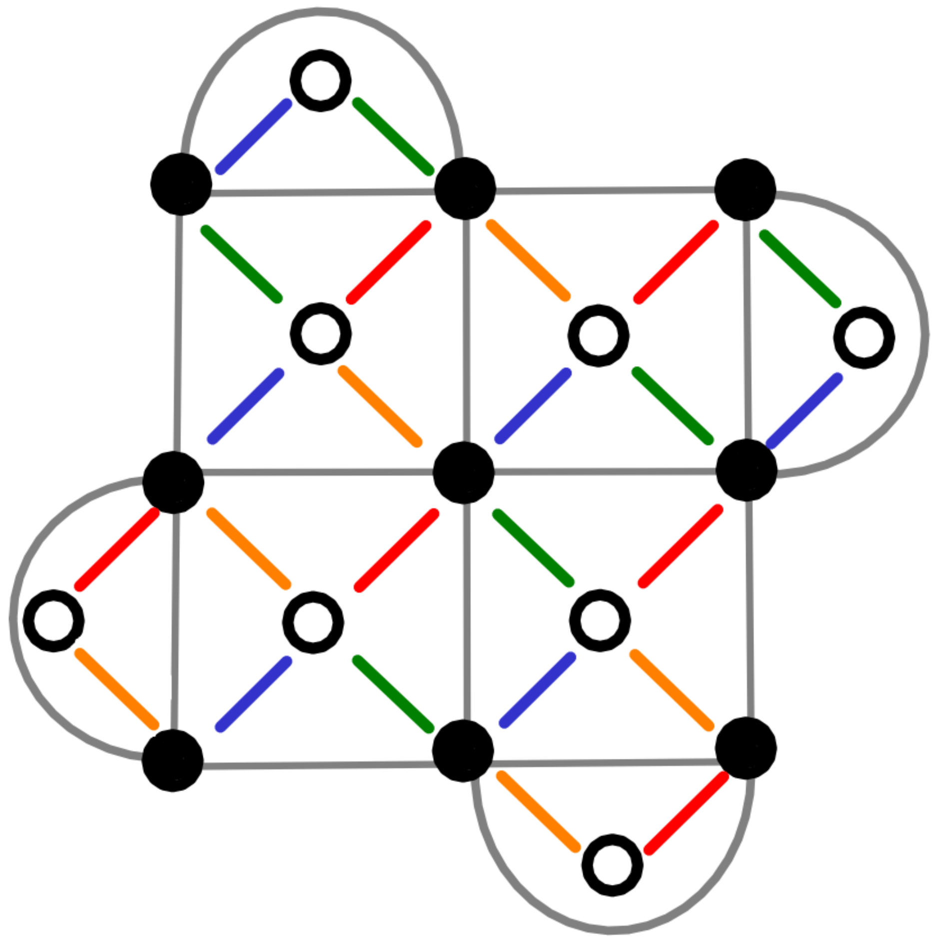
Introduction to the Surface Code

Dr James R. Wootton
University of Basel



UNI BASEL

QSIT Quantum Science and Technology

National Centre of Competence in Research

Towards a better quantum code

- How does the repetition code protect against bit flip noise (σ_x) ?
  - An isolated σ^x creates a pair of *defects* — 0 0 0≠1≠0 0 0 0 0 0 0
  - Further σ^xs can move the defects — 0 0 0≠1 1≠0 0 0 0 0 0
  - Or create new pairs of defects — 0 0 0≠1 1≠0≠1≠0 0 0 0
  - Or annihilate pairs of defects — 0 0 0≠1 1 1 1≠0 0 0 0
  - A distance of >d/2 is needed for a logical error — 0 0 0≠1 1 1 1 1 1≠0 0
  - Then decoding will complete the job, pulling the defects off the ends — ≠1 1 1 1 1 1 1 1 1 1 1≠
- The code is like a ‘universe’ in which the defects are its particles
- Bit flips create and manipulate these particles, but only large scale effects can cause a logical error

Towards a better quantum code

- Why doesn't the repetition code protect against phase flip noise (σ_z) ?
- Measurement is too easy, even when the information is encoded

0 0 0 0 0 0 0 0 0 0 0   1 1 1 1 1 1 1 1 1 1 1

- Once errors are removed, a quick peek at any qubit reveals the stored information
- If it is easy for us to see, it is easy for the environment to dephase
- Consider measuring in the X basis instead

$$|\pm\rangle = \frac{1}{\sqrt{2}}\left(|0\rangle \pm |1\rangle\right) \rightarrow \frac{1}{\sqrt{2}}\left(|000\rangle \pm |111\rangle\right)$$

- Requires multi qubit process for the encoded states

Towards a better quantum code

- This code treats the X and Z basis of the qubit very differently

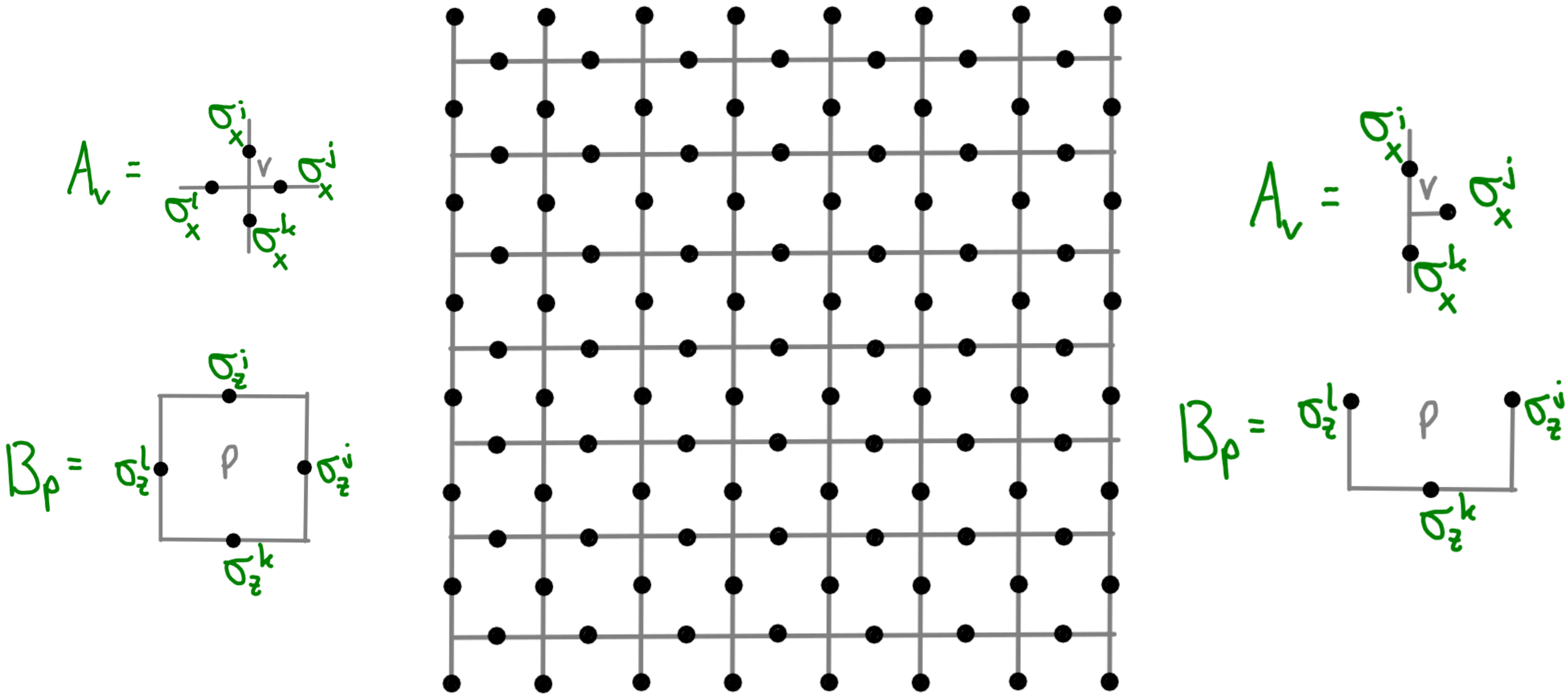
- We need a code that treats them the same

  - ✓ σ_x creates particle-like defects that can be detected
  - ✓ Large scale effects are needed for a logical bit flip
  - ✓ Multiqubit measurement needed to distinguish encoded $|+\rangle, \ |-\rangle$

  - ✓ σ_z creates particle-like defects that can be detected
  - ✓ Large scale effects are needed for a logical phase flip
  - ✓ Multiqubit measurement needed to distinguish encoded $|0\rangle, \ |1\rangle$

- Other methods of generalizing the repetition code also exist, like the Shor code
- But these don't create new universes, and are therefore boring

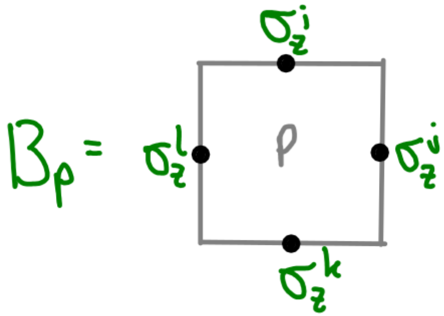
The Surface Code

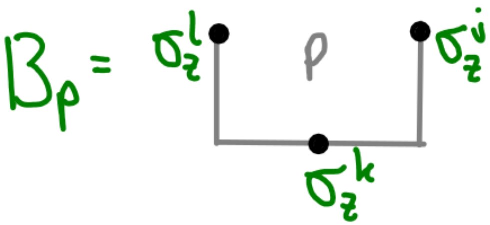


- To do this, we put our grid on a 2D lattice
- The $\sigma_z^j \sigma_z^{j+1}$ observables between neighbouring qubits become ones for qubits around plaquettes
- Similar observables for σ_x are defined on vertices

The plaquette operators

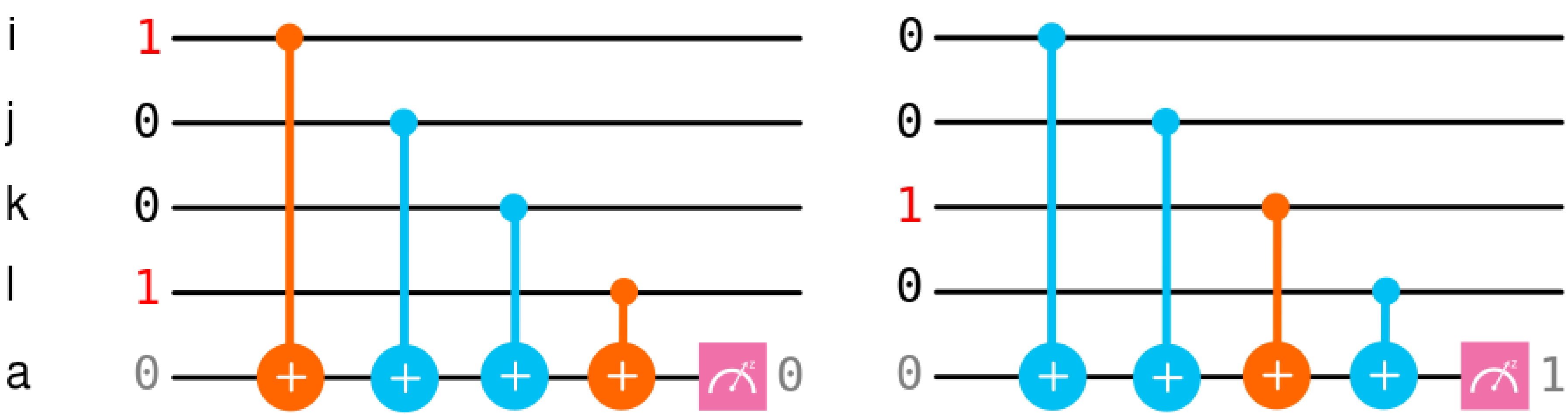
- Let's focus on the plaquette operators





- Generalization of the measurement in the repetition code

- Can be similarly implemented with the controlled-NOT

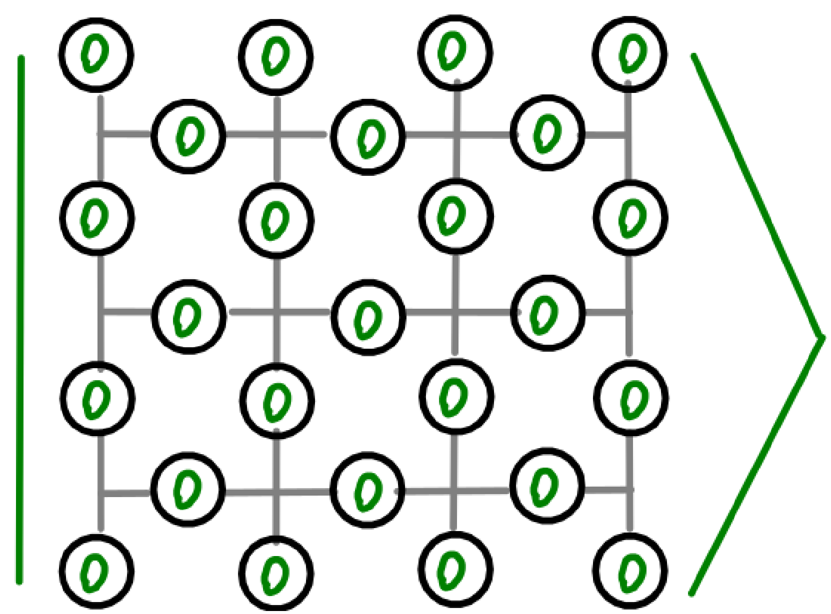


- They tell us whether there is an odd or even # 1s around the plaquette

The plaquette operators

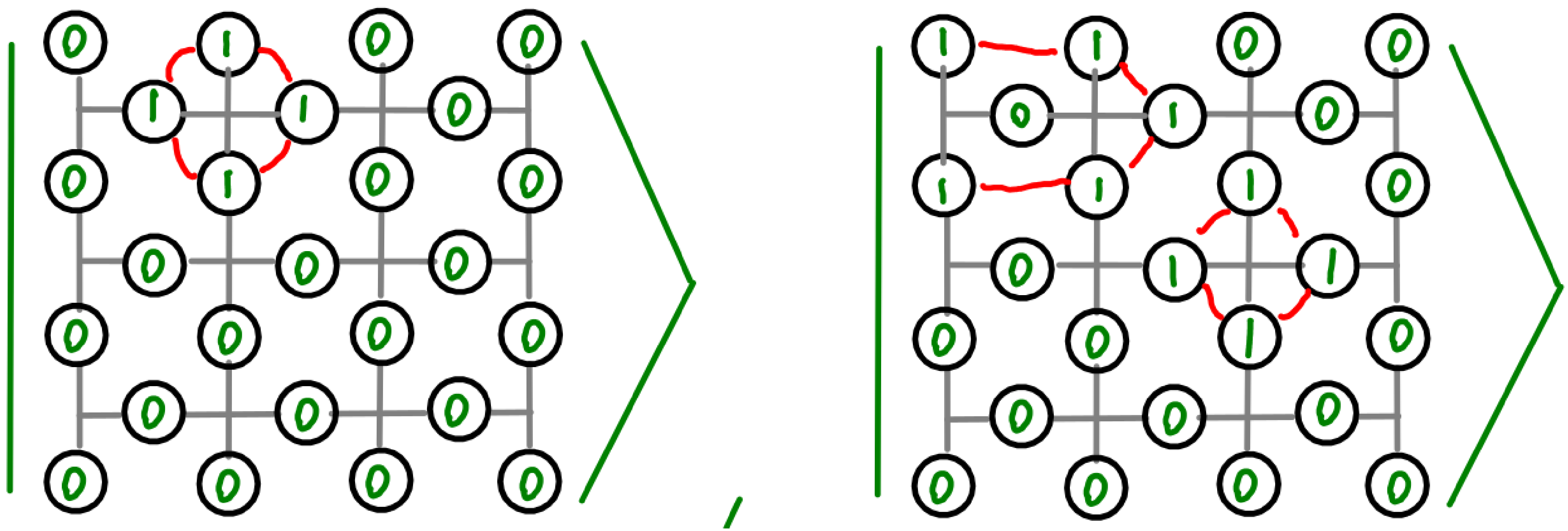
- How do we store a bit in this code?
- Valid encoded states are those for which the measurements don't detect an error
- We associate this with outcome 0, so all plauettes need an even # 1s
- Let's again encode 0 with the 'all qubits are 0' state

$$|0\rangle \rightarrow$$



The plaquette operators

- There are other ‘nearby’ states that have the same results for plaquette measurements



- They can’t be our encoded 1, because they differ by only a few bit flips

- So let’s treat them as other possible ways to encode 0

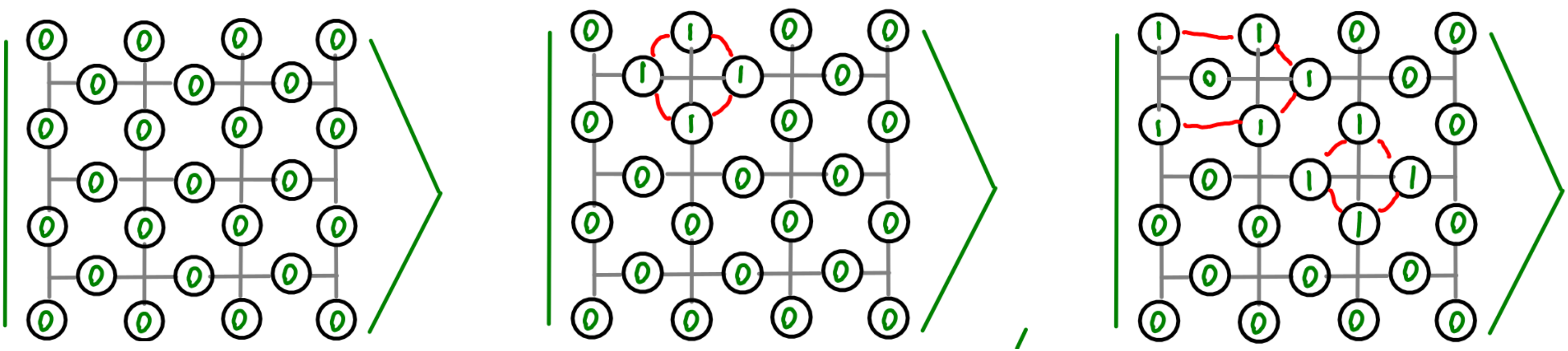
Encoding 0 and 1

- Given any possible encoding for 0:

    1) Pick a vertex
    2) Apply a bit flip on all qubits around the vertex

- Now you have another possible encoding for 1

- This generates an exponentially large family

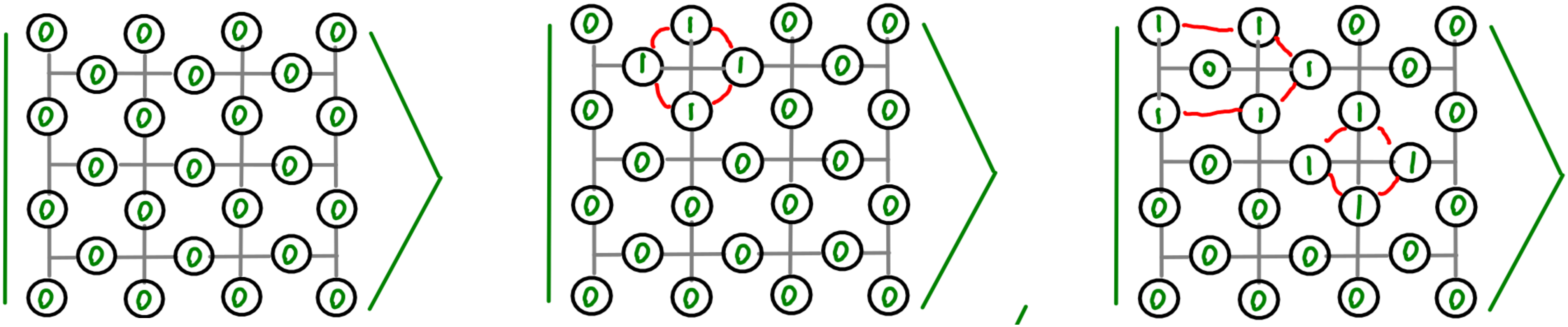


Encoding 0 and 1

- The states in this family can be very different

- But there is one feature shared:
  A line from top to bottom will always have an even number of 1s

- This is how we can measure our encoded 0 state

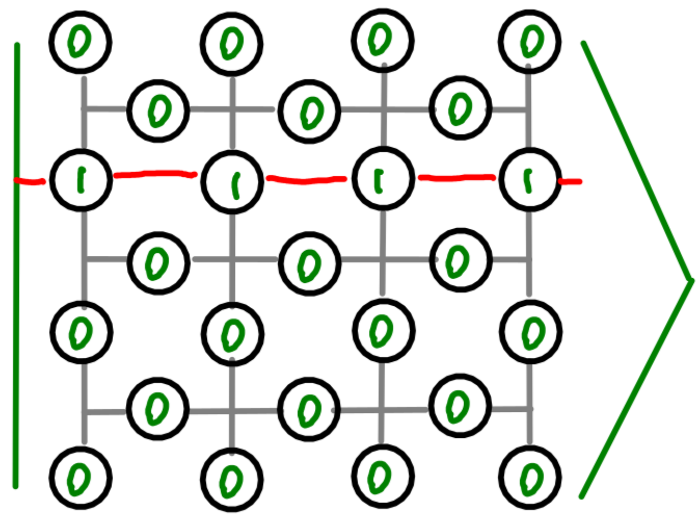


- And it gives us a hint on how to encode a 1

Encoding 0 and 1

- As our basic encoded 1, we can use a bunch of 0s with a line of 1s across

$|1\rangle \rightarrow$



- This also spawns an exponentially large family

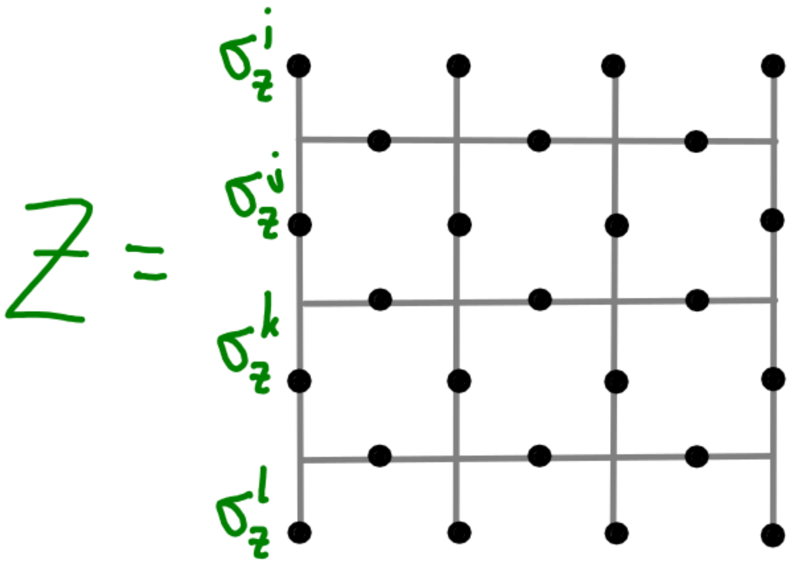
- For each state in that family, the number of 1s on a line from top to bottom is odd

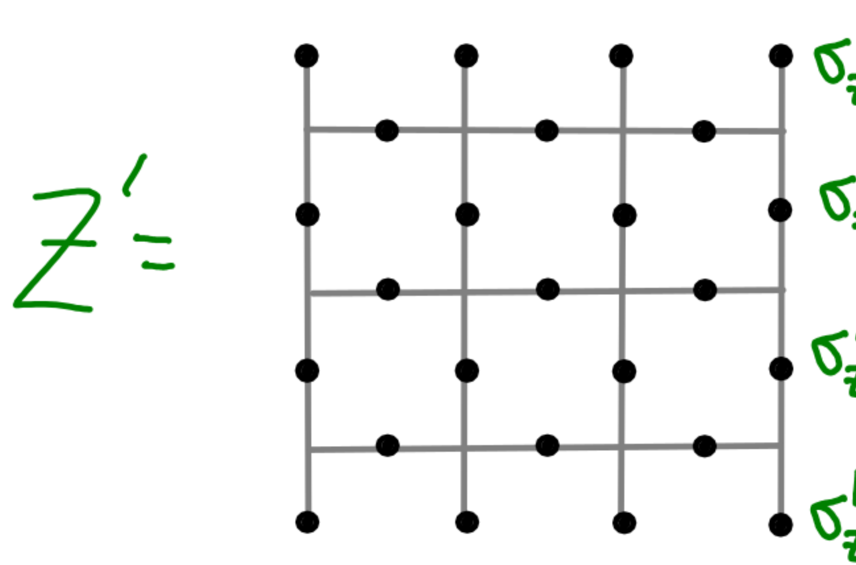
- Measuring our encoded bit has become hard (which is good!)

X and Z for encoded qubit

- Measuring 0 and 1 corresponds to measuring an observable Z for the encoded qubit

- The observable that detects what we need is



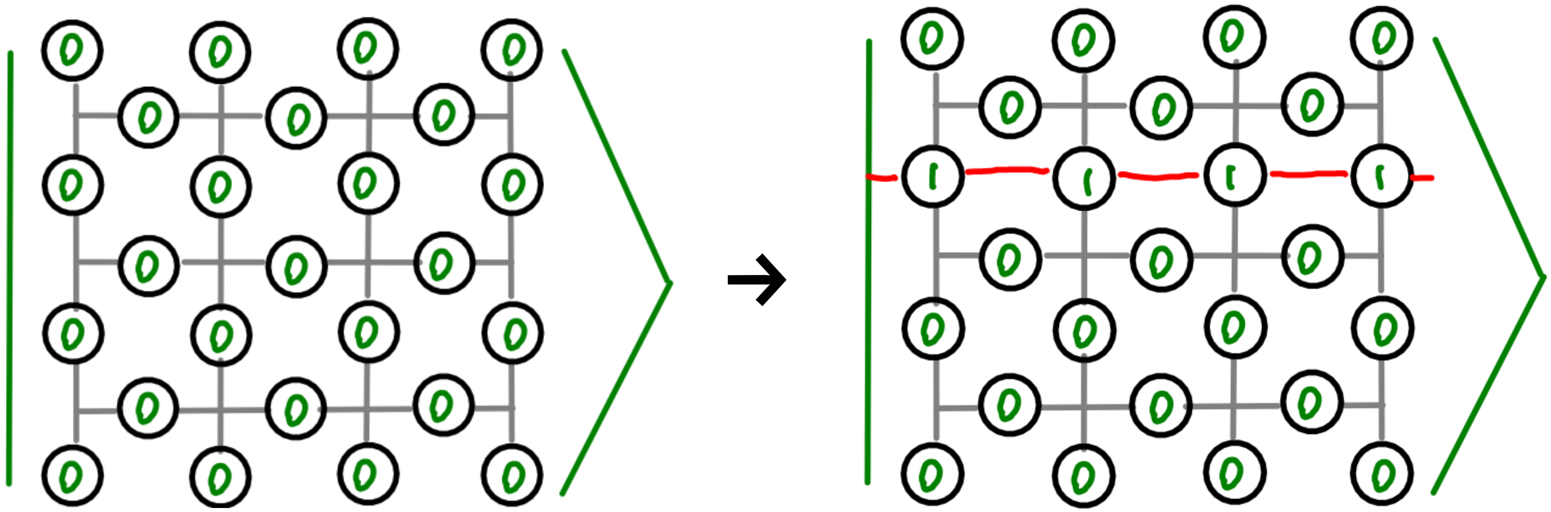


- Or the same on any line from top to bottom

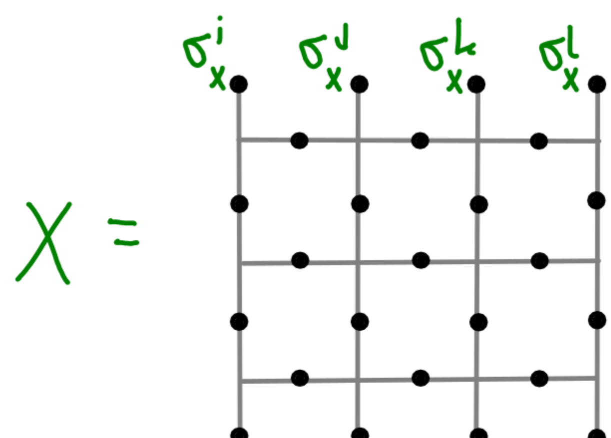
- If we use the edges, we can think of them as large and unenforced plaquettes

X and Z for encoded qubit

- If we want to do a bit flip on the encoded bit, clearly we need a line of flips from left to right



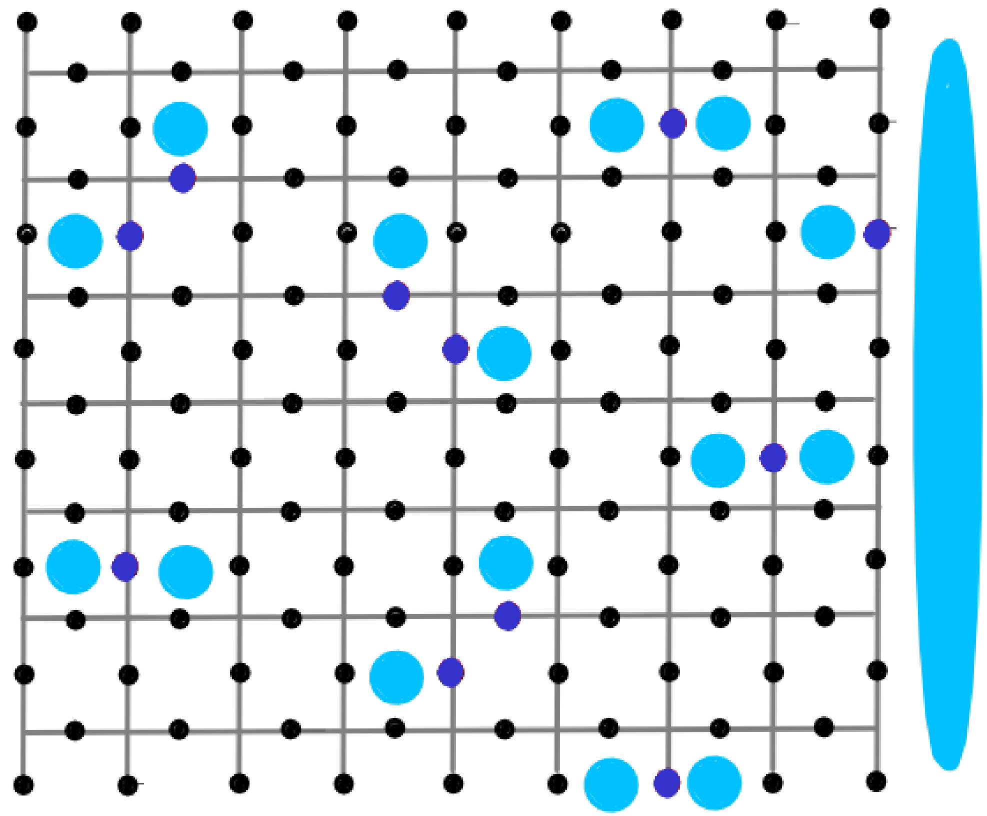
- So the X operation for an encoded qubit is



- Or the same on any other line across

Effects of errors

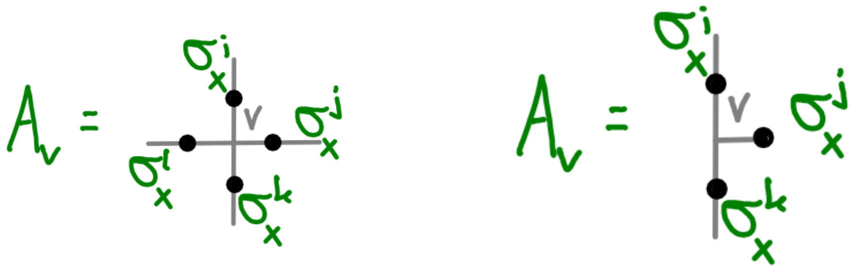
- What happens when a σ_x is applied?
- Changes measurement outcome for neighbouring plaquettes
  - An isolated σ_x creates a pair of *defects*
  - Further σ_xs can move the defects
  - Or create new pairs of defects
  - Or annihilate pairs of defects
  - A distance of *>d/2* is needed for a logical error, where *d* is the width



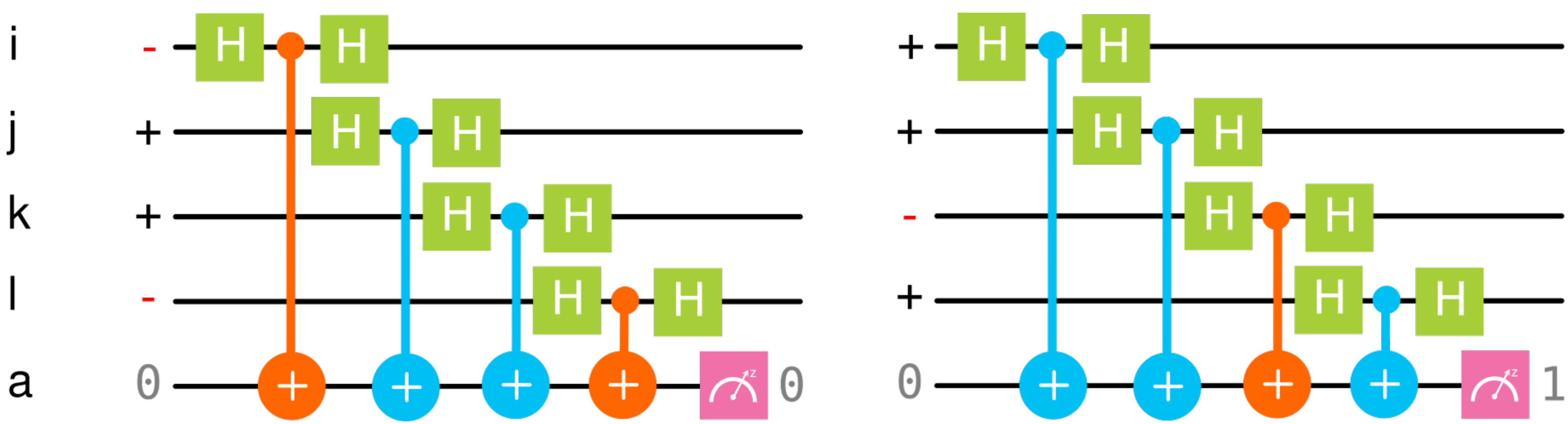
- With the plaquette operators we can encode and protect a bit using Z basis states

Vertex operators

- Now forget the plaquettes, and focus on the vertices



- These can also be measured with controlled ops and an ancilla

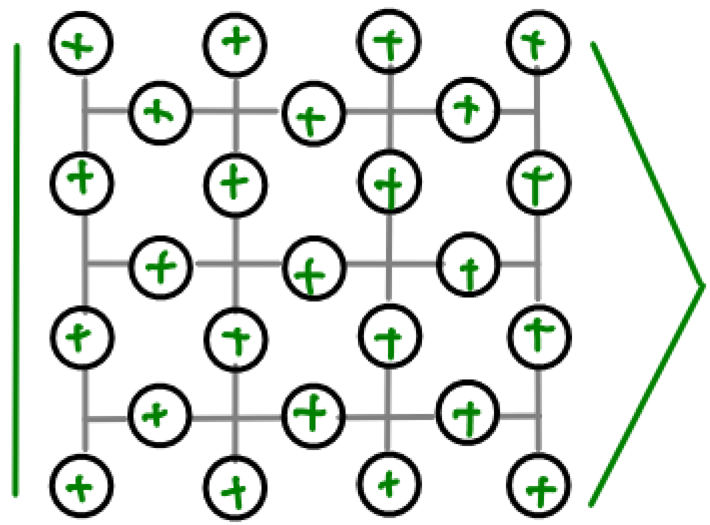


- Looks at $|+\rangle$ and σ^x states, and tell us whether there is an even number of $|-\rangle$ s

Encoding + and -

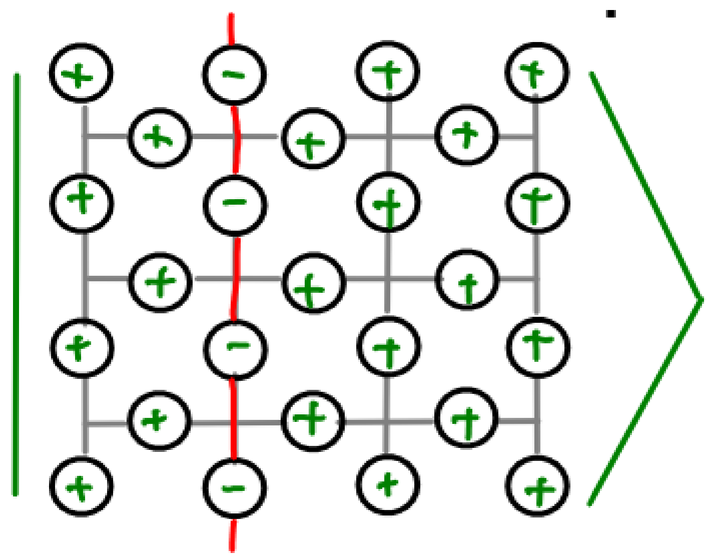
- We can also store a bit using only the vertex ops
- Let’s associate this with the x basis instead

$|+\rangle$ →



+ those that differ by a plaquette of σ_zs

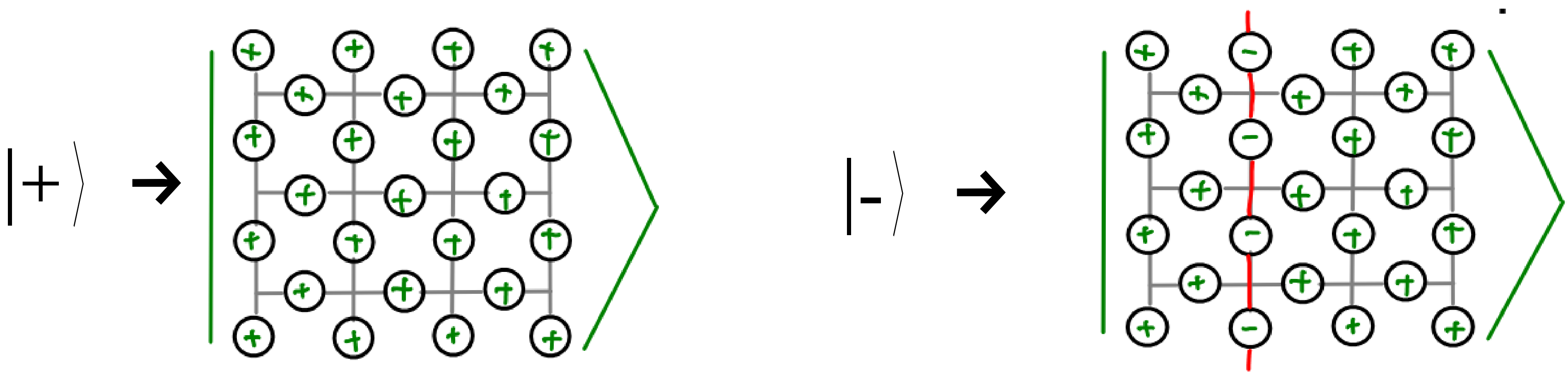
$|-\rangle$ →

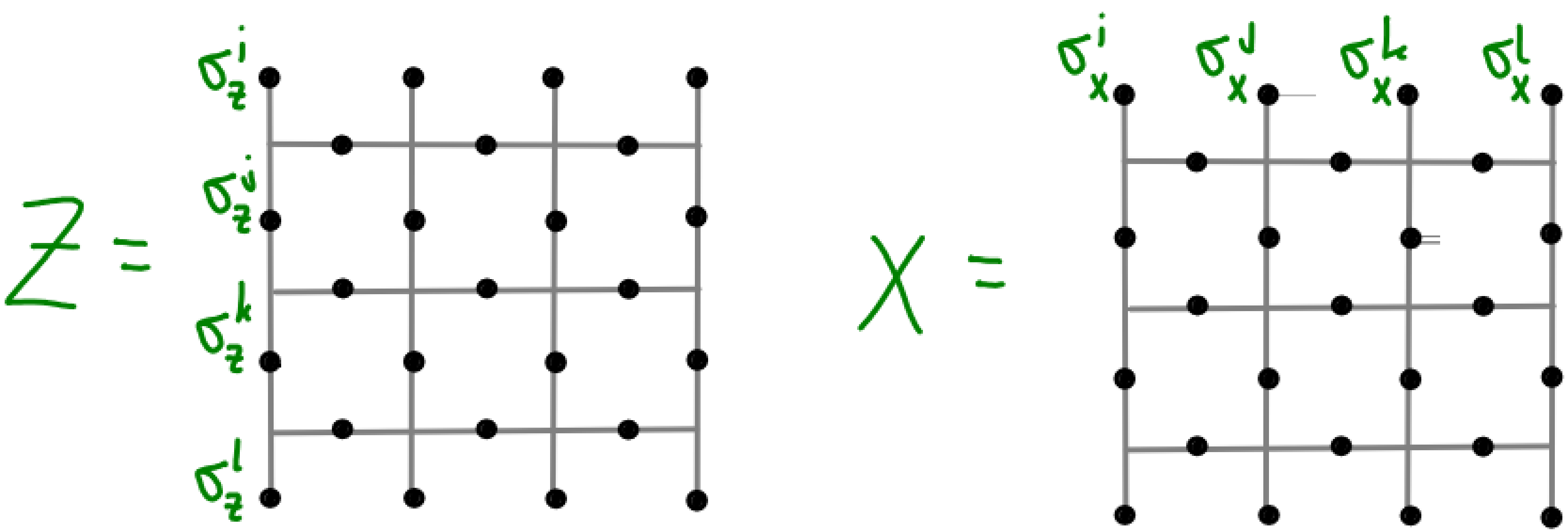


+ those that differ by a plaquette of σ_zs

Encoding + and -

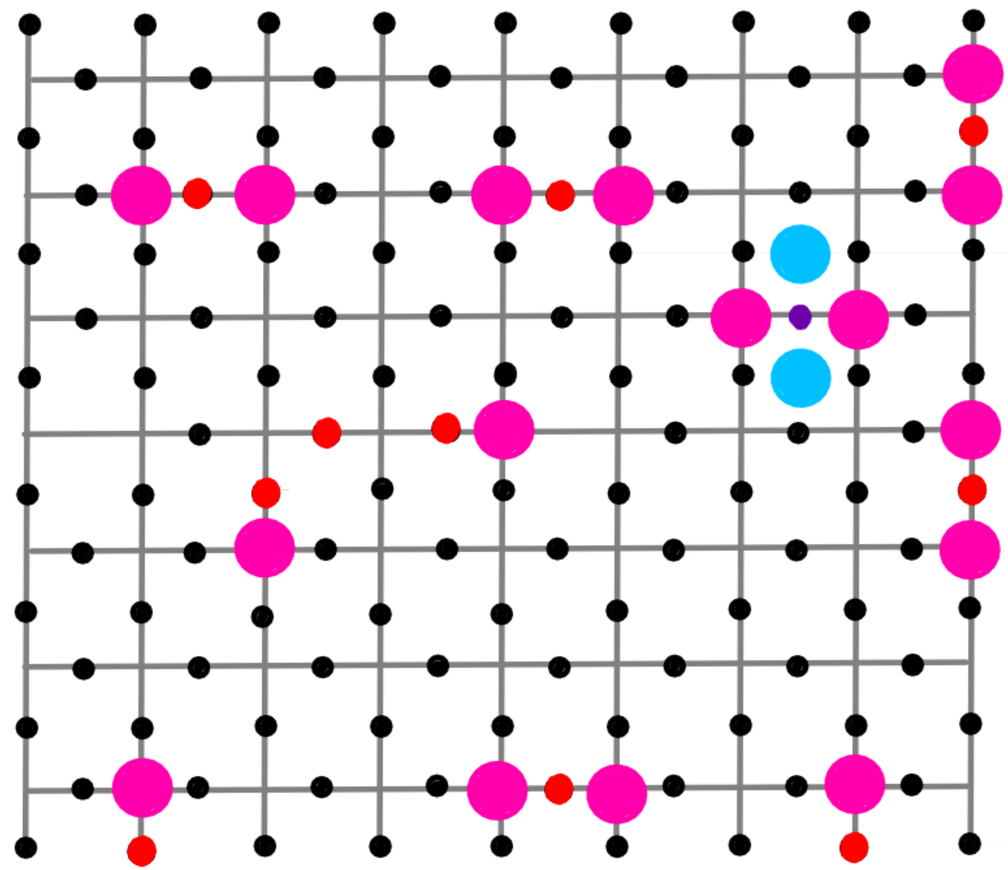
- This leads to exactly the same logical operators as before





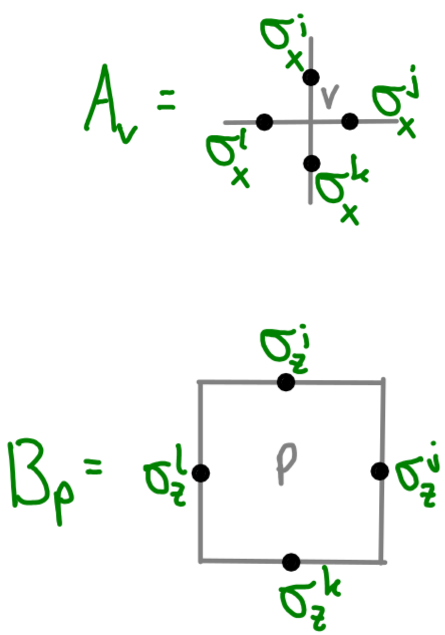
Effects of errors

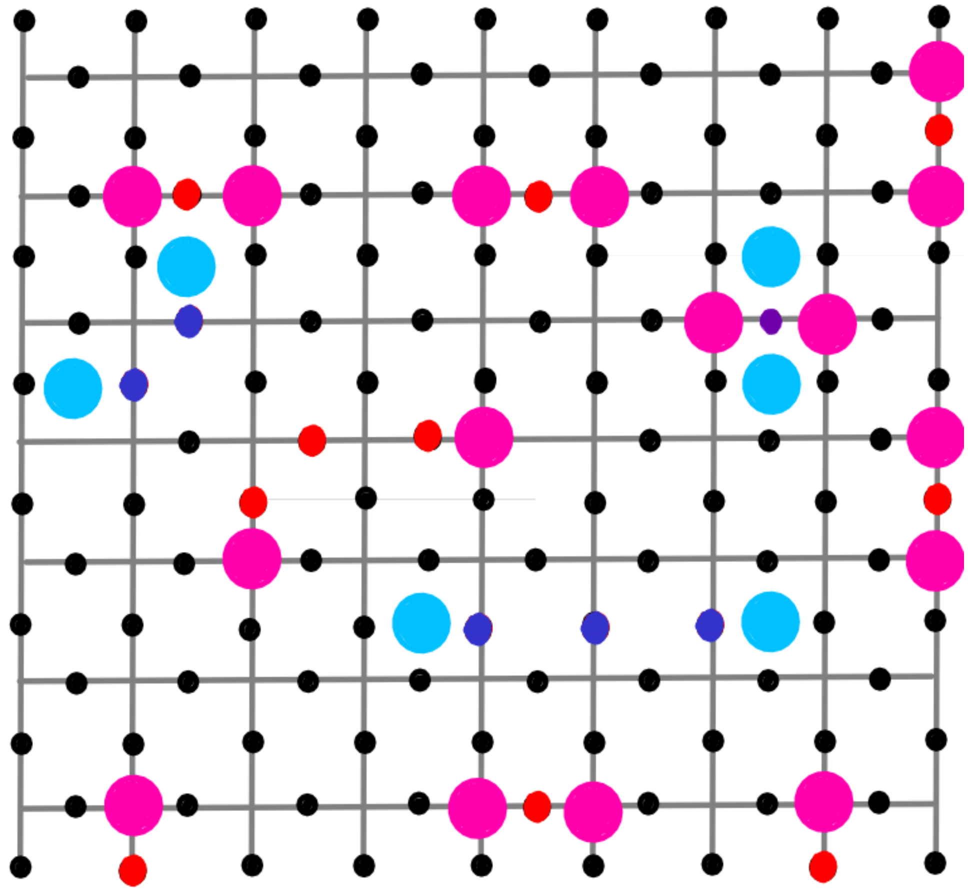
- What happens when a σ_z is applied?
- Changes measurement outcome for neighbouring plaquettes
  - An isolated σ_z creates a pair of *defects*
  - Further σ_zs can move the defects
  - Or create new pairs of defects
  - Or annihilate pairs of defects
  - A distance of *>d/2* is needed for a logical error, where *d* is the height

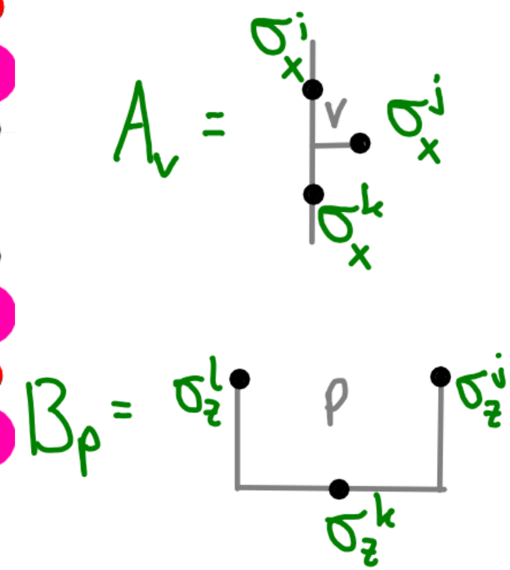


- With the vertex operators we can enocde and protect a bit using X basis states

Vertex and plaquette operators together



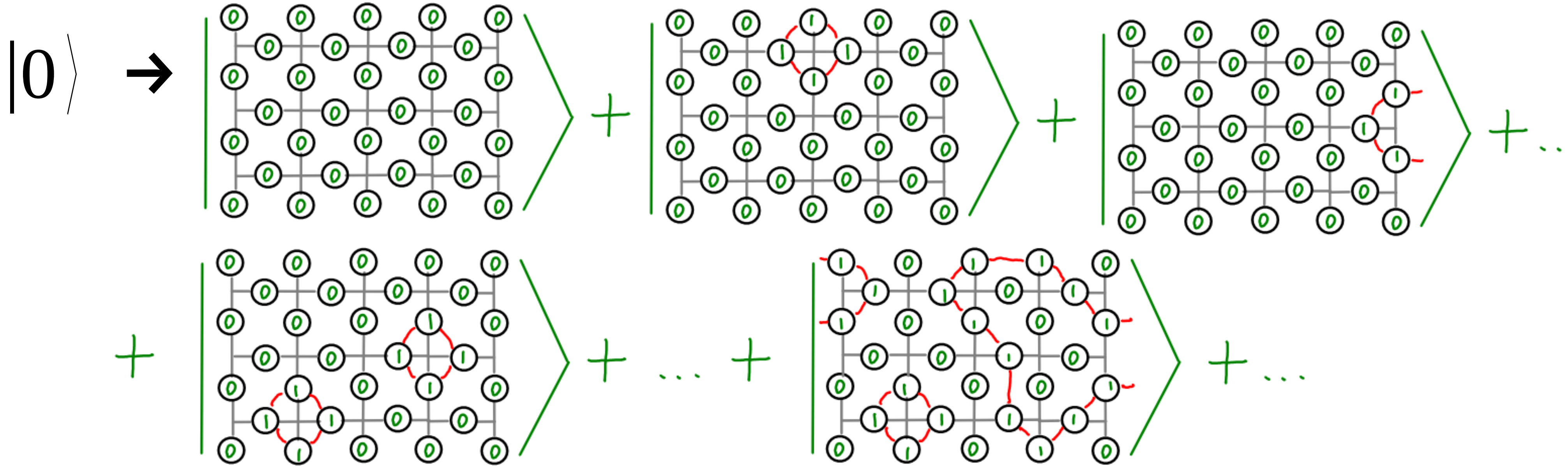




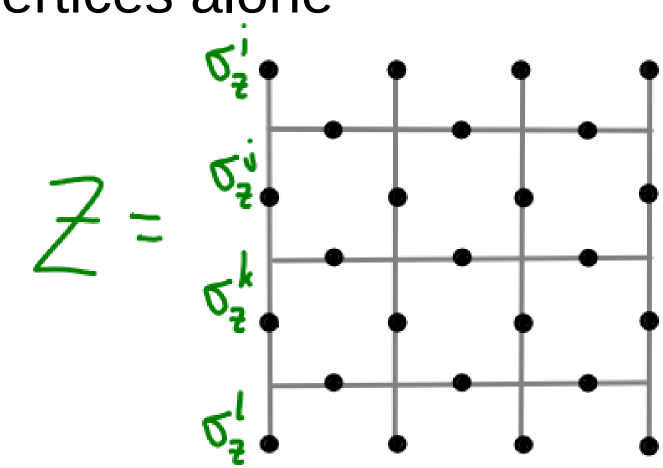
- The vertex and plaquette operators commute
- We can measure these observable simultaneously
- Detect and correct σ_x and σ_z errors simulaneously

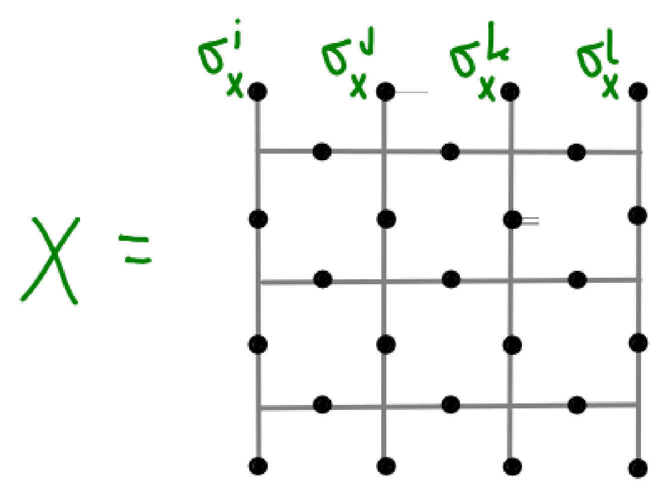
Vertex and plaquette operators together

- Encoded states now unique: superposition of all previous solutions
- Highly entangled states



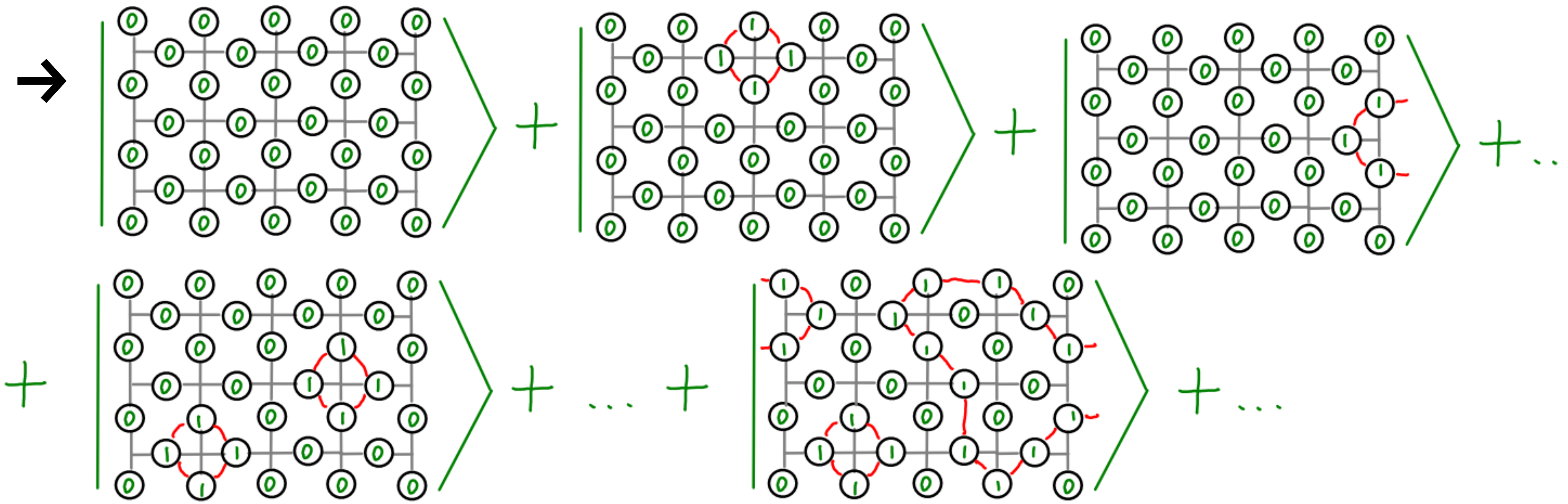
- Pauli X and Z for encoded qubits exactly as they were for plaquettes and vertices alone



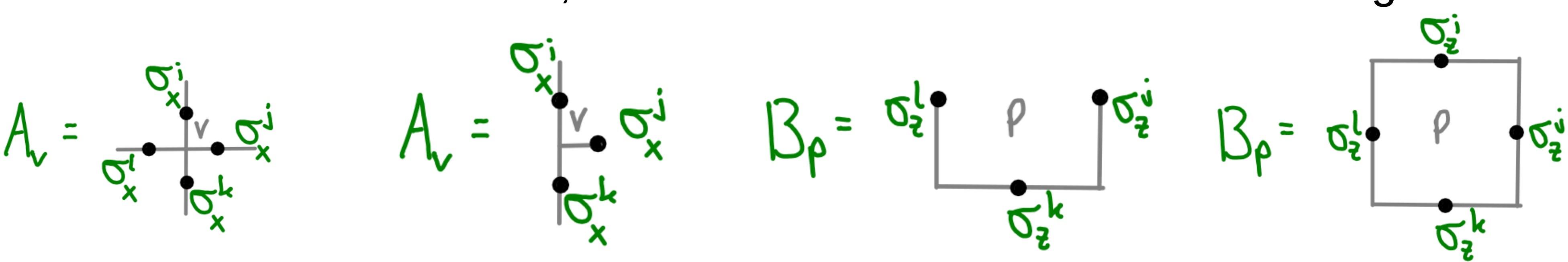


Vertex and plaquette operators together

- But aren't many-body entangled states hard to make?

$|0\rangle \rightarrow$ 

- They are the mutual eigenstates of the observables we measure
- If we can measure them, we can create and maintain the entanglement



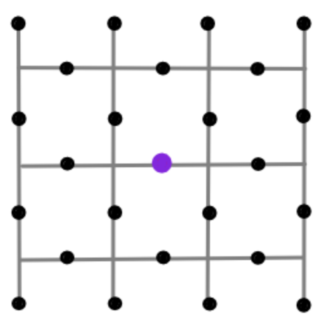
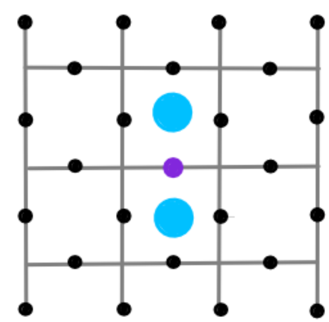
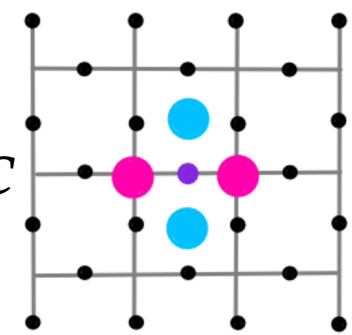
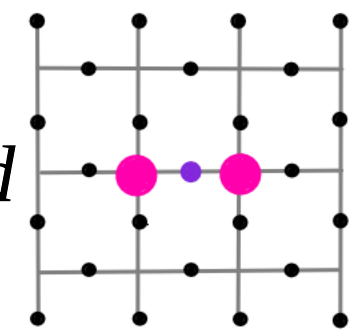
Vertex and plaquette operators together

- We are not just protected against σ_x and σ_z noise, but all local noise

- Any noise operator can be expressed in terms of Paulis

$$M|\psi\rangle = a\sigma_0 + b\sigma_x + c\sigma_y + d\sigma_z$$

- And so creates a superposition of different measurement outcomes for the plaquettes and vertices

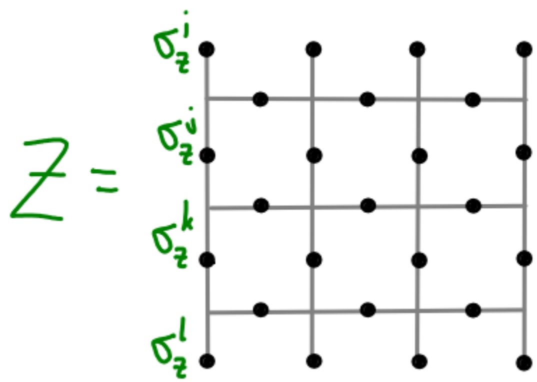
$M|\psi\rangle = a$  $+ b$  $+ c$  $+ d$ 

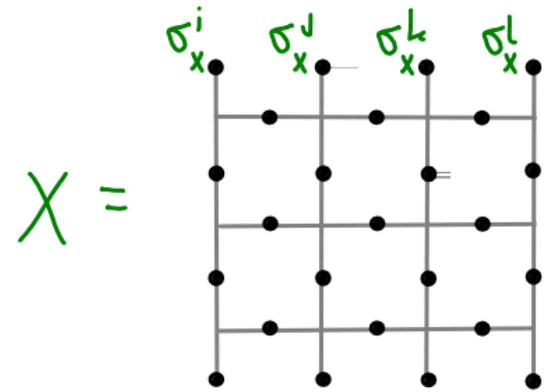
- Measurement collapses this superposition, reducing noise to a simple Pauli

- So any noise can be detected and corrected

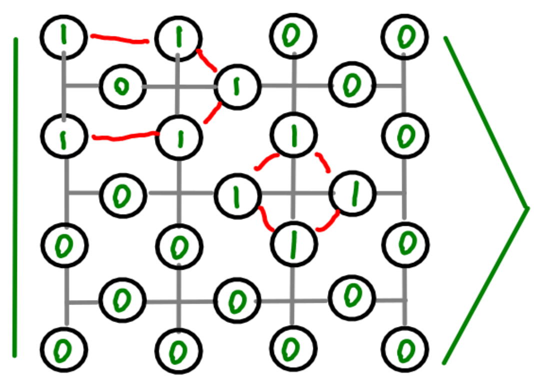
Final Readout

- The logical operators are many body operations





- How do we read out stored information without error?
- When you decide on a basis, you stop caring about one kind of error
- We can just measure in a product basis

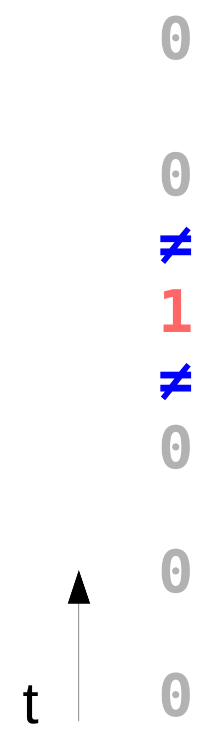


- Logical Z and plaquette info can be constructed from the result

- Imperfect measurement can be corrected like a bit flip

Imperfect measurement

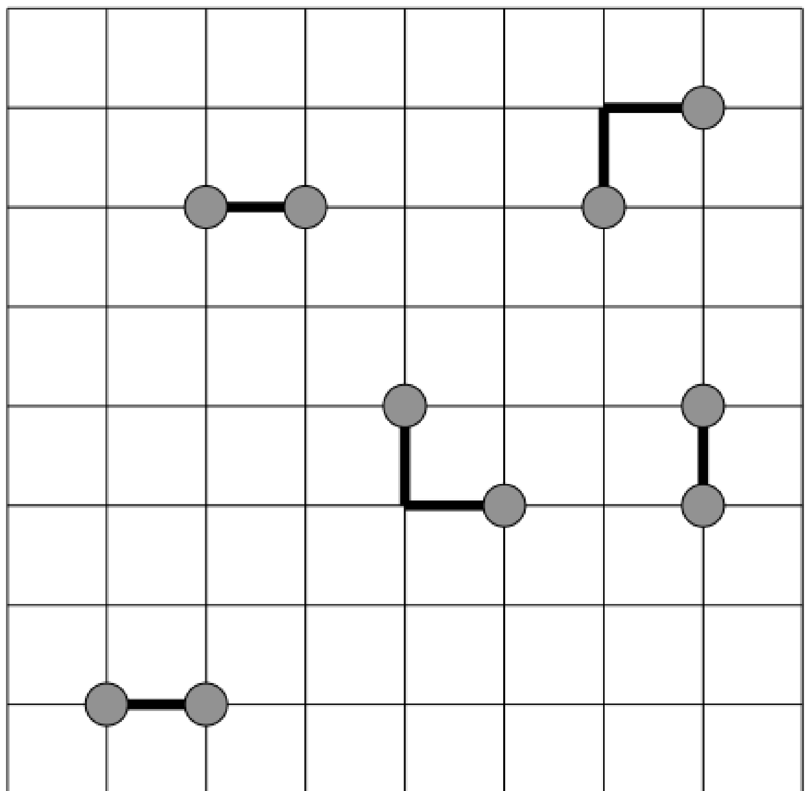
- What about imperfect measurements throughout?
- Consider a measurement of a single qubit that lies with prob. *P*, but doesn't disturb the measured qubit (beyond projection)
- How do we extract information correctly? Repetition!
- Lies create pairs of defects in the time direction



Imperfect measurement

- Combine this with the repetition code or surface code:
  - Defects = changes in ancilla measurement result
  - Bit flips create space-like separated defect pairs
  - Lies create time-like separate defect pairs
  - Combinations create combinations

- Noisy measurements just increase the space of the 'universe' by 1 dimension



arXiv:quant-ph/0110143

The surface code is a good quantum code

- X and Z basis are treated the same
  - σ_x creates particle-like defects that can be detected
  - Large scale effects are needed for a logical bit flip
  - Multiqubit measurement needed to distinguish encoded $|+\rangle$, $|-\rangle$

  - σ_z creates particle-like defects that can be detected
  - Large scale effects are needed for a logical phase flip
  - Multiqubit measurement needed to distinguish encoded $|0\rangle$, $|1\rangle$

- Other good quantum codes also exist
  - Topological codes: Color code, quantum double modes, …
  - Concatenated codes: Shor code, ...