Academic Year 2022 – 2023
Research Unit: UCLA - MQST
Team(s): Stephen, Ryan, and Simon
Supervisor(s): Richard Ross

Lab 3: Two Qubit Gates Using NMR

Ryan Dougherty

March 22, 2023

Abstract

In this lab we built off of the concepts from the second lab to conduct 4 additional experiments to explore constructing quantum gates using 1H and 13C nuclei spins. Quantum gates in NMR can be realized with a set of pulse experiments conducted with pulses along the x and y axes. We also take advantage of the fact that our two spins are coupled by a local magnetic field, which will allow us to observe entanglement of the two spin states when we allow them to freely precess a long the z axes. Our first experiment consisted of building an approximate CNOT gate that is a phase off of the traditional CNOT. This was built with a set of 2 pulses and a free precession. We also explored the concept of exhaustive averaging in order to negate noise the other spin states constitute in our experiments. To do this, we performed a second experiment which built out the 3 possible pseudo state matrices. Lastly, we combined the theory and experimental procedure of the first two to perform a true CNOT gate with no phase error. This was built with 4 pulses and a free precession. We were only able to generate results for the 13C spins since our NMR system had issues with doing 1H readouts. We go over the theory of how we'd expect to see the 1H spins behave in the same experiments and compare our results to the 13C spins. We also discuss the results we obtained and how they compare to the theory.

Contents

1 Introduction

1.1 Spin Dynamics

As mentioned in our previous lab report, we know that there exists a spin coupling between our 1H and 13C nuclei. This coupling is due to the fact that the 1H nucleus is in a magnetic field that is created by the 13C nucleus. This coupling is known as the J-Coupling; we can see its presence in our system's Hamiltonian (**?**, Knill et al, 1998):

$$H_{\text{sys}} = \frac{\hbar\omega_1}{2}\sigma_z \otimes \mathbb{I} + \frac{\hbar\omega_1}{2}\mathbb{I} \otimes \sigma_z + \frac{\pi\hbar^2 J_{1,2}}{4}\sigma_z \otimes \sigma_z$$

The J-Coupling causes a $\sigma_z \otimes \sigma_z$ interaction between our two spins. This interaction is what allows us to create two qubit gates in our system. Importantly, it also causes a slight shift in our spin energy states given by the following table (**?**, Keeler, Ch 2.6):

number	spin state	energy
1	$\alpha\alpha$	$+\frac{1}{2}\nu_{0,1} + \frac{1}{2}\nu_{0,2} + \frac{1}{4}J_{1,2}$
2	$\alpha\beta$	$+\frac{1}{2}\nu_{0,1} - \frac{1}{2}\nu_{0,2} - \frac{1}{4}J_{1,2}$
3	$\beta\alpha$	$-\frac{1}{2}\nu_{0,1} + \frac{1}{2}\nu_{0,2} - \frac{1}{4}J_{1,2}$
4	$\beta\beta$	$-\frac{1}{2}\nu_{0,1} - \frac{1}{2}\nu_{0,2} + \frac{1}{4}J_{1,2}$

Where $\nu_{0,1}$ and $\nu_{0,2}$ are the Larmor frequencies of the 1H and 13C nuclei respectively. Since our magnetic field driving our system is in the +Z direction, we denote our lowest energy spin (spin up) as the α spin state and our highest energy spin (spin down) as the β spin state.

The J-Coupling causes the energy of the $\alpha\alpha$ and $\beta\beta$ states to be shifted by $+\frac{1}{4}J_{1,2}$ and the energy of the $\alpha\beta$ and $\beta\alpha$ states to be shifted by $-\frac{1}{4}J_{1,2}$. From our previous lab, we found our $J_{1,2} \approx 206.21$ Hz. However, for the sake of simplicity, we chose to use $J_{1,2} = 215$ Hz as per the results found in **?**. This is a fine approximation to make since our results don't deviate much from the paper and the $J_{1,2}$ frequency is significantly smaller than the Larmor frequencies of our nuclei. Even though it is a small shift, it is enough to cause a noticeable difference in our spin states which I will discuss in the following section.

1.2 Thermal Equilibrium

A key point of understanding in quantum computing with NMR is the concept of thermal equilibrium. When left alone, our 1H and 13C nuclei spins will relax to their respective ground state given by the Boltzmann distribution. This means that the probability of finding a nucleus in the $|\alpha\rangle$ state is $p_\alpha = 1/e^{\frac{E_\alpha}{kT}} + 1$ and the probability of finding a nucleus in the $|\beta\rangle$ state is $p_\beta = 1/e^{\frac{E_\beta}{kT}} + 1$. The energies E_α and E_α are the energies of the $|\alpha\rangle$ and $|\beta\rangle$ spin states respectively. The temperature T is the temperature of the system, which for our system, is assumed to be room temperature. The Boltzmann constant k is a constant that is equal to $1.38 \cdot 10^{-23}$ J/K. Doing this calculation gives us the density matrix of the spins' ground state:

$$\rho \approx \frac{1}{4}\mathbb{I} + 10^{-4}\begin{bmatrix} 5 & 0 & 0 & 0 \\ 0 & 3 & 0 & 0 \\ 0 & 0 & -3 & 0 \\ 0 & 0 & 0 & -5 \end{bmatrix}$$

In depth calculations of this thermal state can be seen in my prelab.

1.3 Exhaustive Averaging to Achieve the $|00\rangle$ State

The goal of this lab is to do basic quantum computation with our NMR system. To do this, we need to be able to prepare our spin states in some initial state. Say we wish to prepare our spins in the $|00\rangle\langle 00|$ state. In this case the other $|01\rangle\langle 01|$, $|10\rangle\langle 10|$, and $|11\rangle\langle 11|$ constitute as noise. To remove this noise, we can take advantage of the fact that our computation and observation are *linear* in the input. Therefore, we can simply perform three different experiments that will permute our undesired noise states. We then take an average of these three experiments to get an experiment operating only on the desired $|00\rangle\langle 00|$ state. This is called *exhaustive averaging* and is a common technique used in NMR quantum computation.

1.3.1 P0

The first permutation matrix is the P_0 matrix. The P_0 matrix is simply just the identity matrix and leaves our thermal state alone:

$$P_0 = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

1.3.2 P1

The second permutation matrix is the P_1 matrix. This permutes the $|01\rangle\langle 01|$, $|10\rangle\langle 10|$, and $|11\rangle\langle 11|$ states into the $|11\rangle\langle 11|$, $|01\rangle\langle 01|$, and $|10\rangle\langle 10|$ states respectively. The P_1 matrix is given in the form:

$$P_1 = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 1 & 0 & 0 \end{bmatrix}$$

This can be achieved by applying the following sequence of pulses before running our experiment:

$$I_y^2(\pi/2) \rightarrow 2I_z^1 I_z^2 \rightarrow I_y^1(\pi/2) \rightarrow I_x^2(\pi/2) \rightarrow 2I_z^1 I_z^2 \rightarrow I_x^1(\pi/2)$$

1.3.3 P2

The third pseudo state matrix is the P_2 matrix. This permutes the $|01\rangle\langle 01|$, $|10\rangle\langle 10|$, and $|11\rangle\langle 11|$ states into the $|10\rangle\langle 10|$, $|11\rangle\langle 11|$, and $|01\rangle\langle 01|$ states respectively. The P_2 matrix is given in the form:

$$P_2 = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \end{bmatrix}$$

This can be achieved by applying the following sequence of pulses before running our experiment:

$$I_y^2(\pi/2) \rightarrow 2I_z^1 I_z^2 \rightarrow I_y^1(\pi/2) \rightarrow I_x^2(\pi/2) \rightarrow 2I_z^1 I_z^2 \rightarrow I_x^1(\pi/2) \rightarrow I_y^1(\pi/2)I_y^2(-\pi/2) \rightarrow I_y^1(-\pi/2)I_y^2(\pi/2)$$

This is very similar to the P_1 matrix, but we change up the ordering of I_x and I_y pulses. We also apply an appropriate Hadamard gate at the end of the sequence - created by applying $I_y^1(\pi/2)I_y^2(-\pi/2)$ and $I_y^1(-\pi/2)I_y^2(\pi/2)$.

1.4 Taking an Exhaustive Average

Lastly, we can apply each of the pseudo state matrices to our thermal state and run our algorithm for each pseudo state. We take an average of the results and we effectively get a result that is prepared in the $|00\rangle$ state. The full exhaustive averaging procedure is given by the following:

$$\begin{aligned}\bar{\rho} &= \frac{1}{3}\sum_{i=0}^{2} P_i \rho P_i \\ &= \frac{1}{4}\begin{bmatrix}1&0&0&0\\0&1&0&0\\0&0&1&0\\0&0&0&1\end{bmatrix} + 10^{-5}\begin{bmatrix}1&0&0&0\\0&-\frac{1}{3}&0&0\\0&0&-\frac{1}{3}&0\\0&0&0&-\frac{1}{3}\end{bmatrix} \\ &= (\frac{1}{4}-\frac{1}{3}\times 10^{-5})\begin{bmatrix}1&0&0&0\\0&1&0&0\\0&0&1&0\\0&0&0&1\end{bmatrix} + 10^{-5}\begin{bmatrix}\frac{4}{3}&0&0&0\\0&0&0&0\\0&0&0&0\\0&0&0&0\end{bmatrix}\end{aligned}$$

After this computation, the average of the measurements (in the σ_z basis) is given by

$$tr(C\bar{\rho}C^{\dagger}\sigma_z^1) = \frac{4}{3}\times 10^{-5} tr(C|00\rangle\langle 00|C^{\dagger}\sigma_z^1)$$

Where C is our algorithm we wish to run. We can see from this example that all the undesired input states are averaged out, and we are left with the desired $|00\rangle\langle 00|$ state. (**?**, Jones, pg 95) (**?**, Knill et al, pg 57)

1.4.1 Exhaustive Averaging of Other States

This is of course how we prepare the $|00\rangle\langle 00|$ state. However, we can also prepare other states by simply applying an $I_x(\pi)$ on each of the spins we want to prepare. For example, if we want to prepare the $|01\rangle$ state, we would simply do an $I_x^2(\pi)$ pulse before applying our desired algorithm.

1.5 Reading our Results

We can read our results with respect to their corresponding density matrix by simply measuring the integral of our two spectrum peaks. As discussed in the last lab, these two peaks correspond to the energy transitions of either the 1H or 13C spin. If we think of our density matrix in the form:

$$\rho = \begin{bmatrix}a&0&0&0\\0&b&0&0\\0&0&c&0\\0&0&0&d\end{bmatrix}$$

The two 1H peaks are given by the difference of $a - c$ and $b - d$. The two 13C peaks are given by the difference of $a - b$ and $c - d$. We can use these two values to confirm that our experiment produced correct results based on applying our desired algorithm to the above generalized density matrix. We know that more specifically these a, b, c, d values correspond to the thermal state values (**?**, MIT, pg 1):

$$\rho \approx \frac{1}{4}\mathbb{I} + 10^{-4}\begin{bmatrix}5&0&0&0\\0&3&0&0\\0&0&-3&0\\0&0&0&-5\end{bmatrix}$$

For example, if we ran a simple FID experiment on our system, we would expect to see the integrals of 13C proportional to $5 - 3 = 2$ and $3 - (-5) = 8$.

1.6 Quantum Gates

1.6.1 Hadamard Gate

The Hadamard is the foundational gate that can be applied to any qubit to generate equal super position. In our NMR system, we can create a Hadamard gate on qubit j by applying the following sequence of pulses:

$$I_y^j(\pi/4) \rightarrow I_x^j(\pi) \rightarrow I_y^j(-\pi/4)$$

The resulting matrix is:

$$\text{H(j)} = \frac{1}{\sqrt{2}} \begin{bmatrix} 1 & 1 \\ 1 & -1 \end{bmatrix}$$

1.6.2 Free Rotation $I_z^1 I_z^2$

One of the most important pieces of multi-qubit quantum computation is entanglement. In our NMR system, we can take advantage of the fact that our two spins are coupled together within the magnetic field to create a two qubit gate capable of entanglement; This gate is the $I_z^1 I_z^2$ gate. It is achieved by simply letting the spins freely precess about the z-axis for a period of time. This gate is effectively just a matrix exponential of $\sigma_z \otimes \sigma_z$ for a given angle θ (**?**, Knill et al):

$$I_z^1 I_z^2(\theta) = e^{-i\theta/2\sigma_z \otimes \sigma_z}$$

We can find the appropriate time delay τ for a desired angle θ using our J-Coupling J between the two spins:

$$\tau = \frac{\theta}{J\pi}$$

For example, if we want to create an $I_z^1 I_z^2$ gate with an angle θ of $\pi/2$ we can use the time delay τ value of:

$$\tau = \frac{\pi/2}{J\pi} = \frac{1}{2J}$$

1.6.3 Single Qubit I_z

In our NMR system we know that both qubits are constantly precessing about the z-axis when left alone. This makes it quite difficult to create a single qubit z-rotation gate. However, we can create a single qubit z-rotation gate on qubit j with the following set of pulses:

$$I_x^j(\pi/2) \rightarrow I_y^j(\pi) \rightarrow I_x^j(\pi/2)$$

This is simply a decomposition of the I_z gate into a series of rotations about the x and y axes. The resulting matrix is:

$$I_z = \begin{bmatrix} 1 & 0 \\ 0 & -1 \end{bmatrix}$$

1.6.4 Controled Z Gate

The CZ gate applies a phase of -1 to the target qubit if the control qubit is in the $|1\rangle$ state. This gate is useful for creating entanglement between two qubits. We use the $I_z^1 I_z^2$ pulse sequence above with $\tau = \frac{3}{2J}$ ($\theta = 3\pi/2$) to create the CZ gate. The pulse sequence is as follows (**?**, Jones):

$$I_z^1(\pi/2) \rightarrow I_z^2(\pi/2) \rightarrow 2I_z^1 I_z^2(3\pi/2)$$

This of course gives us a matrix in the following form with a global phase of $e^{-i\pi/4}$:

$$\text{CZ(1,2)} = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & -1 \end{bmatrix}$$

1.6.5 Approximate CNOT

An approximate form of the CNOT gate can be formed from a $I_z^1 I_z^2$ gate with a time delay τ corresponging to a θ of $\pi/2$.The approximate CNOT is similar to the true CNOT, but is off by a phase in the first, third, and fourth columns. We can create this gate by applying the following sequence of pulses:

$$I_y^2(\pi/2) \to 2I_z^1 I_z^2(\pi/2) \to I_x^2(\pi/2)$$

And its matrix is in the following with a global phase of $e^{-i\pi/4}$:

$$\text{Approximate CNOT} = \begin{bmatrix} -i & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & -i \\ 0 & 0 & -1 & 0 \end{bmatrix}$$

1.6.6 CNOT

Much like the approximate CNOT and CZ gate, the CNOT gate can be formed from a $I_z^1 I_z^2$ gate with a time delay of $\tau = \frac{1}{2J}$. There's a couple different methods to generate the CNOT gate, but the method our group used was with the following sequence of pulses:

$$I_z^1(\pi/2) \to I_z^2(-\pi/2) \to I_x^2(\pi/2) \to 2I_z^1 I_z^2(\pi/2) \to I_y^2(-\pi/2)$$

Where qubit 1 is our control qubit and qubit 2 is our target qubit (1H and 13C respectively). The pulse decomposition of the Hadamard and CZ gates are shown in the above sections. The matrix form of this gate is of course the following:

$$\text{CNOT(1,2)} = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 1 & 0 \end{bmatrix}$$

2 Methods and Procedure

2.1 Time Delay Calculations

From our previous lab, we found our J-Coupling term to be $J = 215$ Hz. We can use this value to calculate the time delay τ mentioned in our previous sections. For our $I_z^1 I_z^2$ gate with an angle θ of $\pi/2$ we can use the time delay τ value of:

$$\tau = \frac{\pi/2}{J\pi} = \frac{1}{2 \cdot 215} \approx 2.33\,\text{ms}$$

And for our $I_z^1 I_z^2$ gate with an angle θ of $3\pi/2$ we can use the time delay τ value of:

$$\tau = \frac{3\pi/2}{J\pi} = \frac{3}{2 \cdot 215} \approx 7\,\text{ms}$$

We simply delay this long between pulses to create the desired two qubit z-rotation gate. (**?**, Jones, pg 97)

2.2 Building the Approximate CNOT Gate

As mentioned in Section 1.6.5, we can create an approximate CNOT gate by applying the following sequence of pulses:

$$I_y^2(\pi/2) \rightarrow 2I_z^1 I_z^2(\pi/2) \rightarrow I_x^2(\pi/2)$$

We inputted these pulses into our NMR system's pulse editor and created a pulse sequence with the following code:

```
# Pulse sequence
initpp(dir) # Reset internal parameter list

cleardata(n1)

# === Rx pi/2 13C ===
pulse(2,a90C,p2,d90C)

# === Time evolution delay (1/2J) ===
delay(dEvolution)

# === Rx pi/2 13C ===
pulse(2,a90C,p1,d90C)

delay(0.25)

# Acquisition pulse
pulse(2,a90C,p2,d90C)

# Acqusition delay
delay(dAcq)

acquire("overwrite", n1)

lst = endpp(mode) # Return parameter list

# Phase cycle list
phaseList  = [1,3,1,3;   # p1:  y
              0,2,0,2;   # p2:  x
              3,1,3,1;   # p3: -y
              2,0,2,0;   # p4: -x
              0,2,0,2]   # acquire phase
```

2.3 Building the CNOT Gate

As mentioned in Section 1.6.6, we can create the CNOT gate by applying the following sequence of gates:

$$I_z^1(\pi/2) \rightarrow I_z^2(-\pi/2) \rightarrow I_x^2(\pi/2) \rightarrow 2I_z^1 I_z^2(\pi/2) \rightarrow I_y^2(-\pi/2)$$

Here is the respective pulse sequence code:

```
# ======= CNOT =======

# === Rz pi/2 1H and Rz -pi/2 13C ===
pulse(1,a90H,p4,d90H)
delay(0.25)
pulse(1,a90H,p3,d90H)
delay(0.25)
pulse(1,a90H,p2,d90H)

delay(0.25)

pulse(2,a90C,p4,d90C)
delay(0.25)
pulse(2,a90C,p1,d90C)
delay(0.25)
pulse(2,a90C,p2,d90C)
delay(0.25)

# === identity 1H and Rx pi/2 13C ===
pulse(2,a90C,p2,d90C)

# === Time evolution delay ===
delay(dEvolution)

# === identity 1H and Ry -pi/2 13C ===
pulse(2,a90C,p3,d90C)

# ======= End CNOT =======

delay(0.25)

# Acquisition pulse
pulse(2,a90C,p2,d90C)

# Acqusition delay
delay(dAcq)

acquire("overwrite", n1)

lst = endpp(mode) # Return parameter list

# Phase cycle list
phaseList  = [1,3,1,3;   # p1:  y
              0,2,0,2;   # p2:  x
              3,1,3,1;   # p3: -y
              2,0,2,0;   # p4: -x
              0,2,0,2]   # acquire phase
```

2.4 Creating our Pseudo-Pure States (Perumation Matrices)

We were able to create our pseudo-pure states with the pulse sequences referenced in sections 1.3.2 and 1.3.3. Its worth noting that $P0$ is simply just the identity matrix and $P1$ and $P2$ are effectively built with interchanging CNOT gates. This is shown in Figure 1.

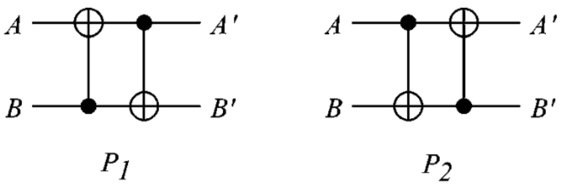

Figure 1 – Circuit representations of exact $P1$ and $P2$ sequences

Our NMR code to create these states is as follows:

```
    # ======= P0 =======
    if pVal == "P0"
       print("Running␣with␣P0\n")
    # ======= P1 =======
    elseif pVal == "P1"
       print("Running␣with␣P1\n")

       # 90y-H pulse
       pulse(1,a90HC,p1,d90C)

       # Evolution delay
       delay(dEvolution)

       # 90x-H and 90y-C pulse
       pulse(1,a90HC,p2,freq1H,2,a90C,p1,freq13C,d90C)

       # Evolution delay
       delay(dEvolution)

       #  90x-C pulse
       pulse(2,a90C,p2,d90C)
    # ======= P2 =======
    elseif pVal == "P2"
       print("Running␣with␣P2\n")

       #  90y-C pulse
       pulse(2,a90C,p1,d90C)

       # Evolution delay
       delay(dEvolution)

       # 90y-H and 90x-C pulse
       pulse(1,a90HC,p1,freq1H,2,a90C,p2,freq13C,d90C)

       # Evolution delay
       delay(dEvolution)

       # 90x-H pulse
       pulse(1,a90HC,p2,d90C)
```

```

        delay(0.25)

        #hadamard sequence
        pulse(1,a90HC,p1,freq1H,2,a90C,p3,freq13C,d90C)
        delay(0.25)
        pulse(1,a90HC,p3,freq1H,2,a90C,p1,freq13C,d90C)
    endif
```

We simply had a variable that was set to either ‘P0‘, ‘P1‘, or ‘P2‘ and then ran the appropriate code.

2.4.1 Exact Matrix Representation

Lastly, its worth mentioning that these pulse sequences actually create $P1$ and $P2$ up to a local phase. Their respective exact matrix representations are as follows:

$$P1 = \begin{bmatrix} -i & 0 & 0 & 0 \\ 0 & 0 & 0 & -i \\ 0 & -1 & 0 & 0 \\ 0 & 0 & 1 & 0 \end{bmatrix}$$

$$P2 = \begin{bmatrix} -i & 0 & 0 & 0 \\ 0 & 0 & -1 & 0 \\ 0 & 0 & 0 & -i \\ 0 & 1 & 0 & 0 \end{bmatrix}$$

Even though these have a local phase, they are still effective at creating our pseudo-pure states. This is because the local phase is irrelevant for our measured results.

3 Results and Discussion

3.1 Experiment 1: Approximate CNOT

For our first experiment we constructed an approximate form of the CNOT gate. We achieved this using the pulse sequence and code shown in section 1.6.5. Our resulting spectra for 13C is shown in figure 2 for approximate CNOT on the thermal state.

From our results, we see a negative peak at about 80 ppm and a positive peak at about 70 ppm. This is consistent with the expected results for the Approximate CNOT gate up to a global phase. The resulting density matrix we'd expect is as follows:

$$\rho_{\text{CNOT Approx}} = \begin{bmatrix} a & 0 & 0 & 0 \\ 0 & b & 0 & 0 \\ 0 & 0 & d & 0 \\ 0 & 0 & 0 & c \end{bmatrix}$$

Using our approximate density matrix a, b, c, and d values, we can calculate the expected values of our peaks using our theory from section 1.5. From this we'd expect to see a 13C peak integrals proportional to values of $5 - 3 = 2$ and $-5 - (-3) = -2$. Summing them together we get a total of 0 which is what we approximately see in our results.

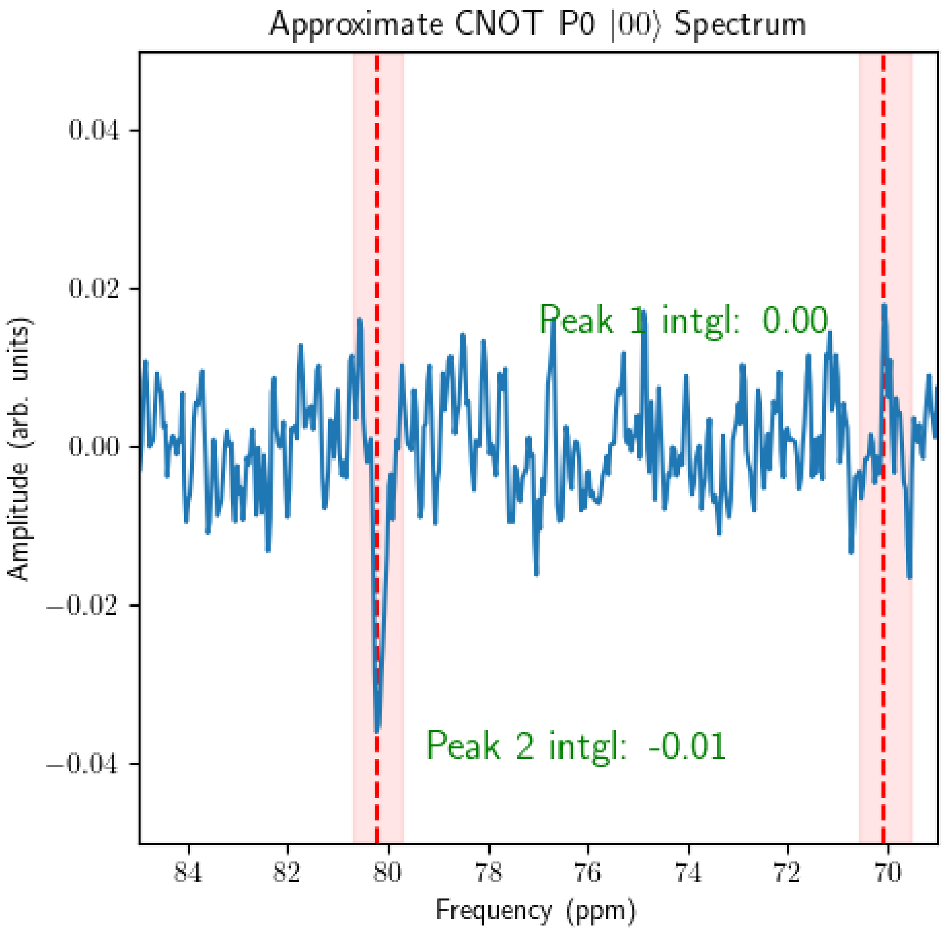

Figure 2 – Results for Approximate CNOT

3.2 Experiment 2: True CNOT

Our second experiment was to create a true CNOT gate using a modified set of pulses from our approximate CNOT gate. We achieved this using the pulse sequence and code shown in section 2.3. Our resulting spectra for 13C is shown in the left-most graph in figure 6. Again, we expect to see 13C peak integrals proportional to values of $5 - 3 = 2$ and $-5 - (-3) = -2$. We see this correctly up to a global phase. Summing our integrals together gives us a value of -0.17, which is quite close to our expected value of 0.

3.3 Experiment 3: Pseudo-Pure States

In order to prepare for doing an exhaustive average of our CNOT gates, we needed to prepare pseudo-pure states. We did this by using the pulse sequences and code shown in section 2.4. Our resulting spectra for 13C is shown in figures 4 and 5. The resulting density matrices we'd expect to see for $P1$ is as follows:

$$\rho_{\text{P1}} = \begin{bmatrix} a & 0 & 0 & 0 \\ 0 & d & 0 & 0 \\ 0 & 0 & b & 0 \\ 0 & 0 & 0 & c \end{bmatrix}$$

This corresponds to proportional integral values of $3 - (-3) = 6$ and $5 - (-5) = 10$ for the P1 13C integrals. In our spectral results we see two positive peaks with integrals of 0.2 and 0.4 respectively. This is seems to be consistent with our expected results, since we'd expect to see the peaks proportional to $\frac{2}{5}$ in value and we get peaks proportional to $0.2/0.4 = \frac{1}{2}$ in value. The resulting density matrices we'd expect to see for $P2$ is as follows:

$$\rho_{\text{P2}} = \begin{bmatrix} a & 0 & 0 & 0 \\ 0 & c & 0 & 0 \\ 0 & 0 & d & 0 \\ 0 & 0 & 0 & b \end{bmatrix}$$

This corresponds to proportional integral values of $-3-5 = -8$ and $5-(-3) = 8$ for the P2 13C integrals. In our spectral results we see two large positive peaks with integrals of 0.6 and -0.5 respectively. Here it seems we are correct up to a global phase. Are values are relatively 1 to -1 proportional to each other, which is what we'd expect.

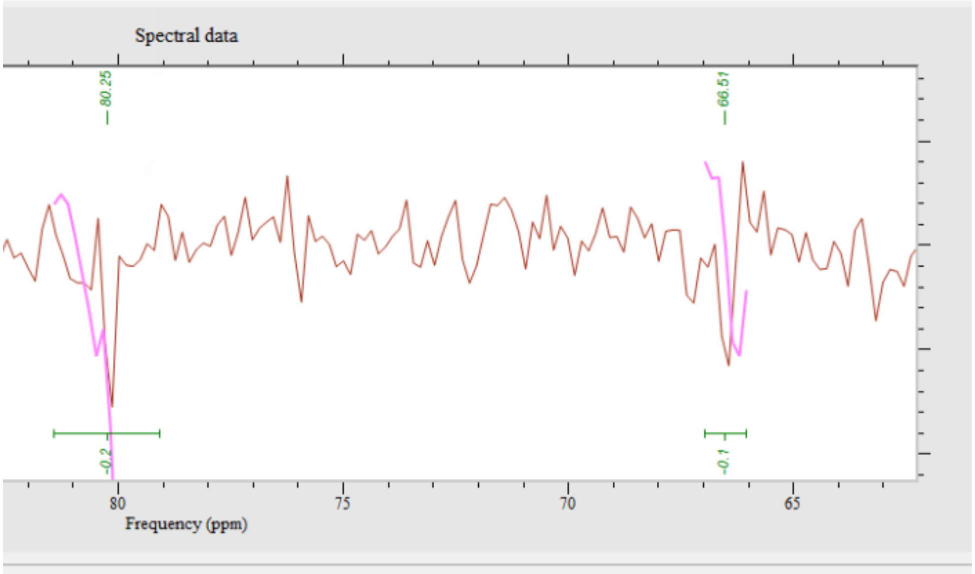

Figure 3 – Results for P0

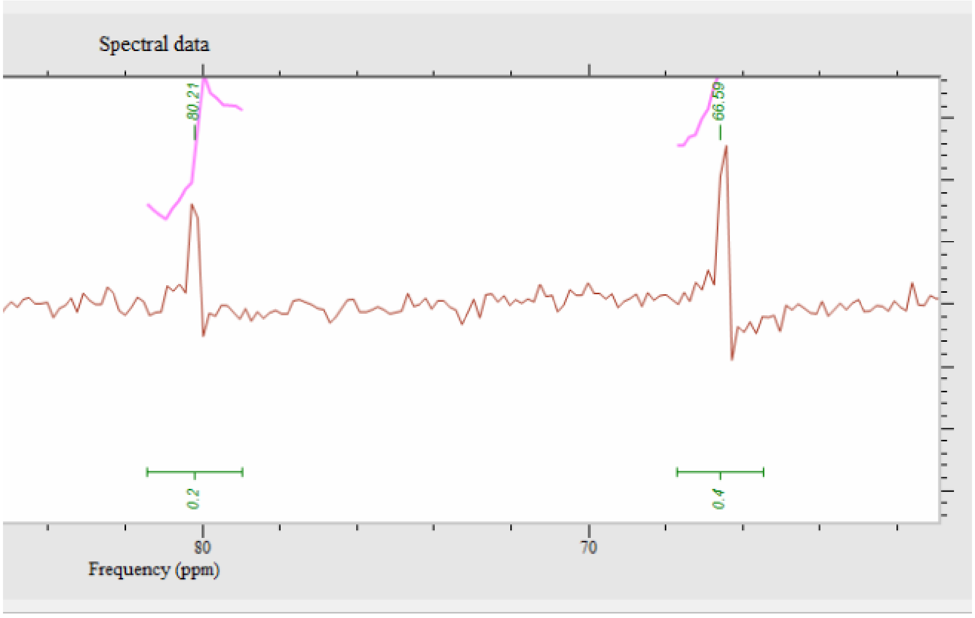

Figure 4 – Results for P1

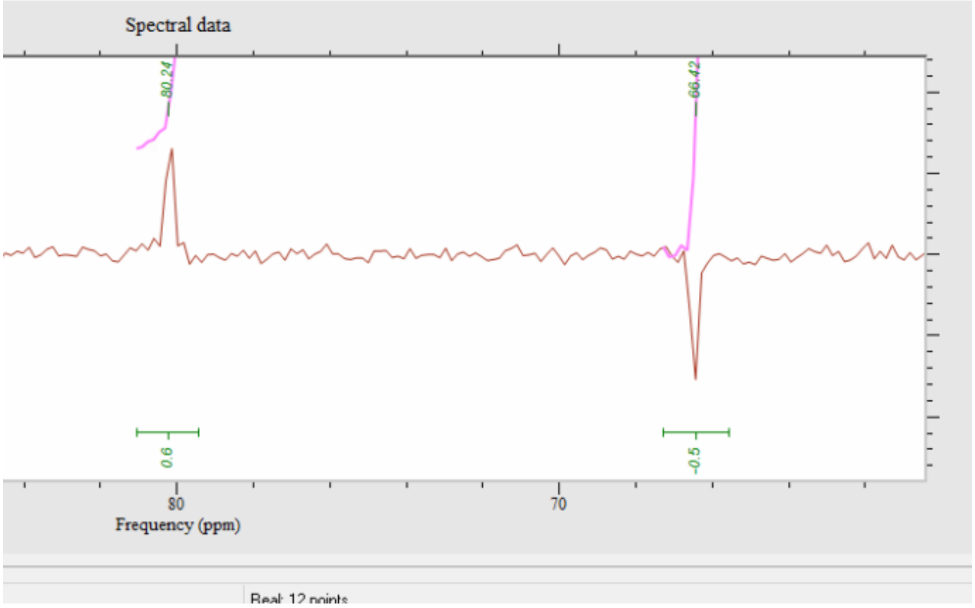

Figure 5 – Results for P2

3.4 Experiment 4: CNOT Truth Table

Unfortunately, we were unable to create a full CNOT truth table because we could not get readout of the 1H channel. This was mainly due to our ignorance of how our NMR machine's code works. We were however, able to get readout of the 13C channel.

We ran the true CNOT gate on the pseudo-pure states $P0$, $P1$, and $P2$. Our resulting graphs (along with their computed integrals) are shown in figures 6, 7, 8, and 9. There is a lot to unpack in these graphs. Our results overall seem fairly consistent with what we'd expected from our theory. We do seem to off by a global phase in some of our results - most notable in the $P0$ and $P1$ results for each state. We do see a bit of error in our results for our $P0$ and $P2$ states. For $P0$ we would expect to see peaks that are equal and opposite in value and sum to 0. For $P2|00\rangle$ and $P2|11\rangle$ we would also expect to see the peaks be exactly equal in value. This slight bit of error is likely due to experimental error of our NMR machine. It is also likely caused by the low Larmor frequency of 13C compared to 1H. This means that our 13C spins are more susceptible to noise and other experimental error.

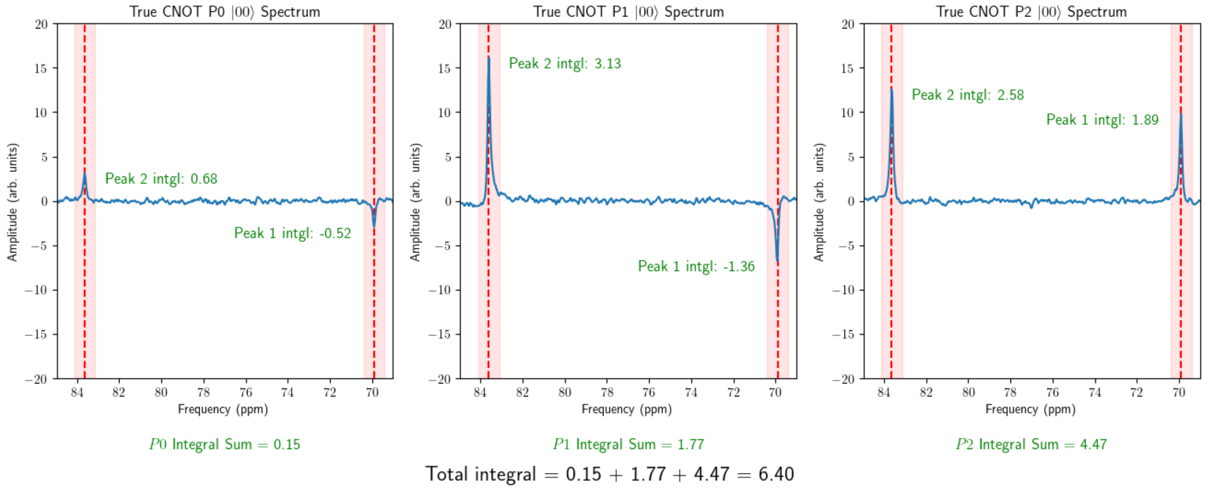

Figure 6 – Results for CNOT on $|00\rangle$

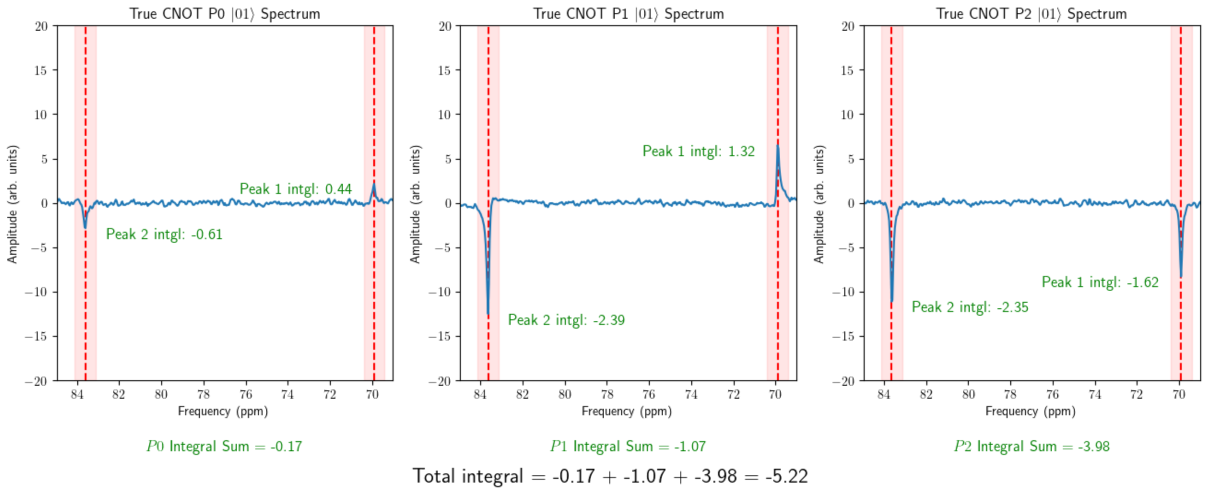

Figure 7 – Results for CNOT on $|01\rangle$

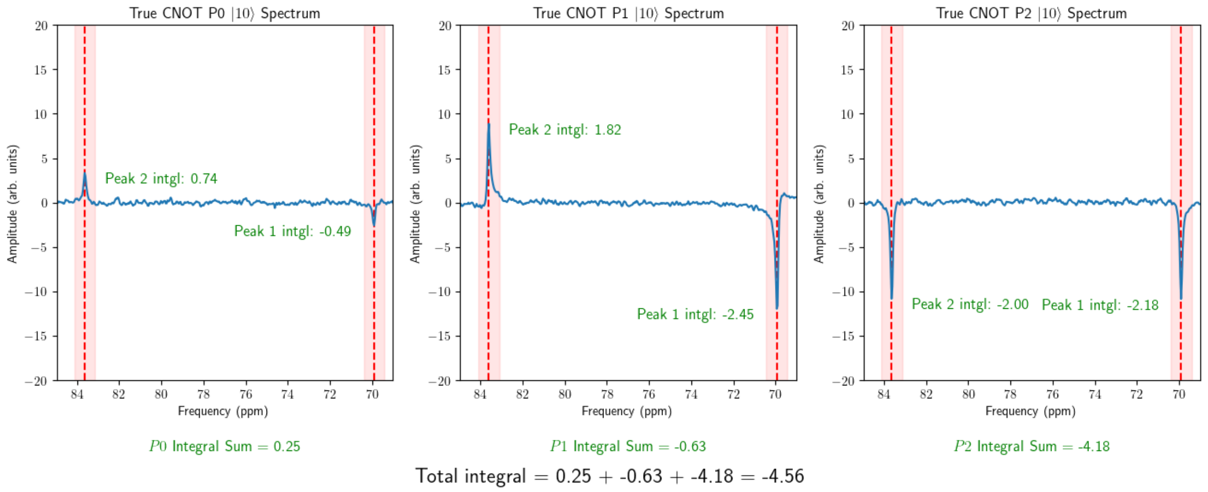

Figure 8 – Results for CNOT on $|10\rangle$

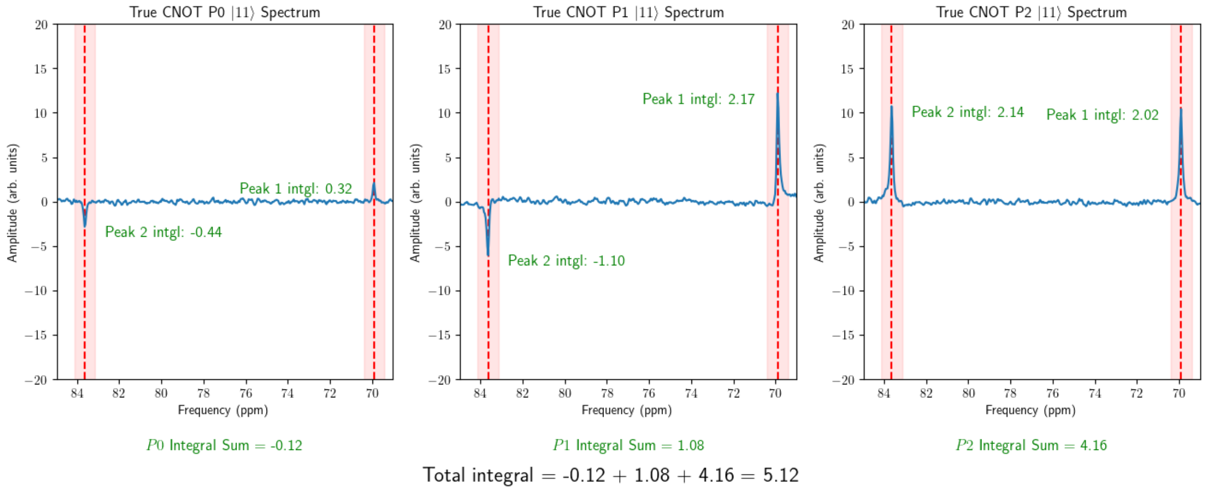

Figure 9 – Results for CNOT on $|11\rangle$

4 Conclusion

Due to time constraints, we were also unable to get a full readout of our approximate CNOT (only our true CNOT). However, we would expect to see nearly identical results to our true CNOT based on our theory. Had we had more time and better understanding of our NMR system going into the lab, then we would have been able to fully perform this experiment alongside our other experiments.

We did seem to have gotten quite good results for our CNOT and pseudo-pure states. Our peak integrals matched theory quite well and correct up to a global phase. We did have some error in our results where our peaks' proportionality was slightly off from what we would have expected. This is likely due to experimental error of our NMR machine. It is also likely caused by the low Larmor frequency of 13C, which makes is more susceptible to noise and other experimental error. There is also a possibility that our pulse sequences aren't perfectly correct. Had we had more time to cross-referenced our pulse sequence and code with the TAs and other groups in the lab, we would have been more confident in our results.

We also didn't have time to perform a full exhaustive average over our CNOT results to average out the other spins states that constitute noise. Doing this would have most likely given us better results. We have the proper theory and understanding of how to do this, and plan to apply the method of exhaustive averaging in our next lab.

References

Jones, J. A. Quantum computing with nmr. *Progress in Nuclear Magnetic Resonance Spectroscopy 59*, 2 (Nov 2010), 91–120.

Keeler, J., and Keeler, J. *Chapters 2-8*. Wiley, 2012, p. 2.1–8.1.

Knill, E., Chuang, I., and Laflamme, R. Effective pure states for bulk quantum computation. *Physical Review A 57*, 5 (1998), 3348–3363.

of Physics, M. D. vol. 180. 2016.