Lecture 23: Fault-Tolerant Quantum Computation

Scribed by: Jonathan Hodges

Department of Nuclear Engineering, MIT

December 4, 2003

1 Introduction

Before von Neumann proposed classical fault-tolerance in the 1940's, it was assumed that a computational device comprised of more than 10^6 components could not perform a computation wihout encountering a fatal hardware error. Von Neumann proved that one could indeed make the computation work, as long as a certain degree of overhead was tolerable. Thus follows the classical fault-tolerance theorem:

Theorem 1 (Classical Fault-Tolerance). *A computation of length n using perfect computational components can be executed reliably (i.e. with probability* $1-\frac{1}{n^\alpha}$ *for polynomial* α *) using* $\mathcal{O}(n \log n)$ *steps, provided the components work with probability 1 -* ϵ *of the time and that the faults encountered are independent.*

We can sketch von Neumann's proof of Fault-Tolerance as follows: Given classical AND,NOT, and OR gates let us encode a 0 into many 0's for $c \log n$ times, where c is some constant.

$$0 \rightarrow 0000000 \tag{1}$$

Now take two identical copies of "0", call them a and b, and put them through the AND gate. Ideally one should get 1111111. Instead, suppose the strings received are 1110101 on a and 0111111 on b. If one performs a bit-wise AND on each successive bit of the bit strings a and b, the result is 0110101. Taking "triples" of bits of this resulting addition, one performs a majority vote. Thus, if one bit has an error probablity of p , two bits in a triple being erroneous occurs with probabiltiy $3p^2$. As long as p is small, one can perform operations with fault-tolerance. The same type of proof can be shown for NOT and OR gates, thus giving universality.

In short, if one has compenents of a computer whose fidelity are high enough, and adding additional compents is relatively easy, then the computation is indeed plausible. As it turns out the critics were too critical. Your desktop computer does not even use a software error-correction for doing computations, as the ϵ for our silicon-based hardware has become increasingly small.

2 Using classical techniques for quantum computation

Classically, four methods exist for dealing with fault-tolerant computing, but only one of these will prove feasible. Consistency checks, like the parity of a bit string, work classically, but in the quantum world are simply not powerful enough. Checkpoints require stoping the computation at a

specific point, checking the result, then starting the computation again. The probablistic nature of quantum mechanics and the no-cloning theorem make this technique useless for QC. Classically, one might make many copies of the computation to perform a massive redundancy scheme; however, this errs like consistency checks, as it is not "powerful enough" for quantum computations. Thus, one is left with error correcting codes, which have previously been shown portable from the classical to the quantum domain.

3 Quantum Fault Tolerance

In order to take an errorless quantum computation to a fault-tolerant computation, one first encodes each qubit into a quantum error correcting code. Every gate in the circuit should then be replaced by a fault tolerant version of it. Finally, insert an error correction step after every gate. Above we argued that fault-tolerance in classical computations need only AND, NOT, OR gates. For universal quantum computation only the CNOT and single-qubit gates are required, but formulating fault-tolerant operations becomes easier with a finite gate set. CNOT, Hadamard, σ_x, σ_z, T, Toffoli, and $\frac{\pi}{8}$ will be proven useful.

Theorem 2 (Kitaev-Solovay Theorem). *Given a set of gates on SU(2) (or SU(k) generally) that generates a dense set in SU(2), then any gate $U \in SU(2)$ can be approximated to ϵ using $\mathcal{O}(log^c \frac{1}{\epsilon})$ gates where $1 \leq c \leq 2$. See Appendix 3 of Nielsen and Chuang for more details.*

3.1 Fault Tolerance of σ_x

In order to show that a σ_x gate can be done with fault tolerance, let us encode 1 qubit into k qubits using a CSS Code (the Steane Code).

$$|0\rangle \longrightarrow \frac{1}{\sqrt{|C_2|}} \sum_{x \in C_2} |x\rangle \tag{2}$$

$$|1\rangle \longrightarrow \frac{1}{\sqrt{|C_2|}} \sum_{x \in C_2} |x + v\rangle \tag{3}$$

Since $\dim(C_1)$ is k, $\dim(C_2)$ is k-1, this code will encode a single qubit, and satisfies the inequality $0 \subseteq C_2 \subseteq C_1$. The codewords v are those not overlapping the two classical codes; $v \in C_1 - C_2$. Since the encoded $|0\rangle$ and $|1\rangle$ are orthogonal, a σ_x should just interchange the two encodings. These two states are separated by v, which amounts to peforming a σ_x on each individual qubit. Now suppose an error is made in performing σ_x on one of the qubits, where the errors on each qubit are uncorrelated. Then this code will be able to correct for these errors, resulting in a quantum error correcting code and operation that performs σ_x with fault-tolerance.

3.2 Fault Tolerance of σ_z

By using the Steane code above, the equivalent of σ_z on an unencoded qubit is to apply σ_z to those individual qubits with a 1 in the codeword w where $w \in C_2^{\perp} - C_1^{\perp}$. This results in a state $|a\rangle$ acquiring a phase of $(-1)^{a.w}$. Under such a transformation the code words become

$$\frac{1}{\sqrt{|C_2|}} \sum_{x \in C_2} |x\rangle \longrightarrow \frac{1}{\sqrt{|C_2|}} \sum_{x \in C_2} (-1)^{x.w} |x\rangle = \frac{1}{\sqrt{|C_2|}} \sum_{x \in C_2} |x\rangle \tag{4}$$

$$\frac{1}{\sqrt{|C_2|}}\sum_{x\in C_2}|x+v\rangle \longrightarrow \frac{1}{\sqrt{|C_2|}}\sum_{x\in C_2}(-1)^{x.w}(-1)^{v.w}|x+v\rangle = -\frac{1}{\sqrt{|C_2|}}\sum_{x\in C_2}|x+v\rangle \tag{5}$$

since at least 1 vector in C_1 gives $v.w = 1$.

3.3 Fault Tolerance of Hadamard Gate

The fault tolerance of the Hadamard gate under this CSS encoding can be seen under the additional constraint $C_1 = C_2^{\perp}$. If the function $E(x)$ represents the act of encoding the bit, the action of a Hadamard on an encoding qubit must follow the transformations:

$$E(|0\rangle) \longrightarrow \frac{1}{\sqrt{2}}(E(|0\rangle) + E(|1\rangle)) \tag{6}$$

$$E(|1\rangle) \longrightarrow \frac{1}{\sqrt{2}}(E(|0\rangle) - E(|1\rangle)) \tag{7}$$

A Hadamard transformation of each individual qubit, $H^{\otimes k}$, applied to $E(|0\rangle)$ will give precisely the correct encoded transformation.

$$\begin{aligned} H\frac{1}{\sqrt{|C_2|}}\sum_{x\in C_2}|x\rangle &= \frac{1}{2^{\frac{k}{2}}\sqrt{|C_2|}}\sum_{y,x\in C_2}(-1)^{x.y}|x\rangle \\ &= \frac{1}{\sqrt{|C_2^{\perp}|}}\sum_{y\in C_1}|y\rangle \\ &= E(\frac{|0\rangle+|1\rangle}{\sqrt{2}}) \end{aligned}$$

The last line follows because $E(|0\rangle)$ is composed of all codeword in C_2 and $E(|1\rangle)$ is everything in C_1, but not in C_2 by definition. A Hadamard transformation on $E(|1\rangle)$, simply adds in the phase factor of $(-1)^{y.v}$, which obviously follows from above. This phase factor will be unity if $y \in C_2$, but -1 if $y \in (C_1 - C_2)$, thus appropriately adding a phase to the states in the code that are $E(|1\rangle)$.

3.4 Fault Tolerance of CNOT Gate

The σ_x, σ_z, and H gates can all be performed on a single encoded qubit with fault-tolerance because these gates are always applied to single qubits. Likewise, given two single-qubit encoded states, one can perform CNOT operations between the k^{th} qubit of one set, with the k^{th} qubit of the other. Thus there are at most two qubits interacting for a single gate, making errors independent among the sets of qubits, and thus correctible with the CSS Code. This can be shown as follows:

$$\begin{aligned} U_{CNOT}^{\otimes k}\frac{1}{|C_2|}\sum_{x\in C_2}|x+v_a\rangle\otimes\sum_{y\in C_2}|y+v_b\rangle &= \frac{1}{|C_2|}\sum_{x\in C_2}|x\rangle\otimes\sum_{y\in C_2}|x+y+v_a+v_b\rangle \\ &= \frac{1}{|C_2|}\sum_{x\in C_2}|x\rangle\otimes\sum_{y\in C_2}|y+v_a+v_b\rangle \end{aligned}$$

If $v_a = v_b$, then $v_a + v_b = 0$ in binary addition and the vector is unchanged. Otherwise, the resulting state will be a string of 1's added to all states $|y\rangle$, which is just the encoding $E(|1\rangle)$. If an error occurs in any of the two-qubit CNOT operations, this will result in v_a or v_b not being all 0's or all 1's, and the CSS code will correct the the appropriate state.

4 Error Correction With Fault-Tolerant Precision

Both classical and quantum error correction schemes require encoding information into a code, computing the syndrome of the code after errors may have occurred, then applying a syndrome-dependent correction step to the coding to recover the information. A simple means of checking the syndrome would be to find the parity of a subset of qubits in a code. One could thus perform a series of CNOT gates where a single target qubit will be flipped depending on the states of the controlled qubits. Measurement of this ancilla qubit would unveil the syndrome, but not with fault-tolerant precision.

This can easily be seen under a Hadamard transformation, $H^{\otimes k+1}$, which reverses the direction of the CNOT gates and gives the dual CSS code in the $\mathcal{H}^\perp$ space. ($\mathcal{H}$ is the parity check matrix of the code C_1.) Due to the reversed CNOT, if any of the gates have an error, the error will propagate forward in time due to the back-action of the CNOT. The stringent requirement of each error not affecting more than a single qubit (or pair in the FT CNOT construction) is not fulfilled.

Using the idea of single failure points between qubits, as seen in the FT CNOT construction, we start our parity check register on k qubits in the state:

$$|\psi\rangle = \frac{1}{2^{k-1}} \sum_{s \in \mathbb{Z}_2^k} |s\rangle \tag{8}$$

Measurement of $|\psi\rangle$ in the canonical basis will result in either an odd parity bit string, indicating a syndrome of 1, or an even parity string for a 0 syndrome. The state $|\psi\rangle$ can be created by applying the Hadamard transformation to the "cat" state.

$$H^{\otimes k} |\psi\rangle = \frac{1}{\sqrt{2}}(|\mathbf{0}\rangle + |\mathbf{1}\rangle) \tag{9}$$

(The state $|\mathbf{x}\rangle$ represents a k length string of $\mathbf{x}$'s.)

Now suppose a maximally entangled state can be created and verified by performing a few CNOT gates between the bits of the cat state and an ancilla. Fault tolerance is not an issue here; one only wants to know if the state is maximally entangled. Measuring $|0\rangle$ on the ancilla for a reasonable number of test qubits ensures that the state is in some superposition of the states $|\mathbf{0}\rangle$ and $|\mathbf{1}\rangle$. The Hadamard transform of this state:

$$H^{\otimes k}(\alpha |\mathbf{0}\rangle + \beta |\mathbf{1}\rangle) = (\frac{\alpha+\beta}{\sqrt{2}})\frac{1}{2^{k-1}} \sum_{s \in even} |s\rangle + (\frac{\alpha-\beta}{\sqrt{2}})\frac{1}{2^{k-1}} \sum_{s \in odd} |s\rangle \tag{10}$$

Thus if $\alpha = \beta$, the state is all zeros and no backaction will occur. The all ones state simply adds the ones vector to the qubits. Thus, fault-tolerant measurements can be obtained.