Lecture 3 the surface patch

Some ideas from last time

- saw the Stabilizer formalism
- saw a fully quantum error correcting code
- understood that errors would show up as -1 eigenvalues of stabilizer elements (and would then have to fix them)
- saw logical gates (even a two-qubit gate!)

What we've been lacking

- a notion of geometry,
  - how are we going to realise this in a fridge
  - locality of stabilizer elements
- protection against multi-qubit errors

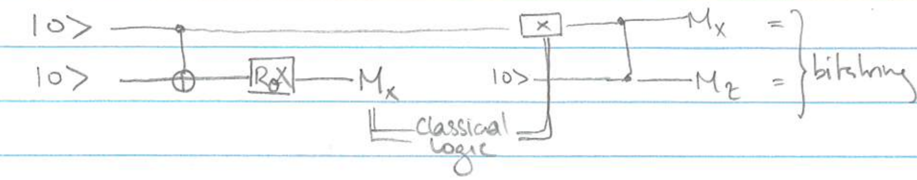
Solution

- surface patch (not surface code)
- today only single logical qubit
- decoding X, Z errors
- more errors allowed if willing to use more physical qubits

  there is a balancing act here, more p qubits implies more noise, but also more protection against noise.

  (notion of threshold)
- only noise on "code qubits" not on ancillas/measurement

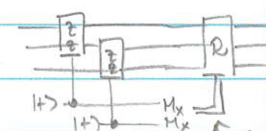
A recap of our first two lectures



We studied the identity operator

$|\psi\rangle$ = ——— got turned into $|\bar{\psi}\rangle$ ≡

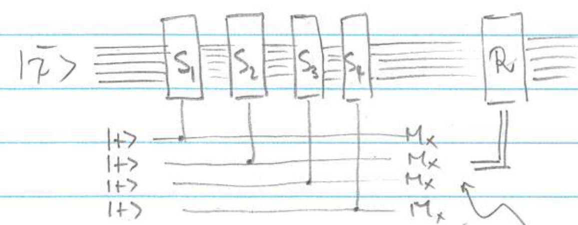
and ancillas were used to stabilize the noise



2^2 lookup table

But those codes only protected either / or against bit/phase flips.
So we introduced a 5-code-qubit code

$|\bar{\psi}\rangle$ ≡ which required 4 ancillas to check "stabilizers"



2^4 possible bit strings

$S_1 = XZZX1$
$S_2 = 1XZZX$
$S_3 = X1XZZ$
$S_4 = ZX1XZ$
1 2 3 4 5

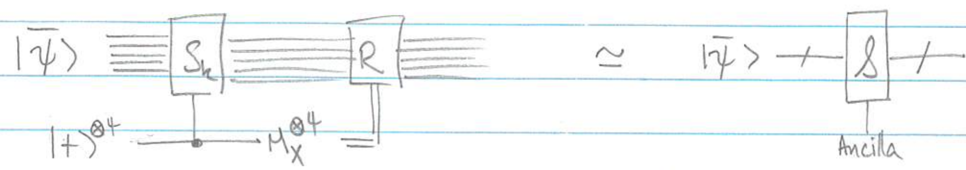
e.g $\underline{X_2 \text{ error}}$ $[X_2, S_k] = 0 \quad k \in \{2,3,4\}$
$[X_2, S_k] = 1 \quad k \in \{1\}$

$\Longrightarrow$ if bit string = 1000
then error was X_2 and recovery is X_2

e.g Z_5 error $[Z_5, S_k] = 0 \quad k \in \{1,2\}$
$[Z_5, S_k] = 1 \quad k \in \{3,4\}$

$\Longrightarrow$ if bit string = 1100
then error was Z_5 and recovery is Z_5

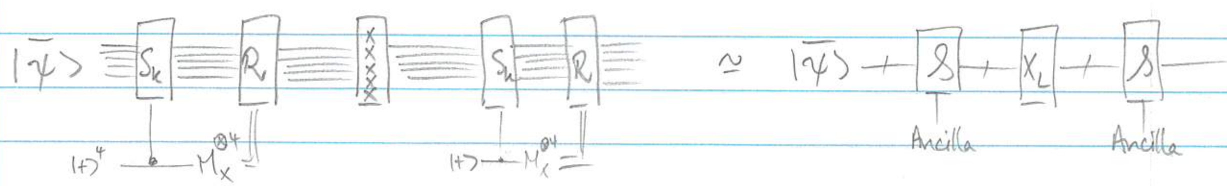
Lets write this extremely compactly



We need to apply gates (not just the identity) and this is very hard to do if we want to respect the code

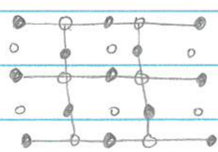
One way of respecting code is to apply gates "transversally"

- the operators XXXXX, ZZZZZ both preserve the code space (they each commute with all the S_k)
- they also interact with themselves in the same way as our old friends X, Z.
  - before $\{X, Z\} = 0$ (anti commute)
  - now $\{XXXXX, ZZZZZ\} = 0$
- transversal because if one error happens when I apply XXXXX then my code will correct for this



- we'll look at the original surface patch of Kitaev - (Bravyi)
- uses roughly 2x the number of qubits than an optimised rotated version
- qubits are split up into 3 types playing different roles
  - many different languages can be used to talk about the surface patch setup :
    - random bond ising model
    - stabilizer code
    - chain complex
  - each has their own notation / language / vocabulary.

- We build a square lattice (degree 2 regular square simplicial complex)
- place qubits on vertices / edges / faces
- careful about the boundary of our lattice (two rough/smooth components)



- code qubits
- ancilla qubits

- ancilla qubits on edges / faces will measure X/Z stabilizers of all code qubits adjacent to said ancilla.

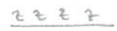
- let d = # of horizontal lines = # of vertical lines + 1 .
  - number of code qubits $= d^2 + (d-1)^2$
    $= 2d^2 - 2d + 1$
  - number of ancilla qubits = one less.

number of degrees of freedom is therefore $1 \equiv 1$ logical qubit hidden here.

there's actually a hamiltonian for this surface patch :

$$A_v = \prod_{e \in \delta v} X_e \qquad B_f = \prod_{e \in \partial f} Z_e \qquad H_{s.p.} = -\sum_v A_v - \sum_f B_f$$

- logical operators

· $X =$  $Z =$ 

· deformation of Z_L by A_f operators.

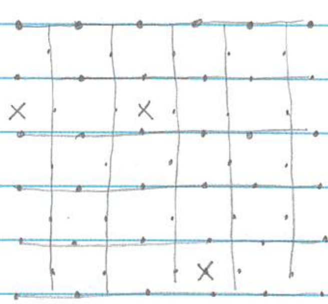
· measurement of X, Z

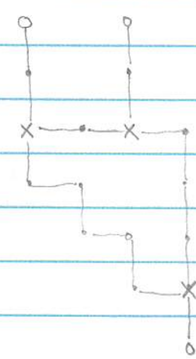
eg X: note $X_L = X_1 X_2 \cdots X_d$

so if I measure $X_1, \dots, X_d$, then the parity will determine X_L

- single code qubit errors
- multi code qubit errors
- minimum weight perfect matching (draw following example)

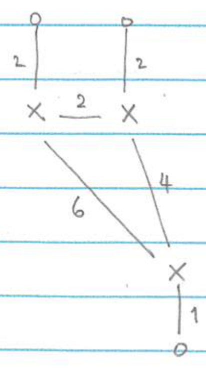
distance 6 code





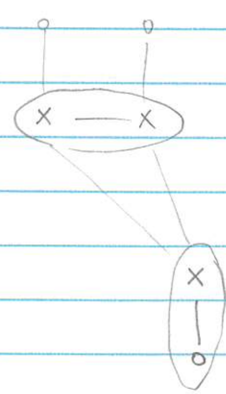
· 3 Z-stabilizer errors detected.

· find probable X-errors which occurred (many possible pictured above)

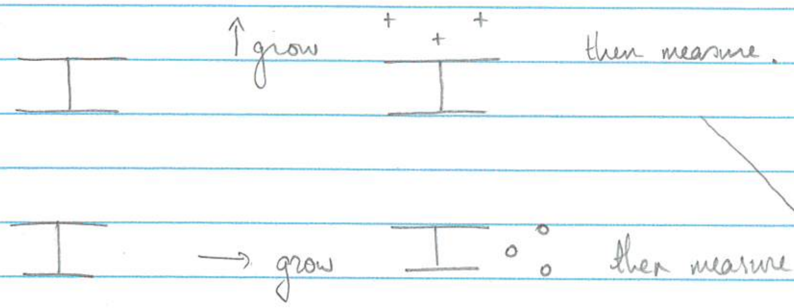


· run MWPM

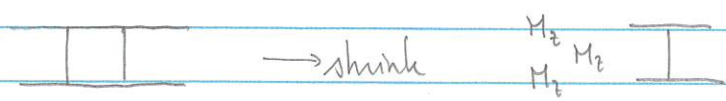
· apply recovery algorithm.



Moving surface patches by growing/shrinking.



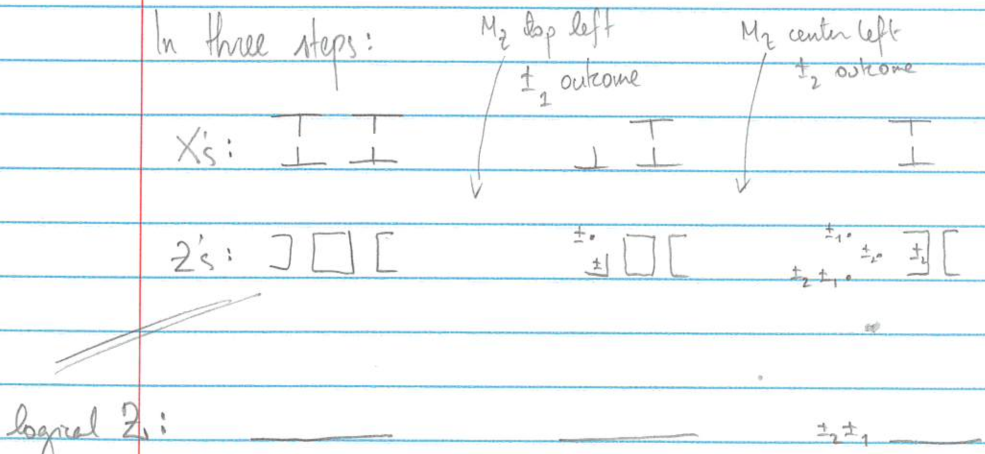
- X stabs = +1
- Z stabs = ±1
- fix up Z stabs with X-gates
- logical Z down bottom won't be affected.



may need to apply logical X gate

- X stabilizers will be good
- left most Z-stab might pick up −1 factor
- logical Z might also pick up −1 factor
- last M_z measurement redundant

In three steps:



if $\pm_2 = -1$ then can apply X to top left code qubit of new surface patch.

if $\pm_2 \pm_1 = -1$ then need to apply logical X gate