Quantum Error Correction

Peter Young

(Dated: March 6, 2020)

I. INTRODUCTION

Quantum error correction has developed into a huge topic, so here we will only be able to describe the main ideas.

Error correction is essential for quantum computing, but appeared at first to be impossible, for reasons that we shall see. The field was transformed in 1995 by Shor[1] and Steane[2] who showed that quantum error correction *is* feasible. Before Shor and Steane, the goal of building a useful quantum computer seemed clearly unattainable. After those two papers, while building a quantum computer obviously posed enormous challenges, it was not *necessarily* impossible.

Some general references on quantum error correction are Refs. [3–6].

Let us start by giving a simple discussion of classical error correction which will motivate our study of quantum error correction. In classical computers error correction is not necessary. This is because the hardware for one bit is huge on an atomic scale and the states 0 and 1 are so different that the probability of an unwanted flip is tiny. However, error correction is needed classically for transmitting a signal over large distances where it attenuates and can be corrupted by noise.

To do error correction one needs to have redundancy. One simple way of doing classical error correction is to encode each *logical* bit by three *physical* bits, i.e.

$$|0\rangle \to |\bar{0}\rangle \equiv |0\rangle|0\rangle|0\rangle \equiv |000\rangle\,, \tag{1a}$$

$$|1\rangle \to |\bar{1}\rangle \equiv |1\rangle|1\rangle|1\rangle \equiv |111\rangle\,. \tag{1b}$$

The sets of three bits, $|000\rangle$ and $|111\rangle$, are called *codewords*. One monitors the codewords to look for errors. If the bits in a codeword are not all the same one uses "majority rule" to correct. For example

$$\begin{aligned} &|010\rangle \text{ is corrected to } |000\rangle \\ &|110\rangle \text{ is corrected to } |111\rangle. \end{aligned} \tag{2}$$

This works if no more than one bit is corrupted and so the error rate must be sufficiently low that the probability of two or more bits in a codeword being corrupted is negligible.

In quantum error correction one also uses multi-qubit codewords and monitoring. However, there are several major differences compared with classical error correction:

(i) *Error correction is essential.* Quantum computing requires error correction. This is because the physical systems for a single qubit are very small, often on an atomic scale, so any small outside interference can disrupt the quantum state.

(ii) *Measurement destroys quantum information.* In contrast to the classical case checking for errors is problematic. Monitoring means measuring, and measuring a general quantum state alters it. Thus it seems that any attempt at error correction must destroy important quantum information.

(iii) *More general types of error can occur.* Bit flips are not the only possible errors. For example one can have phase errors where $\frac{1}{\sqrt{2}}(|0\rangle + |1\rangle) \rightarrow \frac{1}{\sqrt{2}}(|0\rangle + e^{i\phi}|1\rangle)$.

(iv) *Errors are continuous.* Unlike all-or-nothing bit flip errors for classical bits, errors in qubits can grow continuously out of the uncorrupted state.

One might imagine that point (ii), in particular, would be fatal. Amazingly this is not so as we shall see.

II. CORRECTING BIT FLIP ERRORS

We start our discussion of quantum error correction by considering how one might be able to correct just for bit flip errors. If the error rate is low we might hope to correct them by tripling the number of bits as in the classical case, Eq. (1).

The tripling of the qubits can be accomplished by the circuit in Fig. 1. To see how this works suppose that the input qubit, $|x\rangle$, is $|0\rangle$. Then none of the CNOT gates act on their target qubit so all three qubits are $|0\rangle$ at the end (i.e. on the right). However, if the input qubit $|x\rangle$ is $|1\rangle$ then the CNOT gates act so all three qubits are 1 at the end.

By linearity a linear combination of $|0\rangle$ and $|1\rangle$ is transformed as we want:

$$\alpha|0\rangle + \beta|1\rangle \rightarrow \alpha|000\rangle + \beta|111\rangle\,, \tag{3}$$

see Fig. 2. Note that this is not a clone of the input state which would be

$$(\,\alpha|0\rangle + \beta|1\rangle\,)^{\otimes 3} = \alpha^3|000\rangle + \alpha^2\beta\,(\,|001\rangle + |010\rangle + |100\rangle\,) + \alpha\beta^2\,(\,|110\rangle + |101\rangle + |011\rangle\,) + \beta^3|111\rangle\,. \tag{4}$$

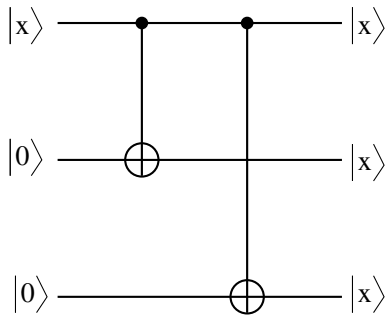

FIG. 1: Circuit to encode the 3-qubit bit-flip code. Here $|x\rangle$ is $|0\rangle$ or $|1\rangle$ in the computational basis. The effect of this circuit on a linear combination of $|0\rangle$ and $|1\rangle$ is shown in Fig. 2.

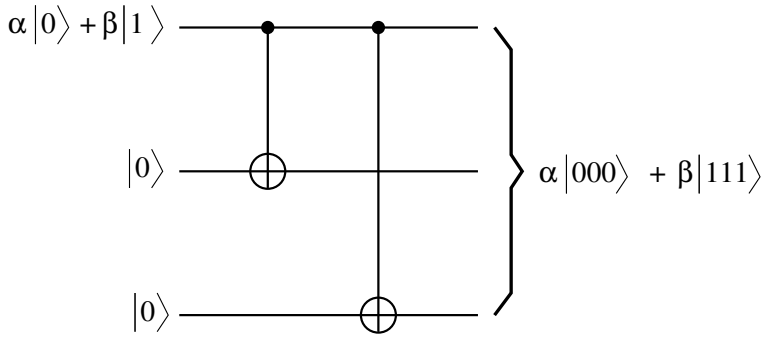

FIG. 2: Circuit to encode the 3-qubit bit-flip code acting on a linear combination of $|0\rangle$ and $|1\rangle$.

We recall that cloning an arbitrary unknown state is impossible according to the no-cloning theorem.

Next an aside on notation. The CNOT gate is usually written with a $\oplus$ symbol (as in Figs. 1 and 2) to indicate the XOR operation but it is often more illuminating to write it, in an equivalent way, with an X symbol (in a square) to indicate that the NOT (i.e. bit-flip) operation is performed with the operator X (recall that in a CNOT gate the target qubit is flipped if the control qubit is 1.) From now on in this handout we shall use the symbol X inside a square when drawing a CNOT gate, see e.g. Fig. 3.

Now we have to check if any of the three qubits generated by the circuit in Fig. 2 are flipped so the situation is that shown in Fig. 3. We assume that no more than one has been flipped, which is a reasonable approximation if the error rate is small.

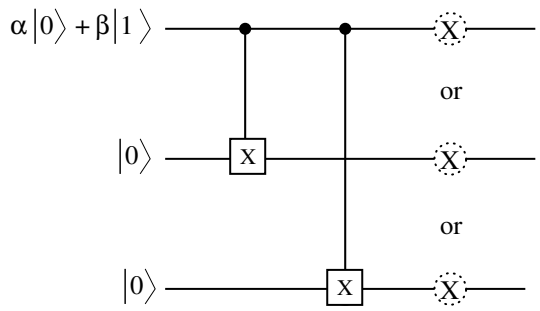

FIG. 3: Circuit indicating that at most one of the three bits generated by the circuit in Fig. 2 has flipped due to an error. The goal will be to determine whether any have flipped, if so which one, and then correct the error. Note that the Control-X gates here are identical to the CNOT gates in Figs. 1 and 2. (Control-X and CNOT are just different ways of describing the same gate.)

We have therefore one uncorrupted state and three corrupted states:

$$|\,\psi\,\rangle = \alpha|000\rangle + \beta|111\rangle\,, \tag{5a}$$

$$|\psi_1\rangle = \alpha|100\rangle + \beta|011\rangle \quad \text{(qubit 1 flipped)}\,, \tag{5b}$$

$$|\psi_2\rangle = \alpha|010\rangle + \beta|101\rangle \quad \text{(qubit 2 flipped)}\,, \tag{5c}$$

$$|\psi_3\rangle = \alpha|001\rangle + \beta|110\rangle \quad \text{(qubit 3 flipped)}\,. \tag{5d}$$

These four states are called the "syndromes". Note that we denote the left hand qubit as the first qubit, the one to its right as the second qubit, and so on, e.g. $|x_1x_2x_3\rangle$. Hence in Eq. (5) $|\psi_i\rangle$ refers to the state in which qubit i is flipped relative to the uncorrupted state $|\psi\rangle$.

Classically, to determine if one of the bits is flipped we just have to look at them. However, quantum mechanically, if we measure $|\psi\rangle$, say, we get $|000\rangle$ with probability $|\alpha|^2$ and $|111\rangle$ with probability $|\beta|^2$, which destroys the coherent superposition. It might therefore seem that quantum error correction is impossible.

Amazingly this is not so. The secret is to couple the codeword qubits to ancillary qubits and measure only these. This will give enough information to determine which syndrome the state is in *without destroying the coherent superposition.*

Here we need two ancillary qubits. The circuit including them is shown in Fig. 4. The three codeword qubits are at the bottom and the ancillary qubits are at the top. The ancillary qubits are measured and give values x and y. We shall now see that each of the four possible pairs of values for x and y corresponds to one of the syndrome states in Eq. (5).

Both ancillas are targeted by two of the codeword qubits.

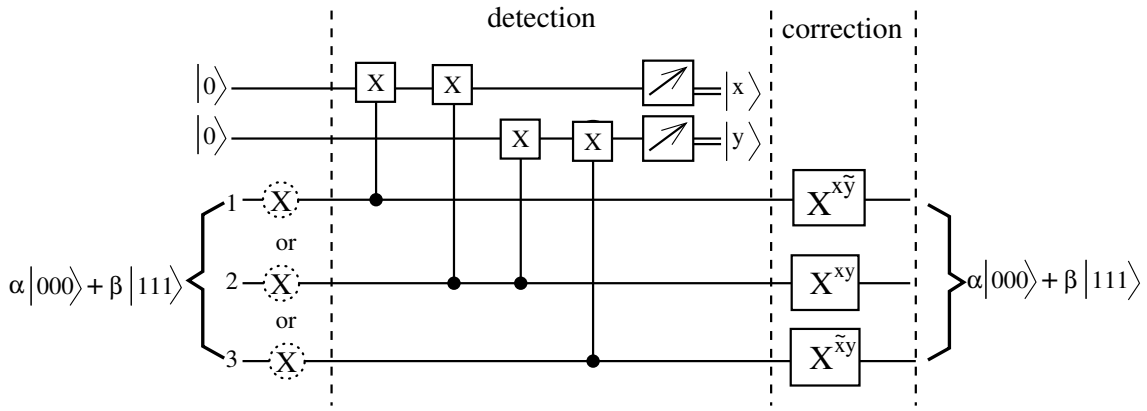

FIG. 4: Circuit to determine the syndrome for the 3-qubit bit-flip code, and correct if necessary. A box with an arrow denotes a measurement. The double lines indicate that the result of a measurement is a classical bit.

1st (upper) ancilla (x) is targeted by codeword qubits 1 and 2.

2nd (lower) ancilla (y) is targeted by codeword qubits 2 and 3.

Let's see what happens for the four syndrome states.

$|\psi\rangle$ Codeword $|000\rangle$. No ancilla flipped so $x = 0, y = 0$.

Codeword $|111\rangle$. Both ancillas are flipped twice so again $x = 0, y = 0$.

Note that the result of the measurement is the same for both the $|000\rangle$ and $|111\rangle$ parts of the state $|\psi\rangle$. Hence the coherent superposition of $|\psi\rangle$ is not destroyed by the measurement on the ancillas.

$|\psi_1\rangle$ Codeword $|100\rangle$. x is flipped once, and y is not flipped, so $x = 1, y = 0$.

Codeword $|011\rangle$. x is flipped once and y is flipped twice so again $x = 1, y = 0$.

Recall that the qubits are ordered such that qubit 1 is on the right.

$|\psi_2\rangle$ Codeword $|010\rangle$. x and y are both flipped once so $x = 1, y = 1$.

Codeword $|101\rangle$. x and y are both flipped once so again $x = 1, y = 1$.

$|\psi_3\rangle$ Codeword $|001\rangle$. x is not flipped and y is flipped once so $x = 0, y = 1$.

Codeword $|110\rangle$. x is flipped twice and y is flipped once so again $x = 0, y = 1$.

Hence we get the table of results shown in Table I. Note that in all cases the coherent superposition of the syndrome state is not destroyed by the measurement of the ancillas.

syndrome	bit flipped	x	y
$\vert\,\psi\,\rangle$	none	0	0
$\vert\psi_1\rangle$	1	1	0
$\vert\psi_2\rangle$	2	1	1
$\vert\psi_3\rangle$	3	0	1

TABLE I: Results of measurement of the ancillary qubits for the different syndromes of the codeword qubits.

Hence by measuring the auxiliary qubits we can determine which if any of the codeword qubits have flipped and then apply a compensating flip if necessary. The X-gates which perform these compensating flips are shown at the right of Fig. 4. For example the $X^{x\tilde{y}}$ gate on qubit 1 indicates that a flip is done by acting with the X operator only if $x = 1$ and $y = 0$, which corresponds to the appropriate entry in the Table I. ($\tilde{y}$ means the complement of y.)

We have assumed up to now that the state of the system has had a bit flipped with probability one. However, as already noted, errors in quantum circuits can arise continuously from zero, and we are concerned with the situation in which the error rate is small (otherwise we can not error correct). Consider therefore a state $|\psi\rangle$ which has a small amplitude less than one to have a bit flipped, i.e.

$$|\psi\rangle \rightarrow [1 + i(\,\epsilon_1 X_1 + \epsilon_2 X_2 + \epsilon_3 X_3\,)]\,|\psi\rangle\,, \tag{6}$$

where $\epsilon_k \ll 1$, and we have only indicated terms to first order in the ϵ_k (the factor of i is needed so the state is normalized to first order in the ϵ_k). The probability of qubit k being flipped is $|\epsilon_k|^2$ to leading order. When we measure the ancilla qubits we project on to either the uncorrupted state or one of the three corrupted states which have one qubit flipped.

Since the ϵ_k are small, the *probability* that a corrupted state is detected is small, so the most probable situation is that no correction is needed. However, there is a small probability that the projection will be on to one of the corrupted syndromes. The corrupted syndromes differ *substantially* from the uncorrupted state. They are further, in fact, from the uncorrupted state than the original state in Eq. (6). This might, at first, seem like a retrograde step but it is not because the corrupted state is known *precisely* so it is possible to correct it back to to the uncorrupted state.

To summarize this part, quantum error correction is feasible, even though errors arise continuously, because possibly corrupted states are projected on to one of a *discrete* set of states which can be corrected if necessary. We will discuss this important point again in Sec. V when we consider

how general errors arise.

It should be noted that in classical analog computers, where errors also arise continuously, no such projection can be done, and hence error correction can not be performed. This is why we don't have classical analog computers.

Going back to the discussion of Fig. 4, one can avoid explicitly measuring the qubits and instead coherently and automatically correct any bit-flip error by having the ancillas interact back on the codeword qubits as shown in Fig. 5. In that figure, the rightmost three controlled gates have the same effect as the NOT gates in the right of Fig. 4 which depend on the result of measurements of the x and y ancillary qubits. The rightmost gate in Fig. 5 has two control qubits and three target qubits. This gate flips all the target qubits if *both* control qubits are 1. It is a generalization of the Toffoli gate T which has two control qubits, and one target qubit which is flipped if both control qubits are 1, i.e. $T|x\rangle|y\rangle|z\rangle = |x\rangle|y\rangle|z \oplus x\,y\rangle$. If we denote by T^* the rightmost gate in Fig. 5 then $T^*|x\rangle|y\rangle|z\rangle|u\rangle|v\rangle = |x\rangle|y\rangle|z \oplus x\,y\rangle|u \oplus x\,y\rangle|v \oplus x\,y\rangle$. Note that this gate is equivalent to three separate Toffoli gates, in which the two ancilla qubits are the controls, qubit 1 is the target for the first Toffoli, qubit 2 for the second Toffoli, etc. In Vathsan's book[5], Fig. 10.5 shows a set of gates which is equivalent to, but different from, the rightmost three in Fig. 5, including a zero-controlled CNOT gate (indicated by the open circle in her Fig. 10.5). After the error on the computational bits has been corrected the ancilla qubits have to be reinitialized to zero.

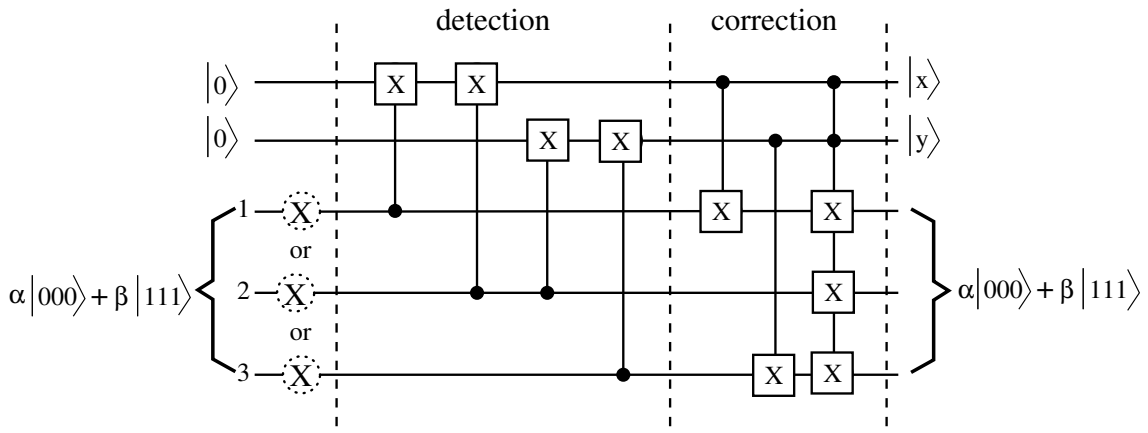

FIG. 5: Automation of the error correction procedure of Fig. 4. The three controlled gates on the right have the same effect as the NOT gates on the right of Fig. 4 which depend on the result of measurements of the x and y ancillary qubits. The rightmost gate, with two control qubits and three target qubits, is discussed in the text. The values of the control bits x and y at the end depend on which of the four syndromes is present (i.e. which if any of the X gates on the left of the figure have acted) according to Table I.

It is instructive to show for the different syndromes in Eq. (5) that the circuits in Figs. 4 and 5 give the same result, i.e. the end product is the uncorrupted state $|\psi\rangle$. The results from the circuit of Fig. 4 have already been discussed above. For the circuit in Fig. 5 we just consider the case of $|\psi_2\rangle$ (so the X gate on codeword qubit 2 on the left has acted), and we have $x = 1, y = 1$ according to Table I. Consider the rightmost three gates in Fig. 5 (these are the ones that do the error correction). For $x = 1, y = 1$, the rightmost gate is active and flips all three codeword qubits. Hence, between them, the rightmost three gates flip codeword qubit 1 twice, flip codeword qubit 2 once, and flip codeword qubit 3 twice. The net result is that only codeword qubit 2 is flipped so we recover the uncorrupted state $|\psi\rangle$. It is useful to check that the circuit in Fig. 5 also works to correct $|\psi_1\rangle$ and $|\psi_3\rangle$.

III. STABILIZER FORMALISM

In order to conveniently generalize the ideas in the previous section to arbitrary errors we need to reformulate them.

For reasons that will shortly become clear, consider the two Hermitian[1] operators Z_1Z_2 and Z_2Z_3. Because $Z_i^2 = \mathbb{1}$ (the identity) and different $Z's$ commute we have

$$(Z_1Z_2)^2 = \mathbb{1}, \quad (Z_2Z_3)^2 = \mathbb{1}\,. \tag{7}$$

An operator whose square is unity has eigenvalues equal to ± 1, since acting twice with the operator on an eigenvector gives the eigenvector, so the square of the eigenvalue is 1. We also we know that Z_1Z_2 and Z_2Z_3 commute with each other.

[1] It is an axiom of quantum mechanics that measurable quantities are represented by Hermitian operators. The motivation for this is that the eigenvalues of Hermitian operators are real, just as the results of a measurement must be.

syndrome		Z_1Z_2	Z_2Z_3	x	y
$\|\psi\rangle$		1	1	0	0
$\|\psi_1\rangle$	$X_1\|\psi\rangle$	-1	1	1	0
$\|\psi_2\rangle$	$X_2\|\psi\rangle$	-1	-1	1	1
$\|\psi_3\rangle$	$X_3\|\psi\rangle$	1	-1	0	1

TABLE II: The eigenvalues of the operators Z_1Z_2 and Z_2Z_3 for the four syndromes for the 3-qubit bit-flip code, and a comparison with the measurements of the ancillary qubits x and y, see Table I. The uncorrupted state has eigenvalue $+1$ for both stabilizers. This is an important property that stabilizers must have in general. Note that $Z_1Z_2 = 1$ corresponds to $x = 0$, and $Z_1Z_2 = -1$ corresponds to $x = 1$. There is a similar connection between Z_2Z_3 and y, so $Z_1Z_2 = (-1)^x, Z_2Z_3 = (-1)^y$. The second column shows how the corrupted state is generated from the uncorrupted state.

One can verify that the syndrome states in Eq. (5) are eigenvectors of Z_1Z_2 and Z_2Z_3 according to Table II. In general we use the term "stabilizers" to denote operators like operators Z_1Z_2 and Z_2Z_3 whose eigenvalues distinguish the different syndromes.

There is a more straightforward way to determine whether the eigenvalue of a stabilizer in a syndrome is $+1$ or -1 than acting with the stabilizer on the syndrome.

We note first that the eigenvalue of all the stabilizers is $+1$ in the uncorrupted syndrome $|\psi\rangle$. This is an essential property that stabilizers must have.

Also note that the operators for the stabilizers will be built out of the single-qubit operators Z_i and X_i. For the 3-qubit, bit-flip code we only have the Z_i but the X_i will also be needed to correct for general errors. Furthermore the syndromes with a single qubit error are obtained by acting on the uncorrupted syndrome with the X_i, Y_i and Z_i operators.[2] Again, for our simple example above, we only had the X_i, but the other operators will be used when we deal with general errors.

The operators, X_i, Y_i, Z_i, have the property that they commute for different qubits i, and anti-commute for the same qubit, e.g.

$$[X_i, Y_j] \equiv X_iY_j - Y_jX_i = 0 \quad (i \neq j)\,, \tag{8a}$$

$$\{X_i, Y_i\} \equiv X_iY_i + Y_iX_i = 0\,. \tag{8b}$$

(Check Eq. (8b) by explicitly working out some cases.)

---

[2] Recall that the Pauli operators X, Y and Z are given by $X \equiv \sigma^x = \begin{pmatrix} 0 & 1 \\ 1 & 0 \end{pmatrix}, Y \equiv \sigma^y = \begin{pmatrix} 0 & -i \\ i & 0 \end{pmatrix}, Z \equiv \sigma^z = \begin{pmatrix} 1 & 0 \\ 0 & -1 \end{pmatrix}$ and so $iY = ZX$.

Hence if we consider a general stabilizer A_α and a syndrome state $|\psi_\beta\rangle = B_\beta|\psi\rangle$ then A_α either commutes or anti-commutes with B_β.

We will now show that if A_α commutes with B_β the eigenvalue of the stabilizer A_α in state $|\psi_\beta\rangle$ is $+1$ and if they anti-commute the eigenvalue is -1. Firstly, if A_α commutes with B_β then

$$A_\alpha|\psi_\beta\rangle = A_\alpha B_\beta|\psi\rangle = B_\beta A_\alpha|\psi\rangle = B_\beta|\psi\rangle = |\psi_\beta\rangle\,, \tag{9}$$

where we used that the eigenvalues of all the stabilizers A_α are $+1$ in the uncorrupted state $|\psi\rangle$ to get the second equality. Hence the eigenvalue of A_α in state $|\psi_\beta\rangle$ is $+1$ if A_α commutes with B_β. Similarly if A_α anti-commutes with B_β then

$$A_\alpha|\psi_\beta\rangle = A_\alpha B_\beta|\psi\rangle = -B_\beta A_\alpha|\psi\rangle = -B_\beta|\psi\rangle = -|\psi_\beta\rangle\,, \tag{10}$$

so the eigenvalue is -1. In Eqs. (9) and (10) we have used that the uncorrupted state has eigenvalue $+1$ for all stabilizers so $A_\alpha|\psi\rangle = |\psi\rangle$.

Next we will see how to determine if a stabilizer commutes or anticommutes with the operator which generates a corrupted syndrome out of the uncorrupted state.

For the case of the 3-qubit, bit-flip code discussed so far the stabilizers are

$$Z_1Z_2 \text{ and } Z_2Z_3\,, \tag{11}$$

and the operators which generate the corrupted syndrome from the uncorrupted state are

$$X_1, X_2 \text{ and } X_3. \tag{12}$$

As an example, we see that X_1 commutes with Z_2Z_3 because there are no sites in common, so the eigenvalue of Z_2Z_3 for $|\psi_1\rangle$ must be $+1$ which agrees with Table II. On the other hand X_2 has one site in common with Z_2Z_3 so

$$X_2\, Z_2Z_3 = -Z_2X_2Z_3 = -Z_2Z_3X_2\,, \tag{13}$$

and the operators anticommute, so the eigenvalue of Z_2Z_3 for $|\psi_2\rangle$ must be -1, which again agrees with Table II. Every time we have to interchange the order of two different operators acting on the same qubit we pick up a minus sign. Hence it is straightforward to deduce the overall sign. Note that operators of the same type, e.g. the Z_i, always commute.

As a more complicated example, which occurs in a scheme for full error correction, a stabilizer is $Z_3X_4X_5Z_1$. For the state which is corrupted by Z_4 the eigenvalue is -1, the minus sign coming from interchanging the order of X_4 and Z_4. However, for the state which is corrupted by X_4 the

eigenvalue is +1 since, for the qubit in common, i.e. 4, both operators are X and so commute. As another example, for the state which is corrupted by X_2 the eigenvalue is +1, because X_2 and the stabilizer commute since they have no qubits in common.

To summarize, the stabilizer formalism we need to construct a mutually commuting set of Hermitian operators (the stabilizers) which square to 1 and for which (i) the syndromes are eigenstates, (ii) the uncorrupted syndrome has eigenvalue +1 for all stabilizers, and (iii) the set of ±1 eigenvalues of the stabilizers uniquely specifies the syndrome. In Sec. VI we will describe an example with full error correction which has codewords with 9 qubits and needs 8 stabilizers.

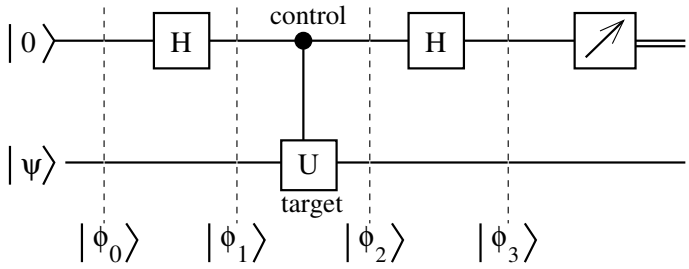

FIG. 6: A circuit with a control-U gate in which the control (upper) qubit is surrounded by Hadamards. U is an operator with eigenvalues ±1 and corresponding eigenvectors $|\psi_+\rangle$ and $|\psi_-\rangle$. As shown in the text, if a measurement of the upper qubit gives $|0\rangle$ then the lower qubit will be in state $|\psi_+\rangle$, and if the measurement gives $|1\rangle$ then the lower qubit will be in state $|\psi_-\rangle$. The states $|\phi_i\rangle$ $(i = 0, 1, 2, 3)$ are described in the text.

In terms of stabilizers, what circuit which will determine which syndrome has occurred? Consider the circuit in Fig. 6 which includes a control-U gate where U is an operator, which, like the stabilizers, has eigenvalues ±1. If the control qubit is 1 the effect on the target qubit is

$$U|\psi_+\rangle = |\psi_+\rangle, \quad U|\psi_-\rangle = -|\psi_-\rangle, \tag{14}$$

where $|\psi_+\rangle$ and $|\psi_-\rangle$ are the eigenvectors with eigenvalue +1 and −1 respectively. If the control qubit is 0 then the target qubit is unchanged.

The lower (target) qubit on the left can be written as a linear combination of the two eigenvectors

$$|\psi\rangle = \alpha_+|\psi_+\rangle + \alpha_-|\psi_-\rangle\,, \tag{15}$$

and so, including the upper (control) qubit which is initially in state $|0\rangle$, the state of the circuit on the left of Fig. 6 is

$$|\phi_0\rangle = \alpha_+|0\,\psi_+\rangle + \alpha_-|0\,\psi_-\rangle\,. \tag{16}$$

Figure 6 includes Hadamards so we remind ourselves of the effects of a Hadamard gate:

$$H|0\rangle = \frac{1}{\sqrt{2}}\left(\,|0\rangle + |1\rangle\,\right), \quad H\left[\frac{1}{\sqrt{2}}\left(\,|0\rangle + |1\rangle\,\right)\right] = |0\rangle, \tag{17}$$

$$H|1\rangle = \frac{1}{\sqrt{2}}\left(\,|0\rangle - |1\rangle\,\right), \quad H\left[\frac{1}{\sqrt{2}}\left(\,|0\rangle - |1\rangle\,\right)\right] = |1\rangle. \tag{18}$$

After the first Hadamard on the upper qubit the state is

$$|\phi_1\rangle = \frac{\alpha_+}{\sqrt{2}}\left(\,|0\,\psi_+\rangle + |1\,\psi_+\rangle\,\right) + \frac{\alpha_-}{\sqrt{2}}\left(\,|0\,\psi_-\rangle + |1\,\psi_-\rangle\,\right)\,. \tag{19}$$

The effect of the control-U gate on the target qubit is given by Eq. (14) when the control qubit is 1 and has no effect if the control qubit is 0. Hence, after the control-U gate, the state is

$$|\phi_2\rangle = \frac{\alpha_+}{\sqrt{2}}\left(\,|0\,\psi_+\rangle + |1\,\psi_+\rangle\,\right) + \frac{\alpha_-}{\sqrt{2}}\left(\,|0\,\psi_-\rangle - |1\,\psi_-\rangle\,\right)\,. \tag{20}$$

Applying the second (rightmost) Hadamard to the upper qubit we get

$$|\phi_3\rangle = \alpha_+|0\,\psi_+\rangle + \alpha_-|1\,\psi_-\rangle\,. \tag{21}$$

Hence if a measurement of the upper qubit gives $|0\rangle$ (which it does with probability $|\alpha_+|^2$) the lower qubit will be in state $|\psi_+\rangle$, and if the measurement gives $|1\rangle$ (probability is $|\alpha_-|^2$) the lower qubit will be in state $|\psi_-\rangle$. We see that measuring the control qubit tells us which eigenstate of U the target qubit is in.

Stabilizers involve more than one codeword qubit so the gates we need will have several target qubits. For the 3-qubit, bit-flip code, the circuit equivalent to Fig. 4 is shown in Fig. 7.

The equivalence of the circuits in Figs. 4 and 7 can also be understood from the simpler case of the equivalences shown in Fig. 8 in which the left-hand equality comes from the fact that the target and control qubits can be exchanged in a control-Z gate,[3] and the right-hand equality is because $HZH = X$ and $H^2 = \mathbb{1}$ (the identity).

The stabilizer formalism will be convenient when devising circuits for full error correction rather than just correcting bit flips as we have done up to now.

IV. PHASE FLIP CODE

Before discussing how to correct general errors, we will briefly mention another special case, a phase flip, which has no classical equivalent since classical bits don't have any property corresponding to phase. In this error model, with some probability p, the relative phase of $|0\rangle$ and $|1\rangle$

[3] Because the only effect of the gate is to change the sign of the state if both target and control qubits are 1.

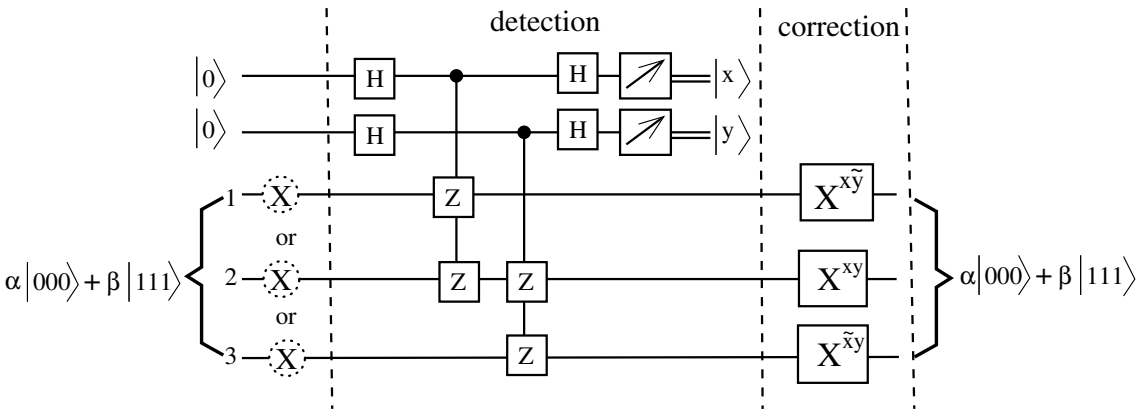

FIG. 7: Circuit equivalent to that in Fig. 4 but in the stabilizer formalism. In this circuit x measures Z_1Z_2, and y measures Z_2Z_3. In other words, if $x = 0$ the state of the codeword bits has $Z_1Z_2 = +1$, whereas if $x = 1$ the state of the codeword bits has $Z_1Z_2 = -1$, with an analogous correspondence between y and Z_2Z_3. Note that Z_1Z_2 and Z_2Z_3 have eigenvalues ± 1 and commute with each other.

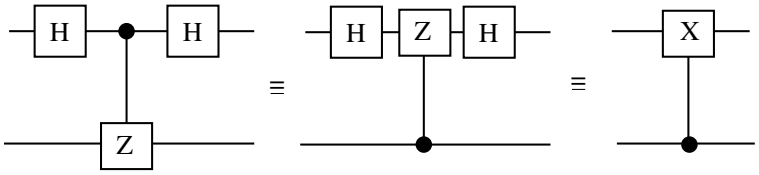

FIG. 8: The equalities in this figure are helpful to understand the equivalence of Figs. 4 and 7. The left-hand equality comes from the fact that the target and control qubits can be exchanged in a control-Z gate, and the right-hand equality is because $HZH = X$ and $H^2 = \mathbb{1}$.

is flipped so

$$|\psi\rangle = \alpha|0\rangle + \beta|1\rangle \;\rightarrow\; \alpha|0\rangle - \beta|1\rangle\,. \tag{22}$$

Equivalently

$$\begin{pmatrix}\alpha\\ \beta\end{pmatrix} \rightarrow Z\begin{pmatrix}\alpha\\ \beta\end{pmatrix} = \begin{pmatrix}\alpha\\ -\beta\end{pmatrix} \quad \text{(computational basis)}. \tag{23}$$

The phase-flip error model can be turned into the already-studied bit-flip model by transforming to the $\pm$ basis[4] where

$$|+\rangle = \frac{1}{\sqrt{2}}\left(|0\rangle + |1\rangle\right), \quad |-\rangle = \frac{1}{\sqrt{2}}\left(|0\rangle - |1\rangle\right), \tag{24}$$

[4] One could also call the computational basis the Z-basis, and the $\pm$ basis the X-basis, since Z is diagonal in the computational basis and X is diagonal in the $\pm$ basis.

One transforms between the $\pm$ basis and the computational basis using Hadamards:

$$H|0\rangle = |+\rangle, \quad H|1\rangle = |-\rangle, \tag{25a}$$

$$H|+\rangle = |0\rangle, \quad H|-\rangle = |1\rangle. \tag{25b}$$

In the $\pm$ basis the roles of X and Z are interchanged since

$$X|0\rangle = |1\rangle, \quad X|1\rangle = |0\rangle, \quad Z|0\rangle = |0\rangle, \quad Z|1\rangle = -|1\rangle, \tag{26a}$$

$$Z|+\rangle = |-\rangle, \quad Z|-\rangle = |+\rangle, \quad X|+\rangle = |+\rangle, \quad X|-\rangle = -|-\rangle. \tag{26b}$$

Thus we shall find in Sec. VI that stabilizers to detect phase errors involve X operators, as opposed to those used to detect bit-flip errors which involve Z operators (see Fig. 7).

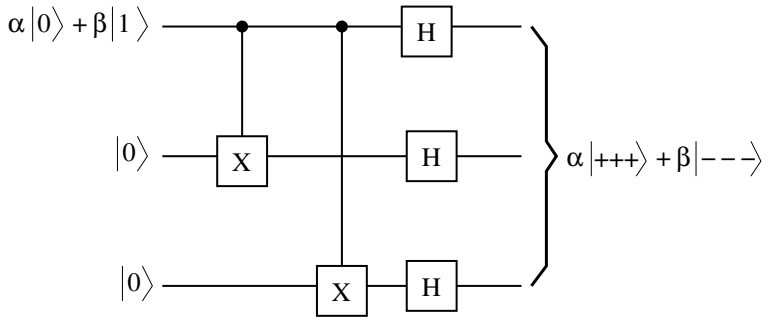

FIG. 9: Encoding circuit for the 3-qubit phase-flip code.

We transform to the $\pm$ basis by adding Hadamards to the encoding circuit for the 3-qubit bit-flip code of Fig. 2, with the result shown in Fig. 9. We shall use this circuit in Sec. VI as part of the encoding circuit in Fig. 10 for a code (due to Shor) which corrects general 1-qubit errors.

V. THE PHYSICS OF ERROR CORRECTION (DISCRETIZATION OF ERRORS)

In our discussion of errors we have so far implicitly assumed that the errors occur because of some malfunction in the circuit. The state has underdone a unitary transformation, but not exactly the right one. Another, and very important, source of error is interaction between the qubits and the environment, which is unavoidable even though quantum computer engineers work very hard to reduce it to a minimum. This can lead to errors due to a *non-unitary* change in the computational qubits (though the combined system of qubits plus environment undergoes unitary time development.) In this section we include the effects of the environment and also consider the most general type of single qubit error.

Consider a single qubit $|x\rangle$, and call the environment $|e\rangle$. Unlike the state of the qubit, the state of the environment is likely to be in a space of very many dimensions. Ideally $|x\rangle$ evolves under the effects of the gates only, independent of the environment. However, interactions with the environment cannot be avoided which leads to a corruption of the qubit and an entangling of the qubit with the environment.

The most general such form of these effects is

$$|e\rangle\,|0\rangle \to |e_0\rangle\,|0\rangle + |e_1\rangle\,|1\rangle, \tag{27a}$$

$$|e\rangle\,|1\rangle \to |e_2\rangle\,|0\rangle + |e_3\rangle\,|1\rangle, \tag{27b}$$

where $|e_i\rangle\,(i = 0, \cdots, 3)$ are possible final states of the environment. The environment states are not normalized, and not orthogonal either. However, the two states on the right hand side of Eqs. (27) must be orthogonal since the time evolution of the combined qubit-environment system is unitary. In other words

$$\langle e_2|e_0\rangle + \langle e_3|e_1\rangle = 0\,. \tag{28}$$

The corruption of the computation by the environment indicated in Eq. (27) is called "decoherence". It is the main source of difficulty in building a practical quantum computer.

In previous sections we have neglected entanglement with the environment. Rather, errors were assumed to occur because of mistakes made in the circuit itself. This corresponds to a special case of Eqs. (27), where all the environment states are the same, apart from normalization, i.e. $|e_i\rangle = c_i|e\rangle$, for $i = 0, \cdots, 3$.

We are interested in the case where the probability of an error is small (otherwise we would not be able to correct for it), i.e.

$$\langle e|e\rangle = 1, \quad \langle e_0|e_0\rangle \simeq 1, \quad \langle e_3|e_3\rangle \simeq 1, \quad \langle e_1|e_1\rangle \ll 1, \quad \langle e_2|e_2\rangle \ll 1. \tag{29}$$

Equations (27) can be combined into one as

$$|e\rangle\,|x\rangle \to \left\{ \left(\frac{|e_0\rangle + |e_3\rangle}{2} \right) \mathbb{1} + \left(\frac{|e_0\rangle - |e_3\rangle}{2} \right) Z + \left(\frac{|e_2\rangle + |e_1\rangle}{2} \right) X + \left(\frac{|e_2\rangle - |e_1\rangle}{2} \right) (iY) \right\} |x\rangle, \tag{30}$$

where $x = 0$ or 1 and, as usual,[5]

$$Z = \begin{pmatrix} 1 & 0 \\ 0 & -1 \end{pmatrix}, \quad X = \begin{pmatrix} 0 & 1 \\ 1 & 0 \end{pmatrix}, \quad iY = ZX = \begin{pmatrix} 0 & 1 \\ -1 & 0 \end{pmatrix}, \quad \mathbb{1} = \begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix}. \tag{31}$$

[5] I prefer to write equations like (30) in terms of $iY(= ZX)$ rather than Y to avoid having explicitly complex coefficients. Many texts on quantum computing write ZX rather than iY. Note that $iY\,(= ZX)$ is not Hermitian (though Y is) but we do not need the Hermitian property here. More importantly, iY, like X, Y and Z is unitary.

Please evaluate Eq. (31) separately for $x = 0$ and 1 to verify that it is equivalent to Eqs. (27). There is nothing special about these environment states so we can write

$$|e\rangle\,|x\rangle \rightarrow (\,|d\rangle\mathbb{1} + |a\rangle X + |b\rangle(iY) + |c\rangle Z\,)\,|x\rangle. \tag{32}$$

Equation (31) applies to both $x = 0$ and $x = 1$. Since time evolution of the combined qubit-environment system follows quantum mechanics and so is unitary and linear, it also applies to a linear superposition $|\psi\rangle = \alpha|0\rangle + \beta|1\rangle$ so

$$|e\rangle\,|\psi\rangle \rightarrow (\,|d\rangle\mathbb{1} + |a\rangle X + |b\rangle(iY) + |c\rangle Z\,)\,|\psi\rangle. \tag{33}$$

We see that the effects of the environment on the uncorrupted state of a single qubit can be expressed entirely in terms of the Pauli operators, $X, (iY)$ and Z. These are characterized as follows:

- X corresponds to a bit-flip error,
- Z corresponds to a phase-flip error, and
- $iY(= ZX)$ corresponds to a combined bit-flip and phase-flip error.

Intuitively, the reason that the new state can be expressed in terms of the Pauli operators and the identity, is that any 2×2 matrix can be written as a linear combination of these operators.

We remind the reader that the environment states are not normalized, and so, in the important case where the initial state is close to the final state, we have

$$\langle a|a\rangle \ll 1, \quad \langle b|b\rangle \ll 1, \quad \langle c|c\rangle \ll 1, \tag{34}$$

in Eq. (33),

We now extend this discussion the situation where we have expanded a single qubit into an n-qubit codeword which we write as $|\psi\rangle_n$. In this course we just consider how to correct single-qubit errors, so we neglect the possibility that two or more of the qubits in the codeword are corrupted. From Eq. (33), we see that all single qubit errors are incorporated by

$$|e\rangle\,|\psi\rangle_n \rightarrow \left(\,|d\rangle\mathbb{1} + \sum_{k=1}^{n}|a_k\rangle X_k + \sum_{k=1}^{n}|b_k\rangle(iY_k) + \sum_{k=1}^{n}|c_k\rangle Z_k\,\right)|\psi\rangle_n. \tag{35}$$

Based on Eq. (35), single qubit quantum error correction involves the following steps:

- Expand the logical qubit to an n-qubit codeword.

- Project the possibly corrupted state to *one* of the $3n+1$ states on the right hand side of Eq. (35), with information indicating *which* one.

- Correct, if necessary, the 1-qubit error by acting with the appropriate X_k, Y_k or Z_k.

Note:

(i) The whole *continuum* of errors can be represented by a finite set of *discrete* errors. Errors emerge continuously from the uncorrupted state by increasing from zero the size of the terms in Eq. (35) involving X_i, Y_i and Z_i, which are characterized by $\langle a_i|a_i\rangle^{1/2}$, $\langle b_i|b_i\rangle^{1/2}$ and $\langle c_i|c_i\rangle^{1/2}$ respectively. However, the projection is always to one of the $3n+1$ discrete states. If the amplitude of the error is small then, with high probability, the projection will be to the uncorrupted state (which needs no correction) but with small but non-zero probability the projection will be to one of the $3n$ corrupted states (which do need correction).

(ii) An *arbitrary* error on a single qubit will be corrected, not just bit-flip (X), or phase-flip (Z), or combined bit- and phase-flip (iY) errors but also *any combination of them.* For example, suppose that the k-th qubit has been reinitialized to zero, i.e.

$$|0_k\rangle \rightarrow |0_k\rangle,\ |1_k\rangle \rightarrow |0_k\rangle. \tag{36}$$

The matrix which accomplishes this transformation is[6]

$$\begin{pmatrix} 1 & 1 \\ 0 & 0 \end{pmatrix} \tag{37}$$

which can be written as

$$\frac{\mathbb{1} + X_k + iY_k + Z_k}{2}. \tag{38}$$

Hence the state of the codeword qubits and environment has been transformed as follows:

$$|e\rangle\,|\psi\rangle_n \rightarrow |e'\rangle\,|\psi'\rangle_n = |e'\rangle\,\frac{1}{2}\,(\mathbb{1} + X_k + iY_k + Z_k)\,|\psi\rangle_n. \tag{39}$$

[6] The reader will notice that the transformation in Eqs. (37), which involves a *linear combination* of X, iY and Z on a single qubit, are not unitary. Now the evolution of an isolated (closed) system *is* unitary, However, qubits are coupled to the environment. If we consider a system coupled to the environment (called an open system), and subject the combined system+environment to a unitary transformation, and finally consider the behavior of just the system by tracing out over the environment, the resulting transformation of the system is not necessarily unitary[4, 6]. A proper treatment of this situation requires the use of density matrices[4, 6, 7], but we won't go into the details here.

The codeword qubits are now in a linear combination of four syndromes, corresponding to the four terms in this equation. A general syndrome measuring circuit, such as the Shor 9-qubit code discussed in the next section, will detect these syndromes and obtain a *unique* set of values for the ancilla qubits for each of them. Hence, even for this non-unitary error, measuring the ancillas will project on to one of the syndromes which can then be corrected if necessary.

(iii) A full discussion of how the entanglement of qubits with the environment generates errors and how they can subsequently be corrected, requires the formalism of the density matrix[4, 6, 7]. This advanced treatment of quantum error correction[4, 6] is beyond the scope of the course.

VI. CORRECTING ARBITRARY ERRORS: THE SHOR CODE

In the section we discuss a code, due to Peter Shor[1], for correcting arbitrary 1-qubit errors. This code needs code words of nine qubits to represent one logical qubit. It is not the most efficient code, there are others which use smaller code words and so don't need as many physical qubits, but the structure of Shor's code follows quite naturally from the discussion we have already given of 1-qubit bit-flip, and 1-qubit phase-flip errors, so will discuss it here.

Essentially it combines bit-flip (X) and phase-flip (Z) codes, which turns out to then automatically correct for combined bit-flip, phase-flip (iY) errors. As discussed in the previous section, it then also corrects *arbitrary* 1-qubit errors.

We first encode for phase flips:

$$|0\rangle \to |+\,+\,+\rangle, \quad |1\rangle \to |-\,-\,-\rangle, \tag{40}$$

and then encode for bit-flip errors

$$|+\rangle \to \frac{1}{\sqrt{2}}\left(\,|000\rangle + |111\rangle\,\right), \quad |-\rangle \to \frac{1}{\sqrt{2}}\left(\,|000\rangle - |111\rangle\,\right). \tag{41}$$

The final result is the 9-qubit encoding

$$|0\rangle \to |\overline{0}\rangle = \frac{1}{2^{3/2}}\left(\,|000\rangle + |111\rangle\,\right)\left(\,|000\rangle + |111\rangle\,\right)\left(\,|000\rangle + |111\rangle\right), \tag{42a}$$

$$|1\rangle \to |\overline{1}\rangle = \frac{1}{2^{3/2}}\left(\,|000\rangle - |111\rangle\,\right)\left(\,|000\rangle - |111\rangle\,\right)\left(\,|000\rangle - |111\rangle\right). \tag{42b}$$

These two equations can be combined as

$$|x\rangle \to |\overline{x}\rangle = \frac{1}{2^{3/2}}\left(\,|000\rangle + (-1)^x|111\rangle\,\right)\left(\,|000\rangle + (-1)^x|111\rangle\,\right)\left(\,|000\rangle + (-1)^x|111\rangle\right), \tag{43}$$

or more concisely as

$$|\overline{x}\rangle = \frac{1}{2^{3/2}} \left(|000\rangle + (-1)^x |111\rangle \right)^{\otimes 3}. \tag{44}$$

Such a code is called a *concatenated* code. The circuit to achieve this encoding is obtained by concatenating the phase flip and the bit flip encodings as shown in Fig. 10. Note the labeling of the qubits. The qubits in each of the three blocks in Eq. (42) have labels $123, 456$ and 789.

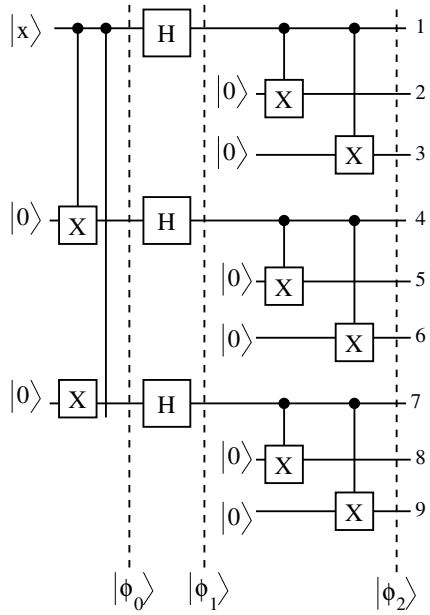

FIG. 10: Encoding for the Shor 9-qubit code. The initial state at the top left, $|x\rangle$, is equal to $|0\rangle$ or $|1\rangle$ in the computational basis, so $\phi_0 = |xxx\rangle$ and $\phi_1 = 2^{-3/2}(|0\rangle + (-1)^x|1\rangle)(|0\rangle + (-1)^x|1\rangle)(|0\rangle + (-1)^x|1\rangle)$ since $H|x\rangle = 2^{-1/2}(|0\rangle + (-1)^x|1\rangle)$. We see by comparison with Fig. 1, that if $x = 0$ then the final state $|\phi_2\rangle$ is equal to $|\overline{0}\rangle$ given by Eq. (42a), while if $x = 1$ the final state is $|\overline{1}\rangle$ given by Eq. (42b). If the initial state at the top left is a linear combination $\alpha|0\rangle + \beta|1\rangle$ then, by linearity, the final state at the right is $\alpha|\overline{0}\rangle + \beta|\overline{1}\rangle$. The numbers at the right are the labels of the nine qubits. Note that this circuit is a concatenation of the encoding circuit for phase-flips shown in Fig. 9, and that for bit-flips in Fig. 1.

The form of the 1-qubit corruption in Eq. (35) simplifies a little here because if $|\psi\rangle$ is a linear combination of the codeword states in Eq. (42) then

$$Z_1|\psi\rangle = Z_2|\psi\rangle = Z_3|\psi\rangle, \tag{45a}$$

$$Z_4|\psi\rangle = Z_5|\psi\rangle = Z_6|\psi\rangle, \tag{45b}$$

$$Z_7|\psi\rangle = Z_8|\psi\rangle = Z_9|\psi\rangle, \tag{45c}$$

since a Z_i operator changes *one* of the $+$ signs in Eq. (42a) into a $-$ sign, or one of the $-$ signs in Eq. (42b) into a $+$ sign.

Hence, the general form of a 1-qubit corruption contains only 22 independent syndromes rather than $28 = (3 \times 9) + 1$:

$$|e\rangle\,|\psi\rangle \rightarrow \left(|d\rangle I + |c\rangle Z_1 + |c'\rangle Z_4 + |c''\rangle Z_7 + \sum_{i=1}^{9} |a_i\rangle X_i + \sum_{i=1}^{9} |b_i\rangle i Y_i \right) |\psi\rangle. \tag{46}$$

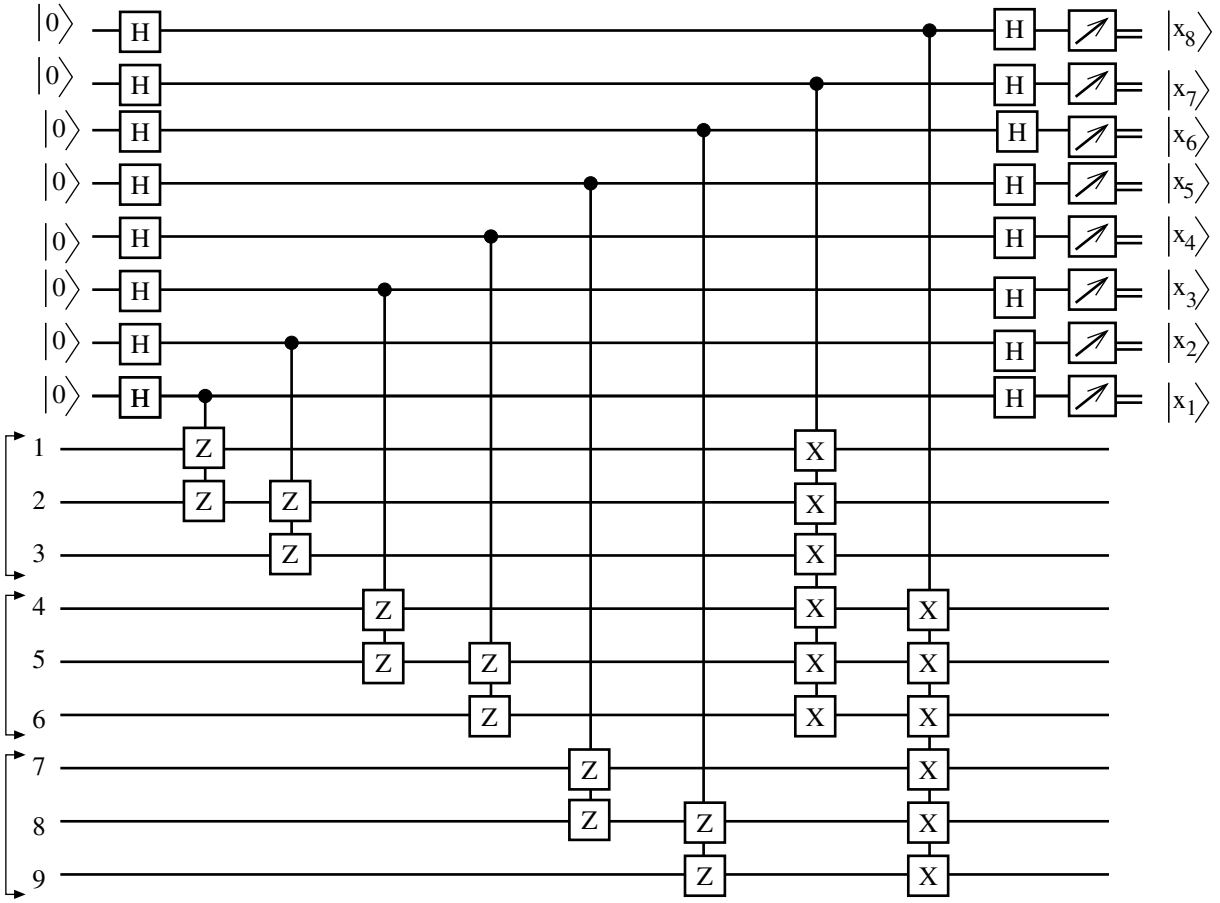

FIG. 11: A circuit to measure the error syndrome for the Shor 9-qubit code. The nine codeword qubits are at the bottom and the eight ancillary qubits at the top. The ancillary qubits determine the values of the eight, mutually commuting stabilizers in Eq. (47), $M_1 = Z_1Z_2, M_2 = Z_2Z_3, M_3 = Z_4Z_5, M_4 = Z_5Z_6, M_5 = Z_7Z_8, M_6 = Z_8Z_9, M_7 = X_1X_2X_3X_4X_5X_6$ and $M_8 = X_4X_5X_6X_7X_8X_9$. The nine codeword qubits can be conveniently grouped into three groups of three as indicated. The measured value of the i-th ancilla x_i (= 0 or 1), is related to the value of the corresponding stabilizer M_i by $M_i = (-1)^{x_i}$. The values of the M_i determine which syndrome in Eq. (46) is projected out, as discussed in the text, see Table III. If a corrupted syndrome is found, it can be corrected back to the uncorrupted state by acting with the appropriate X_i, Y_i or Z_i.

The eight stabilizers which we use to diagnose the error are

$$M_1 = Z_1 Z_2, \quad M_2 = Z_2 Z_3, \quad M_3 = Z_4 Z_5, \quad M_4 = Z_5 Z_6, \quad M_5 = Z_7 Z_8, \quad M_6 = Z_8 Z_9,$$
$$M_7 = X_1 X_2 X_3 X_4 X_5 X_6, \quad M_8 = X_4 X_5 X_6 X_7 X_8 X_9 \,. \tag{47}$$

Note that the nine qubits can conveniently be grouped into three blocks of three, containing qubits $123, 456$ and 789 respectively. M_1 and M_2 act entirely on the first block, and do so in the same way as the stabilizers of the 3-qubit, bit flip code shown in Fig. 7. Similarly M_3 and M_4 act on the second block and M_5 and M_6 act on the third block. M_7 acts on all qubits in blocks 1 and 2, while M_7 acts on all qubit in blocks 2 and 3.

The circuit for determining the syndrome eigenvalues is shown in Fig. 11.

We will now see that the M_i have the desired properties:

- They all square to unity (since each of the Z's and X's square to unity and the X's commute amongst each other as do the Z's). Hence their eigenvalues are ± 1.

- They mutually commute. The six Z-stabilizers trivially commute with each other as do the two X-stabilizers. Comparing the indices on the Z-stabilizers with the X-stabilizers one sees that either they have none in common (in which case this X-stabilizer and Z-stabilizer trivially commute) or they have two in common, in which case there are two minus signs when one pulls one of the stabilizers through the other so the overall sign is positive, and again the X-stabilizer and the Z-stabilizer commute).

- The eigenvalue of the uncorrupted codewords $|\overline{0}\rangle$ and $|\overline{1}\rangle$ is $+1$ for all stabilizers.

  This is trivially seen for M_1–M_6 which involve pairs of Z operators, since, for each pair, both qubits are 0 or both are 1 in the codewords. Note that the pairs are entirely within the blocks of three adjacent qubits in Eq. (42), see Fig. 10.

  Next consider M_7 and M_8 which involve a product of six X operators, each spanning two of the three blocks shown in Fig. 10. For example, M_7 is a product of the X operators for the qubits in the first two blocks. We have

$$\begin{aligned} M_7|\overline{0}\rangle = X_1 X_2 X_3 X_4 X_5 X_6 \frac{1}{2^{3/2}} \left(|000\rangle + |111\rangle\right) \left(|000\rangle + |111\rangle\right) \left(|000\rangle + |111\rangle\right) \\ = \frac{1}{2^{3/2}} \left(|111\rangle + |000\rangle\right) \left(|111\rangle + |000\rangle\right) \left(|000\rangle + |111\rangle\right) \\ = \frac{1}{2^{3/2}} \left(|000\rangle + |111\rangle\right) \left(|000\rangle + |111\rangle\right) \left(|000\rangle + |111\rangle\right) \\ = |\overline{0}\rangle, \end{aligned} \tag{48}$$

and

$$\begin{aligned} M_7|\overline{1}\rangle = X_1X_2X_3X_4X_5X_6\frac{1}{2^{3/2}}\left(|000\rangle-|111\rangle\right)\left(|000\rangle-|111\rangle\right)\left(|000\rangle-|111\rangle\right) \\ =\frac{1}{2^{3/2}}\left(|111\rangle-|000\rangle\right)\left(|111\rangle-|000\rangle\right)\left(|000\rangle-|111\rangle\right) \\ =\frac{1}{2^{3/2}}\left(|000\rangle-|111\rangle\right)\left(|000\rangle-|111\rangle\right)\left(|000\rangle-|111\rangle\right) \\ =|\overline{1}\rangle, \end{aligned} \tag{49}$$

so M_7 has eigenvalue $+1$ for both uncorrupted codewords. The argument for M_8 is virtually identical.

- The ±1 eigenvalues of the stabilizers allow one to determine which of the 22 syndromes in Eq. (46) the system has projected on to. Recalling the discussion in Sec. III, the eigenvalue is $+1$ if the stabilizer commutes with the operator which caused the 1-qubit corruption, and is -1 if it anti-commutes. Each time two different operators on the same qubit are pulled through each other to perform the commutation one generates a minus sign. The operators which generate the corruption are the 21 X_i, Y_i and Z_i in Eq. (46). A table of the eigenvalues of the stabilizers for all 22 syndromes is given in Table III.

Let's make sure that we understand how the syndrome-detection circuit in Fig. 11 works. Firstly we remind the reader that if the measurement of an auxiliary qubit, x_i say, is 0, then the value of the corresponding stabilizer M_i is $+1$, while if the measurement is 1, then the value of M_i is -1. Thus we can say that x_i measures M_i, see the discussion of Fig. 6 on page 11. Next we discuss how each of the stabilizers works.

- We consider first M_1–M_6, the stabilizers involving Z operators.
  The ancilla qubits x_1 and x_2 measure $M_1 = Z_1Z_2$ and $M_2 = Z_2Z_3$ respectively, and so detect a bit-flip error in the first group of three qubits in the 9-qubit encoding of Eq. (42), in exactly the same way as for the 3-qubit, bit-flip code shown in Fig. 7. Similarly x_2 and x_3 detect a bit-flip error in the second group of three qubits (qubits 4–6) and the third group of three qubits respectively.

- Next we consider M_7 and M_8, the stabilizers involving X operators.
  The ancilla x_7 measures $M_7 = X_1X_2X_3X_4X_5X_6$ and the ancilla x_8 measures $M_8 = X_4X_5X_6X_7X_8X_9$. These detect phase flips. M_7 acts on the first two groups of three qubits (qubits 1–6) while M_8 acts on the second and third groups of three qubits (qubits 4–9).

Syndrome	M_1	M_2	M_3	M_4	M_5	M_6	M_7	M_8
$\mathbb{1}$	+	+	+	+	+	+	+	+
X_1	−	+	+	+	+	+	+	+
X_2	−	−	+	+	+	+	+	+
X_3	+	−	+	+	+	+	+	+
X_4	+	+	−	+	+	+	+	+
X_5	+	+	−	−	+	+	+	+
X_6	+	+	+	−	+	+	+	+
X_7	+	+	+	+	−	+	+	+
X_8	+	+	+	+	−	−	+	+
X_9	+	+	+	+	+	−	+	+
Y_1	−	+	+	+	+	+	−	+
Y_2	−	−	+	+	+	+	−	+
Y_3	+	−	+	+	+	+	−	+
Y_4	+	+	−	+	+	+	−	−
Y_5	+	+	−	−	+	+	−	−
Y_6	+	+	+	−	+	+	−	−
Y_7	+	+	+	+	−	+	+	−
Y_8	+	+	+	+	−	−	+	−
Y_9	+	+	+	+	+	−	+	−
$Z_1\,(=Z_2=Z_3)$	+	+	+	+	+	+	−	+
$Z_4\,(=Z_5=Z_6)$	+	+	+	+	+	+	−	−
$Z_7\,(=Z_8=Z_9)$	+	+	+	+	+	+	+	−

TABLE III: The eigenvalues of the 8 stabilizers defined in Eq. (47) for the 22 syndromes of Shor's 9-qubit error correcting code. A + sign indicates eigenvalue +1 and a − sign indicates eigenvalue −1. Each stabilizer M_i is measured by an ancilla qubit x_i, see Fig. 11, such that if $M_i = +1$ then $x_i = 0$ and if $M_i = -1$ then $x_i = 1$. An essential feature is that each of the 22 rows, i.e. syndromes, has a unique pattern of + and − signs.

Let's suppose that there is a phase flip in one of the qubits in the first group (it doesn't matter which one; the resulting state is the same). In other words

$$|\psi\rangle = \alpha|\overline{0}\rangle + \beta|\overline{1}\rangle \tag{50}$$

has been transformed to

$$|\psi'\rangle = \frac{\alpha}{2^{3/2}} \left(|000\rangle - |111\rangle \right) \left(|000\rangle + |111\rangle \right) \left(|000\rangle + |111\rangle \right) + \frac{\beta}{2^{3/2}} \left(|000\rangle + |111\rangle \right) \left(|000\rangle - |111\rangle \right) \left(|000\rangle - |111\rangle \right). \tag{51}$$

Acting with the product of the three X_i in a group has the effect

$$\begin{aligned} |000\rangle + |111\rangle &\rightarrow |111\rangle + |000\rangle = |000\rangle + |111\rangle \\ |000\rangle - |111\rangle &\rightarrow |111\rangle - |000\rangle = -\left(|000\rangle - |111\rangle \right). \end{aligned} \tag{52}$$

Recalling that we are considering here a phase flip in the first group, on which M_7 acts but M_8 does not, it follows that

$$\begin{aligned} M_7|\psi'\rangle &= X_1X_2X_3X_4X_5X_6|\psi'\rangle = -|\psi'\rangle \\ M_8|\psi'\rangle &= X_4X_5X_6X_7X_8X_9|\psi'\rangle = |\psi'\rangle. \end{aligned} \tag{53}$$

Hence M_7 has eigenvalue -1 and M_8 has eigenvalue $+1$. These values are shown in Table III.

Similarly, if the phase-flip is in the second group, both M_7 and M_8 have eigenvalue -1 whereas if phase-flip is in the third group, M_7 has eigenvalue $+1$ and M_8 has eigenvalue -1. These results are also shown in Table III.

We now illustrate in more detail how Table III was obtained by working through a few cases. (Eigenvalues are taken to be $+1$ unless otherwise stated.)

(a) $\boldsymbol{Z_2}$: Clearly Z_2 commutes with all the Z-stabilizers. It anticommutes with M_7 (because it has one qubit in common and X and Z anticommute) and commutes with M_8 because it has no qubits in common. Hence M_7 has eigenvalue -1 while all other stabilizers have eigenvalue $+1$.

(b) $\boldsymbol{Z_4}$: Both M_7 and M_8 have eigenvalue -1 since they have one qubit in common with Z_4 (and X and Z anticommute).

(c) $\boldsymbol{X_4}$: Clearly X_4 commutes with both X-stabilizers. It anticommutes with M_3 because it has one qubit in common (and Z and X anticommute). Hence M_3 has eigenvalue -1.

(d) $\boldsymbol{Y_5}$: We note that Y anticommutes with both X and Z so we have to consider all the stabilizers. Y_5 has a qubit in common with M_3, M_4, M_7 and M_8 so these stabilizers have eigenvalue -1.

We recall that each syndrome gives rise to a unique set of $+1$ and -1 eigenvalues of the stabilizers, see Table III. Thus, measuring the eigenvalues of the eight stabilizers in Eq. (47) projects the corrupted state on to one of the 22 syndromes in Eq. (46), and the set of eigenvalues determines which one it is. One then applies an appropriate unitary transformation to correct the state if necessary. Note that the Shor code is *explicitly* designed to detect and correct bit-flip (X) and phase-flip (Z) errors, but then *automatically* detects and corrects combined bit- and phase-flip ($ZX \equiv iY$) errors.

Not only that, it also corrects *arbitrary* errors on a single qubit, which, as discussed in Sec. V, can be expressed as *linear combinations* of bit-flip, and phase-flip, and combined bit- and phase-flip errors. As an example consider the situation mentioned in Eq. (39) in Sec. V in which a qubit has been reset to $|0\rangle$. This is an example of a *non-unitary*[7] operation on the qubit. Let's take it to be qubit 1 and indicate the codeword qubits by putting the first on the left, the last on the right (we will use the same ordering below for the ancilla qubits). In other words

$$|\psi\rangle = \alpha|\overline{0}\rangle + \beta|\overline{1}\rangle \tag{54}$$

has been transformed to

$$\begin{aligned} |\psi'\rangle =& \frac{\alpha}{2^{3/2}} \left(|000\rangle + |011\rangle \right) \left(|000\rangle + |111\rangle \right) \left(|000\rangle + |111\rangle \right) + \\ & \frac{\beta}{2^{3/2}} \left(|000\rangle - |011\rangle \right) \left(|000\rangle - |111\rangle \right) \left(|000\rangle - |111\rangle \right). \end{aligned} \tag{55}$$

According to Eq. (39) this can be written as

$$|\psi'\rangle = \frac{1}{2} \left(\mathbb{1} + X_1 + iY_1 + Z_1 \right) |\psi\rangle, \tag{56}$$

where

$$|\psi\rangle = \alpha \left(\quad |000\rangle + |111\rangle \right) (\cdots)_+ (\cdots)_+ + \beta \left(\quad |000\rangle - |111\rangle \right) (\cdots)_- (\cdots)_- \tag{57a}$$

$$X_1|\psi\rangle = \alpha \left(\quad |100\rangle + |011\rangle \right) (\cdots)_+ (\cdots)_+ + \beta \left(\quad |100\rangle - |011\rangle \right) (\cdots)_- (\cdots)_- \tag{57b}$$

$$iY_1|\psi\rangle = \alpha \left(-|100\rangle + |011\rangle \right) (\cdots)_+ (\cdots)_+ + \beta \left(-|100\rangle - |011\rangle \right) (\cdots)_- (\cdots)_- \tag{57c}$$

$$Z_1|\psi\rangle = \alpha \left(\quad |000\rangle - |111\rangle \right) (\cdots)_+ (\cdots)_+ + \beta \left(\quad |000\rangle + |111\rangle \right) (\cdots)_- (\cdots)_- , \tag{57d}$$

in which

$$\begin{aligned} (\cdots)_+ &= \left(|000\rangle + |111\rangle \right) \\ (\cdots)_- &= \left(|000\rangle - |111\rangle \right). \end{aligned} \tag{58}$$

[7] In footnote 6 we noted that, while a transformation of the combined system+environment *is* unitary, if the system is coupled to the environment, then a unitary operation applied to system+environment followed by a trace over the environment leaves the system in a new state which is not, in general, related by a unitary transformation to its initial state.

One can verify that adding Eqs. (57) (and dividing by 2 according to Eq. (56)) does indeed give Eq. (55).

Equation (56) is the input to the syndrome measurement circuit. According to Table III, after the syndrome measurement circuit in Fig. 10 has acted, the state of the system is

$$\frac{1}{2}\left[\,|\psi\rangle\,|00000000\rangle_A + (X_1|\psi\rangle)\,|10000000\rangle_A + (iY_1|\psi\rangle)\,|10000010\rangle_A + (Z_1|\psi\rangle)\,|00000010\rangle_A\,\right], \quad (59)$$

where $|\cdots\rangle_A$ denotes the ancillas, which are ordered from 1 on the left to 8 on the right. Measuring the ancillas will project on to one of the four syndromes, $|\psi\rangle, X_1|\psi\rangle, iY_1|\psi\rangle, Z_1|\psi\rangle$, which can them be corrected if necessary.

Thus, Shor's 9-qubit code, and other codes designed to correct both bit-flip and phase-flip errors, actually correct *arbitrary* 1-qubit errors. I find this amazing.

VII. OTHER ERROR-CORRECTING CODES

The Shor code uses nine physical qubits to encode one logical qubit. What is the minimum number of physical qubits needed to correct all 1-qubit errors? If we encode using n qubits the dimension of the space of states is 2^n. This must be sufficient to contain $3n+1$ mutually orthogonal 2-d subspaces for the syndromes (the 1 is for the uncorrupted state and there are n possible corruptions with each of the X, iY or Z operators). Hence we need

$$2^n \geq 2(3n+1)\,, \quad (60)$$

so the smallest value is $n = 5$ which satisfies this condition as an equality.

There *is* a 5-qubit code, but it turns out to be difficult to construct the necessary gates. A more popular choice is a 7-qubit code due to Steane[2]. The Shor code is now mainly of pedagogical interest.

We now state, without much discussion, the codewords and stabilizers for the 5-qubit code. Further details are in Mermin[3].

For the 5-qubit code we have $(3 \times 5) + 1 = 16$ mutually orthogonal, two-dimensional subspaces. We therefore need four stabilizers since they each have two eigenvalues (± 1) and the number of

distinct sets of eigenvalues is $2^4 = 16$. These stabilizers are

$$M_1 = Z_2 X_3 X_4 Z_5, \tag{61a}$$

$$M_2 = Z_3 X_4 X_5 Z_1, \tag{61b}$$

$$M_3 = Z_4 X_5 X_1 Z_2, \tag{61c}$$

$$M_4 = Z_5 X_1 X_2 Z_3. \tag{61d}$$

The circuit to measure the M_i is shown in Fig. 12.

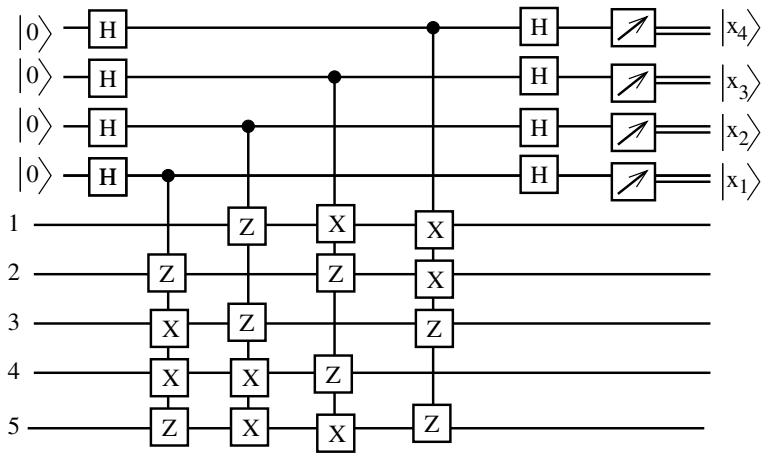

FIG. 12: A circuit to measure the error syndrome for the 5-qubit code. The five codeword qubits are at the bottom and the four ancillary qubits at the top. The ancillary qubits determine the values of the four, mutually commuting stabilizers in Eq. (61), $M_1 = Z_2 X_3 X_4 Z_5, M_2 = Z_3 X_4 X_5 Z_1, M_3 = Z_4 X_5 X_1 Z_2, M_4 = Z_5 X_1 X_2 Z_3$.

The 5-qubit codewords are most conveniently expressed in terms of the M_i:

$$|\overline{0}\rangle = \frac{1}{4}(1 + M_1)(1 + M_2)(1 + M_3)(1 + M_4)|00000\rangle, \tag{62a}$$

$$|\overline{1}\rangle = \frac{1}{4}(1 + M_1)(1 + M_2)(1 + M_3)(1 + M_4)|11111\rangle. \tag{62b}$$

Note that $|\overline{0}\rangle$ is composed of the 16 basis states with an even number of 1's, while $|\overline{1}\rangle$ is composed of the 16 basis states with an odd number of 1's, so the two codewords are orthogonal. It is not completely trivial to generate these codewords, see Mermin [3] for details.

Furthermore the M_i square to unity, are mutually commuting and each has eigenvalue $+1$ for the uncorrupted codewords in Eq. (62). Each of them commutes or anti-commutes with the X_i, Y_i and Z_i error operators, so the 15 corrupted syndromes and the uncorrupted state are distinguished by the set of ± 1 eigenvalues of the M's, as shown in Table IV.

Syndrome	$M_1 = Z_2X_3X_4Z_5$	$M_2 = Z_3X_4X_5Z_1$	$M_3 = Z_4X_5X_1Z_2$	$M_4 = Z_5X_1X_2Z_3$
$\mathbb{1}$	+	+	+	+
X_1	+	−	+	+
X_2	−	+	−	+
X_3	+	−	+	−
X_4	+	+	−	−
X_5	−	+	+	−
Y_1	+	−	−	−
Y_2	−	+	−	−
Y_3	−	−	+	−
Y_4	−	−	−	+
Y_5	−	−	−	−
Z_1	+	+	−	−
Z_2	+	+	+	−
Z_3	−	+	+	+
Z_4	−	−	+	+
Z_5	+	−	−	+

TABLE IV: For each of the four stabilizers M_i for the 5-qubit error correcting codes we show whether they commute (+) or anti-commute (−) with the 15 operators X_i, Y_i and Z_i, $i = 1, 2, \cdots, 5$ (which generate a corruption of the codeword) as well as with the identity. Each of the 16 rows has a unique pattern of + and − signs. A + sign corresponds to an eigenvalue +1 while a − sign indicates an eigenvalue −1.

The 7-qubit code due to Steane will be gone through in a homework assignment.

A different approach to quantum error correction, but one that seems the most promising, is to use "surface codes" in which the physical qubits are arranged in a square array and the values of the logical qubits are encoded in complicated entangled states of the square array. Unfortunately, I have not been able to find a simple introduction to this topic.

VIII. FAULT TOLERANT QUANTUM COMPUTING

So far we have assumed that an error has occurred in some way and that we can correct it by *perfect* gates which do not introduce any further errors. This is, of course unreasonable since all aspects of quantum computing can introduce errors: acting with gates, measurements, or simply waiting. Looking at the number of gates for Shor's 9-qubit syndrome-detection code in Fig. 11 we might imagine that this circuit could introduce more errors than it corrects. Of particular

importance is that the circuit does not spread an error initially in one qubit into multiple qubits which would then be much harder to correct. A circuit which does not spread errors is said to be "fault tolerant".

An important result in quantum error correction is the "threshold theorem" which states that if the intrinsic error rate in an individual gate in a fault tolerant circuit is less than a critical value p_c then the overall error rate in the circuit can be reduced to arbitrary low levels by quantum error correction. This means that errors are being corrected faster than they are being generated. However, getting the error rate down to an acceptable level will still require a considerable increase in the number of physical qubits.

Suppose that the intrinsic error rate is p and we have a fault tolerant error correction scheme which corrects 1-qubit errors. This means that the error rate after error correction is[8] cp^2 for some constant c. If $pc < 1$ then we have decreased the errors, so the threshold error rate is $p_c = 1/c$. How can we go decrease the errors further? Suppose the error correction procedure requires n physical qubits for each logical qubit. We can then take each of the n qubits and error correct these with the same code. This procedure is known as *concatenation*. We then have n^2 physical qubits and the error rate is $c(cp^2)^2 = c^{-1}(cp)^{2^2}$. Generalizing, if we concatenate l times, then the number of qubits is n^l while the resulting error rate is $c^{-1}(cp)^{2^l}$. Note that while the number of qubits increases exponentially with the level of concatenation l, the error rate decreases *doubly* exponentially with l. As an example, to get a feel for what this means, consider the case $p = 1/8, c = 2$, so $cp = 1/4$ and also suppose that $n = 7$ (corresponding to the Steane code). Then successive concatenations give the numbers in Table V.

no. of concatenations (l)	error rate (formula)	error rate (numeric)	no. of qubits
0	p	$1/2^3 = 0.125$	1
1	$cp^2 = c^{-1}(cp)^2$	$1/2^5 = 0.03125$	n $(= 7)$
2	$c(cp^2)^2 = c^{-1}(cp)^{2^2}$	$1/2^9 = 1.953 \times 10^{-3}$	$n^2\,(= 49)$
3	$c((cp^2)^2)^2 = c^{-1}(cp)^{2^3}$	$1/2^{17} = 7.629 \times 10^{-6}$	$n^3\,(= 343)$
4	$c(c((cp^2)^2)^2)^2 = c^{-1}(cp)^{2^4}$	$1/2^{33} = 1.164 \times 10^{-10}$	$n^4\,(= 2401)$
5	$c(c(c((cp^2)^2)^2)^2)^2 = c^{-1}(cp)^{2^5}$	$1/2^{65} = 2.711 \times 10^{-20}$	$n^5\,(= 16807)$

TABLE V: Parameters for the concatenation of a fault tolerant circuit with an (artificial) choice of parameters discussed in the text.

[8] The crucial point is that the new error rate is proportional to the *square* of the old error rate. I don't think it's obvious that one can design a circuit with this property, but a detailed study indicates that one can [4, 6].

These numbers are not realistic. They correspond to a threshold value of $p_c = 1/c = 1/2$ and any realistic circuit would have a much smaller value. However, they do show, and this is the main point, that the error rate goes down much faster than the number of physical qubits goes up. Of course, the number of physical qubits per logical qubit will still have to be very large to get the error rate down to an acceptable value for computation.

Various calculations have estimated the threshold for 7-qubit Steane code at around 10^{-5}. To perform error correction one would need individual circuit elements with an error rate significantly less than this, which, to my knowledge, is not feasible at present. Surface codes, which were briefly mentioned above, are estimated to have a higher threshold, of around 10^{-2}, and it does seem feasible to make gates with a lower error rate than this. For example, at the end of a very long and technical paper, Ref. [8] estimates that to factor, using Shor's algorithm, an integer which is too large to be factored on a classical computer (2000 bits), would require no less than around 220×10^6 qubits with then state-of-the-art superconducting qubits using quantum error correction with surface codes. At present, quantum computers (using the "gate" model of quantum computing which is the main topic of this course) have at most a few tens of qubits, so a huge increase in scale will be required. However, who is to say that this cannot happen in a few decades? An example of a comparable increase in scale which has *already* happened is the number of transistors on a modern chip compared with the number on early integrated circuits.

Thus, in my view, in the next few years, we may see quantum computers with a modest number of logical qubits which perform error correction. However, quantum computers with error correction *having enough logical qubits to outperform classical computers* for some *useful* problem such as integer factorization are for the distant future, if ever.

Acknowledgments

I thank Eleanor Rieffel for a helpful email exchange on quantum error correction.

---

[1] P. Shor, *Scheme for reducing decoherence in quantum computer memory*, Phys. Rev. A **52**, 2493 (1995).

[2] A. M. Steane, *Error correcting codes in quantum theory*, Phys. Rev. Lett. **77**, 793 (1996).

[3] N. D. Mermin, *Quantum Computer Science* (Cambridge University Press, Cambridge, 2007).

[4] M. A. Nielsen and I. L. Chuang, *Quantum Computation and Quantum Information* (Cambridge University Press, Cambridge, England, 2000).

[5] R. Vathsan, *Introduction to Quantum Physics and Information Processing* (CRC Press, Boca Raton, 2016).

[6] E. Rieffel and I. Polak, *Quantum Computing; A Gentle Introduction* (MIT Press, Massachusetts Institute of Technology, 2014).

[7] A. P. Young, *The Density Matrix* (2020), https://young.physics.ucsc.edu/150/density-matrix.pdf.

[8] A. Fowler, M. Mariantoni, J. Martinis, and A. Cleland, *Surface codes: Towards practical large-scale quantum computation*, Phys. Rev. A **86**, 032324 (2012).