Quantum Information Processing with Liquid-State NMR

Pranjal Vachaspati, Sabrina Pasterski*

MIT Department of Physics
(Dated: May 18, 2013)

We demonstrate the use of a Bruker Avance 200 NMR Spectrometer for simulating a two-qubit quantum computer. By manipulating magnetic fields in two directions, we are able to create quantum states with a fidelity of close to 0.9, as well as CNOT and Hadamard gates to implement the Deutsch-Jozsa algorithm and Grover's algorithm. We also find NMR properties of a chloroform solution, including a T_1 time of $18.3 \pm 0.3\,\text{s}$ for the hydrogen nucleus and $19.1 \pm 0.6\,\text{s}$ for the carbon nucleus.

I. INTRODUCTION

In the 1930s, Alan Turing, along with others including Kurt Gödel and Alonzo Church, published a series of papers that laid the foundations for the field of theoretical computer science. One of the most important ideas from that time was the idea of "universal computation", which suggested that anything that could be computed by something in the universe could be computed by a thought-experiment machine called a "Turing Machine"[6]. Later, this was extended into the idea that no machine could compute anything faster than a Turing machine by more than a polynomial factor.

However, in 1994, Daniel Simon published an algorithm[5] that relied on quantum mechanics, that could only run on a new machine called a quantum computer, the idea of which had been put forward in the early 1980s by Richard Feynman as a tool to simulate quantum physics[2]. Since then, the field of quantum information theory has exploded, promising great value for many different applications as well as providing deep insights into fundamental problems in computer science and physics.

II. THEORY

II.1. NMR Spin Manipulation

We present an implementation of a quantum computation using liquid-state NMR spectroscopy. In an NMR system, molecules are placed in a strong magnetic field that causes them to precess with a frequency proportional to the applied magnetic field. Then, applying weak magnetic field pulses in perpendicular directions makes the spins rotate.

---

* pranjal@mit.edu, fermilab@mit.edu

These spins relax back into their original state over time through a variety of mechanisms, including spin-spin interactions (T_2^*) and spin-lattice interactions (T_1). It is crucial for single measurements to be fast enough that the relaxation mechanisms do not play a significant part, and for consecutive measurements to be separated by enough time that each measurement starts in the same state.

The key to performing quantum computations is correlations between the states of multiple atoms. If an atom's neighbor has its spin in the same direction as the external field, the first atom will be in a slightly stronger magnetic field, and its Larmor frequency will increase; if the neighbor is pointing in the opposite direction as the external field, the first atom will have a slightly lower frequency.

In a liquid NMR system, most of the spins are randomly distributed, since the temperature $k_B T$ is higher than the spin energy $\hbar\omega$. The randomness is perturbed slightly by the Hamiltonian due to the magnetic field and spin-spin coupling, which leads to the phenomena observed.

In general, quantum states are denoted by a ket with two numbers like $|01\rangle$, which represents the spin of the carbon and hydrogen atoms. The state of the system is represented by a density matrix. The starting state of the system, for example, is given by

$$\rho = \frac{I}{4} + \begin{pmatrix} 5 & 0 & 0 & 0 \\ 0 & 3 & 0 & 0 \\ 0 & 0 & -3 & 0 \\ 0 & 0 & 0 & -5 \end{pmatrix}$$

II.1.1. Temporal Averaging

Since most quantum computations require the system to be in a pure starting state, some manipulation is required to derive a starting $|00\rangle$ state. Given the thermal state above, the same pulse sequence is run three times

after using a unitary transformation given by delays and rotations each time to achieve three initial states that sum to

$$
\begin{pmatrix} 5 & 0 & 0 & 0 \\ 0 & 3 & 0 & 0 \\ 0 & 0 & -3 & 0 \\ 0 & 0 & 0 & -5 \end{pmatrix} + \begin{pmatrix} 5 & 0 & 0 & 0 \\ 0 & -3 & 0 & 0 \\ 0 & 0 & -5 & 0 \\ 0 & 0 & 0 & 3 \end{pmatrix} + \begin{pmatrix} 5 & 0 & 0 & 0 \\ 0 & -5 & 0 & 0 \\ 0 & 0 & 3 & 0 \\ 0 & 0 & 0 & -3 \end{pmatrix}
$$

$$
= \begin{pmatrix} 20 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{pmatrix} + \begin{pmatrix} -5 & 0 & 0 & 0 \\ 0 & -5 & 0 & 0 \\ 0 & 0 & -5 & 0 \\ 0 & 0 & 0 & -5 \end{pmatrix}
$$

Only traceless observables are measured in NMR, so this is equivalent to starting with a pure $|00\rangle$ state. Additionally, the number of states required to get an initial pure state increases exponentially with the number of qubits, so this quantum computing method can never be faster than classical computing. Thus, this is more of a simulation of quantum computing than true quantum computing.

II.2. Quantum Computation

We consider the gate model of quantum computation, similar to the logic circuit model of classical computing. A series of unitary transformations are applied to an initial state, with a readout at the end. Depending on the algorithm, this readout either deterministically or probabilistically encodes the solution to a problem. It turns out that only a small set of gates are needed to create any unitary matrix to arbitrary precision; the two most commonly used are the Hadamard gate, which acts on one qubit, and the CNOT gate, which acts on two qubits, given by

$$
H = \frac{1}{\sqrt{2}} \begin{pmatrix} 1 & 1 \\ 1 & -1 \end{pmatrix} ; CNOT = \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 1 & 0 \end{pmatrix}
$$

II.2.1. Deutsch-Jozsa Algorithm

The first algorithm we implement is the Deutsch-Jozsa algorithm, which is used to determine whether a coin is fair. Given an oracle that either returns the same output for all possible inputs, or returns 1 for half the inputs and 0 for the other half, the Deutsch-Jozsa algorithm decides which kind of oracle is given with only a single query of the oracle. This is remarkable, because a
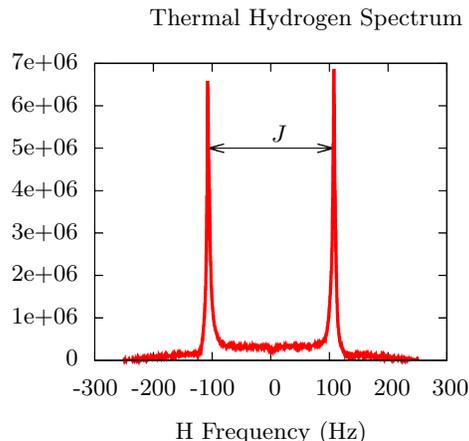

FIG. 1. Applying a $90°$ pulse to the hydrogen atom produces a spectrum of this form. The coupling J is given by the distance between the peaks, and the T_2^* relaxation time is given by the width of the peaks.

classical computer would require $2^{n-1} + 1$ calls to make this determination with certainty for an n bit oracle.

II.2.2. Grover's Algorithm

The second algorithm implemented is Grover's algorithm. While commonly referred to as an algorithm that searches a list, it is more accurate to think of it as reversing a function. Given an oracle that returns 1 for precisely one of n inputs and 0 for all other inputs, Grover's algorithm can determine the input that returns 1 in only $\sqrt{n}$ calls to the oracle! Of course, on a classical computer, this would take $\frac{n}{2}$ calls on average.

It is important to note that neither of these algorithms disprove the extended Church-Turing hypothesis or prove anything about important complexity theory relationships like $P = NP$. Both these algorithms are formulated with respect to oracles, which distinguishes them from standard problems which are formulated with respect to running time.

III. EQUIPMENT

We use a Bruker Avance 200 NMR Spectrometer prepared by the Junior Lab staff with a 7% solution of chloroform ($CHCl_3$) in a deuterated acetone solution. The spectrometer uses a large superconducting magnet to produce a strong constant B field in the z direction

with a proton Larmor frequency of 200 MHz. Magnetic field pulses can be created in the x and y directions to rotate the spins with smaller coils; these coils are also used in a superheterodyne receiver that can measure the frequency and complex phase of the signal.

These coils are controlled by a Linux PC running the XWINNMR program, which receives commands from a MATLAB instance running on a second Linux PC which can be accessed directly by the user or over the network.

After a pulse sequence ("program") is sent to the device and run, the x and y coils turn off and act as receivers and amplify the magnetic field from the rotation of the atoms. This is proportional to the polarization perpendicular to the z axis, since the atoms precess about the z axis. This free induction decay (FID) response is recorded and Fourier transformed to produce a spectrum. A characteristic spectrum for hydrogen is shown in Figure 1. The two Lorentzian peaks separated by 214 Hz correspond to the portion of molecules with aligned and inverse-aligned carbon and hydrogen spins. The most important part of the spectrum is the size of the peaks, given by fitting each of the peaks to a Lorentzian. Occasionally, especially for small peaks, the Levenberg-Marquardt algorithm used for fitting was unable to find a reasonable fit; in these cases the data point was ignored.

IV. PROCEDURE

First, certain NMR parameters must be determined in order to successfully model quantum gates. The pulse width needed for a $90°$ rotation can be determined by repeatedly applying a pulse of varying length in the x direction and measuring the response. The response is maximized for a pulse width corresponding to a $90°$ rotation.

Next, several parameters can be discerned from the Fourier transformed spectrum (Figure 1). The coupling factor J between the hydrogen and carbon atoms is given by the distance between peaks, and the T_2^* decay parameter is given by the width of the Lorentzian peak.

The final NMR parameter that must be measured is the T_1 relaxation time. An inversion recovery pulse sequence is used, where an initial $180°$ pulse which puts the spins in the $-z$ direction is followed by a varying time delay that returns the spin partway into the $+z$ direction; a final $90°$ pulse rotates the spins once more and free induction decay is measured. The magnitude of the decay varies exponentially as a function of the delay time, giving T_1. When running experiments, the time
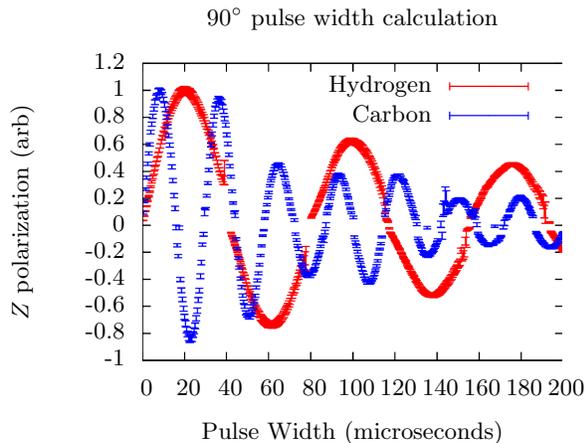

FIG. 2. The $90°$ pulse width is determined by varying the pulse width and maximizing the response

between pulse programs must be several times T_1, or else the particles will not be in a reliable starting state each time.

Following this, quantum states are created with a temporal averaging technique as described in II.1.1. These states can be rotated using the standard rotations, and CNOT gates can be applied using an $x-\overline{y}-\overline{x}$ rotation on the hydrogen atom and a $x - y - \tau - \overline{y}$ sequence on the carbon atom, where τ corresponds to a time delay determined by the coupling constant J.

CNOT gates and rotations can then be combined to implement the Deutsch-Jozsa and Grover algorithms described in II.2.

V. RESULTS

V.1. $90°$ **Pulse Width**

The pulse width was determined by varying the pulse length for both hydrogen and carbon from 0 to 200 microseconds, as shown in Figure 2. Various methods were used for extracting an optimal $90°$ pulse with from this (see VI.1), but the most successful was to find the maximum of a second-degree polynomial fit to the top of the first peak. We found the optimum width to be $20.3 \pm 0.2\,\text{s}$ for hydrogen and $8.2 \pm 0.1\,\text{s}$ for carbon. These values were used for the remainder of the experiment, with some corrections as the machine drifted slightly over time.
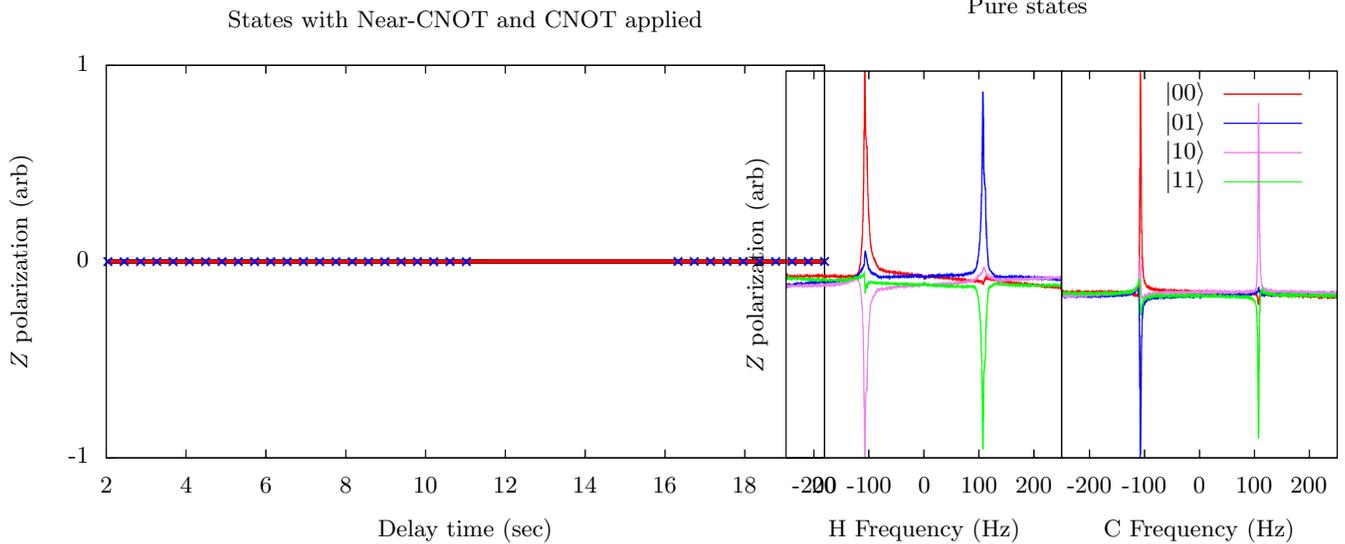
States with Near-CNOT and CNOT applied

Pure states

FIG. 3. Varying the delay in a $180° - \tau - 90°$ inversion recovery sequence allows for measurement of the T_1 relaxation time

FIG. 4. Pure states are created with temporal averaging

V.2. Determination of NMR Parameters

Before implementing the quantum algorithms, we determined certain NMR parameters of the system. First, we found the coupling constant J, determined by the difference in the centers of the two peaks, to be $213.8 \pm 0.3\,\mathrm{s}^{-1}$. Next, we determined T_2^* by measuring the characteristic time of the free induction decay. We found the hydrogen relaxation time to be $0.15 \pm 0.03\,\mathrm{s}$ and the carbon relaxation time to be $0.5 \pm 0.2\,\mathrm{s}$. These are sufficiently high that T_2^* relaxation is not a significant effect during the pulse sequences, which typically last at most a few milliseconds.

Next, the T_1 time was determined using an inversion recovery pulse sequence. The response to the pulse sequence was fit to an exponential as shown in Figure 3, determining a hydrogen T_1 of $18.3 \pm 0.3\,\mathrm{s}$ and a carbon T_1 of $19.1 \pm 0.6\,\mathrm{s}$. The delay between two consecutive pulse programs was therefore set to be at least $60\,\mathrm{s}$, and in some cases was set as high as $400\,\mathrm{s}$. Note that the signal passes through zero at $T_1 \ln 2$. This corresponds to a relaxation equal to the $90°$ pulse.

V.3. Creation of NMR States and Gates

Following the determination of the NMR parameters, quantum states were created using the temporal averaging technique. The quality of the states and gates was evaluated by measuring the fidelity, given by the ratio of the higher peak to the sum of the peaks.

The four possible pure states are shown in Figure 4. The hydrogen states had an average fidelity of 0.87 ± 0.04, and the carbon states had an average fidelity of 0.90 ± 0.01.

Next, the Deutsch-Jozsa algorithm was run on each of the four possible 1-bit functions, as described in II.2.1. The outputs look similar to the pure states in Figure 4, with fidelity 0.86 ± 0.07 for hydrogen and 0.90 ± 0.07 for carbon. This indicates that the Deutsch-Jozsa algorithm itself did not contribute significantly to the loss of fidelity; rather, much of the lack of fidelity came from the creation of the state itself.

Finally, Grover's algorithm was run for each of the four possible oracle functions. Instead of simply measuring the oscillatory behavior after a single application of Grover's algorithm, the algorithm was applied multiple times and the oscillatory behavior was confirmed, as seen in 5. The decay observed in the oscillating signal is due to the increasing effect of the T_2^* relaxation, which becomes relevant here, as each Grover iteration takes approximately four milliseconds.
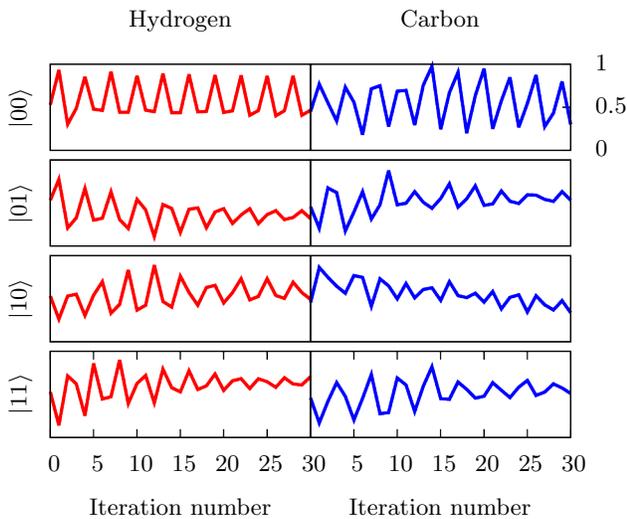
FIG. 5. Applying Grover's algorithm multiple times to a $|00\rangle$ state results in an oscillation with a period proportional to the number of qubits.

VI. ERRORS

VI.1. 90° Pulse Width

By far the largest source of low fidelity in the states was due to trouble calculating the 90° pulse width. After running hundreds of trials of the pulse width tests, we discovered that there was an intrinsic fluctuation in the transmitting and receiving coils, with a period of approximately 77 minutes. However, it was unclear whether the fluctuation was in the transmitting or receiving circuits. If the fluctuation was in the transmitting circuits, there would be a true variation in the needed pulse width; however, if it was in the receiving circuits, the apparent variation would not be representative of an actual change in the behavior of the spins. There were a number of other problems in determining the appropriate pulse width; in the end, this was the greatest source of our error.

VII. CONCLUSION

We successfully use liquid NMR techniques to find properties of a chloroform solution and implement quantum gates and algorithms. We find that we are able to create the states and apply gates while maintaining a fidelity of 0.86 ± 0.04 for hydrogen and 0.9 ± 0.01 for carbon.

[1] David Deutsch, Richard Jozsa, David Deutsch, and Richard Jozsa. Rapid solution of problems by quantum computation. *Proceedings of the Royal Society of London. Series A: Mathematical and Physical Sciences*, 439(1907):553–558, 1992.

[2] Richard P Feynman. Simulating physics with computers. *International journal of theoretical physics*, 21(6):467–488, 1982.

[3] Lov K Grover. A fast quantum mechanical algorithm for database search. In *Proceedings of the twenty-eighth annual ACM symposium on Theory of computing*, pages 212–219. ACM, 1996.

[4] MIT Department of Physics. Quantum information processing with nmr. August 2010.

[5] Daniel R Simon. On the power of quantum computation. In *Foundations of Computer Science, 1994 Proceedings., 35th Annual Symposium on*, pages 116–123. IEEE, 1994.

[6] Alan Turing. On computable numbers, with an application to the entscheidungsproblem (1936). *B. Jack Copeland*, page 58, 2004.