Fall End of Term Progess Report

Peter Karalekas

Thesis Advisor: Robert Schoelkopf

1 Overview

As stated in my thesis proposal, I set for myself a few goals for the fall semester. To begin, the overall theme of my research is to investigate simple quantum systems by implementing numerical simulations. More specifically, I wanted to learn more about how to properly set up a simulated quantum system and explore ways in which we could manipulate it. In relation to setting up a quantum system, I explored the effects of manipulating a multi-level system, which I then investigated further for the first half of the semester. For the second half, I set out to gain a better understanding of certain errors that can occur in a physical system by recreating them in simulation, specifically using a scheme to diagnose errors called randomized benchmarking.

In line with these goals, my first task was to get a handle on the tools necessary to perform quantum simulations. Using the QuTiP simulation and pulse sequence libraries developed by members of the lab, I learned how to set up a single-qubit simulation environment and introduce time-dependent driving terms. To set up the system, we must first define its starting Hamiltonian. The Hamiltonian will define the evolution of the quantum system. For a two-level system this equation is represented by 2x2 operators called Pauli matrices. However, for physical quantum systems like superconducting qubits, a two-level system is only an approximation of reality. This is one of the major difficulties with determining a physical qubit's response to a time-dependent drive—although we want to represent physical qubits as merely a ground and excited state, we are in fact dealing with a multi-level quantum system. One way to combat this problem is by numerically simulating an approximation of the physical Hamiltonian. A superconducting qubit is well approximated by an anharmonic quantum oscillator. By altering the energy levels of the various photon number states, we can create a set of non-uniform energy differences between the states. These non-uniform differences in effect decrease the probability that a driving pulse will result in an undesirable photon number state (i.e. one that is higher than the first excited state).

After understanding how to prepare the quantum system, our next goal was to investigate the possible operations we could perform on the system. When dealing with qubits, we typically use what is called the Bloch sphere to visualize its state. The Bloch sphere is a three-dimensional representation of all the possible superposition states of a qubit, with the north and south poles representing the excited and ground states of the system. The vector from the center of the sphere to a point on the surface represents the current state of the qubit, and being able to correctly manipulate this state is a critical step in building a system capable of performing quantum computations. To manipulate a physical qubit, one can alter its state by applying pulses that act as rotations around the Bloch sphere. The lab uses microwaves to perform these pulses on the qubit, where a π pulse rotates the state 180 degrees around the Bloch sphere, and a $\pi/2$ pulse rotates the state 90 degrees. However, because quantum systems are extremely sensitive to errors, these pulses do not always result in the desired state. Thus, for the second half of the semester I began to investigate these errors.

2 Anharmonicities

For the first part of my research, I set out to develop a better understanding of how to introduce in simulation the anharmonicity described above, and also how their presence effects the final state of a qubit after a pulse sequence has been applied. In addition to the starting Hamiltonian, the entire Hamiltonian

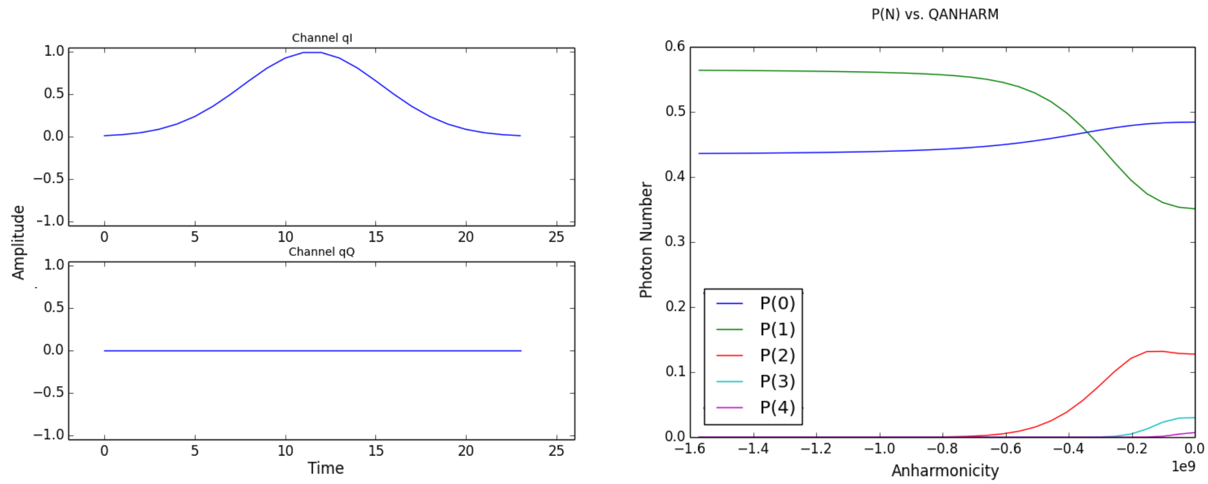

Figure 1: Gaussian driving pulse (left) and photon number distribution as a function of anharmonicity (right). For the pulse, Channel qI shows x amplitude and Channel qQ shows y amplitude. Notice that as the anharmonicity term approaches zero the drive creates photons that are distributed across all levels of the multi-level system.

for our single-qubit five-level system includes an additional term, H_{drive}, which represents the driving pulse used to apply qubit rotations to the system. In this case, we have chosen to use a Gaussian with a standard deviation of 4 nanoseconds for the driving pulse. Considering both terms, our Hamiltonian becomes:

$$H = H_0 + H_{\text{drive}}$$

And, expanding out the H_0 and the H_{drive} terms gives us:

$$H = \frac{\alpha}{2}(a^\dagger)^2 a^2 + \frac{\varepsilon_I(t)}{\sqrt{2}}(a + a^\dagger)$$

Looking at the above formula, α represents the amplitude of the anharmonicity we are introducing into the system, and ε is the amplitude envelope for the driving pulse (seen in Figure 1). In addition, the a and $a^\dagger$ terms represent the creation and annihilation operators, respectively. In order to gain insight into how the anharmonicity term affects the quantum system, I developed a simulation that plots the resulting photon state distributions of a variety of systems with different anharmonicities. These anharmonicities range from zero (meaning that the energy level differences are all uniform) to -$2\pi * 250 * 10^6$ Hz (a typical anharmonicity in a physical system). The resulting plot can be seen above in Figure 1. At our desired value, the distribution is equivalent to that of a two-level system (seen in Figure 2). This agrees with intuition because the anharmonicity acts as a barrier between the system and the higher level energy states. At zero, the photon distribution is spread across the different energy levels, behaving much like a typical anharmonic oscillator (also seen in Figure 2). Thus, we see that decreasing the amplitude of the anharmonicity from 0 has a drastic effect on the photon number distribution, and can allow use to use a qubit more effectively.

3 Randomized Benchmarking

For the second part of my research, I began to investigate the errors associated with performing qubit rotations. To do this, we decided to simulate a technique known as randomized benchmarking, a process in which a known but random series of pulses of various lengths (called a pulse train) is applied to a physical or virtual qubit (virtual in the case of our simulation), and the final state of the qubit is compared to the state we would expect if no errors had occurred. To determine this ideal state, we model the qubit as a two-level system, where its state is a two-dimensional vector and the pulses are 2x2 matricies. The final state can

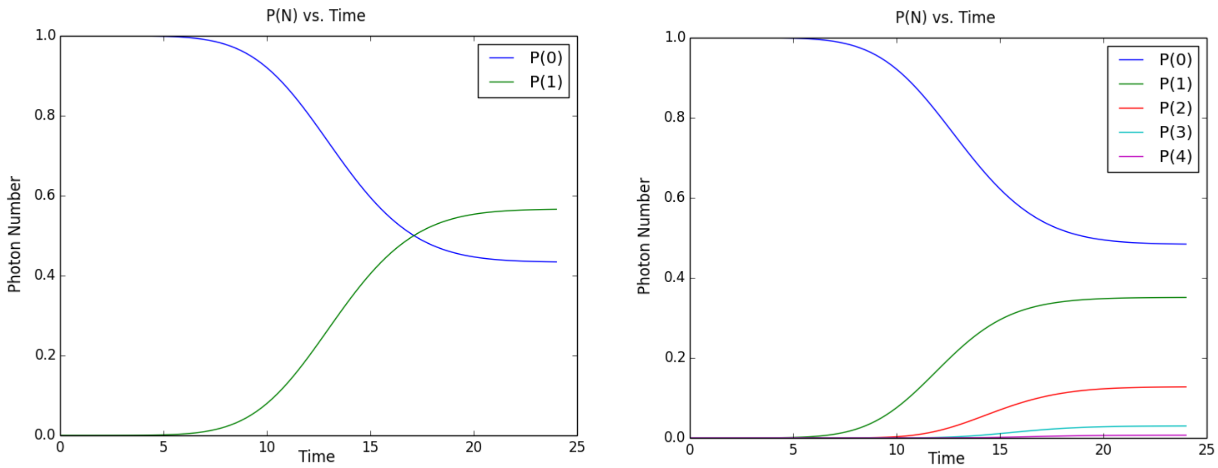

Figure 2: Photon number state distributions as a function of the length of the driving pulse for both a two-level system (left) and a five-level anharmonic oscillator (right).

then be determined simply by performing matrix multiplication on the initial state of the qubit. Once the actual and ideal final states are determined, they can be compared to determine the fidelity associated with a certain pulse train length. When repeated multiple times with various pulse train lengths, we can then analyze how the fidelity of an operation varies as the sequence of pulses grows longer and longer.

To begin, we need to understand the details of how our two-level system and the various pulses can be represented mathematically. As mentioned before, the state of the system is represented by a two-dimensional vector. The basis for our system includes the following two states:

$$\text{Ground state} = |0\rangle = \begin{pmatrix} 1 \\ 0 \end{pmatrix} \qquad \text{Excited state} = |1\rangle = \begin{pmatrix} 0 \\ 1 \end{pmatrix}$$

Starting in the ground state, we randomly select pairs of pulses to form the pulse train, which altogether is represented as $\prod_i P_i C_i$, where P_i is a Pauli matrix (a π pulse) with $i = \sigma_x, \sigma_y, \sigma_z, \mathbb{1}$ and C_i is a Clifford group generator (a $\pi/2$ pulse) with $i = \pm x, \pm y$. These matricies are as follows:

$$P_{\mathbb{1}} = \mathbb{1} = \begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix} \qquad P_x = \sigma_x = \begin{pmatrix} 0 & 1 \\ 1 & 0 \end{pmatrix} \qquad P_y = \sigma_y = \begin{pmatrix} 0 & -i \\ i & 0 \end{pmatrix} \qquad P_z = \sigma_z = \begin{pmatrix} 1 & 0 \\ 0 & -1 \end{pmatrix}$$

$$C_{\pm x} = (\mathbb{1} \mp i\sigma_x)/\sqrt{2} = \frac{1}{\sqrt{2}} \begin{pmatrix} 1 & \mp i \\ \mp i & 1 \end{pmatrix} \qquad C_{\pm y} = (\mathbb{1} \mp i\sigma_y)/\sqrt{2} = \frac{1}{\sqrt{2}} \begin{pmatrix} 1 & \mp 1 \\ \pm 1 & 1 \end{pmatrix}$$

Now that we have our pulse train, we can determine our expected state by applying the pulse train to the ground state, which can be represented as $|\psi\rangle = \prod_i P_i C_i |0\rangle$. With this expected state in mind, we can then find the fidelity of our series of processes by taking the square of the inner product of the expected and actual final states. In quantum mechanics, fidelity is a metric that denotes the distance between two quantum states. A value of 1 means that two states are identical while a value of 0 means the two states are orthogonal to each other. Calling the actual state $|\phi\rangle$, this means that the fidelity of our pulse train is $|\langle \phi | \psi \rangle|^2$.

With an understanding of the mathematics behind qubit rotations, I then set out to develop a simulation to analyze the fidelity of randomized benchmarking experiments of various lengths. First, I developed a dictionary of the possible matricies and pulses, from which the simulation would randomly select pairs of

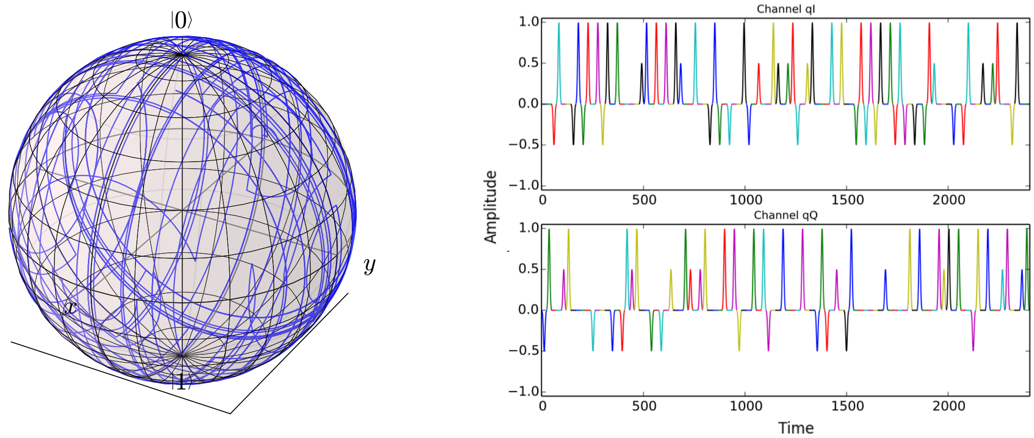

Figure 3: Sample Bloch sphere (left) showing the effect of a pulse train (right) on a qubit in the ground state. The blue lines on the Bloch sphere represent the trajectory of the state over the course of the operations. For the pulse train, Channel qI shows x pulses and Channel qQ shows y pulses. Looking at the amplitudes of these pulses, the larger amplitudes represent π pulses and the smaller amplitudes represent $\pi/2$ pulses.

Pauli and Clifford pulses. Using an imperfectly tuned pulse (to more easily observe its behavior) with shape and size similar to that from the previous section, I then linked up the simulation's sequencer with the randomized pulse train, and could then find the evolution of its state over the course of the sequence. A sample pulse train and state evolution can be seen in Figure 3. Comparing these states to the ideal states produced after each matrix multiplication, I could then find how the fidelity of the process changed as the pulse train grew in length. One can see the effect of running this simulation 50 times with a max pulse train length of 100 in Figure 4. These plots agree with intuition because the fidelity clearly decreases monotonically as the length of the pulse train increases, but the fact that the decrease is not linear warrants further study (and could just be a result of too few runs). In addition, for that particular set of simulations, the average fidelity of the final state was 0.64774966.

4 Spring Plans

Now, having implemented a working simulation of randomized benchmarking, the next immediate step is to make my simulation code both object-oriented and modular, and by doing so turn it into an importable library so that the lab can easily use it when performing randomized benchmarking. This requires separating the program into different functions to accomplish its various tasks, and building a randomized benchmarking simulation object to encapsulate it all. In addition to fine-tuning my randomized benchmarking simulation, there are a couple more areas I plan to investigate in the spring. First off is the idea of pulse tuning. For this simulation I estimated the parameters of the Gaussian pulses for the sake of time, but in actuality the pulses should undergo a process in which a number of tests are run to tune the parameters to their desired values. Thus, this pulse tuning will need to be added to my simulation to increase its level of accuracy. Last, now that I have begun to get a handle on the ways that errors manifest themselves during qubit rotations, the final step would be to figure out a way to fix those errors. Thus, I want to gain a better understanding of the current methods for correcting these errors, and hopefully with that knowledge develop a simulation where I can test those methods.

5 References

Superconducting Circuits for Quantum Information: An Outlook
M. H. Devoret and R. J. Schoelkopf. *Science* **339** 1169-1174 (2013)

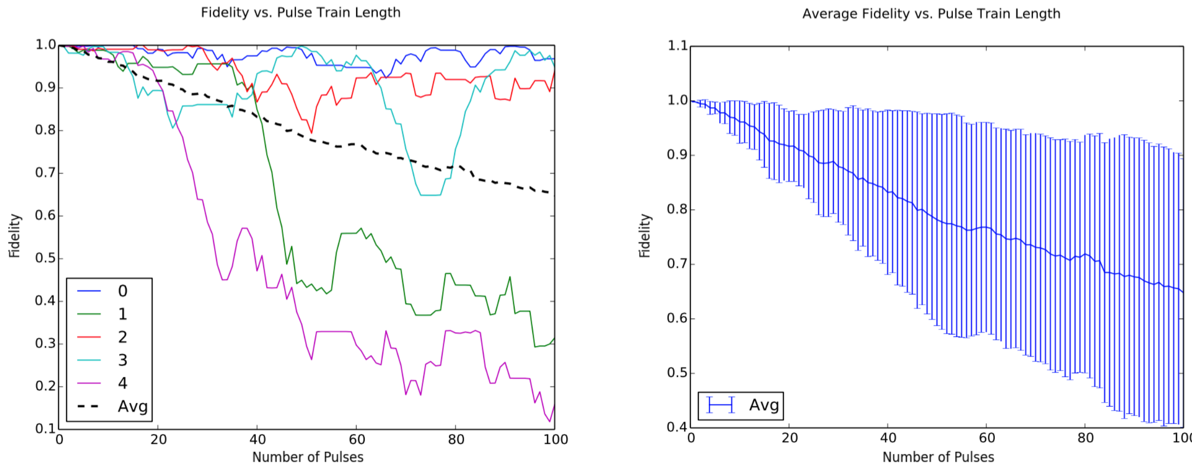

Figure 4: Sample set of fidelity traces as a function of pulse train length (left) and the average fidelity across all the 50 traces as a function of pulse train length (right). The average fidelity plot has error bars that represent $\pm\sigma$.

Randomized Benchmarking and Process Tomography for Gate Errors in a Solid-State Qubit
J. M. Chow, J. M. Gambetta, L. Tornberg, Jens Koch, Lev S. Bishop, A. A. Houck, B. R. Johnson, L. Frunzio, S. M. Girvin, and R. J. Schoelkopf. *Phys. Rev. Lett.* **102**, 090502 (2009)

Circuit QED and engineering charge based superconducting qubits
S. M. Girvin, M. H. Devoret, and R. J. Schoelkopf. *Phys. Scr. T* **137**, 014012 (2009)

Quantum Information Processing with Circuit Quantum Electrodynamics
A. Blais, J. Gambetta, A. Wallraff, D. I. Schuster, S. M. Girvin, M. H. Devoret, and R. J. Schoelkopf. *Phys Rev A* **75**, 032329 (2007)