

Report of Introduction to Quantum Computing Quantum Error Correction

Student: Diego Arcelli
Student ID: 647979
Professors: Gianna Maria Del Corso,
Anna Bernasconi, Roberto Bruni

Academic Year 2021-2022



UNIVERSITÀ DI PISA

1 Introduction

The hardware devices used to implement quantum computers are highly prone to noise, which can affect the state of the qubits during the execution of a quantum circuit. For this reason being able to detect and correct those errors is a topic of major relevance in quantum computing. Error correction has been vastly studied for classical computers, so one might think to apply the knowledge that we already have for classical error correction to correct quantum errors. Sadly, in the quantum case there are some more problems which make error correction more difficult:

- We cannot introduce redundancy by copying arbitrary qubits, because of the no cloning theorem
- We cannot measure a qubit to check its state, because this would destroy the superposition
- An error on a qubit is a continuous error, while in classical computing we only have discrete bit flip errors

2 General Framework

We start by describing a general framework for quantum error correction, and in the next section we'll show some of its applications. The first step to protect a quantum state $|\psi\rangle$ from a possible error is to define an error model, which is a description of the type of error that might occur on the state. Once we have the error model, in order to defend a $|\psi\rangle$, we'll define an encoding operator U_{enc} which, by adding some extra ancillary qubits to the state, will produce an encoded state $|\psi_{enc}\rangle$:

$$U_{enc} |\psi\rangle |0 \dots 0\rangle = |\psi_{enc}\rangle$$

The space of all the possible encoded states will form a subspace of the Hilbert space, that we'll call quantum error correcting code or simply code. If $|\psi\rangle$ is composed of k qubits, and with the encoding we add n qubits, then the code space \mathcal{C} will be a subspace of the 2^{k+n} -dimensional Hilbert space. The rationale of encoding the input state in a higher dimensional state, is that if the error occurs in the encoded state, it will leave some information which can be used to identify its effects, and we refer to this information as error syndrome. Then we define a recovery operator \mathcal{R} that, if an error occurs on the encoded state $|\psi_{enc}\rangle$, will undo the effect of the error. Finally we'll decode $|\psi_{enc}\rangle$ back to $|\psi\rangle$ with a decoding operator U_{dec} :

$$U_{dec} |\psi_{enc}\rangle = |\psi\rangle |0 \dots 0\rangle$$

In general given a quantum code $\mathcal{C} \subset \mathcal{H}$, we say that a noise operator \mathcal{E} is recoverable if there exists a recovery operator \mathcal{R} such that $\mathcal{R}(\mathcal{E}|\psi\rangle) = |\psi\rangle$ for every state $|\psi\rangle \in \mathcal{C}$.

3 Correcting single qubit errors

3.1 Correcting bit flip errors

The simplest error that we might have is the bit flip error, where a generic qubit in state $\alpha|0\rangle + \beta|1\rangle$ it's transformed in $\beta|0\rangle + \alpha|1\rangle$. This can be seen as an application of the X Pauli gate to the qubit. The idea of how to correct this error comes from classical computing, where if we have a noisy channel on which a classical bit can flip with probability p , to reduce the probability of receiving the wrong bit, the sender can repeat the bit three times. In this way the receiver can decode the string of three bits simply by choosing the majority bit as the bit to receive. For instance if the sender sends 111, and there's a bit flip on the second bit, the receiver will receive 101, and he will decode it as 1. Note that if two or all three bits get flipped, the receiver will decode the wrong bit, but if the errors are independent this happens with probability $3p^2 + p^3$ instead of p . We can apply the same idea in the quantum case, by choosing the following three qubit encoding for the states $|0\rangle$ and $|1\rangle$:

$$\alpha|0\rangle + \beta|1\rangle \longrightarrow \alpha|000\rangle + \beta|111\rangle$$

We say that the physical states $|0\rangle$ and $|1\rangle$ have been encoded respectively in the logical states $|000\rangle$ and $|111\rangle$. We want to apply the same strategy we used for the classical case, which is checking if the values of the three qubits are the same and if they aren't we impose the value of the majority qubit. Of course we cannot do that by measuring the state, otherwise the superposition will be destroyed. What we can do instead is using some extra qubits, to store in them the information of which qubit of the state has been flipped, and this will be our error syndrome. The full circuit to correct a bit flip error is the following:

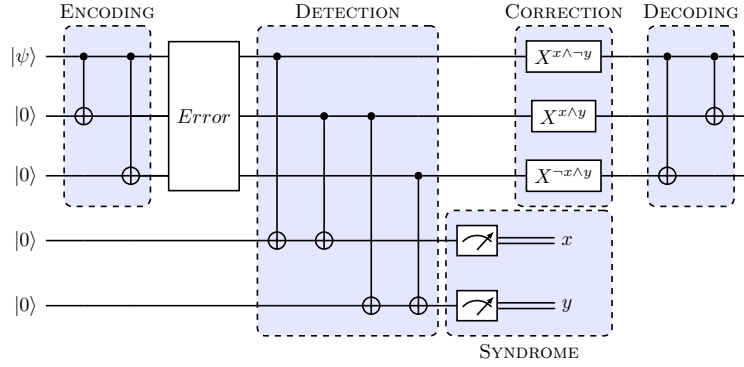


Figure 1: Bit flip error correction circuit

The first three qubits are used for the encoding, while the other two are the syndrome qubits. The first two CNOT gates are used to apply the encoding. Then, with the second two CNOT gates, we basically check if the first two qubits are equal, and if they aren't we set the first syndrome qubit to $|1\rangle$. This is because if the first and the second qubits are equal and they're both $|0\rangle$ then the syndrome qubit is not flipped. If they are equal and they're both $|1\rangle$ the syndrome qubit is flipped twice (so it goes back to $|0\rangle$). If they're different the syndrome qubit is flipped once and so it is set to $|1\rangle$. With the last two CNOT gates we repeat the same procedure, but this time we compare the second and the third qubits and we act on the second syndrome qubit. Finally we measure the syndrome qubits and we get the values x and y . The value of x tells us if the first and the second qubits were equal, while the value of y tells us if the second and the third qubits were equal. Therefore we have four possible syndromes each of which corresponds to a different error, as shown in the below table:

| x | y | Error type |
|-----|-----|--------------------|
| 0 | 0 | No error |
| 1 | 0 | Error on 1st qubit |
| 1 | 1 | Error on 2nd qubit |
| 0 | 1 | Error on 3rd qubit |

Based on the syndrome measured, we can correct the error by applying an X gate to the flipped qubit. Like in the classical case, if a bit flip error occurs in 2 or all the 3 qubits, then the circuit will decode the wrong state $\beta|0\rangle + \alpha|1\rangle$. The implementation of this circuit in Qiskit can be found in the notebook `three_qubit_code_bit_flip.ipynb`.

3.2 Correcting phase flip errors

Another simple type of error is the phase flip error, where a generic qubit $\alpha|0\rangle + \beta|1\rangle$ is transformed in $\alpha|0\rangle - \beta|1\rangle$. This can be seen as an application of the Z Pauli gate to the qubit. To correct this error we'll use the fact that phase flip errors in the computational basis become bit flip errors in the Hadamard basis, since $Z|-\rangle = |+\rangle$ and $Z|+\rangle = |-\rangle$. We can exploit this fact by using the following encoding:

$$\alpha|0\rangle + \beta|1\rangle \rightarrow \alpha|+++\rangle + \beta|---\rangle$$

which can be obtained from the bit flip encoding, by applying to each one of the three qubits the H gate. Suppose that with this encoding a phase flip error occurs in the second qubit, then, since in the Hadamard basis a phase flip error acts like a bit flip error, the encoded state becomes:

$$\alpha|+-+\rangle + \beta|-+-\rangle$$

if we now apply the H gate to each qubit, the state becomes:

$$\alpha|010\rangle + \beta|101\rangle$$

which is like if a bit flip error occurred in the second qubit of the encoded state for the bit flip correction. Therefore we can simply run the circuit for the bit flip correction, and we'll correctly recover the initial state. We can adapt the circuit in figure 1 to protect against phase flip errors, by adding three H gates at the end of the encoding block, and other three H gates at the beginning of the detection block. Like in the bit flip circuit, we can correct a phase flip error only if it occurs in just one of the three qubits. The full circuit is implemented in the notebook `three_qubit_code_phase_flip.ipynb`.

3.3 Nine qubits correcting code

We have seen a circuit which can correct bit flip errors and a circuit which can correct phase flip errors, but both of them fail if a phase flip error and a bit flip error happen at the same time. This type of error can be seen as the application of

the Y Pauli gate, up to a global phase factor of $\pm i$, since $Y = iXZ = -iZX$. A strategy to protect a qubit from this type of error has been presented by Shor in [1]. The intuition is to start from the encoding which protects a qubit from phase flip errors:

$$\alpha|0\rangle + \beta|1\rangle \longrightarrow \alpha|+++ \rangle + \beta|--- \rangle$$

and then we apply to each one of the three qubit the circuit for the bit flip encoding, in order to protect each qubit against bit flip errors. This will produce the following nine qubits encoding:

$$\alpha|0\rangle + \beta|1\rangle \longrightarrow \frac{\alpha}{\sqrt{8}}(|000\rangle + |111\rangle)^{\otimes 3} + \frac{\beta}{\sqrt{8}}(|000\rangle - |111\rangle)^{\otimes 3}$$

where $\frac{1}{\sqrt{8}}(|000\rangle + |111\rangle)^{\otimes 3}$ is the encoding of $|0\rangle$ and $\frac{1}{\sqrt{8}}(|000\rangle - |111\rangle)^{\otimes 3}$ is the encoding of $|1\rangle$. In this encoding we have three blocks of three qubits each. Apart from a ± 1 factor, each block has the structure of the bit flip encoding: $|000\rangle \pm |111\rangle$, therefore if a bit flip error happens on one of the qubit of the block, we can run the bit flip error correction circuit on that block to correct the error. So the protection against bit flip error is made applying the bit flip correction circuit to each block.

For the protection against phase flip errors, consider the sign of each block, which is $+$ for the encoding of $|0\rangle$ and $-$ for the encoding of $|1\rangle$. If there's no error, the sign of every block should be the same, but if there's a phase flip error on one of the qubits of a block, the sign on that block will flip:

$$|000\rangle + |111\rangle \xleftrightarrow{Z} |000\rangle - |111\rangle$$

So to detect a phase flip error, what we can do is compare the sign of every block and impose the majority sign, like we did with the values of the three qubits in the circuit for the bit flip correction. This idea is realized by the following circuit:

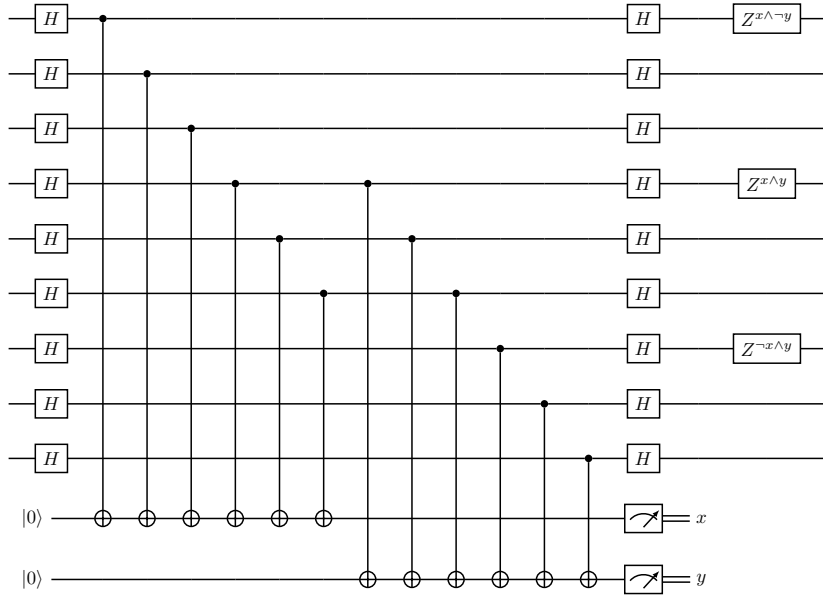


Figure 2: Circuit for phase flip correction in the nine qubits code

The circuit it's quite complicated, but what it does is comparing the sign of the first block with the sign of the second block and if they're different it sets the first syndrome qubit to 1. Then it does the same thing with the second and third blocks, acting on the second syndrome qubit.

Consider the state after the application of the first $H^{\otimes 9}$ gate, every block will be transformed as follows:

$$|000\rangle + |111\rangle \xrightarrow{H^{\otimes 9}} 2|000\rangle + 2|011\rangle + 2|101\rangle + 2|110\rangle = |\psi^+\rangle$$

$$|000\rangle - |111\rangle \xrightarrow{H^{\otimes 9}} 2|001\rangle + 2|010\rangle + 2|100\rangle + 2|111\rangle = |\psi^-\rangle$$

$$\frac{\alpha}{\sqrt{8}}(|000\rangle + |111\rangle)^{\otimes 3} + \frac{\beta}{\sqrt{8}}(|000\rangle - |111\rangle)^{\otimes 3} \xrightarrow{H^{\otimes 9}} \frac{\alpha}{\sqrt{8}}|\psi^+\rangle|\psi^+\rangle|\psi^+\rangle + \frac{\beta}{\sqrt{8}}|\psi^-\rangle|\psi^-\rangle|\psi^-\rangle$$

So the encoding of the state $|0\rangle$ will be composed of three blocks $|\psi^+\rangle$, each of which is composed of four binary strings, where each string is composed of an even number of 1s. The encoding of $|1\rangle$ is instead composed of three blocks $|\psi^-\rangle$, and each block is composed of four binary strings with an odd number of 1s. What the circuit does is basically exploiting this property.

Consider for instance the first six CNOT gates that act on the first and the second blocks. If there's been no phase flip error, the first three CNOT gates will flip the first syndrome qubit an even number of times for the encoding of $|0\rangle$, and an odd number of times for the encoding of $|1\rangle$. Then the second three CNOTs will again flip the first qubits an even number of times for the encoding of $|0\rangle$, and an odd number of times for the encoding of $|1\rangle$. Since an even number plus an even number is an even number, and an odd number plus an odd number is an even number, for both the encoding of $|0\rangle$ and $|1\rangle$ the first ancillary qubit gets flipped an even number of times, and so the value of the qubit remains $|0\rangle$.

Suppose that there's been a phase flip error on the first block. Then the state after the application of $H^{\otimes 9}$ will be:

$$\frac{\alpha}{\sqrt{8}} |\psi^-\rangle |\psi^+\rangle |\psi^+\rangle + \frac{\beta}{\sqrt{8}} |\psi^+\rangle |\psi^-\rangle |\psi^-\rangle$$

This time when we apply the first three CNOTs, the first syndrome qubit will be flipped an odd number of times for the encoding of $|0\rangle$, and an even number of times for the encoding of $|1\rangle$. Then with the second three CNOTs the syndrome qubits will be flipped an even number of times for the encoding of $|0\rangle$ and an odd number of times for the encoding of $|1\rangle$. Since an even number plus an odd number is an odd number, the first syndrome is flipped in total and odd number of times for both the encoding of $|0\rangle$ and $|1\rangle$, and so it will be set to $|1\rangle$. The same reasoning can be applied with the last six CNOTs, which act on the second ancillary qubit.

Finally we measure the syndrome qubits and we apply $H^{\otimes 9}$ to recover the original state:

$$\frac{\alpha}{\sqrt{8}} |\psi^+\rangle |\psi^+\rangle |\psi^+\rangle + \frac{\beta}{\sqrt{8}} |\psi^-\rangle |\psi^-\rangle |\psi^-\rangle \xrightarrow{H^{\otimes 9}} \frac{\alpha}{\sqrt{8}} (|000\rangle + |111\rangle)^{\otimes 3} + \frac{\beta}{\sqrt{8}} (|000\rangle - |111\rangle)^{\otimes 3}$$

Then we correct the block with the different sign, by applying a Z gate to one qubit of that block, according to the value of the syndrome. Note that the nine qubits code can protect the quantum state against simultaneous phase flip error and three bit flip errors, but only if every bit flip error acts different blocks.

The full circuit is implemented in the notebook `nine_qubit_code.ipynb`.

3.4 Correcting generic errors

Since now we only saw bit flip and phase flip errors, but of course there are many other types of errors. Consider a generic error on a single qubit defined by a matrix $E \in \mathbb{C}^{2 \times 2}$, and consider the four Pauli matrices I , X , Y and Z . As we already said X and Z represent respectively the bit flip and the phase flip error, Y can be seen as a bit flip error followed by a phase flip error and the gate I can be simply seen as the no-error gate. These four matrices span the space of the 2×2 complex matrices, so E can be written as:

$$E = \alpha_0 I + \alpha_1 X + \alpha_2 Y + \alpha_3 Z, \quad \alpha_0, \alpha_1, \alpha_2, \alpha_3 \in \mathbb{C}$$

So we've rewritten E as a linear combination of bit flip and phase flip errors, which we already know how to correct. If a state $|\psi\rangle$ is affected by the error E , then we can write the resulting state as:

$$E|\psi\rangle = \alpha_0 |\psi\rangle + \alpha_1 X|\psi\rangle + \alpha_2 Y|\psi\rangle + \alpha_3 Z|\psi\rangle$$

If we apply a circuit to correct the error on this state, we'll get a syndrome for each term of the superposition:

$$\alpha_0 |\psi\rangle |I \text{ syndrome}\rangle + \alpha_1 X|\psi\rangle |X \text{ syndrome}\rangle + \alpha_2 Y|\psi\rangle |Y \text{ syndrome}\rangle + \alpha_3 Z|\psi\rangle |Z \text{ syndrome}\rangle$$

When we measure the syndrome, the state will collapse to one of the four terms of the superposition: $|\psi\rangle |I \text{ syndrome}\rangle$, $X|\psi\rangle |X \text{ syndrome}\rangle$, $Y|\psi\rangle |Y \text{ syndrome}\rangle$ or $Z|\psi\rangle |Z \text{ syndrome}\rangle$, and we can correct the error according to the syndrome measured and recover $|\psi\rangle$.

4 Stabilizer codes

The correcting codes we saw so far and many others can be described using a particular formalism, called the stabilizer formalism, introduced by Gottesman in [2], which makes it easier to define quantum error correcting codes. In the following section we'll introduce the stabilizer formalism and we'll see how we can apply it to quantum error correction.

We say that a unitary operator U stabilizes the state $|\psi\rangle$ if $U|\psi\rangle = |\psi\rangle$, which means that $|\psi\rangle$ is an eigenstate of U with eigenvalue 1. If V and U stabilize $|\psi\rangle$ then also their products and their inverses stabilize $|\psi\rangle$. Since matrix multiplication is an associative operation, the set of unitary matrices that stabilizes $|\psi\rangle$ form a group under the matrix multiplication operation. Consider the set of Pauli operators $\mathcal{P} = \{\pm I, \pm X, \pm Y, \pm Z, \pm iI, \pm iX, \pm iY, \pm iZ\}$, we then define \mathcal{P}^n as:

$$\mathcal{P}^n = \{A_1 \otimes \cdots \otimes A_n : A_j \in \mathcal{P}, j = 1 \dots n\}$$

which is the set $2^n \times 2^n$ matrices which can be obtained by the tensor product of n Pauli matrices. All the operators in \mathcal{P}^n either commute or anti-commute. A stabilizer \mathcal{S} is defined as an abelian subgroup of \mathcal{P}^n .

Let's now see how we can apply stabilizers to quantum error correction. Consider the circuit for the bit flip correction in figure 1. What we were trying to do was checking if the first and the second qubits were equal, and the same thing for the second and the third qubits, and then use this information to correct the error. We can equivalently say that we were verifying if the encoded state $|\psi_{enc}\rangle = \alpha|000\rangle + \beta|111\rangle$ was an eigenstate with eigenvalue 1 of the operators $Z_1 = Z \otimes Z \otimes I$ and $Z_2 = I \otimes Z \otimes Z$, since:

$$(Z \otimes Z \otimes I)(\alpha|000\rangle + \beta|111\rangle) = \alpha|000\rangle + \beta|111\rangle, \quad (I \otimes Z \otimes Z)(\alpha|000\rangle + \beta|111\rangle) = \alpha|000\rangle + \beta|111\rangle$$

If we apply Z_1 and Z_2 to the three errors detectable by the circuit, we get:

$$(Z \otimes Z \otimes I)(\alpha|100\rangle + \beta|011\rangle) = -(\alpha|100\rangle + \beta|011\rangle), \quad (I \otimes Z \otimes Z)(\alpha|100\rangle + \beta|011\rangle) = \alpha|100\rangle + \beta|011\rangle$$

$$(Z \otimes Z \otimes I)(\alpha|010\rangle + \beta|101\rangle) = -(\alpha|010\rangle + \beta|101\rangle), \quad (I \otimes Z \otimes Z)(\alpha|010\rangle + \beta|101\rangle) = -(\alpha|010\rangle + \beta|101\rangle)$$

$$(Z \otimes Z \otimes I)(\alpha|001\rangle + \beta|110\rangle) = \alpha|001\rangle + \beta|110\rangle, \quad (I \otimes Z \otimes Z)(\alpha|001\rangle + \beta|110\rangle) = -(\alpha|001\rangle + \beta|110\rangle)$$

So if there's no error then the encoded state $|\psi_{enc}\rangle$ is a +1 eigenstate for both Z_1 and Z_2 , therefore Z_1 and Z_2 stabilize $|\psi_{enc}\rangle$. If there's a bit flip error in the first or second qubit, the resulting state will be an -1 eigenstate of Z_1 , while if the error is on the second or third qubit the encoded state will be an -1 eigenstate of Z_2 . If we instead consider $|\psi_{enc}\rangle$ after a bit flip error on two qubits, then:

$$(Z \otimes Z \otimes I)(\alpha|101\rangle + \beta|010\rangle) = \alpha|101\rangle + \beta|010\rangle, \quad (I \otimes Z \otimes Z)(\alpha|101\rangle + \beta|010\rangle) = \alpha|101\rangle + \beta|010\rangle$$

which means that the state $\alpha|101\rangle + \beta|010\rangle$ is a +1 eigenstate of both Z_1 and Z_2 , and the same thing holds for every possible error on more than one qubit. So an error E applied to the encoded state $|\psi_{enc}\rangle$ will be detected only if $E|\psi_{enc}\rangle$ is a -1 eigenstate of at least one of Z_1 and Z_2 . Note that Z_1 and Z_2 are not the only stabilizers of $|\psi_{enc}\rangle$, there are also $Z_3 = Z \otimes I \otimes Z$ and $I^{\otimes 3} = I \otimes I \otimes I$, but these two can be obtained as product of Z_1 and Z_2 since $Z_3 = Z_1 Z_2 = Z_2 Z_1$ and $I^{\otimes 3} = Z_1 Z_1 = Z_2 Z_2$. Therefore we say that Z_1 and Z_2 are the generators of the group of stabilizers of $|\psi_{enc}\rangle$, and we write:

$$\begin{array}{c|ccc} G_1 & Z & Z & I \\ G_2 & I & Z & Z \end{array}$$

which has to be read $G_1 = Z \otimes Z \otimes I$ and $G_2 = I \otimes Z \otimes Z$. The same reasoning can be applied to the phase flip correction, but using as generators of the stabilizer group $G_1 = X \otimes X \otimes I$ and $G_2 = I \otimes X \otimes X$.

In this way we can describe a quantum correcting code by listing the generators of its stabilizers group, which is a simple representation and it allows us to immediately check if an error represented by a matrix E can be detected by the code, since if it is detectable it will anti-commute with at least one generators G_i , or equivalently the state $E|\psi_{enc}\rangle$ will be a -1 eigenstate of G_i . In general, consider an error matrix $E \in \mathcal{P}^n$ and a stabilizer group \mathcal{S} . E can either commute or anti-commute with the operators in \mathcal{S} . If $\exists M \in \mathcal{S}$ such that $ME = -EM$ then $ME|\psi\rangle = -EM|\psi\rangle = -E|\psi\rangle$, thus the error defined by E can be detected. If instead $\forall M \in \mathcal{S}$ we have $ME = EM$ then $ME|\psi\rangle = EM|\psi\rangle = E|\psi\rangle$, which means that this error cannot be detected.

It is possible to show that if we encode k physical qubits in n logical qubits, then the corresponding stabilizers group will have exactly $n - k$ generators (for the bit flip code in fact $k = 1$ and $n = 3$). The generators of the stabilizer group for the 9-qubit encoding are the following:

$$\begin{array}{c|cccccccc} G_1 & Z & Z & I & I & I & I & I & I \\ G_2 & I & Z & Z & I & I & I & I & I \\ G_3 & I & I & I & Z & Z & I & I & I \\ G_4 & I & I & I & I & Z & Z & I & I \\ G_5 & I & I & I & I & I & I & Z & Z \\ G_6 & I & I & I & I & I & I & I & Z \\ G_7 & X & X & X & X & X & X & I & I \\ G_8 & I & I & I & X & X & X & X & X \end{array}$$

Using stabilizers has also the advantage that is pretty straightforward to derive a circuit for the correction, once we have the generators of the code. The idea is to use a circuit like the following:

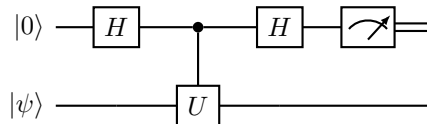


Figure 3: Circuit for error detection with stabilizer codes

If we execute the circuit we get:

$$|0\rangle|\psi\rangle \xrightarrow{H \otimes I} \frac{1}{\sqrt{2}} \left[|0\rangle|\psi\rangle + |1\rangle|\psi\rangle \right] \xrightarrow{I \otimes c - U} \frac{1}{\sqrt{2}} \left[|0\rangle|\psi\rangle + |1\rangle U|\psi\rangle \right] \xrightarrow{H \otimes I} \frac{1}{2} \left[(|0\rangle + |1\rangle)|\psi\rangle + (|0\rangle - |1\rangle)U|\psi\rangle \right]$$

$$= \frac{1}{2} |0\rangle (|\psi\rangle + U|\psi\rangle) + \frac{1}{2} |1\rangle (|\psi\rangle - U|\psi\rangle)$$

If $|\psi\rangle$ is a +1 eigenstate of U then it will only remain the term $|0\rangle |\psi\rangle$ and we'll measure 0 on the first register, while if $|\psi\rangle$ is a -1 eigenstate of U it will only remain the term $|1\rangle |\psi\rangle$ and we'll measure 1 on the first register. In any case we recover the input state $|\psi\rangle$ on the second register and we get on the first register the information of the sign of the eigenvalue associated to the eigenstate $|\psi\rangle$. So if U is a stabilizer of $|\psi\rangle$ we'll always measure 0. In instead $|\psi\rangle$ has been affected by some correctable error E , hence $UE|\psi\rangle = -EU|\psi\rangle = -E|\psi\rangle$, we'll measure 1, and therefore we know that an error occurred.

So the idea is to replicate a circuit like the one in figure 3 for each generator of the code we're dealing with, to check if an error occurred. We'll see an example of this with the five qubit code.

5 Five qubit code

The five qubit code is the smallest quantum error correcting code, in terms of number of qubits used for the encoding, able to detect bit-phase flip errors on a single qubit. It has the following generators:

$$\begin{array}{l|l} G_1 & I \quad Z \quad X \quad X \quad Z \\ G_2 & Z \quad I \quad Z \quad X \quad X \end{array} \quad \begin{array}{l|l} G_3 & X \quad Z \quad I \quad Z \quad X \\ G_4 & X \quad X \quad Z \quad I \quad Z \end{array}$$

The encoding for the state $|0\rangle$ is $|0\rangle_L = \sum_{M \in \mathcal{S}} M |00000\rangle$, where \mathcal{S} is the stabilizer group generated by the above generators, which is composed of 16 operators. In the same way, the encoding of $|1\rangle$ is $|1\rangle_L = \sum_{M \in \mathcal{S}} M |11111\rangle$. We can check that every generator G_i stabilizes $|\psi_{enc}\rangle = \alpha |0\rangle_L + \beta |1\rangle_L$:

$$\begin{aligned} G_i |\psi_{enc}\rangle &= G_i (\alpha \sum_{M \in \mathcal{S}} M |00000\rangle + \beta \sum_{M \in \mathcal{S}} M |11111\rangle) = \alpha \sum_{M \in \mathcal{S}} G_i M |00000\rangle + \beta \sum_{M \in \mathcal{S}} G_i M |11111\rangle \\ &= \alpha \sum_{M \in \mathcal{S}} M |00000\rangle + \beta \sum_{M \in \mathcal{S}} M |11111\rangle \end{aligned}$$

the last equality comes from the fact that multiplying every element of \mathcal{S} by G_i , will produce \mathcal{S} again.

Once we have figured out a circuit for the encoding, for each generator we'll add a syndrome qubit and we'll apply circuit in figure 3, substituting U with the generator:

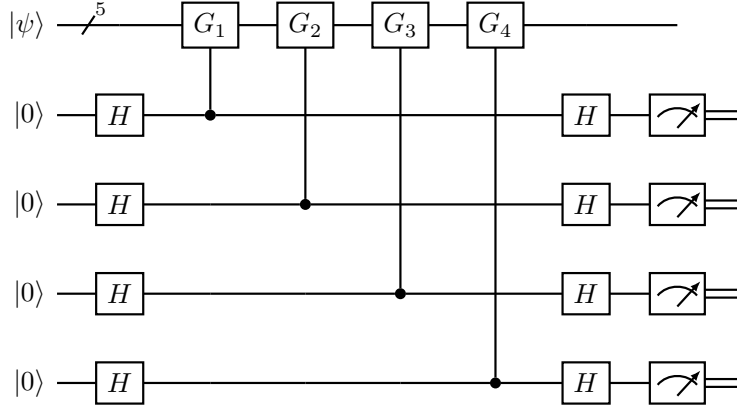


Figure 4: Error detection circuit for the five qubits code

The code has the nice property that each possible X , Y and Z error applied on one of the five qubits, will produce a different syndrome, as shown in below table.

| Error | Syndrome | Error | Syndrome | Error | Syndrome | Error | Syndrome | Error | Syndrome |
|-------|----------|-------|----------|-------|----------|-------|----------|-------|----------|
| X_1 | 0100 | X_2 | 1010 | X_3 | 0101 | X_4 | 0010 | X_5 | 1001 |
| Y_1 | 0111 | Y_2 | 1011 | Y_3 | 1101 | Y_4 | 1110 | Y_5 | 1111 |
| Z_1 | 0011 | Z_2 | 0001 | Z_3 | 1000 | Z_4 | 1100 | Z_5 | 0110 |

The notation X_i means that the gate X is applied to the i -th qubit. In this way we have $2^4 = 16$ distinct syndromes, which can be used to correct the error applying the appropriate gate on the appropriate qubit.

The full implementation of the circuit can be found in the notebook `five_qubit_code.ipynb`.

References

- [1] Peter W. Shor. Scheme for reducing decoherence in quantum computer memory. *Phys. Rev. A*, 52:R2493–R2496, Oct 1995.
- [2] Daniel Gottesman. Stabilizer codes and quantum error correction. *arXiv: Quantum Physics*, 1997.
- [3] Simon J Devitt, William J Munro, and Kae Nemoto. Quantum error correction for beginners. *Reports on Progress in Physics*, 76(7):076001, 2013.
- [4] Michael A Nielsen and Isaac Chuang. Quantum computation and quantum information, 2002.
- [5] Phillip Kaye, Raymond Laflamme, and Michele Mosca. *An Introduction to Quantum Computing*. Oxford University Press, Inc., USA, 2007.