Error Correction Lecture 3

with Dan Browne

March 27, 2016

Contents

1 Correction of a General Error

We can write the density matrix for a general error in terms of Kraus operators ,

$$\rho \to \sum_j E_j \rho E_j^\dagger = \Sigma(\rho) \tag{1}$$

The limitation of this approach is we cannot write down every possible error. For example, an environment that just replaces ρ by any random state cannot be written down in this formalism.

Here, we will assume that ρ is pure, so that $\rho = |\psi\rangle \langle\psi|$. This is a fair assumption, because the codewords should be pure states. Mixed states cannot be perfectly distinguished. thus, the channel has the following effect:

$$\Sigma\left(|\psi\rangle \langle\psi|\right) = \sum_j E_j |\psi\rangle \langle\psi| E_j^\dagger \tag{2}$$

Therefore, we can understand the entire behaviour by studying $E_j |\psi\rangle$ only. If we can correct this, we are good.

Let's now assume that the system is a qubit. This is not necessary, but it will simplify our analysis. We then know that I, X, Y, Z is a basis for linear operators that act on qubits. So we can write

$$E_j |\psi\rangle = (aI + bX + cY + dZ) |\psi\rangle \tag{3}$$

This could perhaps be though of as a 'superposition' of operators. We create a superposition of faulty states that contain Pauli errors. We assume that we have a code that can correct for X and Z errors. We also assume tat X, Y, Z all lead to distinct syndromes.

All errors are orthogonal, since they are also our basis operators. We can detect them and distinguish between them. However, a distinguishable syndrome leads to distinguishable states. Then, in a sense, we 'collapse' to a single X, Y or Z error.

For a formal treatment of this situation, see Nielsen and Chuang. This is also called error quantisation, or error discretisation.

2 The Knill-Laflamme condition

Here, we will introduce a general view of quantum error correction. Here, we imagine a circuit with an error $\mathcal{E}$ and an operator R that allows for both measurement and detection. Let us start with a state $|\psi\rangle$ that is a codeword in the codespace. The error is a general Kraus operator error,

$$\mathcal{E}(\rho) = \sum_j E_j \rho E_j^\dagger \tag{4}$$

The final state of the circuit should ideally be some $\alpha |\psi\rangle$ where α is some proportionality constant.

Then, given a code, which we call C, and an error channel $\mathcal{E}$, we can ask the following question: When does recovery exist?

Recovery can be thought of as a projector onto the code space. We can assume that we are encoding the logical bits, $|00\ldots0\rangle$, $|00\ldots1\rangle$ and so on all up to and including $|11\ldots1\rangle$. We can choose to represent these bit strings as integers:

$$|0\rangle_L, |1\rangle_L, \ldots |k-1\rangle_L \tag{5}$$

Then we can create a projector that projects onto the codespace:

$$P = \sum_{x=0}^{2k-1} |x\rangle \langle x| \tag{6}$$

This is a projector onto the space of all codewords.

Question: Why does the projector have $2k$ element? This could be a copying mistake.

Then, given a code C with code space projector P and error $\mathcal{E}$, a recovery R exists if and only if

$$PE_j^\dagger E_k P = \alpha_{jk} P \tag{7}$$

So an error model can be corrected if this expression is proportional to P, where α_{jk} are the elements of some Hermitian matrix. We can rewrite the above to find

$$\sum_x |x\rangle \langle x| E_j^\dagger E_k \sum_y |y\rangle \langle y| = \alpha_{jk} \sum_z |z\rangle \langle z| \tag{8}$$

Here, x, y, z are orthogonal codewords. Now, consider two additional codewords, $|p\rangle$ and $|q\rangle$. We apply these on each side of the expression above and find

$$\langle p| E_j^\dagger E_k |q\rangle = \alpha_{jk} \delta_{pq} \tag{9}$$

Let us look at the two different situations.

If $q \neq p$, then $\delta_{pq} = 0$ and so

$$\langle p| E_j^\dagger E_k |q\rangle = 0 \tag{10}$$

What we are saying then is that the states $E_k |q\rangle$ and $E_j |p\rangle$ are orthogonal. This is telling us that no error can make us mix up the codewords because we could always distinguish between them .

Question: However, the error could still change one codeword into another. So we are simply saying that this error channel cannot create codewords that are mixed, or superpositions of other codewords.

The next situation is $p = q$ where $\delta_{pq} = 1$. So then

$$\langle p| E_j^\dagger E_k |q\rangle = \alpha_{jk} \tag{11}$$

What we conclude here is that the inner product of $E_k |q\rangle$ and $E_j |p\rangle$ is independent of the state $|p\rangle$. However, note also that α_{jk} could be zero.

Essentially, we want to prevent the environment from learning about the codeword. Let us look at an example.

2.1 Example: The Repetition Code

Recall that it cannot correct Z errors. However, we can also have a value of a qubit by measuring just one single qubit. Imagine that the environment measures one of our qubits by implementing a CNOT gate.

If we trace out the environment, we find that we have the operators

$$E_0 = \frac{I}{\sqrt{2}} \tag{12}$$

$$E_1 = \frac{Z}{\sqrt{2}} \tag{13}$$

We will show that this fails the condition that we said out before. Let E act on the 1st qubit. We obtain

$$\langle 000| (ZII)(III) |000\rangle = 1 \tag{14}$$

$$\langle 111| (ZII)(III) |111\rangle = -1 \tag{15}$$

Note that the two outcomes are different. Thus this code fails the Knill-Laflamme condition. Look at section 10.3 in Nielsen and Chuang for more information.

3 The Steane Code

This code has $(n = 7, k = 1, d = 3)$. It has logical states

$$|0\rangle_L = |0000000\rangle + |1010101\rangle + |0110011\rangle + |1100110\rangle + |000111\rangle + |1011010\rangle + |0111100\rangle + |1101001\rangle \tag{16}$$

There is a large difference in the Hamming weight compared to what we had before. Here, the Hamming weight is 4. This code is actually using superpositions of codewords for a classical Hamming code. We obtain the other codeword by

$$|1\rangle_L = XXXXXXX |0\rangle_L \tag{17}$$

To detect phase flips, we can perform the following measurements:
$IIIXXXX$
$IXXIIXX$
$XIXIXIX$

In order to detect phase flips, we can apply the measurement $\begin{matrix} IIIZZZZ \\ IZZIIZZ \\ ZIZIZIZ \end{matrix}$

Note the symmetry of the detection methods. Furthermore, the logical operators are given by

$$\bar{X} = X^{\otimes 7} \tag{18}$$

$$\bar{Z} = Z^{\otimes 7} \tag{19}$$

Note that every time we write down logical operators, they must retain the original commutation and anti-commutation properties as the original operators. This is the case here as we have an odd number of operators.

4 Example of a CSS Code

CSS stands for Calderbank-Shor-Steane. A CSS code is a special type of stabiliser code (more on this later) which is fundamentally constructed based on classical codes.

Error detection operators are design to only detect one signle type of error. The Steane code is non-degenerate. This means that all detectable errors have a unique syndrome. The Steane code is also smaller than the Shor code that we looked at before, which is another advantage.

In classical code, we have the **Hamming bound**, which denotes a bound on the size of the code.

The Hamming bound is a limit on the parameters of an arbitrary block code.

5 Quantum Hamming Bound

For a non-degenerate code, every error has a unique syndrome. This can be used as a proof of a classical Hamming bound.

For a degenerate code, the above is not true. Recall that a non-degenerate code, for example the Steane code, can correct the following number of errors

$$t = \frac{d-1}{2} \tag{20}$$

where d is the Hamming distance. Then, we can show that a degenerate code has the following bound

$$\sum_{j=0}^{t} \binom{n}{j} 3^j 2^k \leq 2^n \tag{21}$$

Every error here leads to a distinct syndrome.

Let us here outline the idea instead of looking at the actual proof. The proof can be found in Section 10.3.4 in Nielsen and Chuang.

First, note that we have a number of 2^k codewords. We can cound the maximum number of orthogonal states to see this.

The first thing we then do is count the number of different errors. We can have up to t errors of the form X, Y, Z on t different qubits. Then, define $j \leq t$, where j is the number of errors that occurred for this specific case.

Let us count how many error combinations there are. For a number of n physical qubits and j errors, we have a number of

$$\binom{n}{j} \tag{22}$$

errors. On every location location we have a X, Y, Z error. Then, we have

$$3^j \binom{n}{j} \tag{23}$$

distinctive errors. Thus in total, if we consider all values of j, we have

$$\sum_{j=0}^{t} 3^j \binom{n}{j} \tag{24}$$

And given 2^k codewords, we need at least

$$\sum_{j=0}^{t} 3^j \binom{n}{j} 2^k \tag{25}$$

orthogonal codewords. These codewords need to be distinguishable, due to the Knill-Laflamme condition.

However, we have 2^n orthogonal states in the code space, and hence we obtain the inequality. Note that this only applies to non-degenerate codes, whereas for degenerate codes it is still an open question.

So, if we have $k = 1$ and $t = 1$, we find

$$2(1 + 3n) = 2^n \tag{26}$$

Thus, we find

$$n > 4 \tag{27}$$

which means that a 5-qubit code is the smallest code that we can have.

Question: I'm not entirely sure what this condition tells us.

6 Ingredients for the Stabiliser Formalism

Before we look closer at the stabiliser formalism, let us outline what we need. There are two main components that we will make use of: Pauli operators and Group Theory.

Recall that a group is an abstraction of symmetry operations. One example of this is a square. Symmetry transformations that leave the square invariant are rotations of 90, 180, 270 and 360 degrees, and reflections. What features are there that we need to take into account?

We can compose two symmetry operations to et another symmetry. For example, the identity is an example of this.

Recall that symmetries are invertible. In an abelian group, the group elements all commute.

We also have that operations are associative. That is,

$$A(BC) = (AB)C \tag{28}$$

We can write these conditions succinctly as follows: A group G is a set of elements $\{g_i\}$ with a binary operation defined, $g_k \circ g_l$ for all elements that satisfies the following requirements.

1 Closure. For every element in the group, their combination using the binary group combination rule is

$$g_k \circ g_l \in G \tag{29}$$

2 An identity element e, such that

$$e \circ g_j = g_j \tag{30}$$

3 An inverse, such that

$$g_j (g_j)^{-1} = e \tag{31}$$

4 Composition rule is associative.

For us, the last three properties are automatically satisfied by using matrices as group elements. The only thing we need to really concern ourselves with is the requirement of closure.

Recall also that a subgroup is a subset of group elements that is itself a group. For example, it must include an identity.

Another important concept is that of **group generators**. These can be thought of as a sort-of group-theoretical basis.

generators are a subset of elements of the group G such that every element of G can be expressed as a product of generators. For the square, this would be the 90 rotation element – from this we could generate all other rotations. However, we also need one more generator, a reflection. If we combine these, we can generate the full group. We sometimes write the group in terms of its generators:

$$G = \langle a_1, a_2 \dots a_n \rangle \tag{32}$$

Note that the set of generators is not unique. There can be many sets of generators that can create the full group.

The group that we shall study is the Pauli group, or more closely, the N-qubit Pauli group. We write

$$P_N = \langle i, X_j, Z_j \rangle \tag{33}$$

for all $j = 1 \dots n$.

For the $n = 1$ case, we have already seen that

$$Y = iXZ \tag{34}$$

So in order to create Y, we do indeed need to include i as our generators. Therefore, we get the following group elements:

$$\begin{array}{cc} \pm I & \pm iI \\ \pm X & \pm iX \\ \pm Y & \pm iY \\ \pm Z & \pm iZ \end{array}$$

The N-qubit group consists of all N-fold tensor products of X, Y, Z, i with prefactors. Note that $N = 1$ has 16 elements. In general, we get 4^{N+1} elements for the N-qubit group.