Quantum Error Correction

Ketan Patel

December 6, 2001

Outline

- Motivation
- Introduction to Classical Error Correction
- Examples of Quantum Codes
- Properties of Quantum Codes
- Stabilizer Codes
- Fault Tolerant Computing
- References

Motivation

- Decoherence

  In practice sustaining entangled state is difficult

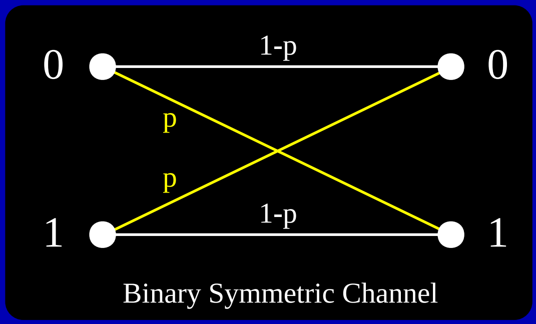
- Error Build-up

  Quantum algorithms involve computations w/o intermediate measurements

- Faulty Gates

  Quantum gates tend to be “noisy”

Classical Error Correction Codes



$P_e = p$

Repetition Code $\begin{cases} 0 \longrightarrow 000 \\ 1 \longrightarrow 111 \end{cases}$ Decode by majority vote

Bit flips in one out of three are correctable $\Rightarrow P_e \sim p^2$

...but rate reduced to $\frac{1}{3}$

Classical Codes (cont.)

A code maps information bits to codewords

Example:

Info Bits			Codewords				
0	0	$\longrightarrow$	0	0	0	0	0
0	1	$\longrightarrow$	1	1	1	0	0
1	0	$\longrightarrow$	0	0	1	1	1
1	1	$\longrightarrow$	1	1	0	1	1

The farther apart (Hamming distance) the codewords, the more errors can be corrected

A code with minimum distance d corrects up to $t = \lfloor \frac{d-1}{2} \rfloor$ errors

This code has length 5, encodes 2 info bits, and has minimum distance 3
We say this is a $[5, 2, 3]$ code

Linear Code Representations

A code is linear if the sum of any two codewords is also a codeword

Generator Matrix (G) Code is span of rows of G

$$C = \left\{ xG \quad \text{for all } x \in \{0,1\}^k \right\}$$

Parity Check Matrix (H) Code is null space of H

$$Hc = \underline{0} \quad \text{if and only if } c \in C$$

Example: for repetition code

$$G = \left. \overbrace{\begin{bmatrix} 1 & 1 & 1 \end{bmatrix}}^{n} \right\} k \qquad H = \left. \overbrace{\begin{bmatrix} 1 & 1 & 0 \\ 1 & 0 & 1 \end{bmatrix}}^{n} \right\} n-k$$

Error Correction via Syndrome

c - original codeword
e - error vector (bits flipped by channel)

Noisy codeword is $c+e$

$$H(c+e) = He \quad \Leftarrow \text{Syndrome}$$

Syndrome identifies error (which can then be corrected)

Note: Syndrome independent of codeword

Example: Repetition Code

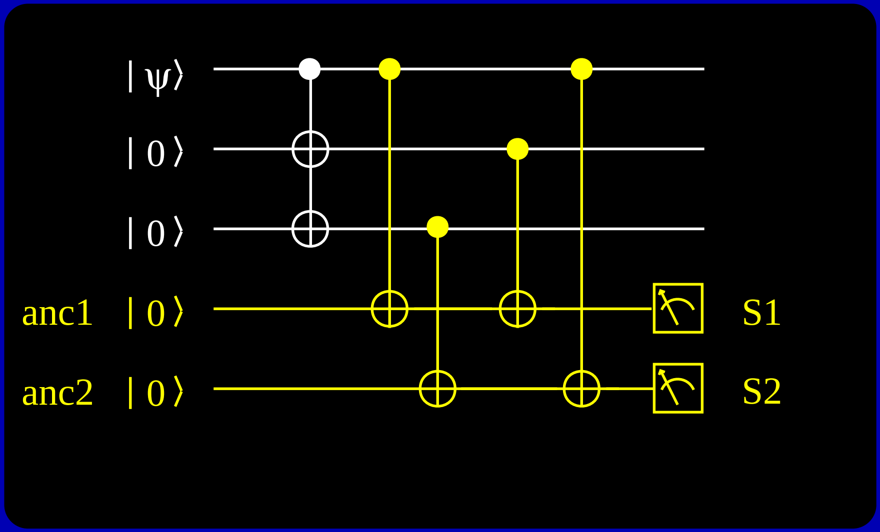
$$\underbrace{\begin{bmatrix} 1 & 1 & 0 \\ 1 & 0 & 1 \end{bmatrix}}_{H} \cdot \underbrace{\begin{bmatrix} 0 \\ 1 \\ 0 \end{bmatrix}}_{c+e} = \begin{bmatrix} 1 \\ 0 \end{bmatrix} \qquad \begin{bmatrix} 1 & 1 & 0 \\ 1 & 0 & 1 \end{bmatrix} \cdot \begin{bmatrix} 1 \\ 0 \\ 1 \end{bmatrix} = \begin{bmatrix} 1 \\ 0 \end{bmatrix}$$

Quantum Issues

- No cloning $|\psi\rangle \not\Rightarrow |\psi\rangle \otimes |\psi\rangle \otimes |\psi\rangle$
- Continuum of Errors e.g. $\alpha|0\rangle + \beta|1\rangle \Rightarrow \alpha|0\rangle + e^{i\phi}\beta|1\rangle$
- No Peeking! Measurement can collapse superpositions

Quantum "Repetition" Code

$$|0\rangle \rightarrow |000\rangle$$
$$|1\rangle \rightarrow |111\rangle$$



S1	S2	action
0	0	none
0	1	flip Q3
1	0	flip Q2
1	1	flip Q1

So we can correct one bit flip

But a single phase error $\begin{bmatrix} 1 & 0 \\ 0 & -1 \end{bmatrix}$ is uncorrectable:

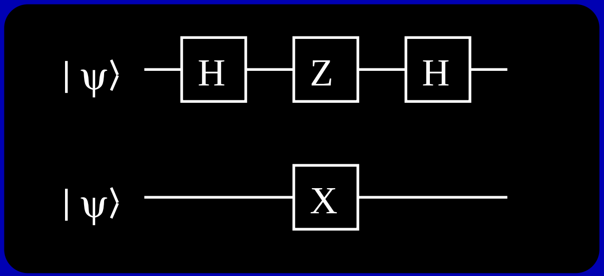
$$\underbrace{|000\rangle + |111\rangle}_{|0\rangle+|1\rangle} \Rightarrow \underbrace{|000\rangle - |111\rangle}_{|0\rangle-|1\rangle}$$

Phase-Flip Code

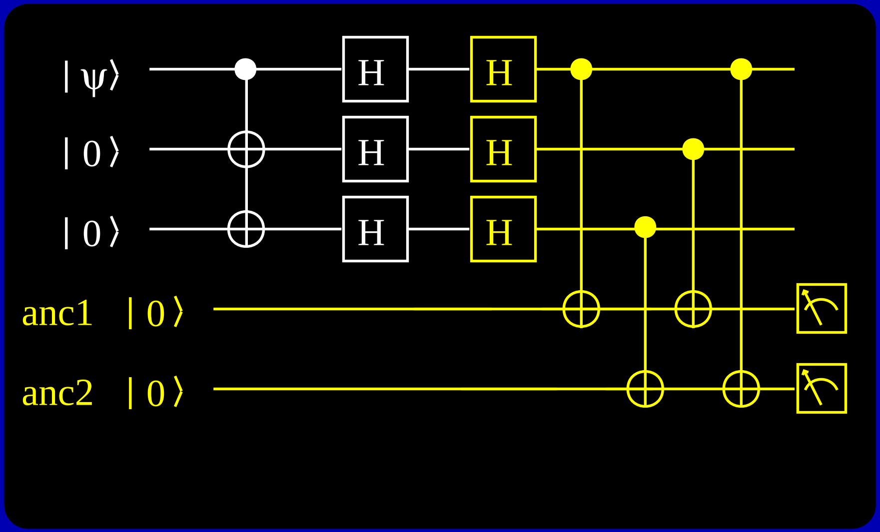
Phase Flips in Hadamard domain

⇕

Bit Flips



So we can correct one phase flip



But now bit flip is uncorrectable!

Shor $[[9, 1, 3]]$ Code

Concatenate Bit-flip and Phase-flip codes

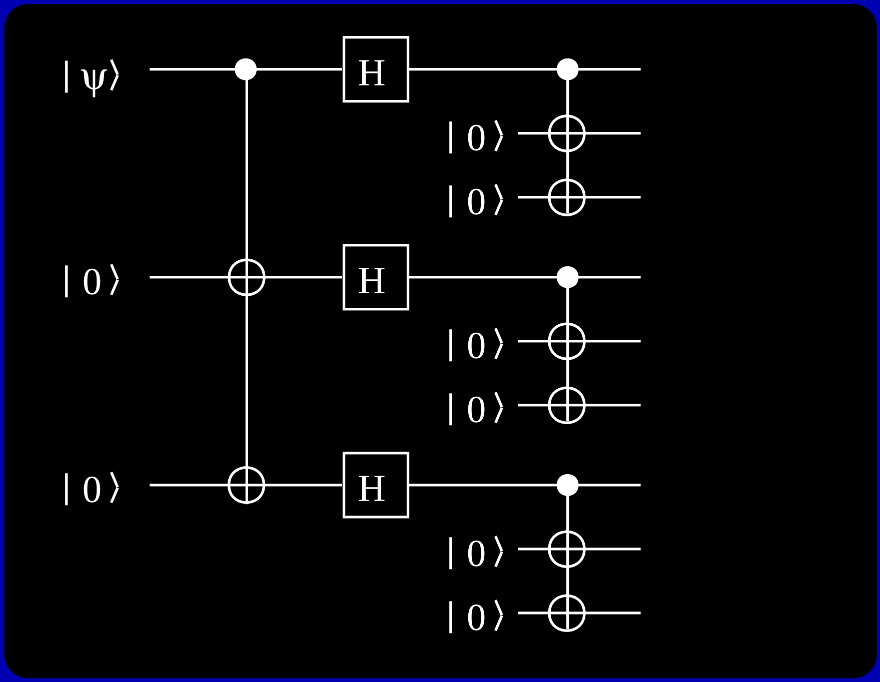
Bit flips are corrected by inner code

Phase flips are corrected by outer code

Example:

Phase flip on Q1-3 $\equiv$ Phase flip on Q1

This can be corrected by the outer code



But what about errors other than bit-flips & phase-flips...

Error Discretization

Four operations form a basis for all 2×2 matrices:

X - Bit-Flips
Z - Phase-Flips
Y - Combined Bit/Phase-Flips
I - No Error

Therefore if we can correct these, we can correct any single qubit error

Intuition: Measuring the error forces it to discretize!

Similarly $\{E_a\} \equiv \{I, X, Y, Z\}^{\otimes n}$ form a basis for errors on n qubits

The weight of an error is the number of qubits acted on by X,Y, or Z

So a t-error correcting code corrects all weight $\leq t$ errors

Conditions for a Quantum Code

- Must not confuse two codewords even in presence of errors
  $\langle c_1|E^\dagger F|c_2\rangle = 0$ (for $\langle c_1|c_2\rangle = 0$)
  Errors map orthogonal subspaces to orthogonal subspaces

- Measurement must not reveal info about codewords
  $\langle c_1|E^\dagger F|c_1\rangle = \langle c_2|E^\dagger F|c_2\rangle$
  The error subspaces must "look" the same for all codewords

These conditions are both necessary and sufficient

Stabilizers

Define G_n as the Pauli Group on n qubits

$$G_n \equiv \{\pm I, \pm iI, \pm X, \pm iX, \pm Y, \pm iY, \pm Z, \pm iZ\}^{\otimes n}$$

All elements of G_n either commute or anticommute

An operator $O \in G_n$ stabilizes a state ψ if

$$O|\psi\rangle = |\psi\rangle$$

Let $S \subset G_n$ and V_S set of states stabilized by every element in S

$$V_S = \{\psi : \quad s|\psi\rangle = |\psi\rangle \quad \text{for all } s \in S\}$$

S is the stabilizer of the vector space V_S

Can consider V_S a stabilizer code

Independent set $\{g_1, \ldots, g_i\}$ generates S

Stabilizer Codes

Generators are analogous to rows of the parity check matrix

$$g_i|c\rangle = |c\rangle \quad \text{but} \quad g_i E|c\rangle = -E g_i|c\rangle = -E|c\rangle$$

Measuring eigenvalues of generators gives syndrome

Example: $[[9,1,3]]$ Shor Code

Phase flip in Q3 ($I \otimes I \otimes Z \otimes I \otimes I \otimes I \otimes I \otimes I \otimes I$)
has syndrome (Recall $ZX = -XZ$)

$$\begin{bmatrix} 1 & 1 & 1 & 1 & 1 & 1 & -1 & 1 \end{bmatrix}^t$$

Note: Phase flip in Q1 gives same syndrome

	Operator								
$g1$	Z	Z	I	I	I	I	I	I	I
$g2$	I	Z	Z	I	I	I	I	I	I
$g3$	I	I	I	Z	Z	I	I	I	I
$g4$	I	I	I	I	Z	Z	I	I	I
$g5$	I	I	I	I	I	I	Z	Z	I
$g6$	I	I	I	I	I	I	I	Z	Z
$g7$	X	X	X	X	X	X	I	I	I
$g8$	I	I	I	X	X	X	X	X	X

Fault-Tolerant Computing

Errors can be introduced during computation

Fault-tolerance: Single failure in any component causes at most one error in each encoded block of qubits at output

Uncorrectable error occurs only if > 1 components fails

If probability of component failure is p, then overall error probability is $\sim cp^2$

FT elementary operations & FT error correction $\Rightarrow$ FT circuit

Fault-Tolerant Logic

Sufficient set of fault-tolerant procedures

- FT Hadamard gate
- FT phase gate
- FT C-NOT gate
- FT $\pi/8$ gate
- FT Measurement
- FT State Preparation

Universal Set

Implementation of fault-tolerant procedures dependent on error correction code

For $[[7,1,3]]$ Steane Code
Bitwise implementation of Hadamard, phase, and C-NOT gates is fault-tolerant

Concatenation

Recursively replace components with FT components & qubits with encoded qubits

Original circuit:

error occurs if single component fails: $P_f \sim p$

Apply procedure once:

error occurs if fault-tolerant component fails

$\Rightarrow$two of its components fail: $P_f \sim cp^2$

Apply procedure twice:

error occurs if fault-tolerant fault-tolerant component fails

$\Rightarrow$two of its fault-tolerant components fail

$\Rightarrow$two of their components fail: $P_f \sim c \cdot (cp^2)^2$

If we concatenate k times

Probability of Failure $\sim (cp)^{2^k}/c$

while circuit size scales by $\sim d^k$

Threshold Theorem

If $p < p_{th} \equiv 1/c$, a circuit with N gates can be simulated with probability of error at most ε using only

$$O\left(\text{poly}(\log N/\varepsilon)N\right) \Leftarrow \text{ polylogorithmically larger circuit}$$

gates.

Threshold is $p_{th} \approx 10^{-5}$–10^{-6}

Application to Factoring

Factoring a 130 digit number (few months on classical)

Quantum algorithm takes ~ 2150 qubits and $\sim 3 \cdot 10^9$ Toffoli gates

[Preskill 97]: Estimates factor of ~ 343 increase for FT implementation

[Steane 98]: Estimates factor of ~ 10 increase

References

- Neilsen & Chuang - Quantum Computation and Quantum Information
- MacWilliams & Sloan - The Theory of Error-Correction Codes
- Gottesman - An Introduction to Quantum Error Correction (2000)
  quant-ph/0004072
- Preskill - Reliable Quantum Computers (1997)
  quant-ph/9705031
- Steane - Efficient fault-tolerant quantum computing (1998)
  quant-ph/980954