CPEN 400Q Lecture 21
Stabilizer codes

Wednesday 26 March 2025

Announcements

- Last content lecture on Monday; presentations for two classes + two tutorials after (attendance expected - come support your classmates!)
- Project rubric available on PrairieLearn
- Quiz 10 on *Tuesday* before presentation
- TA4 due Friday 23:59

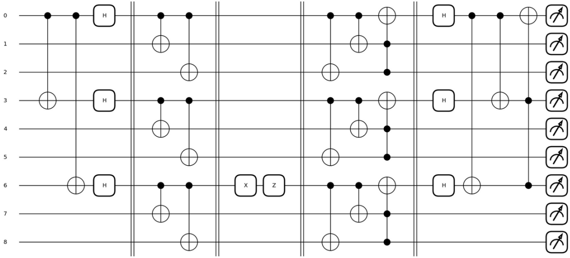
Last time

We made a code to fix phase flip errors by making them look like bit flip errors in a different basis.

$$\mathcal{E}(\alpha|0\rangle + \beta|1\rangle) = \alpha|0\rangle_L + \beta|1\rangle_L = \alpha|+++\rangle + \beta|---\rangle$$

Last time

We *concatenated* the bit and phase flip codes to create a 9-qubit code that corrected *any* single-qubit error.



This worked because if a code can correct a set of error operations $\{E_j\}$, it can also correct linear combinations of them.

Learning outcomes

Today:

1. Outline the conditions under which errors can be corrected
2. Define the stabilizers of a quantum error correcting code
3. Express the bit flip, phase flip, Shor code, and 5-qubit code in the stabilizer formalism

Conditions for quantum error correction

Formal definition of a quantum error correcting code is a *subspace*, C, called the **codespace**.

Example: bit flip code.

Define **projector** onto codespace,

$$P = |000\rangle\langle 000| + |111\rangle\langle 111|$$

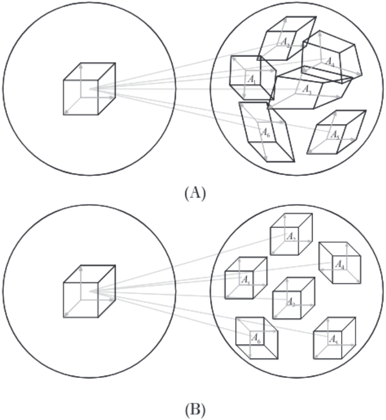


Image: Nielsen & Chuang, Fig. 10.5

Conditions for quantum error correction

$$PE_i^\dagger E_j P = \alpha_{ij} P$$

Theorem 10.1: (**Quantum error-correction conditions**) Let C be a quantum code, and let P be the projector onto C. Suppose $\mathcal{E}$ is a quantum operation with operation elements $\{E_i\}$. A necessary and sufficient condition for the existence of an error-correction operation $\mathcal{R}$ correcting $\mathcal{E}$ on C is that

(10.16)

for some Hermitian matrix α of complex numbers.

If such an $\mathcal{R}$ exists, $\{E_i\}$ is called a *correctable set of errors*.

Bit flip code: recovery revisited

Last week / last class, we considered two recovery circuits for the bit flip code.

Stabilizers

We can consider a more general invariant than parity: eigenvalues w.r.t. special subsets of the Pauli group.

The n-qubit Pauli group is

$$\mathcal{P}_n = \{I, X, Y, Z\}^{\otimes n} \times \{1, -1, i, -1\}$$

Example: Which two-qubit Paulis is this Bell state a $+1$ eigenstate of?

$$|\psi\rangle = \frac{1}{\sqrt{2}} \left(|00\rangle + |11\rangle\right)$$

Bit flip code: stabilizers

Consider our logical states:

$$|0\rangle_L = |000\rangle$$
$$|1\rangle_L = |111\rangle$$

Which three-qubit Paulis are these states $+1$ eigenstates of?

Stabilizers

Let S be a subgroup of $\mathcal{P}_n$.

Let V_S be a set of states that are $+1$ eigenstates for all $P \in S$.

Then, S is the **stabilizer** of V_S, and V_S is stabilized by S.

Facts about S:

- $-I$ is never in S
- all items of S commute
- choosing S uniquely defines the fixed subspace, V_S

Bit flip code: stabilizers generators

Let's determine a *minimal* representation of the group in terms of its **stabilizer generators**.

Bit flip code: stabilizers and logical operations

Logical Z: need a Pauli that

- stabilizes the logical state but is *not* in S
- *commutes* with everything in S
- has the right action on the subspace

Logical X: need a Pauli that

- stabilizes the logical state but is *not* in S
- *commutes* with everything in S
- has the right action on the subspace
- anticommutes with logical Z

Bit flip code: stabilizer measurement and error detection

We can use this formalism to construct circuits for error detection and recovery: simply *measure the stabilizer generators*.

Bit flip code: stabilizer measurement and error detection

Correctible errors all anticommute with at least one of the generators, so we can detect their presence in the syndrome measurement.

Bit flip code: stabilizers and logical errors

Danger: error that *commutes* with all elements of S, but isn't in S.

Phase flip code: stabilizer formalism

Shor code: stabilizer formalism

Name	Operator
g_1	$Z\,Z\,I\,I\,I\,I\,I\,I\,I$
g_2	$I\,Z\,Z\,I\,I\,I\,I\,I\,I$
g_3	$I\,I\,I\,Z\,Z\,I\,I\,I\,I$
g_4	$I\,I\,I\,I\,Z\,Z\,I\,I\,I$
g_5	$I\,I\,I\,I\,I\,I\,Z\,Z\,I$
g_6	$I\,I\,I\,I\,I\,I\,I\,Z\,Z$
g_7	$X\,X\,X\,X\,X\,X\,I\,I\,I$
g_8	$I\,I\,I\,X\,X\,X\,X\,X\,X$
$\bar{Z}$	$X\,X\,X\,X\,X\,X\,X\,X\,X$
$\bar{X}$	$Z\,Z\,Z\,Z\,Z\,Z\,Z\,Z\,Z$

Screenshot: Nielsen & Chuang, chapter 10.5.6.

The Hamming bound

What is the smallest number of physical qubits that we can use to make a logical qubit, and correct any single-qubit error?

General bound:

- n physical qubits
- k logical qubits
- up to t errors

$$\sum_{j=0}^{t} (nj)3^j \cdot 2^k \leq 2^n$$

$$(3^0 + 3n)2 = (1 + 3n)2 \leq 2^n \Rightarrow n \geq 5$$

The smallest code: [[5, 1, 3]]

$$
\begin{aligned}
|0_L\rangle = \frac{1}{4}\Big[&|00000\rangle + |10010\rangle + |01001\rangle + |10100\rangle \\
&+|01010\rangle - |11011\rangle - |00110\rangle - |11000\rangle \\
&-|11101\rangle - |00011\rangle - |11110\rangle - |01111\rangle \\
&-|10001\rangle - |01100\rangle - |10111\rangle + |00101\rangle\Big] \\
|1_L\rangle = \frac{1}{4}\Big[&|11111\rangle + |01101\rangle + |10110\rangle + |01011\rangle \\
&+|10101\rangle - |00100\rangle - |11001\rangle - |00111\rangle \\
&-|00010\rangle - |11100\rangle - |00001\rangle - |10000\rangle \\
&-|01110\rangle - |10011\rangle - |01000\rangle + |11010\rangle\Big]
\end{aligned}
$$

Name	Operator
g_1	$XZZXI$
g_2	$IXZZX$
g_3	$XIXZZ$
g_4	$ZXIXZ$
$\bar{Z}$	$ZZZZZ$
$\bar{X}$	$XXXXX$

Screenshot: Nielsen & Chuang, chapter 10.5.6.

Properties of stabilizer codes

Stabilizer code usually described by notation $[[n, k, d]]$:

- code has 2^{n-k} stabilizer generators
- d = distance (minimum weight of Paulis that commute with everything in S but aren't in S)
- a distance d code can correct $(d-1)/2$ errors

Next time

Next class (last class):

- More on stabilizer codes; fault-tolerant quantum computing
- Quiz on Tuesday (+ presentations)

Action items:

1. TA4 due Friday at 23:59
2. Work on project

Recommended reading:

- From this class: Codebook EC; N&C 10.1-10.5
- For next class: Codebook EC; N&C 10.6