Computer Science - Senior Capstone Project
Spring 2019
May 12, 2019

Simulating Quantum Physics with Quantum Computing

Sean Coneys, Kevin Orellana

Supervised by: *Tim Byrnes*
Mentor: *Olivier Marin*

Quantum computing promises to offer great advances in computational speed by harnessing the power of quantum superposition and entanglement. One of the most promising applications of quantum computing is in the simulation of quantum physics, which is known to be a computationally difficult problem. Using classical computers, calculating the quantum state of an N-particle system requires resources that scale exponentially with N, making the problem intractable at the level of 30 or 40 particles. However, using quantum computers one can calculate various properties of the system, such as the time dynamics with an exponential speedup. This project focuses on quantum simulations of the Transverse Ising model.

1 Introduction

One of the best ways to form an understanding of quantum computation is to definite relative to classical computation. Classical bits have two states, either 0 or 1. In quantum computation, we perform operations on quantum bits, or *qubits*. A qubit, may have the state $|\psi\rangle$, which can be expressed as a linear combination of two states in the following form assuming $|\psi\rangle$ is a pure state:

$$|\psi\rangle = \alpha\,|0\rangle + \beta\,|1\rangle \tag{1}$$

Where α and β are complex numbers and $|\alpha|^2 + |\beta|^2 = 1$. The *computational basis states* $|0\rangle$ or $|1\rangle$ may be represented by the following matrices

$$\begin{bmatrix}0\\1\end{bmatrix}, \begin{bmatrix}1\\0\end{bmatrix} \tag{2}$$

in that order.

A qubit, which itself is a quantum system, may be in the state $|0\rangle, |1\rangle$ or a combination of these states. A combination of these two states is known as a *superposition*. Because of a quantum mechanical phenomenon, scientists do not have direct access to α and β, or the *amplitudes*, of the qubits superposition (i.e the state of quantum system). Furthermore, when we perform a *quantum measurement* to deduce the state of the qubit, we may only ascertain the quantum state to be either 0 or 1. While the quantum state $|\psi\rangle$ may have been a composition of the two computational basis states $|0\rangle$ and $|1\rangle$, performing this quantum measurement causes the quantum system to undergo a wave function collapse, rendering only one of the possible basis states. The square of amplitudes alpha and beta, $|\alpha|^2$ and $|\beta|^2$, represent the probability of finding the quantum state to be $|0\rangle$ or $|1\rangle$, respectively.

One way to visualize a qubits state is as a complex vector in a multi-dimensional Hilbert space. A quantum system with N qubits has a Hilbert space represented by 2^N orthogonal states, or basis vectors. Therefore, the state of an N-qubit system can be defined by the equation:

$$|\psi_n\rangle = \sum_{j \in J} c_j\,|j\rangle \tag{3}$$

where J is a set of computational basis states and c_j is the respective amplitude of a computational basis states.

To represent a quantum system on a classical, Turing machine computer, we must store each amplitude c_j. These amplitudes are comprised of a real and imaginary component, usually represented as two floating point numbers. Because of this, to represent a quantum system of size N we need to store a minimum of 2^N 32-bit floating point numbers [1]. Therefore, when using a classical computer, to represent a quantum system of size N we need a minimum of 2^N 32-bit numbers [1]. Therefore, for each additional entity in the quantum system we intend to simulate on a classical computer, we endure exponential growth in computational costs. On a quantum computer, however, for each additional body we intend to model we need add only a single qubit. This suggest that for the simulation of quantum systems where superpositions must be accounted for, quantum computers possess an inherent advantage over classical systems.

The idea of harnessing quantum mechanics for quantum simulation was first suggested in 1982 by Richard Feynman [2]. His theory was later proved correct in 1996 by Seth Lloyd who additionally demonstrated the existence of what one might call universal quantum computers. Lloyd demonstrated that quantum computers can simulate the behavior of arbitrary quantum systems whose dynamics are determined by local interactions. [3]. This shows that we do not need to replicate the exact quantum system that we wish to observe, a process known as *analog simulation*. Through the use of a universal quantum computer, we can recreate and manipulate quantum states in an effort to simulate the quantum system, a process known as *digital simulation*. The creation of problem-specific quantum computers offers the possibility of higher computational efficiency but also presents large overhead in the engineering of such quantum computers. Because of this, our review focuses on digital simulation. There is a large variety of quantum systems capable of universal computation [4] but our project focuses on the superconducting qubit architecture implemented by IBM that is available for public use. The work done by Vartiainen et al [5]. shows that Lloyds method for the simulation of a quantum systems can be efficiently transformed so that it can be implemented on a gate model quantum computer like IBM's IBM-Q quantum computers.

2 Ising Model

This capstone project will focus on the simulation of the Ising model. The Ising model can be used to describe the ferromagnetic[1] properties of an N-particle system. Such a ferromagnetic system may be d-dimensional and represented as a lattice structure [6], as shown in Figure 1 below:

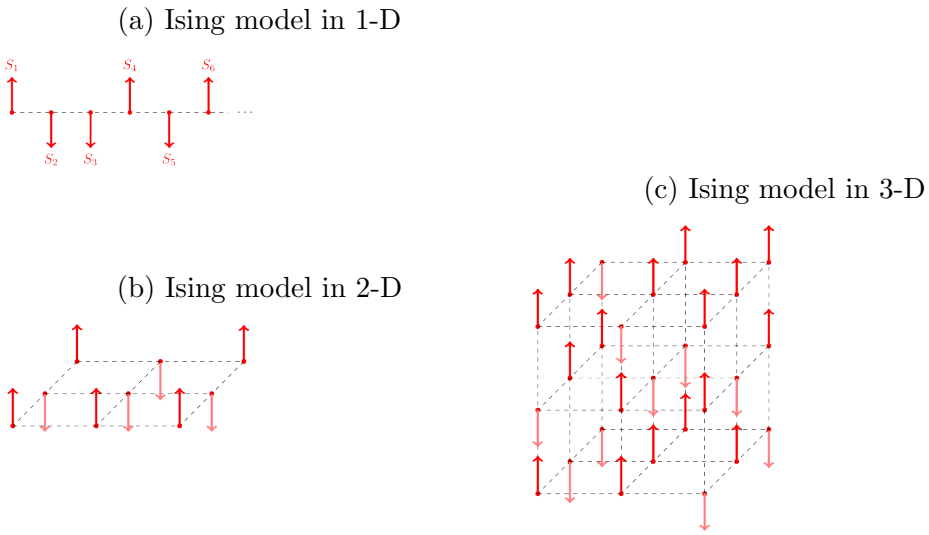


Figure 1: Visual representations of the Ising model in D-dimensions. Images from Leonardo Pacciani, 2016.

Though the Ising model was originally intended to describe the spins of atoms in a magnet, we can also use it to describe systems such as the motion of atoms in a lattice gas or even the activity of neurons in a neural networks. In this capstone project, we use the Ising model to simulate the inter-atomic interactions and energies of an N-body quantum system. This type of Ising model is also known as the Heisenberg model.

We use the Ising model's Hamiltonian H to describe the dynamics of the quantum system. The Hamiltonian H will describe the system's dynamics

[1] A body with a ferromagnetic property is defined as a body or substance that has a high susceptibility to magnetization, the strength of which depends on an applied magnetizing field.

relative to its environment, which may include a magnetic field and other magnetized neighbors.

The Hamiltonian H of a system can be expressed as a summation of its subsystems' Hamiltonians, mathematically expressed as:

$$H = \sum^{N} H_j \tag{4}$$

where j is a subsystem of the N-body quantum system and H_j is the subsystem's Hamiltonian.

An Ising model's Hamiltonian can be described by:

$$H = -\frac{J}{2} \sum_{j,j+1} \sigma_j^x \sigma_{j+1}^x - \frac{h}{2} \sum_j \sigma_j^z \tag{5}$$

where J, h are represent the inter-atomic coupling strength and magnetic field strength, respectively[7]. The Hamiltonian H is the sum of the interactions between nearest neighbors and each body's interaction with the magnetic field. It is important to note that the exchange-forces between nearest neighbors and their environment are orthogonal. We can note this from the fact that the inter-atomic exchange-forces between σ_j^x and σ_{j+1}^x are along the z-axis and the interaction between the j-th body in the quantum system with surrounding magnetic field h is along the x-axis.

3 Methods

This section aims to describe the methods used in the simulation of the Ising model. We used a classical, deterministic Turing machine and quantum computer to simulate the Ising models dynamics. The methods used to simulate these dynamics differ on classical and quantum computers and so we delineate these methods in separate sections below.

The dynamics of a quantum system can be described by the Schrödinger's equation, detailed below:

$$i\hbar \frac{\partial}{\partial t} \Psi = \hat{H} \Psi \tag{6}$$

where Ψ and $\hat{H}$ may be functions of time. The Schrödinger equation is a partial differential equation and can be written in matrix and vector form[8] as:

$$i\hbar\frac{d}{dt}|\psi\rangle = H|\psi\rangle \tag{7}$$

where $|\psi\rangle$ is the state vector and H is the matrix representation of the Hamiltonian.

In the subsections below, we explore the mechanics that govern the Ising model and how they were simulated.

3.1 Classical Numerical Analysis

Classical simulations of quantum systems can be divided into two categories: strong and weak simulations. In a strong simulation, the entire quantum systems state information is stored in memory as the system is evolved. This allows access to information that we might otherwise be unable to observe in a quantum computer simulation. A weak simulation, on the other hand, fails to fully retain the quantum system and instead relies on probabilistic methods to determine the state of the system. In particular, the quantum systems superpositions are of interest as they allow us to manipulate this information to deduce other useful information, such as the ground and excited states of the quantum system.

3.1.1 Ordinary Differential Equation Solver

The classical numerical analysis method we used in this project involves the use of an ordinary differential equation solver. We used the open-source Python toolbox Qutip [9] to simulate the dynamics of the Ising model on a classical computer. Our steps included instantiating a quantum system state vector $|\psi\rangle$, defining the Ising model Hamiltonian H and using Qutip to perform a unitary time-evolution. Qutip does this by solving the Lindblad master equation, in our case the Schrodinger equation. Ultimately, Qutip performs a unitary time-evolution for specified time value tau and returns the expectation values for such tau τ. Because Qutip is a strong simulation and is devoid of errors typically incurred by quantum computers, we consider the expectation values returned by Qutip's Schrodinger equation solver as the most accurate results in our experiment. We will expand more on how we

use these theoretical values to compare the efficiency of quantum simulations later.

3.2 Quantum Simulations

Given an initial quantum state $|\psi(0)\rangle$, we can find the unitary time evolution according to the Hamiltonian H. This time evolution can be found by solving the following equations [3]:

$$\begin{aligned} |\psi(t)\rangle &= e^{\frac{-iHt}{\hbar}} |\psi(0)\rangle \\ &= exp\left(-\frac{it}{\hbar}\sum_j H_j\right)|\psi(0)\rangle \end{aligned} \tag{8}$$

We can perform the unitary time evolution $e^{\frac{-iHt}{\hbar}}$ on a quantum computer through a series of one and two qubit gate operations. These gates can be represented as:

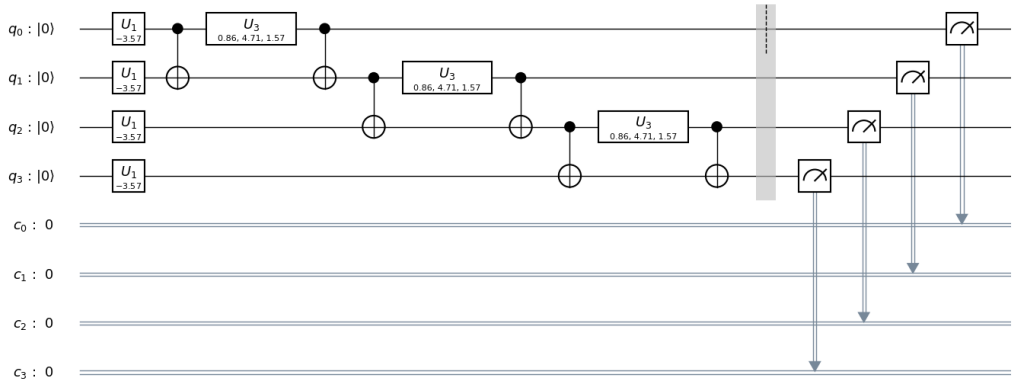
$$U_l(\gamma,\theta) = exp(\frac{-i\theta\gamma * \sigma_l}{2}) \tag{9}$$

$$U_{ll\prime}(\phi) = exp(-i\phi\sigma_l^z\sigma_{l\prime}^z) \tag{10}$$

Where γ is the unit vector specifying the axis of the single qubit gate, and θ, ϕ are angles we can freely choose.

Below is a figure of the quantum circuit used to evolve a 4-qubit quantum system. Note that U_1 and U_3 correspond to U_l and $U_{ll\prime}$, respectively. U_1 applies the gate that reflects the effects of the magnetic field while U_3 applies the coupling strength effects on neighboring qubits.

Figure 2: Transverse Ising Circuit



To decompose this unitary time evolution into a series of elementary gates, we must perform a Suzuki-Trotter expansion[10][11]. The Ising models unitary time evolution only requires a first order Suzuki-Trotter expansion, which can be written as:

$$e^{A+B} = \lim_{n\to\infty} \left(e^{\frac{A}{k}} e^{\frac{B}{k}}\right)^k \tag{11}$$

Here, k is the Suzuki-Trotter number. The unitary evolution[12] then expands to:

$$\begin{aligned} e^{\frac{-iHt}{\hbar}} &= exp\left(\frac{-iHt}{\hbar}\sum_j H_j\right) \\ &= \left[exp(\frac{-i\triangle t}{\hbar}\sum_j H_j\right]^{\frac{t}{\triangle t}} \\ &= \prod_j exp\left(\frac{-H\triangle t}{\hbar}H_j\right)^{\frac{t}{\triangle t}} \end{aligned} \tag{12}$$

This allows us to conduct a sequence of elementary gate operations[12]:

$$U_l(x, \lambda t\ \hbar) = exp(\frac{-i\lambda\triangle t}{\hbar}\sigma_l^x) \tag{13}$$

$$U_{ll\prime}(J\triangle t\ \hbar) = exp(\frac{-iJ\triangle t}{\hbar}\sigma_l^z\sigma_{ll\prime}^x) \tag{14}$$

The number of elementary gate operations that a quantum computer will perform in the decomposition of the unitary time evolution is proportional to the number of terms in the Hamiltonian, which is $2 * N$. On the other hand, a classical simulation will have to diagonalize a 2^N x 2^N matrix[12]. .

3.2.1 Adiabatic Simulations

The adiabatic evolution method is useful in both analog and digital simulations used to reach a desired ground state of a Hamiltonian H that is generally hard to construct or that satisfies a certain combinatorial search problem. The adiabatic evolution method is only useful in certain cases because it is not always efficient, e.g it cannot always be performed in less than exponential time [13]. In practice it is often hard to estimate whether a simulation will be efficient or not.

An adiabatic evolution starts with the Hamiltonian of interest in an easy-to-construct ground state H(0). A total run time τ is then chosen. Following that, H is slowly evolved using the formula for the time dependent Hamiltonian H(t) and Schrodinger's equation, for time τ resulting in H(τ). According to adiabatic theorem, if τ is sufficiently large and if the energy gap between the ground state and the first excited state of H is always non-zero then H(τ) will be very close to the desired ground state of H. [13] The magnitude of τ will vary based off the minimum size of the energy gap and the desired closeness of H(τ) to the ground state of H(τ). This energy gap in many instances of the problem is hard to estimate therefore making it hard to chose a τ or asses whether the simulation will be efficient[13].For the transverse Ising model the definition of the time dependent hamiltonian derived from the time independent definition in figure 5 is:

$$H = -\frac{J(t)}{2} \sum_{j,j+1} \sigma_j^x \sigma_{j+1}^x - \frac{h(t)}{2} \sum_j \sigma_j^z \tag{15}$$

With J(t) and h(t) being time dependent functions defined differently based off of the desired ground state. J(t) and h(t) will move slowly towards towards their final value for the desired ground state e.g. J(T) and h(T) from the initial ground state.

4 Error

Error in quantum simulations comes from a large variety of sources and is generally rather hard to quantify. Research into both error reduction and correction is extremely important and ongoing. Quantum simulation error can largely be broken down into two categories: those caused by something having to do with the physical computer itself and its interactions with its environment and those having to do with the way the simulation is conducted or approximated. For the former the main error categories are type 1 decoherence, type 2 decoherence, gate error, and readout error. These are universal and are significant sources of error in the implementations of a quantum computer. These sources of errors are however not a factor in classical simulations unless a noise, or error model, is present. On the IBM-Q architecture each of these factors is different for each qubit. Iin the case of 2-qubit gate error, each pair of connected qubits also has its own gate error. For the latter category, the induction of error in the simulation is peculiar to the implementation of the simulation.

4.1 Error on the IBM-Q

Type 1: T1 decoherence, also called energy relaxation, is the tendency of the qubit over time to move from an excited state, generally $|1\rangle$, to a lower energy state, generally $|0\rangle$, e.g it represents rotation orthogonal to the z-axis in a bloch sphere visualization [14]. T1 decoherence can be measured by finding the time it takes for the probability of getting a measurement of $|0\rangle$ to be larger than or equal to probability of obtaining a measurement of $|1\rangle$ after a qubit has been placed in the $|1\rangle$ state. e.g. if $|\psi\rangle = |1\rangle$ (here pr($|0\rangle$)=0) it is the time until pr($|0\rangle$) >= pr($|1\rangle$) is true. On the IBM-Q architecture, this is implemented through the repeated application of identity gates to simulate the passage of time. For each time step, this is repeatedly performed and measured to estimate the probability of decoherence occuring at that time [14].

Type 2: T2 and T2* decoherence are different measures of phase decoherence. Both quantify error involving rotations around the the z-axis of the Bloch sphere. Because a state of $|0\rangle$ or $|1\rangle$ has no phase this type of decoherence is only applicable to superpositions [14]. As time passes, the phase of

a qubit changes and over time the probability of a qubit having the correct phase deteriorates to .5.

Gate Error: There are two varieties of quantum gate error. The first variety is caused by the interaction of the system with its environment and does not necessarily preserve the input state (incoherent gate error). The second variety of error is induced by an imperfect transformation performed by a quantum gate (coherent gate error) [15]. Gate error is particular to each qubit and to each set of connected qubits i.e one and two qubit gates respectively. On the IBM-Q architecture any gate is implemented through the use of a U1, U2, U3, or CNOT gate. The U1 is implemented using a frame change, on the IBM-Q and is therefore instant and error less , all other gates have their own respective gate error [16].

Gate error can be quantified by using a process known as random benchmarking [17][18]. In a simplified sense random benchmarking is a series of randomly selected transformations are performed upon a qubit and/or a set of 2 qubits that in the absence of error return the qubit(s) to their original state. The average gate error is then found by repeatedly measuring the fidelity of the resulting vector with the original one using multiple iterations and circuits. The error is then generally reported as (1 - average fidelity).

Unfortunately, average gate fidelity can be a misleading metric. For example, a two qubit gate with 99% fidelity is only guaranteed to have an error rate below 45% [19]. In general average gate fidelity is not thought to be a sufficient measurement for the evaluation fault tolerance in quantum computing [19]. This is because gate error is generally reported as an average, while this is a useful metric for determining general performance it fails to show us the behavior of outliers. The behavior of outliers is important because they are crucial to determining if a quantum simulation satisfies the threshold theorem. According to the threshold theorem if error can be held below a certain given threshold then quantum error correction can be done with polylogarithmic scaling [20]. Diamond distance and Unitaries are other metrics that can help asses gate error but alone none of them are sufficient to fully describe the behavior of gate errors [19].

Readout Error: Readout error is error that is induced by getting the wrong result from a measurement (e.g should be $ket0$ but $ket1$ is measured instead or vice versa). Essentially it can be thought of as the quantum version of a bit flip error. This can have one of two causes, first it can be caused by impercision particular to the instruments used for measuring. The second is that because measurement is not instantaneous there is a chance that due to t1 decoherence a qubit can decohere during measurement [15].

Error Correction: Quantum error correction in the best case scales poly-log with regards to N. This scaling however can only be assumed if a certain threshold for error is met [20]. Therefore the error correction cost for the transverse Ising model is at best PolyLog(N) but likely higher.

4.2 Error induced by Suzuki-Trotter Expansion

The Trotter-Suzuki decomposition of unitary time evolution into elementary gates is a useful in digital quantum simulation. It is an estimation, whose accuracy depends on the Trotter-Suzuki number k. We can observe this from the first-order Suzuki-Trotter expansion [10][11]:

$$e^{A+B} = \lim_{k\to\infty} \left(e^{\frac{A}{k}} e^{\frac{B}{k}}\right)^k \tag{16}$$

We note from the expansion that the left-hand side is equal to the right-hand side only as k approaches infinity. Because we are limited to finite numbers in our quantum simulations, we will surely incur errors.

The error of operation by performing the Suzuki-Trotter decomposition for one of the time steps delta t scales as $O((\triangle t)^2)$ [12]. For an unitary time evolution, the error induced by Suzuki-Trotter is $O(t\triangle t)$.

4.3 Adiabatic Error

In the adiabatic time evolution, our calculated errors are largely related to the size of time τ. If the energy gap between the ground state and the first excited state in the Hamiltonian of interest in a adiabatic evolution is always non-zero, as τ approaches infinity, H(τ) approaches the ground state of H(τ) and H(τ) can be brought arbitrarily close to the ground state of H(τ). The

size of τ required to bring $H(\tau)$ arbitrarily close to the ground state of $H(\tau)$ will vary based off of the minimum size of the energy gap and the desired closeness of $H(\tau)$ to the ground state of $H(\tau)$ [13].

In an analogue simulation there is error related to the specific system and how it is measured and its interaction with its environment. However the only error introduced by the adiabatic method is that between $H(\tau)$ and the ground state of $H(\tau)$.

For digital simulations involving adiabatic evolutions error is induce both by the adiabatic method and by the Trotter Suzuki expansion required to convert the evolution into unitary gates [13]. Error specific to the quantum computer it is performed on applies as well.

5 Results

In this section we demonstrate our findings gathered from simulating the Ising model and analyzing its dynamics.

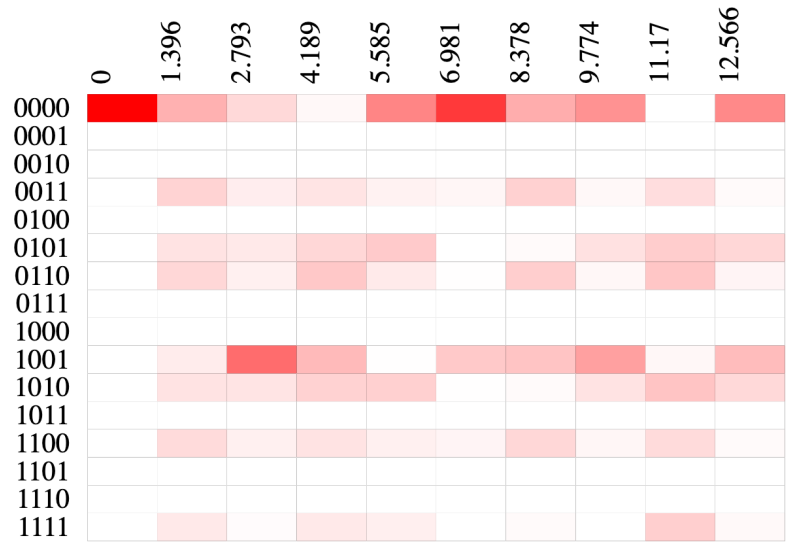
5.1 Time-evolving the Ising Model with a time-independent Hamiltonian

We first performed time evolutions of the quantum system with a time-independent Hamiltonian. An Ising model system's Hamiltonian has variables that may change over time, namely the strength of the magnetic field h and the coupling factor between atoms J. A time-independent Hamiltonian suggests that the values of h and J are fixed and do not vary over time. However, the configuration of the system may change over time.

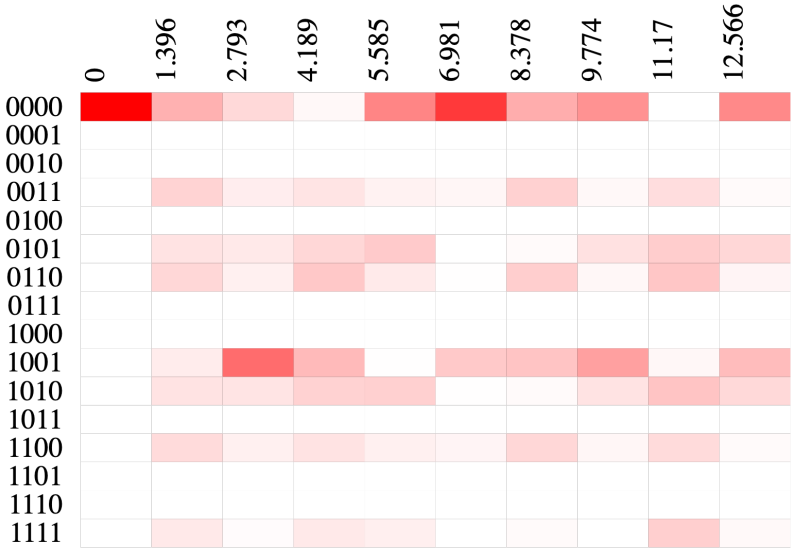
5.1.1 Flipping the sign of J

In the figures below, we depict the evolution of a 4 qubit system over time $\tau = 4\pi$ on QASM. Our simulation conditions were: transverse magnetic field $h = 0.5$, coupling factor $J = -1$ or $J = 1$. We attempted to visualize the evolution of the quantum system by capturing the state of the system from $t = 0$ to $t = \tau$ by splitting τ into $res = 10$ intervals. We then simulated the

evolution of the system for each of these intervals and recorded the probabilities of each state configuration at that $t = \tau_j$ where j is the interval from 0 to res.



(a) QASM, $N = 4, h = 0.5, J = 1, \tau_{final} = 4\pi, res = 10$.



(b) QASM, $N = 4, h = 0.5, J = -1, \tau_{final} = 4\pi, res = 10$.

Figure 3: Matrix representing the probabilities of state configurations over time τ. The rows of the matrix represent the possible state configurations and the columns represent the τ_{step} for which the system was evolved to. The opacity of the matrix's elements represent the probability that the system is in that state with the darkest gradient representing a probability of 1.

Note from the figure above that at $\tau = 0$ the probability of the state having the configuration $|0000\rangle$ is 1. This is as expected as, at the time of the initial quantum system state prepration, we initialize all qubits to 0.

5.2 Comparing QASM and IBMQ Results

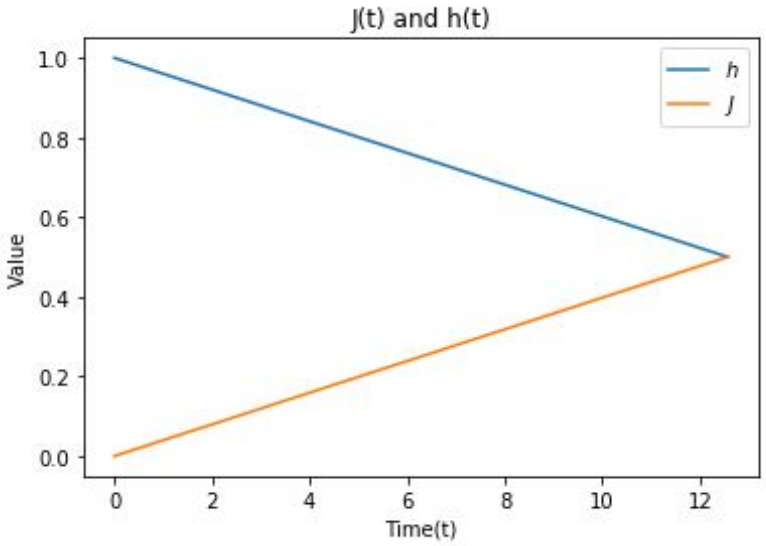
We ran a simulation against the IBM-Q quantum computer simulator *QASM* with a set of fixed parameters for h and J to compare the probabilities for each possible state. The two figures below show matrices that display the probabilities of each possible state configuration over a time evolution.

Figure 4: IBM Q Quantum computer, $N = 4, h = 0.5, J = -1, \tau_{final} = 4\pi, res = 10$.

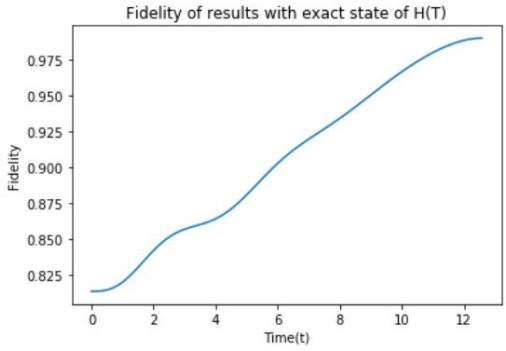
Figure 5: A matrix representing the probabilities of states of the quantum system of τ. We used the the same parameters as those of Figure 3 with J=1.

5.3 Adiabatic Simulation

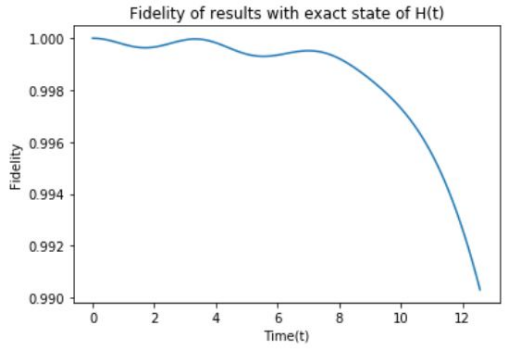
The first step we perform in this simulation is defining the desired parameters for the final ground state. In this experiment we want a ground state defined by $J(\tau) = 0.5$ and $h(\tau) = 0.5$. We select a τ of 4π but since finding the energy gap is essentially the same as solving the simulation, this τ is simply an educated guess as to what will be sufficient. We start with an easier to define ground state H(0) where $J(0) = 0$ and $h(0) = 1$ and slowly move over time towards H(τ). The graph below demonstarates this slow change by showing the relationship of J and h with t.



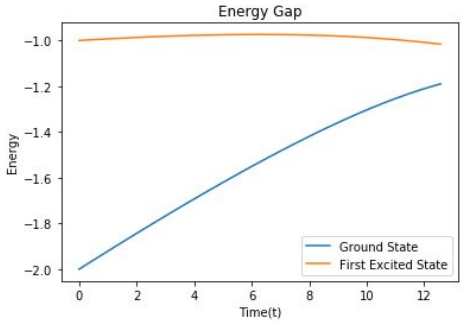
We then use Qutip to calculate what the exact ground state of H(t) would be for all times to compare to our results from the simulation for each time. The next graph shows the fidelity of the results we measure at time t with the exact ground state of H(τ). As the evolution progresses we move closer and closer to the correct ground state and when we arrive at H(τ) the fidelity of our state with the correct state is over 99% indicating that our τ is sufficiently large for our purposes.



The graph below then then shows the fidelity of our results at time t with that of the exact ground state of H(t) for all t. Interestingly, the instantaneous fidelity decreases over time. However, this is largely unimportant as the goal is to find the correct ground state of H(τ) not H(t) for any t.



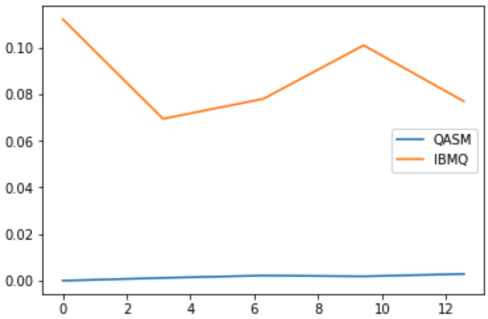
The last graph then shows the energy gap between the ground state and the first excited state for each time t. Here the gap is non-zero for all t, although it appears as though it may converge if we used a larger τ this is not the case. The energy gap is defined by the time dependent hamiltonion and therefore by J and h, not τ.



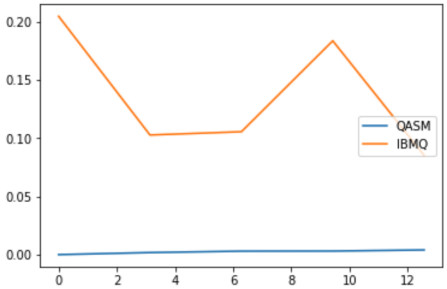
5.4 Analyzing Difference between Classical Numerical Analysis and IBM Q Simulation

We also analyzed the degree of accuracy that quantum computer algorithms provided. The results from the classical, numerical analysis acted as the baseline, as we saw it as the most accurate result of the Ising model simulations.

We define our degree of error as the absolute difference between the classical, numerical results and the quantum computer experiment's expected values. For example, if the probabilities for the state $|01\rangle$ at $\tau = \pi$ given by the Qutip and IBM-Q methods are .10 and .07, respectively, then the absolute difference for the two results is .03. We then calculated the average and standard deviation for the all the differences at a time interval τ_{step} and finally graphed these values over time. The graphs below the average and standard absolute difference for the IBM-Q simulation QASM and the IBM-Q Quantum Computer ibmqx2 *Yorktown*.



(a) Average Absolute Error between Qutip results, and results from IBM-Q QASM and IBM-Q quantum computer ibmqx2 Yorktown



(b) Standard Absolute Error between Qutip results, and results from IBM-Q QASM and IBM-Q quantum computer ibmqx2 Yorktown

Figure 6

We observe a significantly larger average and standard absolute error the quantum computer's results. We can attribute the quantum computer's larger degree of errors incurred to errors caused $T1$, T_2 decoherence, noise, and other types of error. We hypothesize that the QASM error trends are smaller because they do not experience the errors observed on actual computers.

Here we analyze the error observed on the QASM simulator caused by the Suzuki-Trotter decomposition. We first fixed N, h, J, τ and res and varied k from 1 to 100 and then calculated the averages and standard deviation of the absolute errors observed. The graph below shows our results:

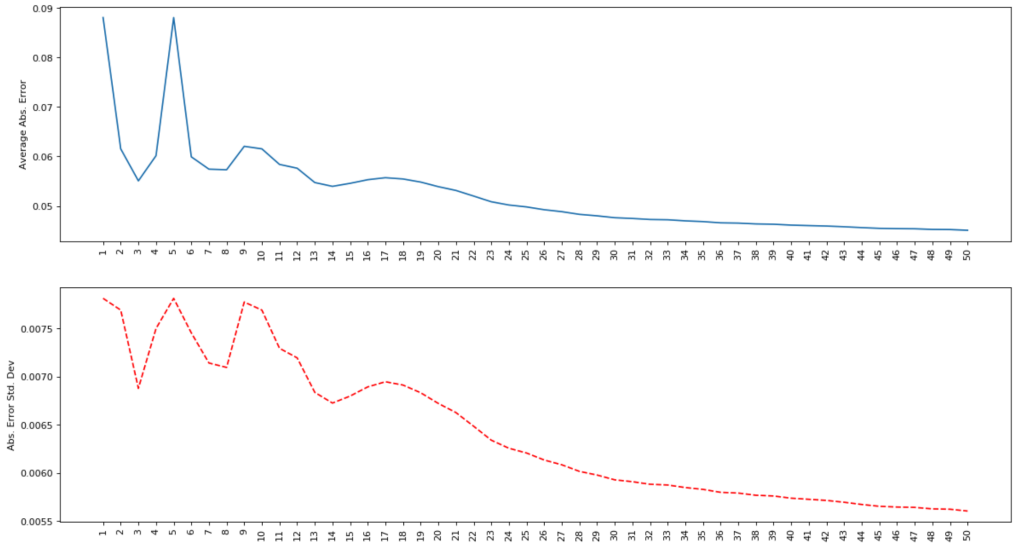


Figure 7: Varying Suzuki-Trotter number k to observe the error caused by Suzuki-Trotter expansion.

6 Conclusion

Our work provides a framework for creating and understanding one-dimensional transverse Ising model simulations. We show how both classical and digital quantum simulations of this model can be performed using time-independent Hamiltonians. The adiabatic section shows how difficult-to-prepare ground states can be prepared for these types of simulations. As the size of experiments we can simulate on quantum computers will soon exceed those whose ground states we can calculate exactly using classical methods, this will become an important step in future simulations. We additionally provide a framework for understanding the error present in these simulations, and where it comes from, in order to provide a better understanding of simulation accuracy and parameter selection. The low error rates these quantum simulations exhibit when compared to their exact classical counterparts shows that these simulations can be done with reasonable accuracy, providing evidence that as we begin to scale to sizes of N that we can no longer simulate classically we can still assume reasonable accuracy.

We found that for the Suzuki-Trotter decomposition, there exists a k for a time-evolution such that the levels of error appear to be negligible for greater values of k. For example, in 6 qubit simulation where $k = 50$ and $\tau = 4\pi$, the average absolute error trendline observed in the lower line graph of Figure 7, increasing k will not change the observed error very much.

We also demonstrated that we can observe a significant difference in levels of error when comparing a quantum simulation on the quantum computer emulator *QASM* and an actual quantum computer, the IBM-Q (this can be observed in Figure 6).

7 References

1. Brown, K., Munro, W. and Kendon, V. (2010). Using Quantum Computers for Quantum Simulation. Entropy, 12(11), pp.2268-2307.

2. Feynman, R.P. Simulating Physics with Computers.Int. J. Theoret. Phys.1982,21, 467488.

3. Lloyd, S. Universal Quantum Simulators.Science 1996,273, 10731078.

4. Nielsen, M. and Chuang, I. (2018). Quantum computation and quantum information. Cambridge: Cambridge University Press.

5. Vartiainen, J.J.; Mottonen, M.; Salomaa, M.M. Efficient Decomposition of Quantum Gates.Phys.Rev. Lett.2004,92, 177902.

6. Onsager, L. (1944). Crystal Statistics. I. A Two-Dimensional Model with an Order-Disorder Transition. Physical Review, 65(3-4), pp.117-149.

7. Suzuki, S., Inoue, J. and Chakrabarti, B. (2013). Quantum ising phases and transitions in transverse ising models. Berlin: Springer.

8. M. A. Nielsen and I. L. Chuang, Quantum computation and quantum information. Royaume-uni: Cambridge University Press, 2016.

9. Users Guide, Users Guide - QuTiP 3.1.0 Documentation. [Online]. Available: http://qutip.org/docs/3.1.0/guide/guide.html. [Accessed: 12-May-2019].

10. Suzuki, M. Improved Trotter-like Formula.Phys. Lett. A1993,180, 232234.

11. Trotter, H.F. On the Product of Semi-Groups of Operators.Proc. Am. Math. Phys.1959,10, 545.

12. T. Byrnes, Quantum information with atom optics. Unpublished.

13. E. Farhi, S. Gutmann, J. Goldstone, and M. Sipser, Quantum Computation by Adiabatic Evolution, MIT CTP 2936

14. Decoherence, Decoherence - Experience Documentation 2.0 documentation. Online. Available: https://quantumexperience.ng.bluemix.net/proxy/tutorial/full-user-guide/002-TheWeirdandWonderfulWorldoftheQubit/006-Decoherence.html. Accessed: 12-May-2019.

15. Noise and Quantum Computation, Noise and Quantum Computation - pyQuil 2.7.1 documentation. [Online]. Available: http://docs.rigetti.com/en/stable/noise.htm [Accessed: 12-May-2019].

16. Advanced Single-Qubit Gates, Advanced Single-Qubit Gates - Experience Documentation 2.0 documentation. [Online]. Available: https://quantumexperience.ng.b user-guide/002-TheWeirdandWonderfulWorldoftheQubit/004-advancedqubitgates.html. [Accessed: 12-May-2019].

17. J. Emerson, R. Alicki, and K. yczkowski, J. Opt. B: Quantum Semiclassical Opt. 7, S347 (2005).

18. E. Magesan, J. M. Gambetta, and J. Emerson, Phys. Rev. A 85, 042311 (2012).

19. Y. R Sanders et al 2016 New J. Phys. 18 012002

20. D. Aharonov and M. Ben-Or, Fault-Tolerant Quantum Computation with Constant Error Rate, SIAM Journal on Computing, vol. 38, no. 4, pp. 12071282, 2008.

21. Devitt, S.J.; Nemoto, K.; Munro, W.J. The Idiots Guide to Quantum Error Correction, 2009.arXiv:0905.2794. arXiv.org e-Print archive. Available online: http://arxiv.org/abs/0905.2794.