

Quantum logic

Quantum information

Quantum computing
models and algorithms

Applications

Solving the NJL
model in 1 + 1
dimensions

Analytical solution

Lattice formulation

Fermion-qubit mapping

Space parametrization
and state preparation

Ground state energy

Conclusions and
prospective work



Sites in the staggered computational lattice are labeled using a parameter $n \in \mathbb{Z}$ such that all evaluations of ψ are made at integer multiples of the distance a :

$$\phi(n) \triangleq \sqrt{a} \begin{cases} \psi_+(\frac{n}{2}a), & 2 \mid n \\ \psi_-(\frac{n-1}{2}a), & 2 \nmid n \end{cases}$$

These newly defined operators obey the **canonical commutation relations for fermions**:

$$[\phi^\dagger(p), \phi(q)]_+ = \delta_{pq}, \quad [\phi(p), \phi(q)]_+ = 0$$

Finally, thanks to the periodic boundary conditions, we can write:

$$H_N^{(K)} = \frac{i}{a} \sum_{n=0}^{2N-1} [\phi^\dagger(n)\phi(n+1) - \phi^\dagger(n+1)\phi(n)]$$

Lattice formulation IV

Introduction

Background in
Quantum
Computation

Quantum logic
Quantum information
Quantum computing
models and algorithms
Applications

Solving the NJL
model in 1 + 1
dimensions

Analytical solution
Lattice formulation
Fermion-qubit mapping
Space parametrization
and state preparation
Ground state energy

Conclusions and
prospective work



From this Hamiltonian, we can now recover the **massless Dirac equation** in the continuum limit; which serves as proof of correctness:

$$\dot{\phi}(n) = i \left[H_N^{(K)}, \phi(n) \right]_- = \frac{\phi(n+1) - \phi(n-1)}{a}$$

In terms of the original fields, this is:

$$\dot{\psi}_+ = \frac{\Delta \psi_-}{\Delta x}, \quad \dot{\psi}_- = \frac{\Delta \psi_+}{\Delta x}$$

Lastly, taking the limit when $a \rightarrow 0$:

$$\frac{\partial}{\partial t} \psi = \hat{\alpha}_1 \frac{\partial}{\partial x} \psi$$

$$\hat{\alpha}_1 \triangleq \gamma_0 \gamma_1 = \gamma^0 \gamma_1 = -\gamma^0 \gamma^1 = \begin{bmatrix} 0 & 1 \\ 1 & 0 \end{bmatrix}$$

Lattice formulation V

Introduction

Background in
Quantum
Computation

Quantum logic

Quantum information

Quantum computing

models and algorithms

Applications

Solving the NJL
model in 1 + 1
dimensions

Analytical solution

Lattice formulation

Fermion-qubit mapping

Space parametrization
and state preparation

Ground state energy

Conclusions and
prospective work



And it is now straight forward to obtain all other components of the Hamiltonian from the expressions in the Hamiltonian density, which are written in terms of **Dirac bilinears**.

$$H_N = H_N^{(M)} + H_N^{(K)} + H_N^{(I)}$$

$$H_N^{(M)} = m \sum_{n=0}^{2N-1} (-1)^n \phi^\dagger(n) \phi(n)$$

$$H_N^{(K)} = \frac{i}{a} \sum_{n=0}^{2N-1} [\phi^\dagger(n) \phi(n+1) - \phi^\dagger(n+1) \phi(n)]$$

$$H_N^{(I)} = -\frac{1}{2a} G_\pi \sum_{n=0}^{N-1} [\phi^\dagger(2n) \phi(2n) - \phi^\dagger(2n+1) \phi(2n+1)]^2$$

Fermion-qubit mapping

Introduction

Background in
Quantum
Computation

Quantum logic
Quantum information
Quantum computing
models and algorithms
Applications

Solving the NJL
model in 1 + 1
dimensions

Analytical solution
Lattice formulation
Fermion-qubit mapping
Space parametrization
and state preparation
Ground state energy

Conclusions and
prospective work



Generally speaking, quantum computers cannot measure any given operator directly. Therefore, in order to simulate this or any other Hamiltonian in a quantum processor, one needs to efficiently map its component operators onto ones suitable for evaluation in such machines. The most commonly used of these sets are Pauli operators along with the identity: this is what we will use and refer to as the **Pauli set**.

It turns out that, in one spatial dimension, spin- $\frac{1}{2}$ particles (i.e. qubits) behave much like fermions. This allows us to implement a mapping between one and the other. The **Jordan-Wigner transform** associates spin “down” and “up” with occupied and unoccupied fermion states (or vice versa). Particularly, we will choose:

$$|\uparrow\rangle \equiv |0\rangle, \quad |\downarrow\rangle \equiv |1\rangle$$

$$|\downarrow\rangle \equiv \phi^\dagger |0\rangle, \quad |\uparrow\rangle \equiv \phi |1\rangle$$

$$\phi \rightarrow \sigma^+, \quad \phi^\dagger \rightarrow \sigma^-$$

Jordan-Wigner transform I

Introduction

Background in
Quantum
Computation

Quantum logic
Quantum information
Quantum computing
models and algorithms
Applications

Solving the NJL
model in 1 + 1
dimensions

Analytical solution
Lattice formulation
Fermion-qubit mapping
Space parametrization
and state preparation
Ground state energy

Conclusions and
prospective work



To prove that this is valid, we can define the following equivalences by drawing inspiration from raising and lowering operators in angular momentum theory:

$$\sigma^1 \equiv \phi^\dagger + \phi$$

$$\sigma^2 \equiv i(\phi^\dagger - \phi)$$

$$\sigma^3 \equiv 1 - 2\phi^\dagger \phi$$

Making use of the properties of fermion creation and annihilation operators, comprised within their canonical commutation relations, these can be shown to behave algebraically like **Pauli matrices**:

$$[\sigma^p, \sigma^q]_- = 2i\epsilon_{pqr}\sigma^r$$

$$[\sigma^p, \sigma^q]_+ = 2\delta_{pq}$$

Jordan-Wigner transform II

Introduction

Background in
Quantum
Computation

Quantum logic
Quantum information
Quantum computing
models and algorithms
Applications

Solving the NJL
model in 1 + 1
dimensions

Analytical solution
Lattice formulation
Fermion-qubit mapping
Space parametrization
and state preparation
Ground state energy

Conclusions and
prospective work



We can recover the usual spin raising and lowering hermitian conjugate operators:

$$\sigma^{\pm} \triangleq \frac{1}{2} (\sigma^1 \pm i\sigma^2) \quad (1)$$

All this allows us to use these spins as a basic model for fermions:

$$[\phi, \phi^\dagger]_+ = 1 \quad \rightarrow \quad [\sigma^+, \sigma^-]_+ = 1 \quad (2)$$

Unfortunately, this only works for single-fermion representations, and needs to be modified once we introduced more than one particle; since independent spins commute, while independent fermions anticommute.

$$[\sigma^+(p), \sigma^-(q)]_- = \delta_{pq}, \quad [\sigma^+(p), \sigma^-(q)]_+ \neq \delta_{pq}$$

$$[\phi(p), \phi^\dagger(q)]_- \neq \delta_{pq}, \quad [\phi(p), \phi^\dagger(q)]_+ = \delta_{pq}$$

Jordan-Wigner transform III

Introduction

Background in
Quantum
Computation

Quantum logic
Quantum information
Quantum computing
models and algorithms
Applications

Solving the NJL
model in $1+1$
dimensions

Analytical solution
Lattice formulation
Fermion-qubit mapping
Space parametrization
and state preparation
Ground state energy

Conclusions and
prospective work



A way to fix this issue is by defining $N(l) = \phi^\dagger(l)\phi(l)$ as the hermitian number operator for state l , and attaching a so called unitary **string operator** $S(n)$ to the fermion operators:

$$\begin{aligned} S(n)\phi(n) &\rightarrow \sigma^+(n) \\ \phi^\dagger(n)S^\dagger(n) &\rightarrow \sigma^-(n) \\ S(n) &= \exp\left[-i\pi \sum_{l < n} [N(l) + s(l)]\right] \end{aligned}$$

The $s(l)$ terms in the string operator are scalars associated to gauge transformations and do not add much to the transform. With this new mapping, Pauli matrices get redefined to:

$$\begin{aligned} \sigma^1(n) &\equiv \phi^\dagger(n)S^\dagger(n) + S(n)\phi(n) \\ \sigma^2(n) &\equiv i\left[\phi^\dagger(n)S^\dagger(n) - S(n)\phi(n)\right] \\ \sigma^3(n) &\equiv 1 - 2\phi^\dagger(n)\phi(n) \end{aligned}$$

Jordan-Wigner transform IV

Introduction

Background in
Quantum
Computation

Quantum logic
Quantum information
Quantum computing
models and algorithms
Applications

Solving the NJL
model in 1 + 1
dimensions

Analytical solution
Lattice formulation
Fermion-qubit mapping
Space parametrization
and state preparation
Ground state energy

Conclusions and
prospective work



We now retrieve the correct statistics. Of course, we would like to express the string operator in terms of the Pauli set so that we can move it to the other side of the transformation. Fortunately, we can do so by expanding the number operators:

$$\exp\left[\pm i\pi \sum_{l < n} N(l)\right] = \prod_{l < n} \exp\left[\pm i\pi \phi^\dagger(l) \phi(l)\right] = \prod_{l < n} [1 - 2\phi^\dagger(l) \phi(l)] = \prod_{l < n} \sigma^3(l)$$
$$S(n) = \prod_{l < n} e^{-i\pi s(l)} \sigma^3(l)$$

Finally, our **choice of gauge** will be such that $s(l) = s \in (-1, 1] \forall l$ and the string operator is hermitian for all values of n . All in all, this can be achieved by making $s = 0$, which gives:

$$\phi(n) \rightarrow \left[\prod_{l < n} \sigma^3(l) \right] \sigma^+(n), \quad \phi^\dagger(n) \rightarrow \left[\prod_{l < n} \sigma^3(l) \right] \sigma^-(n)$$

Refactoring the NJL Hamiltonian I

Introduction

Background in
Quantum
Computation

Quantum logic
Quantum information
Quantum computing
models and algorithms
Applications

Solving the NJL
model in 1 + 1
dimensions

Analytical solution
Lattice formulation
Fermion-qubit mapping
Space parametrization
and state preparation
Ground state energy

Conclusions and
prospective work

Now that we have a way of converting from fermion to spin operators, we can apply it to the NJL Hamiltonian:

$$H_N^{(M)} \rightarrow \frac{m}{2} \sum_{n=0}^{2N-1} (-1)^{n+1} \sigma^3(n)$$

$$H_N^{(K)} \rightarrow \frac{i}{a} \sum_{n=0}^{2N-1} [\sigma^-(n) \sigma^+(n+1) - \sigma^-(n+1) \sigma^+(n)]$$

$$H_N^{(I)} \rightarrow \frac{G_\pi}{4a} \sum_{n=0}^{N-1} [\sigma^3(2n) \sigma^3(2n+1) - N]$$

We will only be interested in studying the **Chiral limit** (i.e. $m = 0$), which justifies dropping the entire mass term. From the interaction Hamiltonian we will get an adiabatic modification term of the form $\frac{G_\pi N}{4a}$; this will also be dropped for the moment.



Refactoring the NJL Hamiltonian II

Introduction

Background in
Quantum
Computation

Quantum logic

Quantum information

Quantum computing
models and algorithms

Applications

Solving the NJL
model in 1 + 1
dimensions

Analytical solution

Lattice formulation

Fermion-qubit mapping

Space parametrization
and state preparation

Ground state energy

Conclusions and
prospective work



Hence, with periodic boundary conditions $\sigma^p(N) = \sigma^p(0)$ and expressed in quantum computing notation $(\mathbb{1}_n, X_n, Y_n, Z_n) \equiv \sigma^p(n)$ for $p = (0, 1, 2, 3)$, our refactored Hamiltonian $P_N \triangleq 2aH_N$ will adopt the form:

$$P_N = \sum_{n=0}^{2N-1} [X_{n+1}Y_n - Y_{n+1}X_n] + \frac{G_\pi}{2} \sum_{n=0}^{N-1} Z_{2n+1}Z_{2n} \quad (3)$$

One important feature of this operator is that its size (i.e. number of terms) **grows polynomially** with the size of the system N . It is also interesting to notice that this operator can be expressed in a recursive layout:

$$\begin{aligned} P_{N+1} = & P_N - Y_{2N-1}X_0 + X_{2N-1}Y_0 + Y_{2N+1}X_0 - X_{2N+1}Y_0 \\ & + X_{2N}Y_{2N-1} - Y_{2N}X_{2N-1} + X_{2N+1}Y_{2N} - Y_{2N+1}X_{2N} + \frac{G_\pi}{2}(Z_{2N+1}Z_{2N}) \end{aligned}$$

Space parametrization and state preparation

Introduction

Background in
Quantum
Computation

Quantum logic
Quantum information
Quantum computing
models and algorithms
Applications

Solving the NJL
model in 1 + 1
dimensions

Analytical solution
Lattice formulation
Fermion-qubit mapping
Space parametrization
and state preparation
Ground state energy

Conclusions and
prospective work



Once we have ways of measuring our Hamiltonian, we need to be able to explore different quantum states; a process known as **state preparation**. This can be achieved by parametrizing the Hilbert/Fock space of states representing the system, and finding a way to prepare the corresponding quantum state in the processor given any combination of those parameters.

For this task we will be employing some of the IBM-Q quantum computers and simulators; accessible through the cloud via the Qiskit framework. Nevertheless, there is one major shortcoming: the dimension of the space at hand grows exponentially with the number of qubits $2N$ used in the representation of the system. This means that the parametrization will become too large to handle both in state of the art and forthcoming machines unless it is treated in a smart way.



Unitary coupled cluster I

Introduction

Background in
Quantum
Computation

Quantum logic
Quantum information
Quantum computing
models and algorithms
Applications

Solving the NJL
model in 1 + 1
dimensions

Analytical solution
Lattice formulation
Fermion-qubit mapping
Space parametrization
and state preparation
Ground state energy

Conclusions and
prospective work



The motivation behind coupled cluster methods arises from the idea of exploring only the regions of phase space close to an initial **reference state**. The choice of said reference state is of great importance for retrieving successful results. Let us begin by choosing the region of study in terms of **Hamming distance** (i.e. bit flips). This is sometimes called **configuration interaction** (CI). Using a notation where $\sigma_n^P \equiv \sigma^P(n)$ we can write any state one spin flip away from the initial reference state as:

$$|CI_1(z^n)\rangle \triangleq \sum_n z^n \sigma_n^+ |SR\rangle$$

This approach can be easily extended to account for more and more states:

$$|CI_k(z)\rangle \triangleq \sum_{j=1}^k T_j(z^{n_1, \dots, n_j}) |SR\rangle$$

$$T_j(z^{n_1, \dots, n_j}) \triangleq \sum_{n_1, \dots, n_j} z^{n_1, \dots, n_j} \sigma_{n_1}^+ \cdots \sigma_{n_j}^+$$

Unitary coupled cluster II

Introduction

Background in
Quantum
Computation

Quantum logic

Quantum information

Quantum computing

models and algorithms

Applications

Solving the NJL
model in 1 + 1
dimensions

Analytical solution

Lattice formulation

Fermion-qubit mapping

Space parametrization
and state preparation

Ground state energy

Conclusions and
prospective work



However, this approach presents a number of deficiencies which render it non-optimal; the biggest one for us being that it presents no clear advantage over classical state preparation. The idea behind **coupled cluster** (CC) consists on using the spin flips as generators instead:

$$|\text{CC}_k(\mathbf{z})\rangle \triangleq \exp \left[\sum_{j=1}^k T_j(z^{n_1, \dots, n_j}) \right] |\text{SR}\rangle$$

In order to make this ansatz suitable for quantum processors, we need to express it in terms of unitary transformations. We have finally arrived at **unitary coupled cluster** (UCC):

$$|\text{UCC}_k(\boldsymbol{\theta}, \boldsymbol{\eta})\rangle \triangleq \exp \left[\sum_{j=1}^k T_j^-(\theta^{n_1, \dots, n_j}) + i \sum_{j=1}^k T_j^+(\eta^{n_1, \dots, n_j}) \right] |\text{SR}\rangle$$
$$T_j^\pm(\theta^{n_1, \dots, n_j}) \triangleq \sum_{n_1, \dots, n_j} \theta^{n_1, \dots, n_j} (\sigma_{n_1}^+ \cdots \sigma_{n_j}^+ \pm \sigma_{n_1}^- \cdots \sigma_{n_j}^-)$$

Unitary coupled cluster III

Introduction

Background in
Quantum
Computation

Quantum logic
Quantum information
Quantum computing
models and algorithms
Applications

Solving the NJL
model in 1 + 1
dimensions

Analytical solution
Lattice formulation
Fermion-qubit mapping
Space parametrization
and state preparation
Ground state energy

Conclusions and
prospective work



Another important version of this ansatz is developed in a **fermionic basis** (i.e. for parametrizing Fock space instead of Hilbert space), and replaces spin-flips with changes in the state of particles. This is interesting because, due to the non-locality of the Jordan-Wigner transform, if we were to apply the unitary coupled cluster parametrization directly onto the Jorda-Wigner's mapping image (i.e. Hilbert space), we might be exploring uninteresting regions of the domain (i.e. Fock space) —such as those containing symmetry-broken states. On top of that, this method is usually developed so that it **conserves the total number of particles** in the system; which is predefined through the fermion reference (FR):

$$|\text{FUCC}_k(\theta, \eta)\rangle \triangleq \exp \left[\sum_{j=1}^k F_j^- (\theta^{p_1, \dots, p_j, q_1 \dots, q_j}) + i \sum_{j=1}^k F_j^+ (\eta^{p_1, \dots, p_j, q_1 \dots, q_j}) \right] |\text{FR}\rangle_Q^F$$
$$F_j^\pm (\theta^{p_1, \dots, p_j, q_1 \dots, q_j}) \triangleq \sum_{\substack{q \in Q \\ p \in \overline{Q}}} \theta^{p_1, \dots, p_j, q_1 \dots, q_j} \left(\phi_{p_1}^\dagger \cdots \phi_{p_j}^\dagger \phi_{q_1} \cdots \phi_{q_j} \pm \phi_{q_1}^\dagger \cdots \phi_{q_j}^\dagger \phi_{p_1} \cdots \phi_{p_j} \right)$$

Custom symmetry-based parametrization ansatz I

Introduction

Background in
Quantum
Computation

Quantum logic
Quantum information
Quantum computing
models and algorithms
Applications

Solving the NJL
model in 1 + 1
dimensions

Analytical solution
Lattice formulation
Fermion-qubit mapping
Space parametrization
and state preparation
Ground state energy

Conclusions and
prospective work

While general, it is easy to notice that the implementation of any variant of the unitary coupled cluster ansatz is still cumbersome and can depend on a large number of parameters based on the chosen order of truncation. We will now introduce a new ansatz inspired by the shape of our refactored Hamiltonian. For this task, we will analyze the two distinct parts in our Hamiltonian independently; since these will dominate in two **different regimes**:

INFINITELY STRONG INTERACTIONS

Interaction term dominates (i.e. $G_\pi \rightarrow \infty$)

$$G_N \triangleq \sum_{n=0}^{N-1} Z_{2n+1} Z_{2n}$$

INFINITELY WEAK INTERACTIONS

Kinetic term dominates (i.e. $G_\pi \rightarrow 0$)

$$K_N \triangleq \sum_{n=0}^{2N-1} [X_{n+1} Y_n - Y_{n+1} X_n]$$

Let us call each computational basis state by the decimal translation of its binary form:

$$|0\rangle \triangleq |0b\ldots0000\rangle, \quad |1\rangle \triangleq |0b\ldots0001\rangle, \quad |2\rangle \triangleq |0b\ldots0010\rangle, \quad \dots$$



Custom symmetry-based parametrization ansatz II

Introduction

Background in
Quantum
Computation

Quantum logic
Quantum information
Quantum computing
models and algorithms
Applications

Solving the NJL
model in 1 + 1
dimensions

Analytical solution
Lattice formulation
Fermion-qubit mapping
Space parametrization
and state preparation
Ground state energy

Conclusions and
prospective work



For the **interaction term** we notice that:

- We can cycle in steps of two computational lattice sites.
- We can change the position of any theoretical lattice site; since the interaction only occurs between positive and negative energy components at the same spot.
- We can interchange positive and negative components in any number of theoretical lattice sites. We call each of these a *swap transformation*.
- We can flip every component $0 \leftrightarrow 1$ in any number of theoretical lattice sites. We call each of these a *flip transformation*.

This means that for $N = 2$ (i.e. $2^{2N} = 16$ basis states):

$$\begin{aligned}|0\rangle &\equiv |3\rangle \equiv |12\rangle \equiv |15\rangle \\|1\rangle &\equiv |2\rangle \equiv |4\rangle \equiv |7\rangle \equiv |8\rangle \equiv |11\rangle \equiv |13\rangle \equiv |14\rangle \\|5\rangle &\equiv |6\rangle \equiv |9\rangle \equiv |10\rangle\end{aligned}$$

Custom symmetry-based parametrization ansatz III

Introduction

Background in
Quantum
Computation

Quantum logic

Quantum information

Quantum computing

models and algorithms

Applications

Solving the NJL
model in 1 + 1
dimensions

Analytical solution

Lattice formulation

Fermion-qubit mapping

Space parametrization
and state preparation

Ground state energy

Conclusions and
prospective work



Which represent the partition in degenerate subspaces of associated eigenvalues $\{+2, 0, -2\}$ respectively. These eigenvalues come from each theoretical lattice site in the spin-Z basis contributing with either ± 1 . Notice that the eigenvalues of this operator are always symmetrically disposed about zero, which means that $\forall N$:

$$\gamma_{\max}^N = -\gamma_{\min}^N$$

The interaction is maximum inside any theoretical lattice site whenever there is equal presence of both positive and negative energy components; conversely, the interaction is minimum whenever there is only one component present. Of course, the maximum (minimum) of the operator occurs when all theoretical lattice sites are maximized (minimized) individually:

$$|\gamma_{\max}\rangle_n \in \{|\theta b00\rangle, |\theta b11\rangle\}, \quad |\gamma_{\min}\rangle_n \in \{|\theta b01\rangle, |\theta b10\rangle\}$$

$$|\gamma_{\max}^N\rangle \equiv |\gamma_{\max}\rangle^{\otimes N}, \quad |\gamma_{\min}^N\rangle \equiv |\gamma_{\min}\rangle^{\otimes N}$$

Custom symmetry-based parametrization ansatz IV

Introduction

Background in
Quantum
Computation

Quantum logic

Quantum information

Quantum computing

models and algorithms

Applications

Solving the NJL
model in 1 + 1
dimensions

Analytical solution

Lattice formulation

Fermion-qubit mapping

Space parametrization
and state preparation

Ground state energy

Conclusions and
prospective work



Moving on to the **kinetic term** we have:

- We can cycle in steps of one computational lattice site. We call this a *cycling transformation*.
- We cannot change the position of any theoretical lattice site; since the computational lattice sites now form a chain with their nearest neighbors.
- If we interchange positive and negative components in all theoretical lattice sites, the resulting expectation value flips its sign (i.e. *global swap transformations* are antisymmetric).
- Flip transformations do not represent any apparent symmetry.

Once more, this operator has its eigenstates symmetrically disposed around zero. Also, global swap transformations are equivalent to complex conjugation:

$$\kappa_{\max}^N = -\kappa_{\min}^N$$

$$K_N^\dagger = (K_N^*)^T = (-K_N)^T = K_N \quad \Rightarrow \quad K_N^T = -K_N = K_N^*$$

Custom symmetry-based parametrization ansatz V

Introduction

Background in
Quantum
Computation

Quantum logic
Quantum information
Quantum computing
models and algorithms
Applications

Solving the NJL
model in 1 + 1
dimensions

Analytical solution
Lattice formulation
Fermion-qubit mapping
Space parametrization
and state preparation
Ground state energy

Conclusions and
prospective work



Table: Global swap transformations for the $N = 2$ basis states according to their number of particles.

$N_{\text{particles}}$	Global swap transformations
0	$ 0\rangle \leftrightarrow 0\rangle$
1	$ 1\rangle \leftrightarrow 2\rangle, 4\rangle \leftrightarrow 8\rangle$
2	$ 3\rangle \leftrightarrow 3\rangle, 5\rangle \leftrightarrow 10\rangle, 6\rangle \leftrightarrow 9\rangle, 12\rangle \leftrightarrow 12\rangle$
3	$ 7\rangle \leftrightarrow 11\rangle, 13\rangle \leftrightarrow 14\rangle$
4	$ 15\rangle \leftrightarrow 15\rangle$

Table: Cycles for the $N = 2$ basis states according to their number of particles.

$N_{\text{particles}}$	Cycles
0	$ 0\rangle$
1	$ 1\rangle \leftrightarrow 2\rangle \leftrightarrow 4\rangle \leftrightarrow 8\rangle$
2	$ 3\rangle \leftrightarrow 6\rangle \leftrightarrow 12\rangle \leftrightarrow 9\rangle, 5\rangle \leftrightarrow 10\rangle$
3	$ 7\rangle \leftrightarrow 14\rangle \leftrightarrow 13\rangle \leftrightarrow 11\rangle$
4	$ 15\rangle$

Custom symmetry-based parametrization ansatz VI

Introduction

Background in
Quantum
Computation

Quantum logic
Quantum information
Quantum computing
models and algorithms
Applications

Solving the NJL
model in 1 + 1
dimensions

Analytical solution
Lattice formulation
Fermion-qubit mapping
Space parametrization
and state preparation
Ground state energy

Conclusions and
prospective work



If an eigenstate is not degenerate, it must comply with the rule that all amplitudes multiplying the states that make up its superposition, and which are related by a cycling transformation, must be the same up to a constant *phase factor*.

$$e^{i\phi} \in \left\{ \exp\left(i\frac{2\pi}{p}n\right) : n \in \mathbb{Z} \text{ and } p \triangleq \text{size of the smaller cycle} \right\}$$

Finally, because this operator is made out of the Pauli X and Y matrices, and these matrices—regardless of any phase factors—flip the state they are applied to (i.e. $0 \leftrightarrow 1$), we can see that any maximum (minimum) eigenstate will necessarily have the **same number of occupied and unoccupied states**. If we naively parametrize the space of states with this number of particles, we will get a prohibitive parametrization that grows exponentially in the number of basis states as $\binom{2N}{N}$. However, if we assume that these eigenstates are not degenerate, we can apply the above mentioned rules to build a simpler general form for them.

Custom symmetry-based parametrization ansatz VII

Introduction

Background in
Quantum
Computation

Quantum logic
Quantum information
Quantum computing
models and algorithms
Applications

Solving the NJL
model in 1 + 1
dimensions

Analytical solution
Lattice formulation
Fermion-qubit mapping
Space parametrization
and state preparation
Ground state energy

Conclusions and
prospective work



Because the smallest cycle for $N = 2$ is of size two, any phase factors between elements of the same cycle can only be ± 1 . Also, we need to make sure that a swap transformation in all theoretical lattice sites —equivalent to complex conjugation— applied to the either the maximum or minimum eigenstate, reproduces its counterpart.

$$\alpha \frac{1}{\sqrt{2}}(|3\rangle + |12\rangle) - \beta \frac{1}{\sqrt{2}}(|6\rangle + |9\rangle) + \delta \frac{i}{\sqrt{2}}(|5\rangle - |10\rangle)$$

Which can then be converted into a general normalized form by resorting to an analogy with spherical coordinates. This results in the **symmetry based parametrization** (SBP):

$$|\text{SBP}_2(\theta, \eta)\rangle \triangleq \sin(\theta) \sin(\eta) |\gamma_{\max}^2\rangle - \sin(\theta) \cos(\eta) |\gamma_{\min,1}^2\rangle + i \cos(\theta) |\gamma_{\min,2}^2\rangle$$

$$|\gamma_{\max}^2\rangle \triangleq \frac{|3\rangle + |12\rangle}{\sqrt{2}}, \quad |\gamma_{\min,1}^2\rangle \triangleq \frac{|6\rangle + |9\rangle}{\sqrt{2}}, \quad |\gamma_{\min,2}^2\rangle \triangleq \frac{|5\rangle - |10\rangle}{\sqrt{2}}$$

Custom symmetry-based parametrization ansatz VIII

Introduction

Background in
Quantum
Computation

Quantum logic

Quantum information

Quantum computing
models and algorithms

Applications

Solving the NJL
model in 1 + 1
dimensions

Analytical solution

Lattice formulation

Fermion-qubit mapping

Space parametrization
and state preparation

Ground state energy

Conclusions and
prospective work



As a matter of fact, this state can indeed evaluate to the minimum and maximum eigenstates of the operator when $N = 2$:

$$|\kappa_{\max}^2\rangle = \frac{1}{2\sqrt{2}}(|3\rangle - |6\rangle - |9\rangle + |12\rangle) - \frac{i}{2}(|5\rangle - |10\rangle)$$

$$|\kappa_{\min}^2\rangle = \frac{1}{2\sqrt{2}}(|3\rangle - |6\rangle - |9\rangle + |12\rangle) + \frac{i}{2}(|5\rangle - |10\rangle)$$

The particular choice when assigning the basis states in the spherical coordinates analogy, was made so that these maximum and minimum eigenstates evaluate for simple values of the parameters:

$$|\kappa_{\max}^2\rangle \equiv |\text{SBP}_2\left(\frac{3\pi}{4}, \frac{\pi}{4}\right)\rangle, \quad |\kappa_{\min}^2\rangle \equiv |\text{SBP}_2\left(\frac{\pi}{4}, \frac{\pi}{4}\right)\rangle$$

$$|\gamma_{\max}^2\rangle \equiv |\text{SBP}_2\left(\frac{\pi}{2}, \frac{\pi}{2}\right)\rangle, \quad |\gamma_{\min,1}^2\rangle \equiv |\text{SBP}_2\left(\frac{\pi}{2}, 0\right)\rangle, \quad |\gamma_{\min,2}^2\rangle \equiv |\text{SBP}_2(0, 0)\rangle$$

Parametrization ansatz implementation I

Introduction

Background in
Quantum
Computation

Quantum logic

Quantum information

Quantum computing
models and algorithms

Applications

Solving the NJL
model in 1 + 1
dimensions

Analytical solution

Lattice formulation

Fermion-qubit mapping

Space parametrization
and state preparation

Ground state energy

Conclusions and
prospective work



In order to implement this parametrization on any of the IBM-Q quantum computers, we need to be able to write it down as a quantum circuit. This means that it has to be expressed through an unitary operator $U(\theta, \eta)$ in the form:

$$|\text{SBP}_2(\theta, \eta)\rangle = U(\theta, \eta) |\text{SR}\rangle \quad (4)$$

At first we will only be interested in obtaining the ground state energy of our system for positive values of the coupling constant (i.e. we will only need the minimum eigenstates), this allows us to simplify even further the parametrization introduced in the previous section. This will allow us to reduce the subspace we are looking into from three dimensions down to two.

$$|\gamma\rangle \equiv \left| \gamma_{\min,2}^2 \right\rangle \triangleq \frac{|5\rangle - |10\rangle}{\sqrt{2}} \equiv \left| \text{SBP}_2\left(0, \frac{\pi}{4}\right) \right\rangle \quad (5)$$

$$|\kappa\rangle \triangleq \frac{|3\rangle - |6\rangle - |9\rangle + |12\rangle}{2} \equiv \left| \text{SBP}_2\left(\frac{\pi}{2}, \frac{\pi}{4}\right) \right\rangle \quad (6)$$

Parametrization ansatz implementation II

Introduction

Background in
Quantum
Computation

Quantum logic

Quantum information

Quantum computing
models and algorithms

Applications

Solving the NJL
model in 1 + 1
dimensions

Analytical solution

Lattice formulation

Fermion-qubit mapping

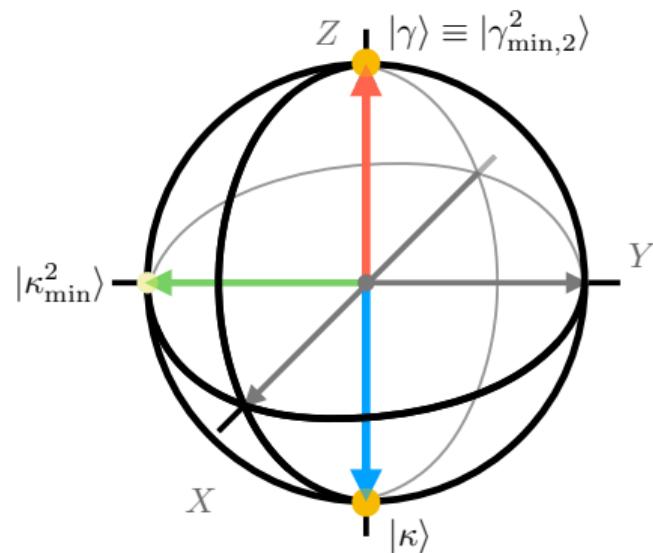
Space parametrization
and state preparation

Ground state energy

Conclusions and
prospective work

This will effectively cut down the degrees of freedom from two to just one (i.e. fixing $\eta = \pi/4$). However, having now only two distinguishable quantum states, we can choose to ease the parametrization to account for the entirety of the Hilbert space associated to the **qubit** they comprise; spuriously increasing the degrees of freedom back to two, but making the implementation as a quantum circuit conceptually easier.

We will parametrize an **ancilla qubit**, and then map its basis states to the two basis states of the subspace that we are interested in exploring.



Parametrization ansatz implementation III

Introduction

Background in
Quantum
Computation

Quantum logic
Quantum information
Quantum computing
models and algorithms
Applications

Solving the NJL
model in 1 + 1
dimensions

Analytical solution
Lattice formulation
Fermion-qubit mapping
Space parametrization
and state preparation
Ground state energy

Conclusions and
prospective work

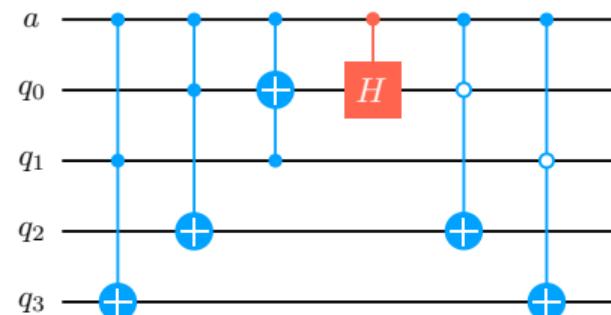
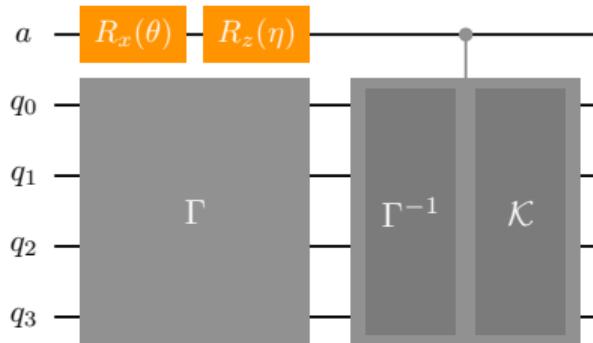


Figure: (Left) Quantum circuit to map the ancilla qubit onto the target qubit Hilbert space in our system. (Right) Simplified controlled $\mathcal{K}\Gamma^{-1}$ gate.



Parametrization ansatz implementation IV

Introduction

Background in
Quantum
Computation

Quantum logic
Quantum information
Quantum computing
models and algorithms
Applications

Solving the NJL
model in 1 + 1
dimensions

Analytical solution
Lattice formulation
Fermion-qubit mapping
Space parametrization
and state preparation
Ground state energy

Conclusions and
prospective work

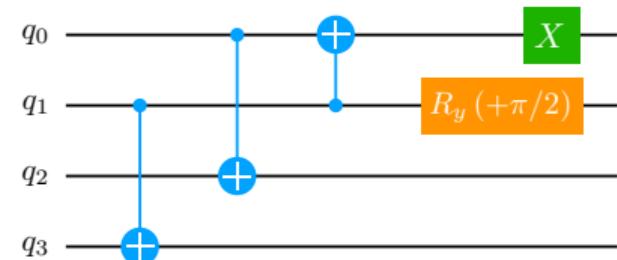
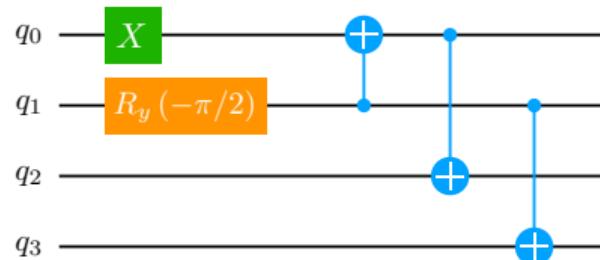


Figure: (Left) Preparation Γ of state $|\gamma\rangle$. (Right) Quantum gate Γ^{-1} for reversing state $|\gamma\rangle$.



Parametrization ansatz implementation V

Introduction

Background in
Quantum
Computation

Quantum logic
Quantum information
Quantum computing
models and algorithms
Applications

Solving the NJL
model in 1 + 1
dimensions

Analytical solution
Lattice formulation
Fermion-qubit mapping
Space parametrization
and state preparation
Ground state energy

Conclusions and
prospective work

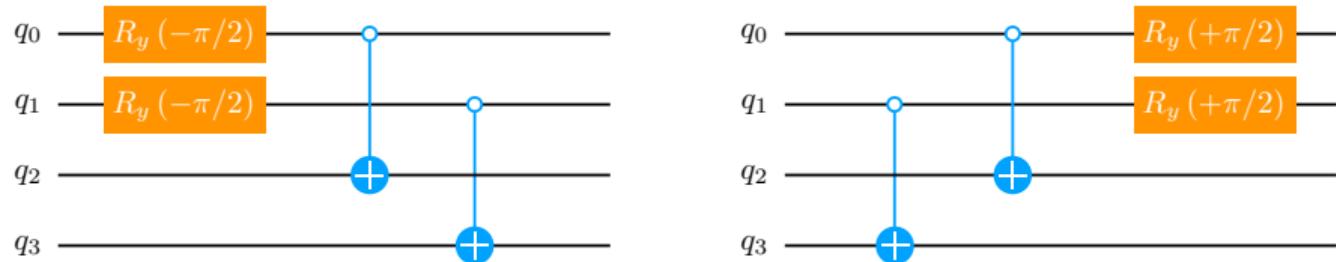


Figure: (Left) Preparation \mathcal{K} of state $|\kappa\rangle$. (Right) Quantum gate \mathcal{K}^{-1} for reversing state $|\kappa\rangle$.



Parametrization ansatz implementation VI

Introduction

Background in
Quantum
Computation

Quantum logic
Quantum information
Quantum computing
models and algorithms
Applications

Solving the NJL
model in 1 + 1
dimensions

Analytical solution
Lattice formulation
Fermion-qubit mapping
Space parametrization
and state preparation
Ground state energy

Conclusions and
prospective work



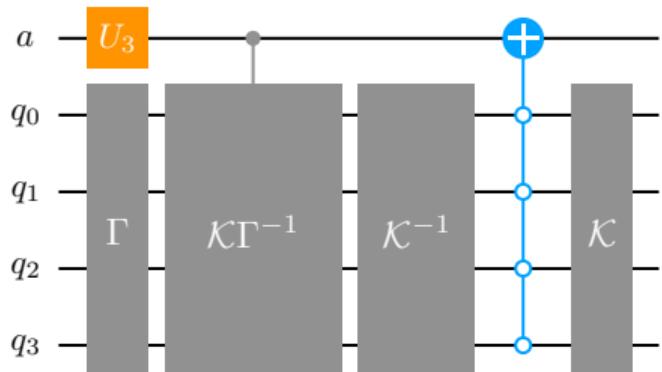
This parametrization will work if we are interested in measuring in the computational basis only (i.e. Pauli-Z measurements). Nonetheless, if we change our basis before measuring (e.g. to Pauli-X or Pauli-Y) we are going to face a problem: the resulting states of our system will be entangled with the ancilla qubit, and so, states that should be indistinguishable from one another will turn out distinct; because of this, the probability distributions will not be correct.

$$\Pr(|\psi\rangle_{\text{distinct}}) = \|\psi_\gamma\|^2 + \|\psi_\kappa\|^2$$

$$\Pr(|\psi\rangle_{\text{indist}}) = \|\psi_\gamma + \psi_\kappa\|^2$$

$$\Pr(|\psi\rangle_{\text{distinct}}) \geq \Pr(|\psi\rangle_{\text{indist}})$$

Therefore, as a last step, we need to **break this entanglement**.



Variational quantum eigensolver algorithm I

Introduction

Background in
Quantum
Computation

Quantum logic

Quantum information

Quantum computing

models and algorithms

Applications

Solving the NJL
model in 1 + 1
dimensions

Analytical solution

Lattice formulation

Fermion-qubit mapping

Space parametrization
and state preparation

Ground state energy

Conclusions and
prospective work

We can now solve for the ground state energy of our system. This can be accomplished by making use of the hybrid quantum-classical algorithm known as the **Variational Quantum Eigensolver** (VQE); which is based in the **variational theorem of quantum mechanics**:

$$\langle H \rangle(\theta^n) \equiv \langle \psi(\theta^n) | H | \psi(\theta^n) \rangle \geq \lambda_{\min}$$

Evaluating the expectation value of the different components making up our Hamiltonian is done through a process known as **operator averaging**:

$$P_N = \sum_{p=1}^{\mathcal{O}(N^q)} w_p P_N^p \quad \Rightarrow \quad \langle P_N \rangle = \sum_{p=1}^{\mathcal{O}(N^q)} w_p \langle P_N^p \rangle$$



Variational quantum eigensolver algorithm II

Introduction

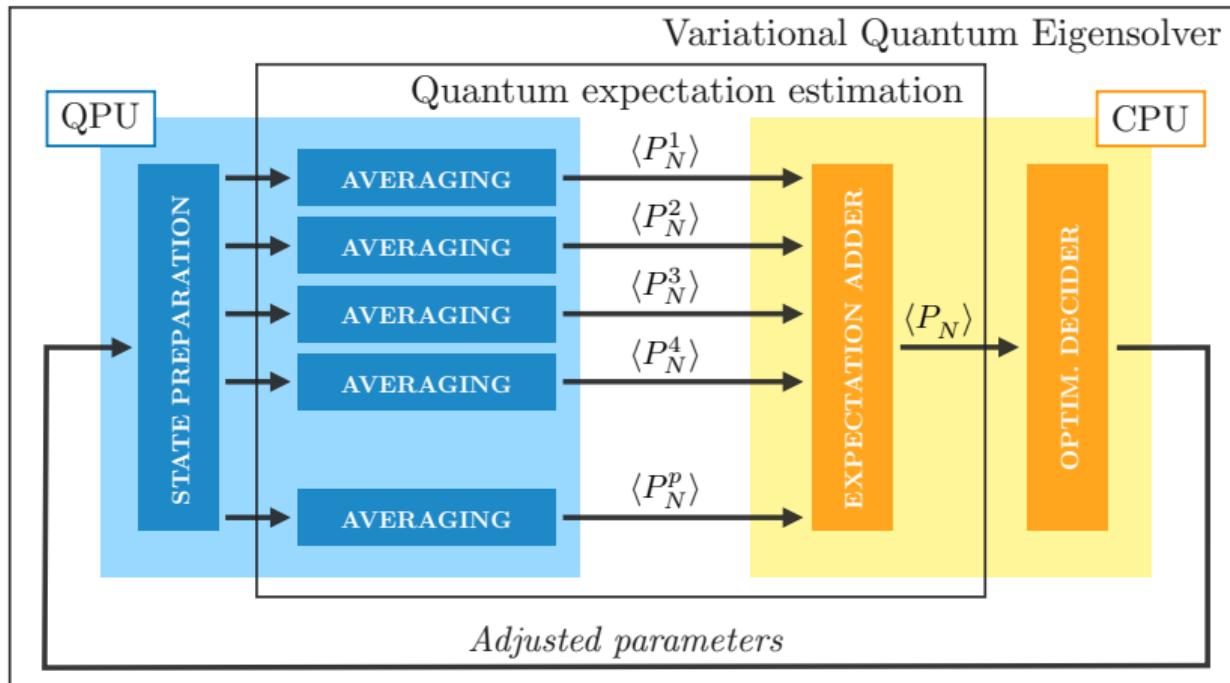
Background in
Quantum
Computation

Quantum logic
Quantum information
Quantum computing
models and algorithms
Applications

Solving the NJL
model in 1 + 1
dimensions

Analytical solution
Lattice formulation
Fermion-qubit mapping
Space parametrization
and state preparation
Ground state energy

Conclusions and
prospective work



Optimal sampling regression algorithm I

Introduction

Background in
Quantum
Computation

Quantum logic

Quantum information

Quantum computing
models and algorithms

Applications

Solving the NJL
model in 1 + 1
dimensions

Analytical solution

Lattice formulation

Fermion-qubit mapping

Space parametrization
and state preparation

Ground state energy

Conclusions and
prospective work

The method that we have used to parametrize space will naturally return cycles in the states that we are parametrizing. Such **periodic nature** will transfer to the expectation value function, which in turn allows us to consistently apply Fourier analysis to fully describe it:

$$f(\theta) \equiv a_0 + \sum_{s=1}^S [a_s \cos(s\theta) + b_s \sin(s\theta)]$$

$$\begin{bmatrix} 1 & \cos(\theta_1) & \sin(\theta_1) & \cos(2\theta_1) & \cdots & \sin(S\theta_1) \\ 1 & \cos(\theta_2) & \sin(\theta_2) & \cos(2\theta_2) & \cdots & \sin(S\theta_2) \\ \vdots & \vdots & \vdots & \vdots & \ddots & \vdots \\ 1 & \cos(\theta_{2S+1}) & \sin(\theta_{2S+1}) & \cos(2\theta_{2S+1}) & \cdots & \sin(S\theta_{2S+1}) \end{bmatrix} \begin{bmatrix} a_0 \\ a_1 \\ b_1 \\ a_2 \\ \vdots \\ b_S \end{bmatrix} = \begin{bmatrix} f(\theta_1) \\ f(\theta_2) \\ \vdots \\ f(\theta_{2S+1}) \end{bmatrix}$$

$$Fc = f \quad \rightarrow \quad F^\dagger Fc = F^\dagger f$$



Optimal sampling regression algorithm II

Introduction

Background in
Quantum
Computation

Quantum logic

Quantum information

Quantum computing

models and algorithms

Applications

Solving the NJL
model in 1 + 1
dimensions

Analytical solution

Lattice formulation

Fermion-qubit mapping

Space parametrization
and state preparation

Ground state energy

Conclusions and
prospective work

Generally $S \rightarrow \infty$, however, if the bandwidth is bounded, S will be finite and it will be possible to evaluate this expression exactly. Theoretically, the power of this method is demonstrated through the **Nyquist-Shannon sampling theorem**; which states that if a function $f(\theta)$ contains no angular frequencies higher than ω_S , it is completely determined by giving its ordinates at a series of points $1/2\omega_S$ apart:

$$\omega_{\text{sampling}} > 2\omega_S$$

Extending these results to higher dimensions is straight forward considering multidimensional Fourier series. In this case, we may have a different bandwidth S_q for each parameter. Calling the total number of parameters Q , and the maximum bandwidth S_{\max} , the total number of samples T required by this method is:

$$T = \prod_{q=1}^Q (2S_q + 1) = \mathcal{O}(S_{\max}^Q)$$



Optimal sampling regression algorithm III

Introduction

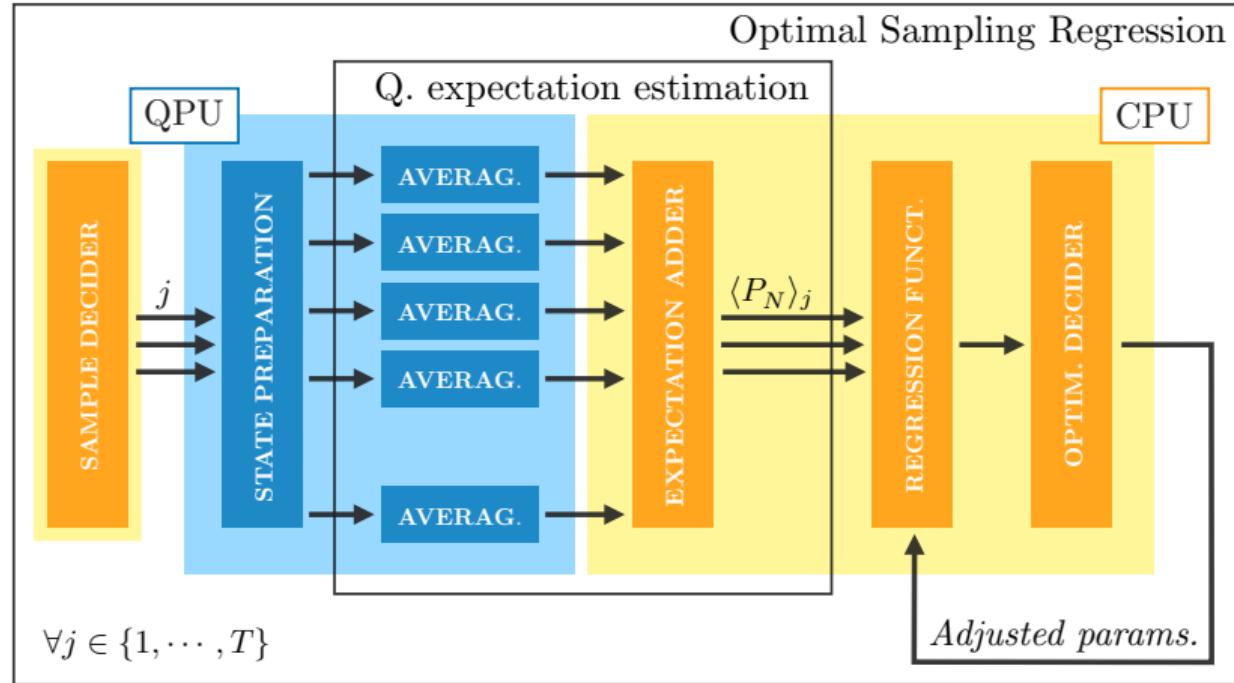
Background in
Quantum
Computation

Quantum logic
Quantum information
Quantum computing
models and algorithms
Applications

Solving the NJL
model in $1 + 1$
dimensions

Analytical solution
Lattice formulation
Fermion-qubit mapping
Space parametrization
and state preparation
Ground state energy

Conclusions and
prospective work



Optimal sampling regression algorithm IV

Introduction

Background in
Quantum
Computation

Quantum logic
Quantum information
Quantum computing
models and algorithms
Applications

Solving the NJL
model in 1 + 1
dimensions

Analytical solution
Lattice formulation
Fermion-qubit mapping
Space parametrization
and state preparation
Ground state energy

Conclusions and
prospective work

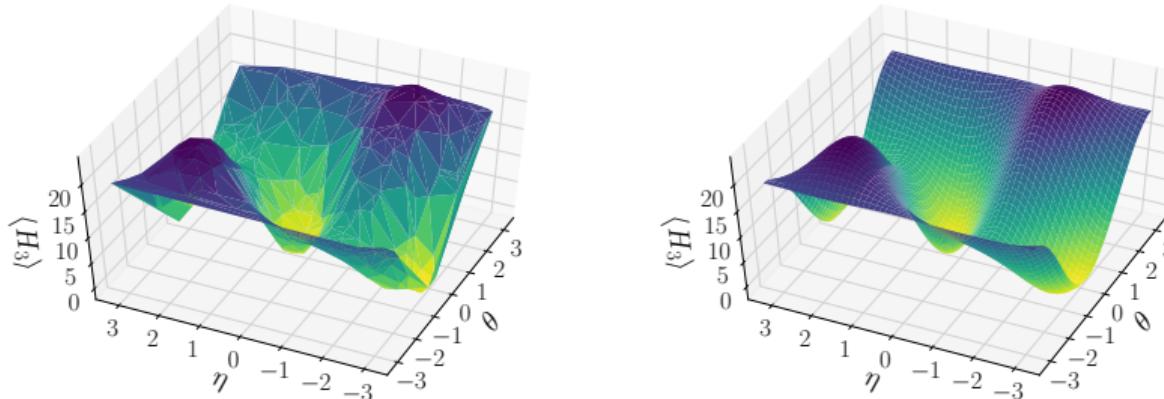


Figure: Comparison between the VQE and OSR algorithms, when reproducing an external model with two parameters. (Left) Triangulation of the expectation value function from raw samples. (Right) Approximate function obtained through the Optimal Sampling Regression method with $S_q = S_{\max} = 2 \forall q$.

N_{params}	VQE samples	OSR samples	VQE error	OSR error
1	24	3	3.5%	1.0%
2	153	25	0.3%	0.2%

Ground state energy I

Introduction

Background in
Quantum
Computation

Quantum logic

Quantum information

Quantum computing
models and algorithms

Applications

Solving the NJL
model in 1 + 1
dimensions

Analytical solution

Lattice formulation

Fermion-qubit mapping

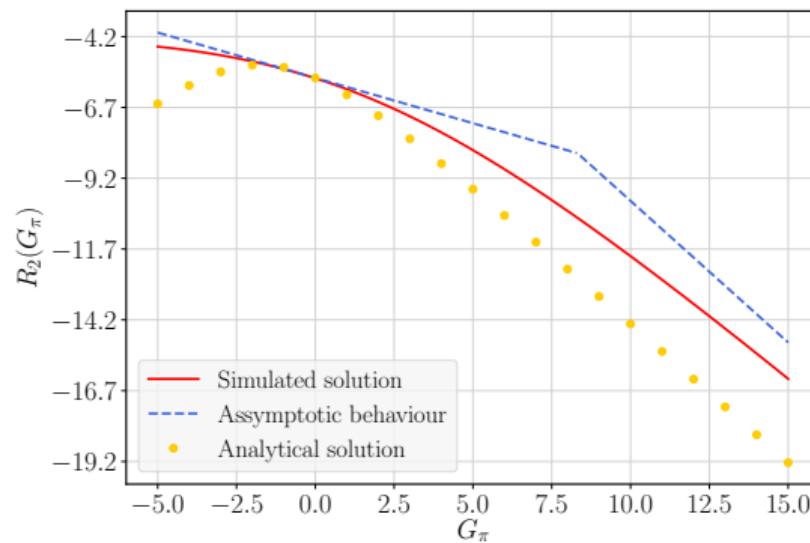
Space parametrization
and state preparation

Ground state energy

Conclusions and
prospective work



At long last, we have everything that we need to solve for the ground state energy of our system using a quantum computer. For simplicity, we will do so first through a **quantum simulator**.



Ground state energy II

Introduction

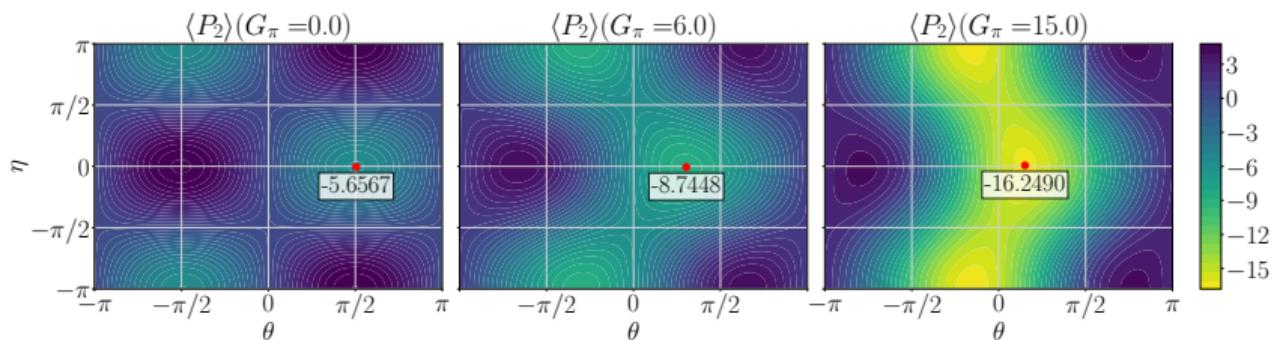
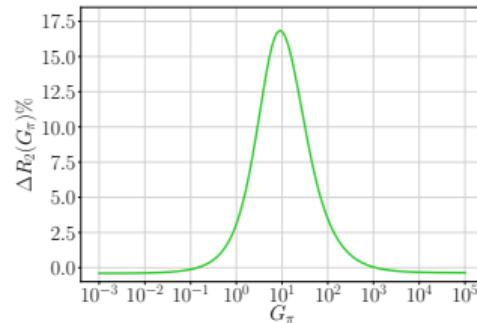
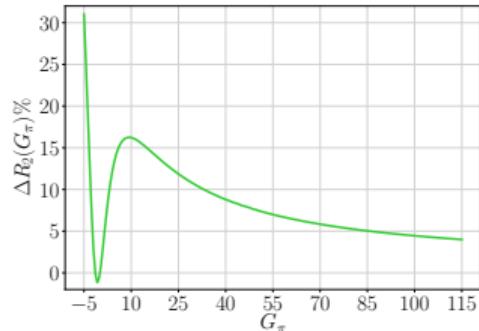
Background in
Quantum
Computation

Quantum logic
Quantum information
Quantum computing
models and algorithms
Applications

Solving the NJL
model in 1 + 1
dimensions

Analytical solution
Lattice formulation
Fermion-qubit mapping
Space parametrization
and state preparation
Ground state energy

Conclusions and
prospective work



Conlcusions

Introduction

Background in
Quantum
Computation

Quantum logic
Quantum information
Quantum computing
models and algorithms
Applications

Solving the NJL
model in 1 + 1
dimensions

Analytical solution
Lattice formulation
Fermion-qubit mapping
Space parametrization
and state preparation
Ground state energy

Conclusions and
prospective work



So far I have:

- Reproduced an instance of mass generation in QCD using the NJL model in $1 + 1$ dimensions, and found the critical value of the coupling constant at which it occurs.
- Discretized a simple version of the NJL Hamiltonian using a staggered fermion lattice.
- Mapped the resulting Hamultonian onto Pauli operators in order to work with it on a quantum processor.
- Developed a custom ansatz for parametrizing certain regions of Hilbert space according to the symmetries of our problem.
- Implemented a simplified version of said ansatz as a quantum circuit.
- Introduced a new variational algorithm to obtain the minimum eigen value of an encoded operator and tested it against the VQE algorithm.
- Used the OSR algorithm to solve for the ground state energy of the NJL Hamiltonian for differernt values of the coupling constant.

Prospective work

Introduction

Background in
Quantum
Computation

Quantum logic
Quantum information
Quantum computing
models and algorithms
Applications

Solving the NJL
model in 1 + 1
dimensions

Analytical solution
Lattice formulation
Fermion-qubit mapping
Space parametrization
and state preparation
Ground state energy

Conclusions and
prospective work



I still have not observed definite evidence of mass generation.

- Implement the full parametrization and compare them with the analytical solution.
- Perform these same calculations on actual quantum computers, analyzing any possible differences in the results.
- Use a Hamiltonian with two flavors that explicitly contains chiral symmetry, and study properties of the resulting domains (e.g. chiral condensate) such as correlation lengths.
- Build a new refactored Hamiltonian which exhibits the finite jumps for arbitrarily small changes in the coupling constant, and backwards engineer a continuum model of QCD.
- Study the NJL model in higher dimensions up to full $3 + 1$ QCD.
- Validate the techniques and results.
- Describe the scalability of these methods: how fast resource requirements grow once we start increasing the complexity of the systems we are simulating.
- Determine some of the limitations of state of the art, as well as short-term NISQ technology.
- Develop a consistent notation for these applications of QIS.

