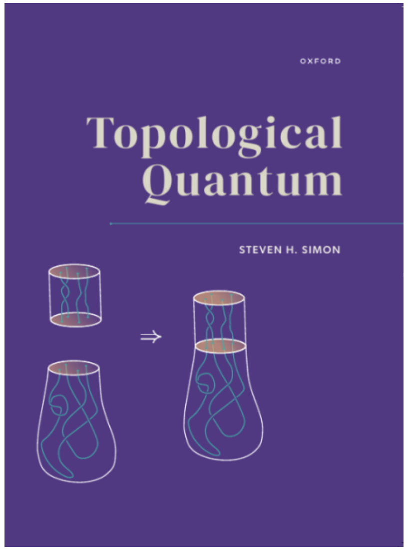
Why are we interested in Anyons?

- Fundamental Interest:
  - What can exist, in principle
  - Is it lurking in plain sight?
  - Surprising connections to: High Energy Physics, Quantum Gravity, Pure Maths/Topology, ...

- Connection to Quantum Memories (A. Kitaev '97)
  and the notion of topological order (Wen)

A. Kitaev



Classical Memory

1 bit $\Rightarrow$ 0 or 1

N bit memory $\Rightarrow$ 2^N possible states

Example 5 bit state = 11010

Quantum Memory

1 qubit $\Rightarrow$ $|\psi\rangle = \alpha|0\rangle + \beta|1\rangle$

N qubit memory $\Rightarrow$ 2^N dimensional Hilbert space

Example 2 qubit State

$$|\psi\rangle = \alpha|00\rangle + \beta|01\rangle + \gamma|10\rangle + \delta|11\rangle$$

Errors

Error = Any process by which the state of your memory is unintentionally changed.

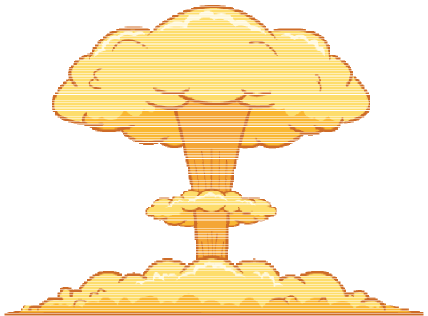
010010001001

Error Correcting Code

One very important bit of information we want to protect

0

1



Error Correcting Code

One very important bit of information we want to protect

logical bit	physical bits
0	000
1	111

Code Space

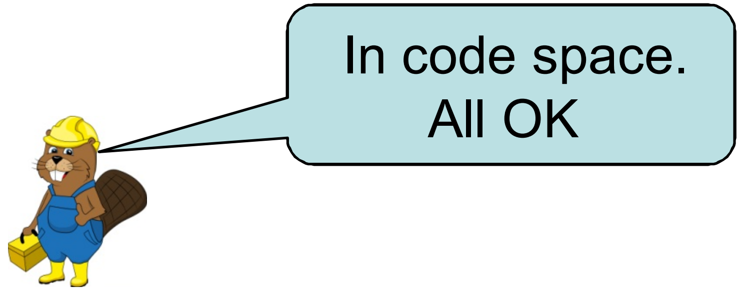
Error Correcting Code

One very important bit of information we want to protect

logical bit	physical bits
0	000
1	111

Code Space

Example:



0 0 0

Error Correcting Code

One very important bit of information we want to protect

logical bit	physical bits
0	000
1	111

Code Space

Example:

0 0 0

Error Correcting Code

One very important bit of information we want to protect

logical bit	physical bits
0	000
1	111

Code Space

Example:

0 0 0

Error Correcting Code

One very important bit of information we want to protect

logical bit	physical bits
0	000
1	111

Code Space

Example:

NOT in code space. ERROR! Must Repair!

0 0 1

Can’t we do the same for qubits?

NOT SO EASY!

1) Quantum No Cloning Theorem!
2) Measuring Disturbs

Can't we do the same for qubits?

Quantum No Cloning Theorem!
(Zurek et al, 1982)

Theorem: Given a qubit in an arbitrary unknown state $|\phi_1\rangle$ and another qubit in an initial state $|\phi_2\rangle$, there does not exist any unitary operator U (i.e., any quantum mecahnical evolution) such that

$$U(|\phi_1\rangle \otimes |\phi_2\rangle) = |\phi_1\rangle \otimes |\phi_1\rangle$$

for all possible input $|\phi_1\rangle$.

Proof of No Cloning Theorem:

Suppose such a copying unitary exists.
Apply unitary to two states |0> and |1>

$$
\begin{aligned}
U(|0\rangle \otimes |\phi_2\rangle) &= e^{i\chi}|0\rangle \otimes |0\rangle \\
U(|1\rangle \otimes |\phi_2\rangle) &= e^{i\chi}|1\rangle \otimes |1\rangle
\end{aligned}
$$

Now apply to a superposition $\alpha|0\rangle + \beta|1\rangle$ and use linearity

$$U(\underline{[\alpha|0\rangle + \beta|1\rangle]} \otimes |\phi_2\rangle) = e^{i\chi}(\alpha|0\rangle \otimes |0\rangle + \beta|1\rangle \otimes |1\rangle)$$

But this is *not* a copy of the superposition which would be

$$e^{i\chi}[\alpha|0\rangle + \beta|1\rangle] \otimes [\alpha|0\rangle + \beta|1\rangle]$$

QED

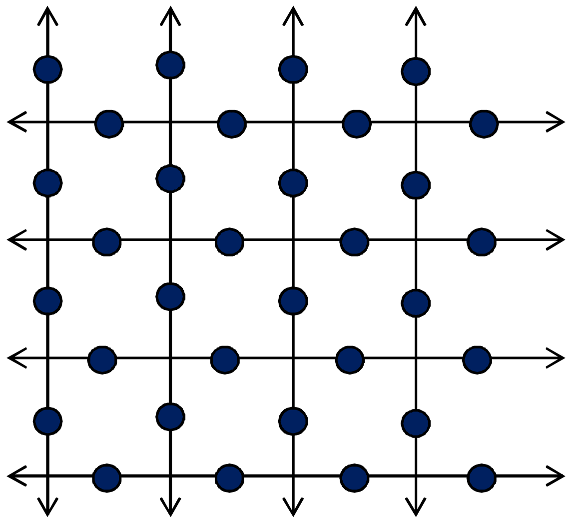
Nonetheless Quantum Error Correction Exists!

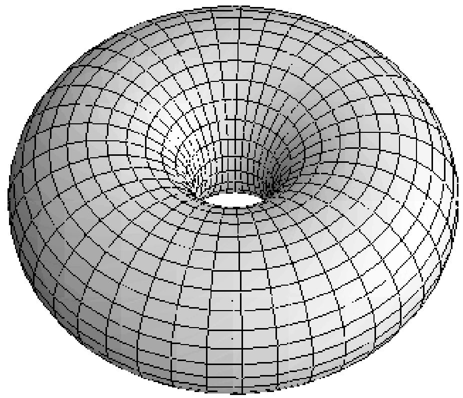
Peter Shor:
Quantum Factoring Algorithm 1994
Quantum Error Correction 1995

Toric Code: Kitaev 1997

Hilbert Space

Dim = $2^{\#spins}$

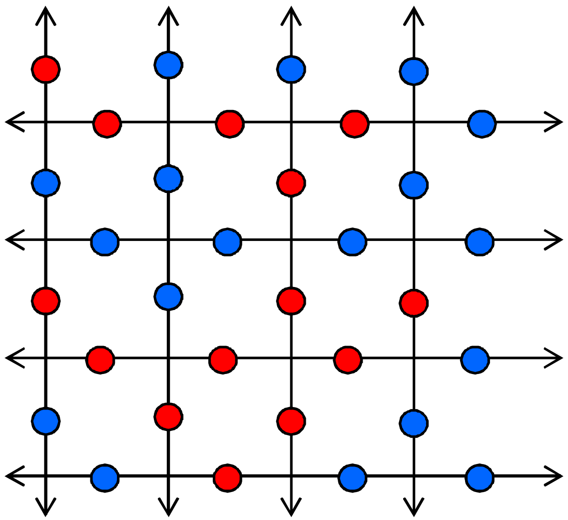


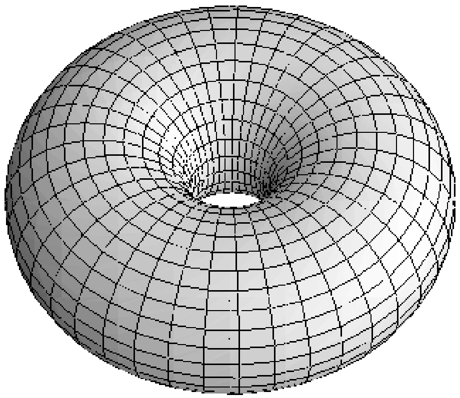


● = spin ½

Hilbert Space

Dim = $2^{\#\text{spins}}$

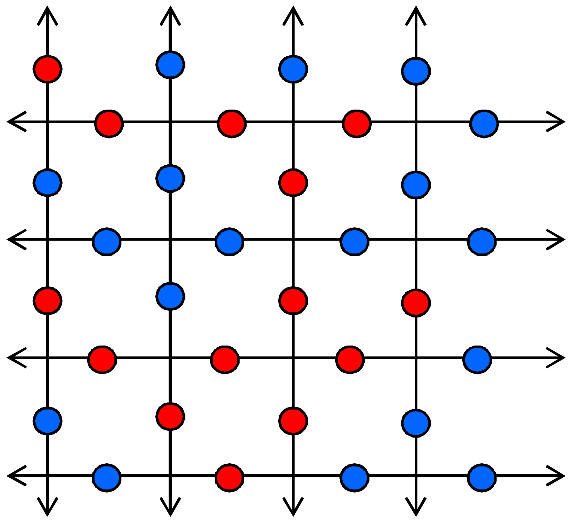




● = spin down = $|1\rangle$

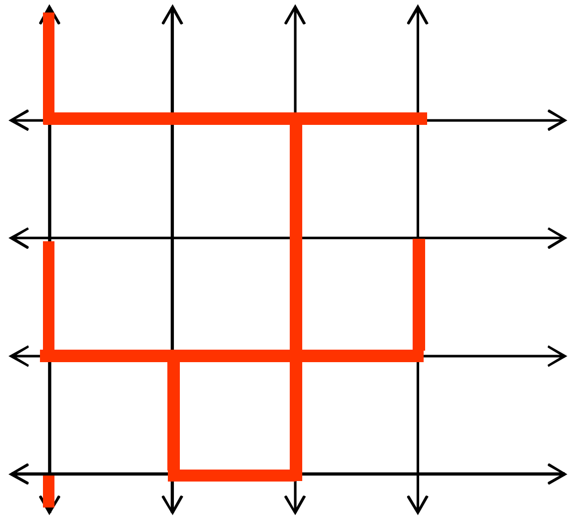
● = spin up = $|0\rangle$

Hilbert Space



● = spin down = $|1\rangle$

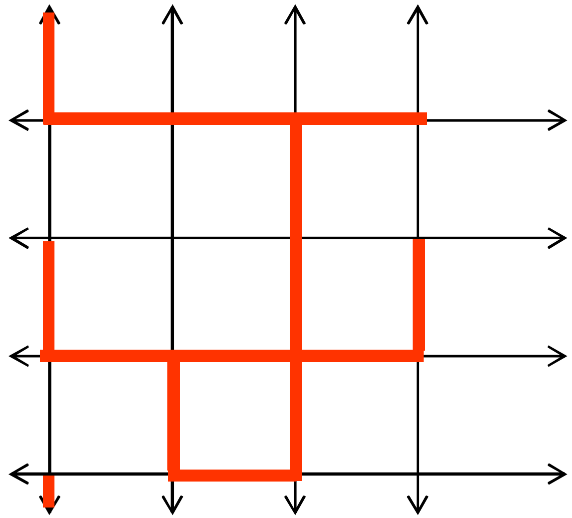
● = spin up = $|0\rangle$



= spin down = $|1\rangle$

= spin up = $|0\rangle$

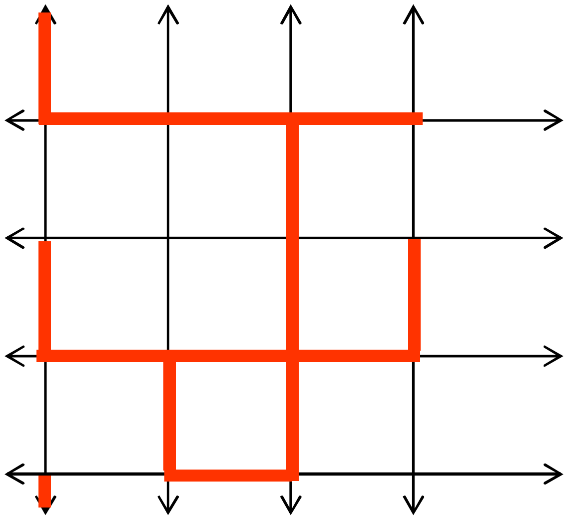
Hilbert Space



= spin down = $|1\rangle$

= spin up = $|0\rangle$

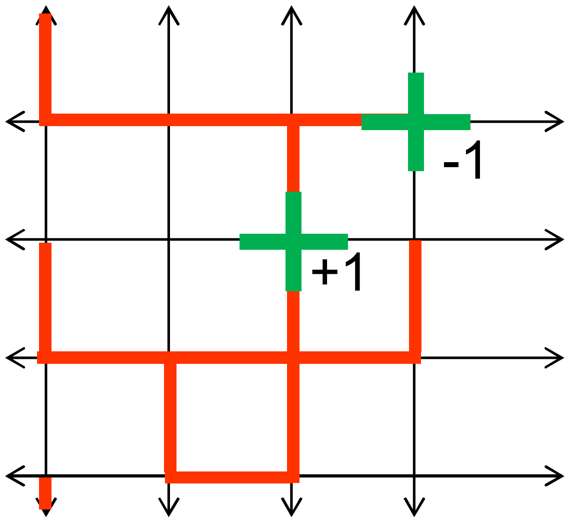
Hilbert Space

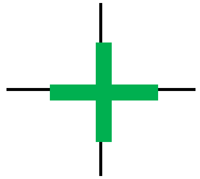


= spin down = $|1\rangle$

= spin up = $|0\rangle$

Some Operators





= Vertex operator

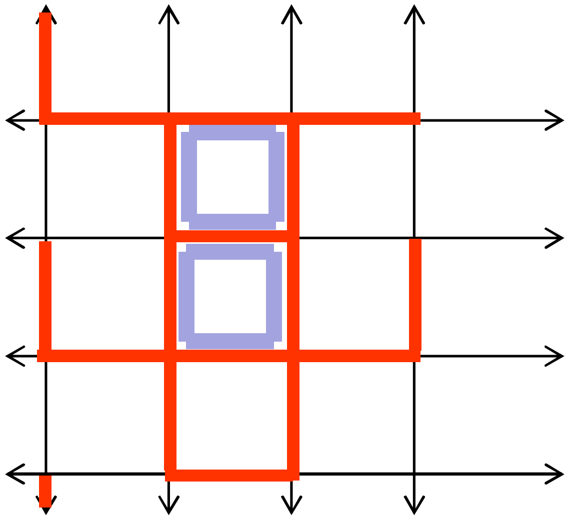
$$V_\alpha = \prod_{i \in vertex\ \alpha} \sigma_i^z = \prod_{i \in vertex\ \alpha} Z_i$$

= +1 if an even # of down spins
= -1 if an odd # of down spins

= spin down = $|1\rangle$

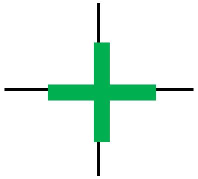
= spin up = $|0\rangle$

Some Operators



= spin down

= spin up

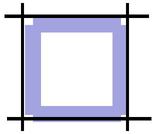


= Vertex operator

$$V_\alpha = \prod_{i \in vertex\ \alpha} \sigma_i^z = \prod_{i \in vertex\ \alpha} Z_i$$

= +1 if an even # of down spins

= -1 if an odd # of down spins

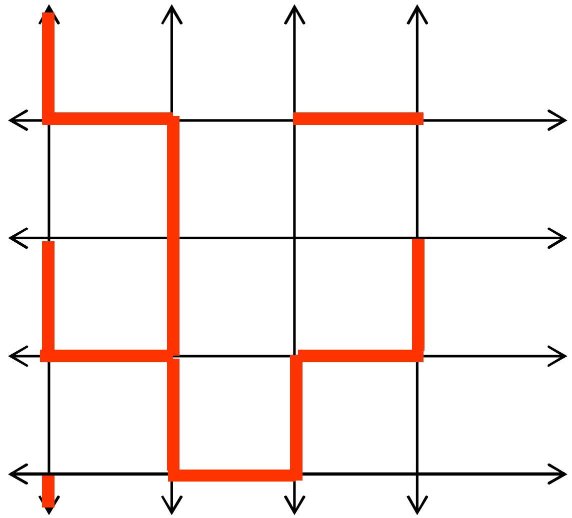


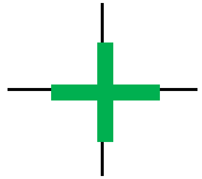
= Plaquette operator

$$P_\beta = \prod_{i \in plaquette\ \beta} \sigma_i^x = \prod_{i \in plauqette\ \beta} X_i$$

= flip spins around a plaquette

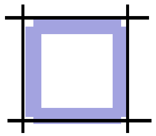
Some Operators



 = Vertex operator

$$V_\alpha = \prod_{i \in vertex\, \alpha} \sigma_i^z = \prod_{i \in vertex\, \alpha} Z_i$$

= +1 if an even # of down spins
= -1 if an odd # of down spins

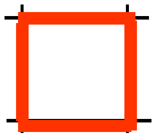
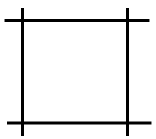
 = Plaquette operator

$$P_\beta = \prod_{i \in plaquette\, \beta} \sigma_i^x = \prod_{i \in plauqette\, \beta} X_i$$

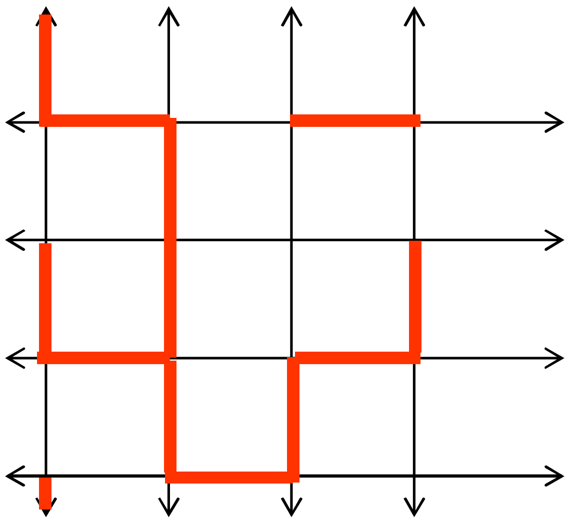
= flip spins around a plaquette

= spin down

= spin up

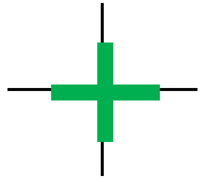
ex $\{$  +  $\}$ is a +1 eigenstate of P_β

Some Operators



= spin down

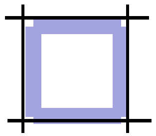
= spin up



= Vertex operator

$$V_\alpha = \prod_{i \in vertex\ \alpha} \sigma_i^z = \prod_{i \in vertex\ \alpha} Z_i$$

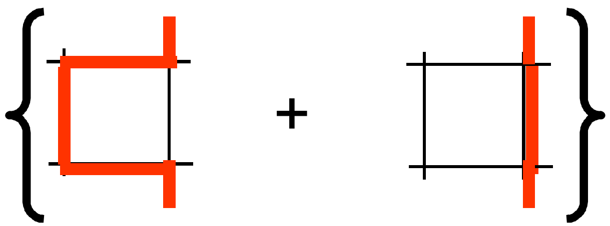
= +1 if an even # of down spins
= -1 if an odd # of down spins



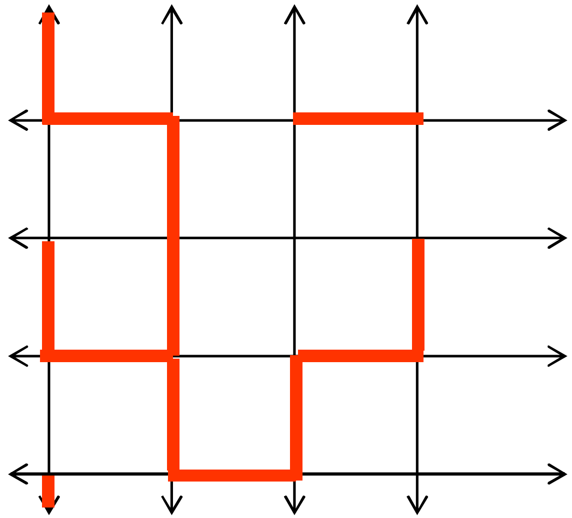
= Plaquette operator

$$P_\beta = \prod_{i \in plaquette\ \beta} \sigma_i^x = \prod_{i \in plauqette\ \beta} X_i$$

= flip spins around a plaquette

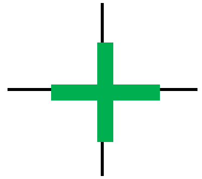
ex  is a +1 eigenstate of P_β

Some Operators



 = spin down

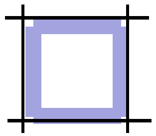
 = spin up

 = Vertex operator

$$V_\alpha = \prod_{i \in vertex\, \alpha} \sigma_i^z = \prod_{i \in vertex\, \alpha} Z_i$$

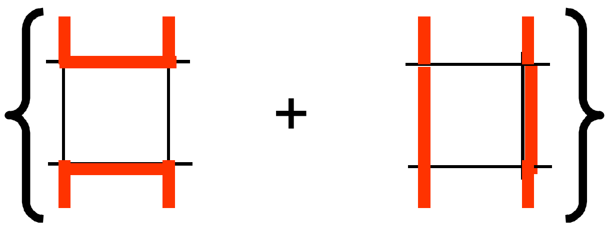
= +1 if an even # of down spins

= -1 if an odd # of down spins

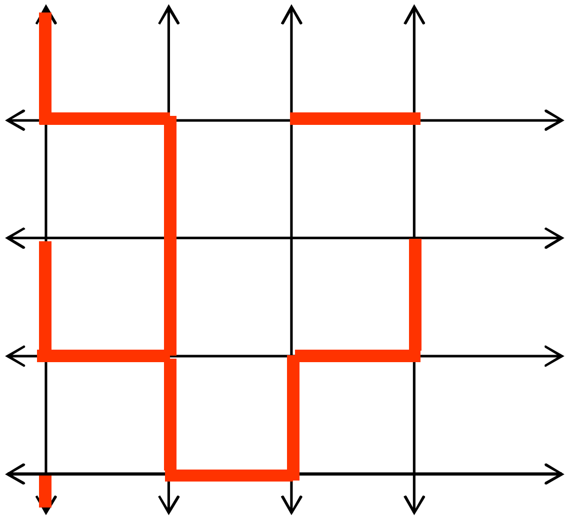
 = Plaquette operator

$$P_\beta = \prod_{i \in plaquette\, \beta} \sigma_i^x = \prod_{i \in plauqette\, \beta} X_i$$

= flip spins around a plaquette

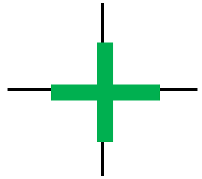
ex  is a +1 eigenstate of P_β

Some Operators



= spin down = $|1\rangle$

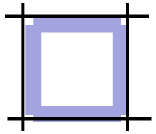
= spin up = $|0\rangle$



= Vertex operator

$$V_\alpha = \prod_{i \in vertex\ \alpha} \sigma_i^z = \prod_{i \in vertex\ \alpha} Z_i$$

= +1 if an even # of down spins
= -1 if an odd # of down spins



= Plaquette operator

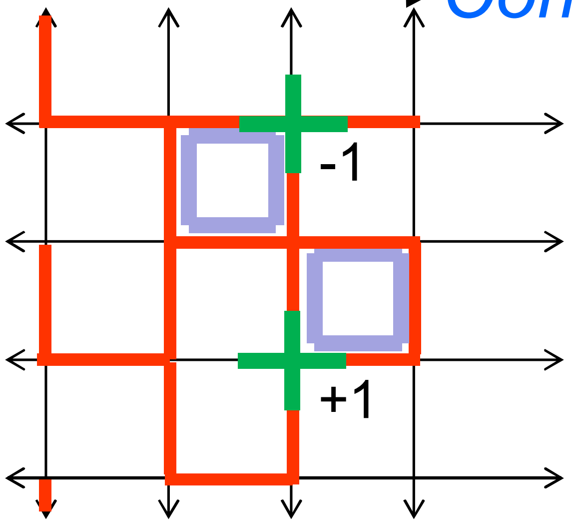
$$P_\beta = \prod_{i \in plaquette\ \beta} \sigma_i^x = \prod_{i \in plauqette\ \beta} X_i$$

= flip spins around a plaquette

Caution: Some refs exchange σ^x and σ^z

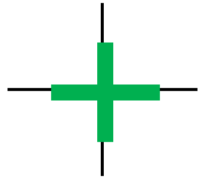
Some Operators

Commuting



= spin down = $|1\rangle$

= spin up = $|0\rangle$

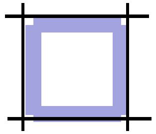


= Vertex operator

$$V_\alpha = \prod_{i \in vertex\ \alpha} \sigma_i^z = \prod_{i \in vertex\ \alpha} Z_i$$

= +1 if an even # of down spins

= -1 if an odd # of down spins



= Plaquette operator

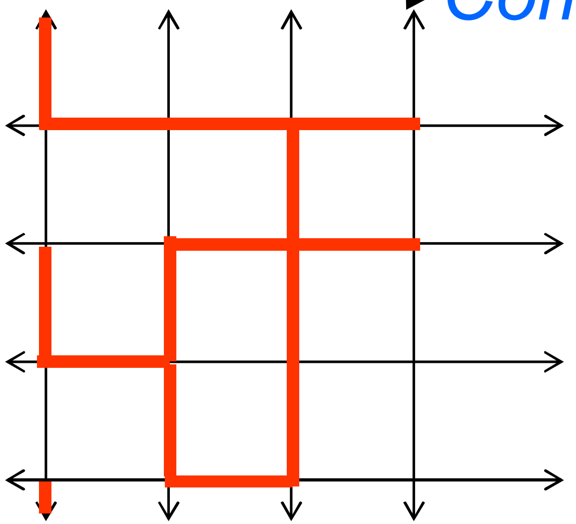
$$P_\beta = \prod_{i \in plaquette\ \beta} \sigma_i^x = \prod_{i \in plauqette\ \beta} X_i$$

= flip spins around a plaquette

$$[V_\alpha, V_{\alpha'}] = 0 \qquad [P_\beta, P_{\beta'}] = 0$$

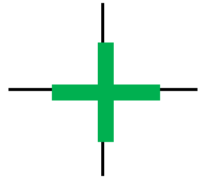
$$[V_\alpha, P_\beta] = 0$$

Some Operators
Commuting



= spin down = |1⟩

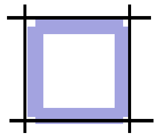
= spin up = |0⟩



= Vertex operator

$$V_\alpha = \prod_{i \in vertex\ \alpha} \sigma_i^z = \prod_{i \in vertex\ \alpha} Z_i$$

= +1 if an even # of down spins
= -1 if an odd # of down spins



= Plaquette operator

$$P_\beta = \prod_{i \in plaquette\ \beta} \sigma_i^x = \prod_{i \in plauqette\ \beta} X_i$$

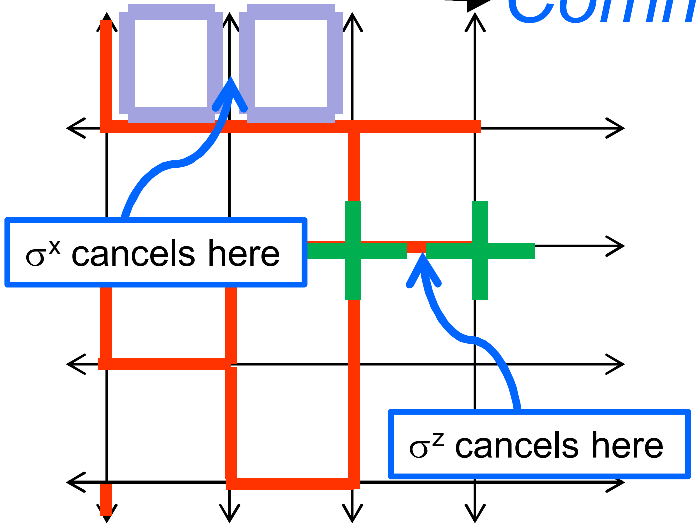
= flip spins around a plaquette

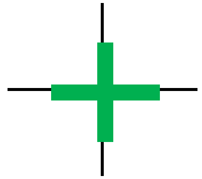
$$[V_\alpha, V_{\alpha'}] = 0 \qquad [P_\beta, P_{\beta'}] = 0$$

$$[V_\alpha, P_\beta] = 0$$

Is this a complete set?

Some Operators
Commuting

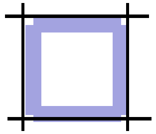




= Vertex operator

$$V_\alpha = \prod_{i \in vertex\ \alpha} \sigma_i^z = \prod_{i \in vertex\ \alpha} Z_i$$

= +1 if an even # of down spins
= -1 if an odd # of down spins



= Plaquette operator

$$P_\beta = \prod_{i \in plaquette\ \beta} \sigma_i^x = \prod_{i \in plauqette\ \beta} X_i$$

= flip spins around a plaquette

# of spins
= # of vertices + # of plaquettes

BUT: Not V's and P's not all independent!

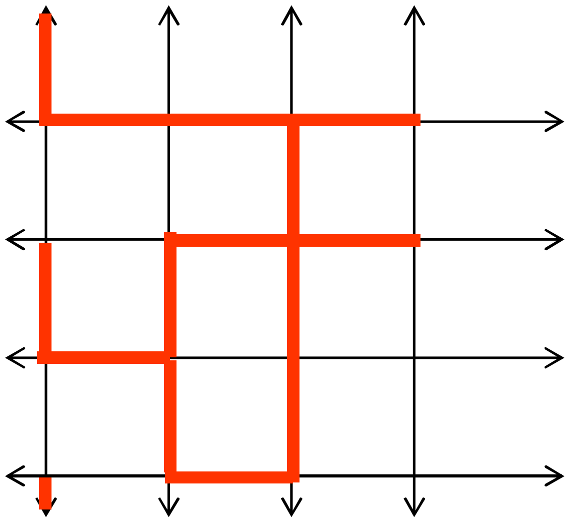
$$\prod_{all\ \beta} P_\beta = 1 \qquad \prod_{all\ \alpha} V_\alpha = 1$$

$$[V_\alpha, V_{\alpha'}] = 0 \qquad [P_\beta, P_{\beta'}] = 0$$

$$[V_\alpha, P_\beta] = 0$$

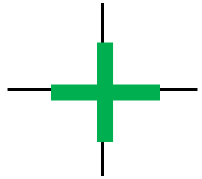
Fixing P's and V's leaves 2 degrees of freedom unspecified

A Code



(code space)

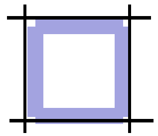
Rule 1) Specify all $V_\alpha=1$



= Vertex operator

$$V_\alpha = \prod_{i \in vertex\ \alpha} \sigma_i^z = \prod_{i \in vertex\ \alpha} Z_i$$

= +1 if an even # of down spins
= -1 if an odd # of down spins

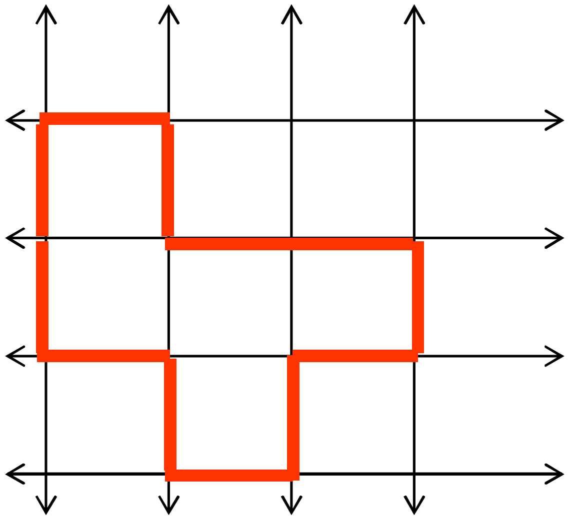


= Plaquette operator

$$P_\beta = \prod_{i \in plaquette\ \beta} \sigma_i^x = \prod_{i \in plauqette\ \beta} X_i$$

= flip spins around a plaquette

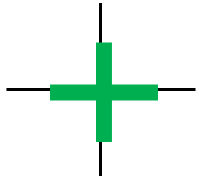
A Code



(code space)

Rule 1) Specify all V_α=1

Loops only

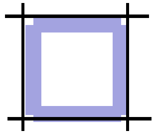


= Vertex operator

$$V_\alpha = \prod_{i \in vertex\ \alpha} \sigma_i^z = \prod_{i \in vertex\ \alpha} Z_i$$

= +1 if an even # of down spins

= -1 if an odd # of down spins

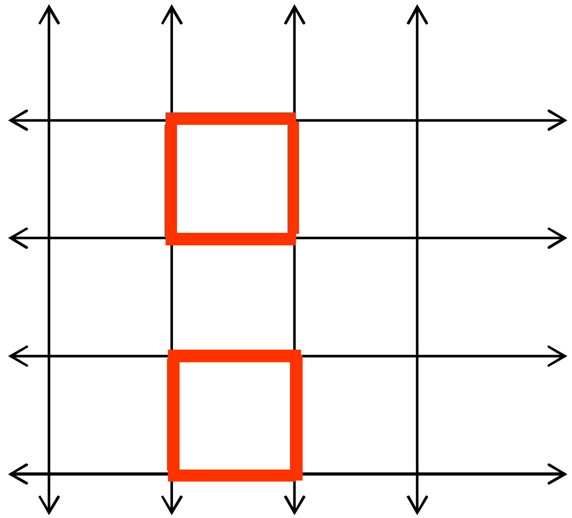


= Plaquette operator

$$P_\beta = \prod_{i \in plaquette\ \beta} \sigma_i^x = \prod_{i \in plauqette\ \beta} X_i$$

= flip spins around a plaquette

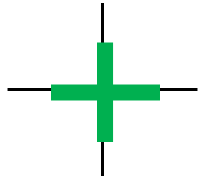
A Code



(code space)

Rule 1) Specify all V_α=1

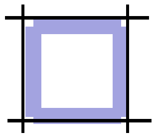
Loops only



= Vertex operator

$$V_\alpha = \prod_{i \in vertex\ \alpha} \sigma_i^z = \prod_{i \in vertex\ \alpha} Z_i$$

= +1 if an even # of down spins
= -1 if an odd # of down spins

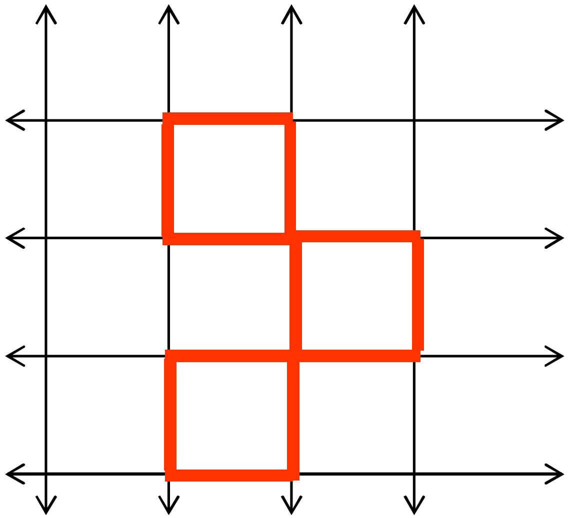


= Plaquette operator

$$P_\beta = \prod_{i \in plaquette\ \beta} \sigma_i^x = \prod_{i \in plauqette\ \beta} X_i$$

= flip spins around a plaquette

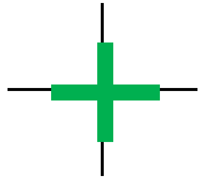
A Code



(code space)

Rule 1) Specify all V_α=1

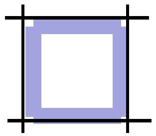
Loops only



= Vertex operator

$$V_\alpha = \prod_{i \in vertex\ \alpha} \sigma_i^z = \prod_{i \in vertex\ \alpha} Z_i$$

= +1 if an even # of down spins
= -1 if an odd # of down spins

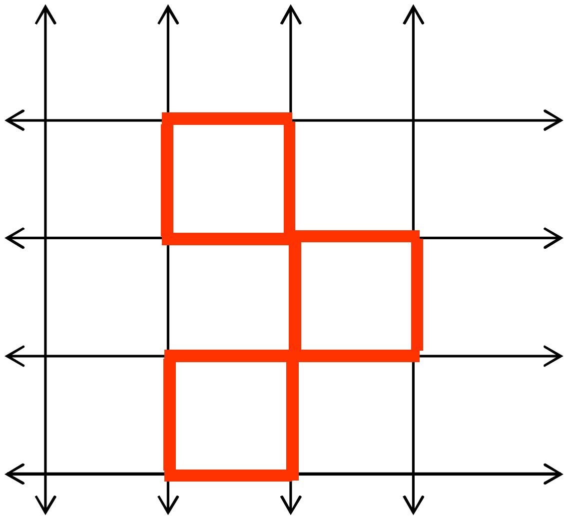


= Plaquette operator

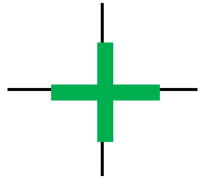
$$P_\beta = \prod_{i \in plaquette\ \beta} \sigma_i^x = \prod_{i \in plauqette\ \beta} X_i$$

= flip spins around a plaquette

A Code



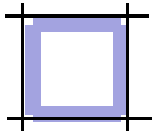
(code space)



= Vertex operator

$$V_\alpha = \prod_{i \in vertex\ \alpha} \sigma_i^z = \prod_{i \in vertex\ \alpha} Z_i$$

= +1 if an even # of down spins
= -1 if an odd # of down spins



= Plaquette operator

$$P_\beta = \prod_{i \in plaquette\ \beta} \sigma_i^x = \prod_{i \in plauqette\ \beta} X_i$$

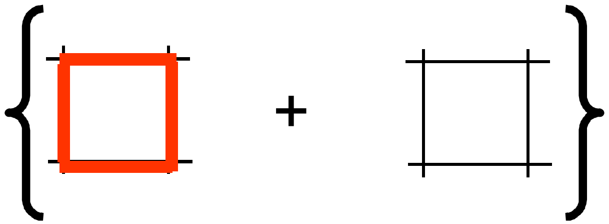
= flip spins around a plaquette

Rule 1) Specify all V_α=1

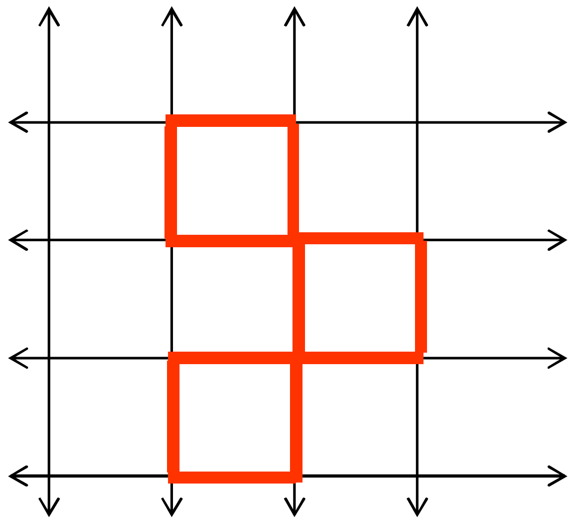
→ Loops only

Rule 2) Specify all P_β=1

→ Equal superposition of flipped plaquettes

ex  is a +1 eigenstate of P_β

A Code



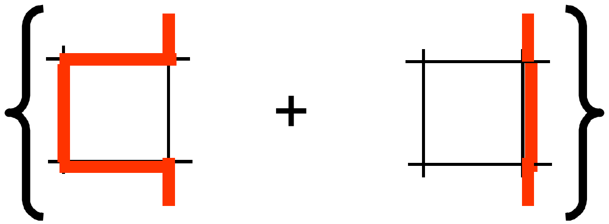
(code space)

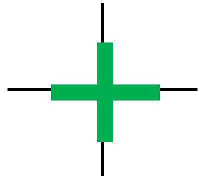
Rule 1) Specify all $V_\alpha=1$

➡ Loops only

Rule 2) Specify all $P_\beta=1$

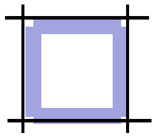
➡ Equal superposition of flipped plaquettes

ex  is a +1 eigenstate of P_β

 = Vertex operator

$$V_\alpha = \prod_{i \in vertex\ \alpha} \sigma_i^z = \prod_{i \in vertex\ \alpha} Z_i$$

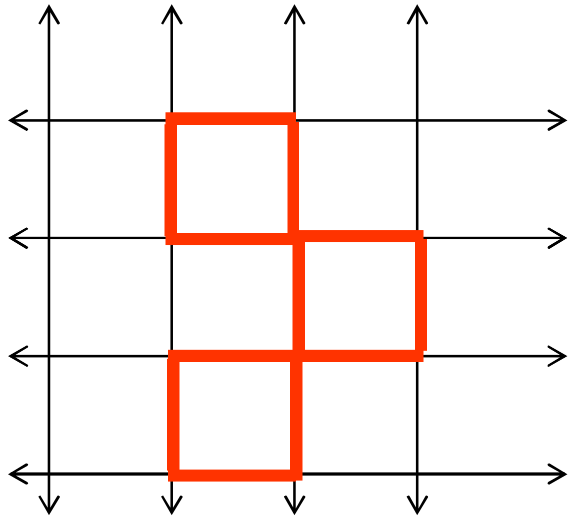
= +1 if an even # of down spins
= -1 if an odd # of down spins

 = Plaquette operator

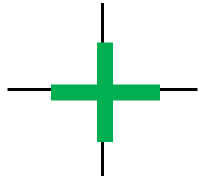
$$P_\beta = \prod_{i \in plaquette\ \beta} \sigma_i^x = \prod_{i \in plauqette\ \beta} X_i$$

= flip spins around a plaquette

A Code



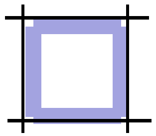
(code space)



= Vertex operator

$$V_\alpha = \prod_{i \in vertex\ \alpha} \sigma_i^z = \prod_{i \in vertex\ \alpha} Z_i$$

= +1 if an even # of down spins
= -1 if an odd # of down spins



= Plaquette operator

$$P_\beta = \prod_{i \in plaquette\ \beta} \sigma_i^x = \prod_{i \in plauqette\ \beta} X_i$$

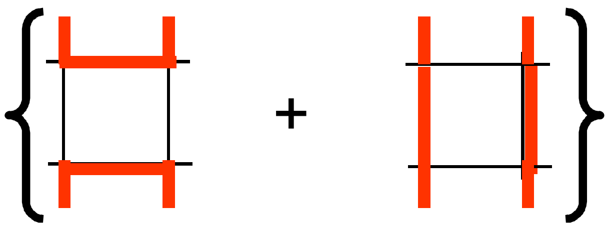
= flip spins around a plaquette

Rule 1) Specify all $V_\alpha=1$

→ Loops only

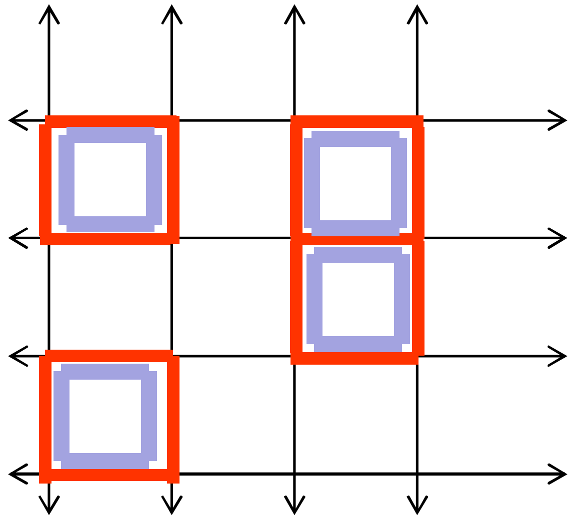
Rule 2) Specify all $P_\beta=1$

→ Equal superposition of flipped plaquettes

ex  is a +1 eigenstate of P_β

A Code

States satisfying rules
(i.e., with correct value of stabilizers)



(code space)

$$|\psi\rangle = \sum |\text{loop config}\rangle$$

Sum is over all loop configs that can be obtained by flipping plaquettes

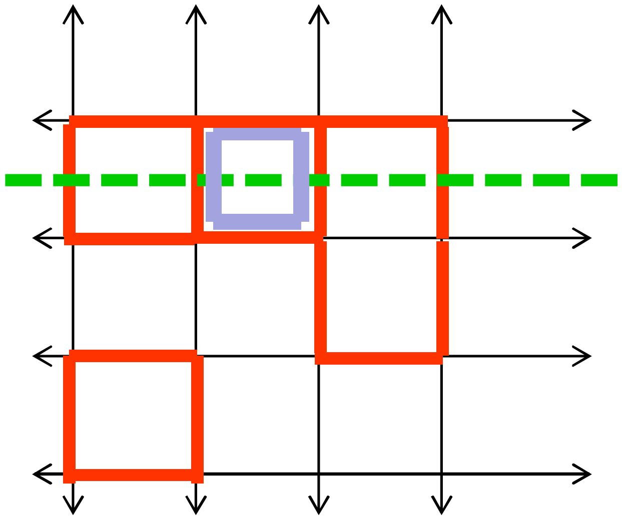
Rule 1) Specify all $V_\alpha=1$

→ Loops only

Rule 2) Specify all $P_\beta=1$

→ Equal superposition of flipped plaquettes

A Code



(code space)

States satisfying rules
(i.e., with correct value of stabilizers)

$$|\psi\rangle = \sum |\text{loop config}\rangle$$

Sum is over all loop configs that can be obtained by flipping plaquettes

Expect 2 “logical” qubits remain

Rule 1) Specify all $V_\alpha=1$

➡ Loops only

Fixing P’s and V’s leaves 2 degrees of freedom unspecified

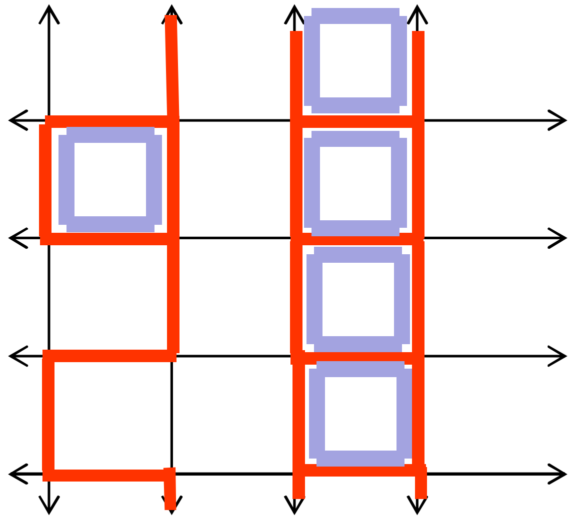
Rule 2) Specify all $P_\beta=1$

➡ Equal superposition of flipped plaquettes

Does flipping obtain all allowed loop configs?

No, Flipping does not change the parity of loops around handle

A Code



(code space)

States satisfying rules
(i.e., with correct value of stabilizers)

$$|\psi\rangle = \sum |\text{loop config}\rangle$$

Sum is over all loop configs that can be obtained by flipping plaquettes

Expect 2 "logical" qubits remain

Rule 1) Specify all V_α=*1*

➡ Loops only

Fixing P's and V's leaves 2 degrees of freedom unspecified

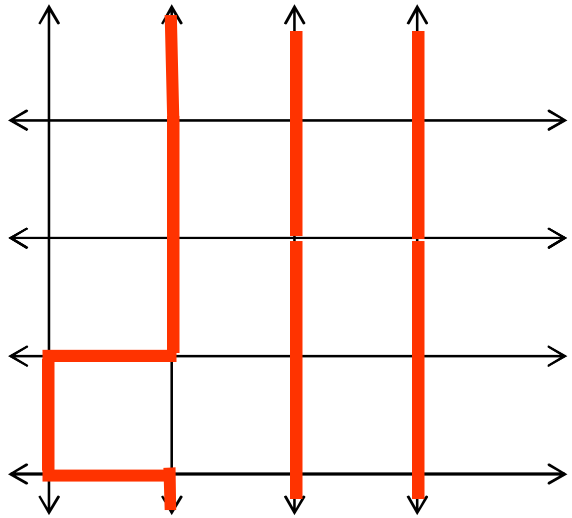
Rule 2) Specify all P_β=*1*

➡ Equal superposition of flipped plaquettes

Does flipping obtain all allowed loop configs?

No, Flipping does not change the parity of loops around handle

A Code



(code space)

States satisfying rules
(i.e., with correct value of stabilizers)

$$|\psi\rangle = \sum |\text{loop config}\rangle$$

Sum is over all loop configs that can be obtained by flipping plaquettes

Expect 2 "logical" qubits remain

Rule 1) Specify all V_α=1

➡ Loops only

Fixing P's and V's leaves 2 degrees of freedom unspecified

Rule 2) Specify all P_β=1

➡ Equal superposition of flipped plaquettes

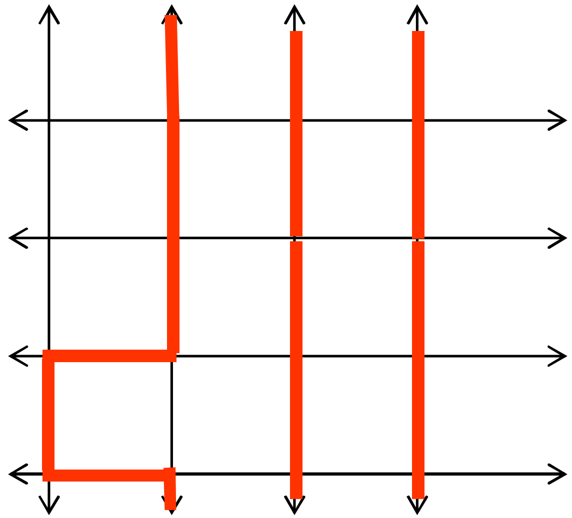
Does flipping obtain all allowed loop configs?

No, Flipping does not change the parity of loops around handle

A Code

$$|\psi\rangle = A_{ee}|\psi_{ee}\rangle + A_{eo}|\psi_{eo}\rangle + A_{oe}|\psi_{oe}\rangle + A_{oo}|\psi_{oo}\rangle$$

Sum of all loop configs with even parity around both handles



(code space)

Expect 2 “logical” qubits remain

Rule 1) Specify all $V_\alpha=1$

→ Loops only

Fixing P’s and V’s leaves 2 degrees of freedom unspecified

Rule 2) Specify all $P_\beta=1$

→ Equal superposition of flipped plaquettes

Does flipping obtain all allowed loop configs?

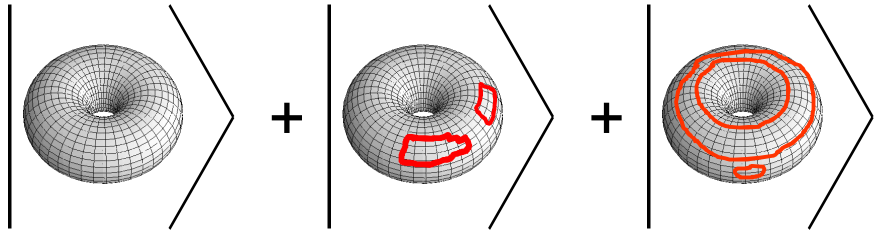
No, Flipping does not change the parity of loops around handle

A Code

$$|\psi\rangle = A_{ee}|\psi_{ee}\rangle + A_{eo}|\psi_{eo}\rangle + A_{oe}|\psi_{oe}\rangle + A_{oo}|\psi_{oo}\rangle$$

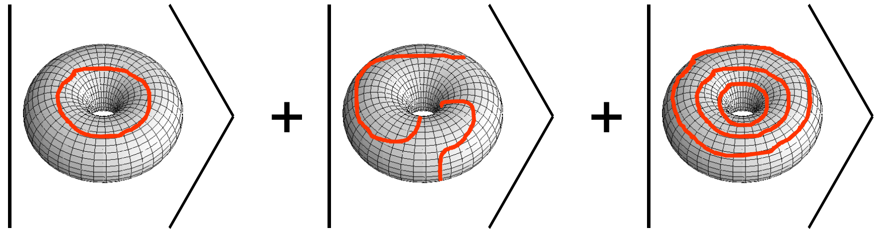
Sum of all loop configs with even parity around both handles

$$|\psi_{ee}\rangle =$$



$+\ldots$

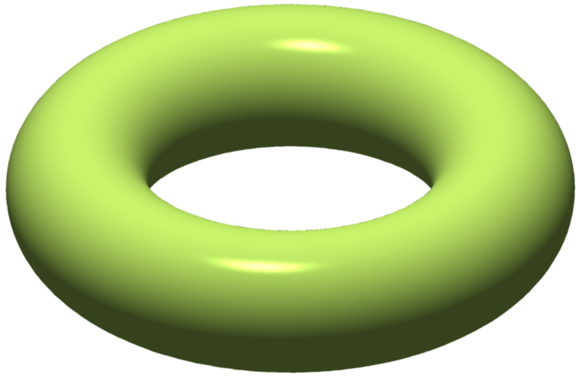
$$|\psi_{eo}\rangle =$$



$+\ldots$

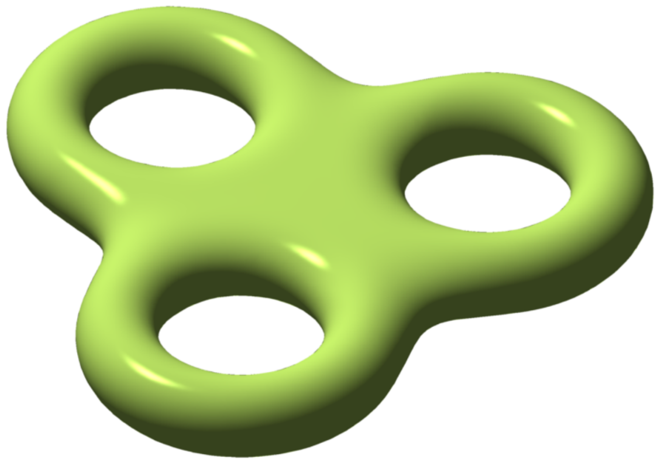
Highly protected code space!

Handle of torus is essential



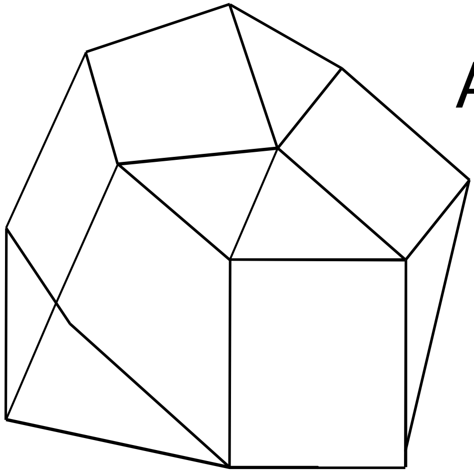
What about g-handle torus? (“genus” = g)

g=2



g=3

How many wavefunctions for toric code on g-handle torus? (g=genus)



Any graph spanning the surface: spins on edges

Euler characteristic:

$$\chi = 2 - 2g = \#\text{vertices} - \#\text{edges} + \#\text{faces}$$

$$2g = \#\text{ edges} - (\#\text{vertices}-1) - (\#\text{faces}-1)$$

degrees of freedom

independent vertex operators

independent plaquette operators

2^{2g} wavefunctions

Consider such a system in code space

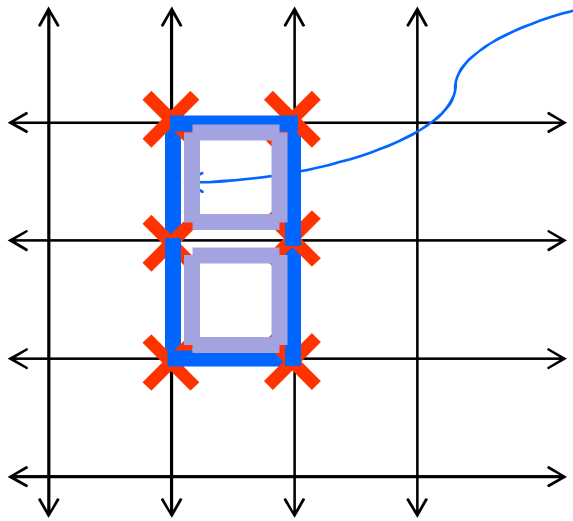
(Here I do not draw spin up and spin down!)

Creating an "error"

$$V_\alpha|\psi\rangle = |\psi\rangle$$

$$\sigma_i^x|\psi\rangle = \sigma_x V_\alpha|\psi\rangle = -V_\alpha\sigma_i^x|\psi\rangle$$

$$V_\alpha[\sigma_i^x|\psi\rangle] = -[\sigma_i^x|\psi\rangle]$$



apply σ^x "error" to this bond

anticommutes with neighboring vertex operators

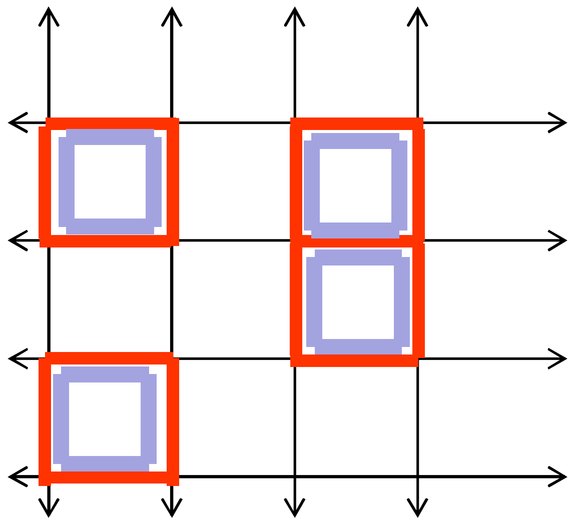
creates two vertex "defects" V_α = -1 instead of +1 (loop endpoints)

To restore to code space bring defects back together to re-annihilate

Closed blue loop = $\prod_{loop} \sigma_x = \prod_{\beta\ enclosed} P_\beta$ = +1 in code space

Another Way to Think About Errors

Start with reference state (all up)



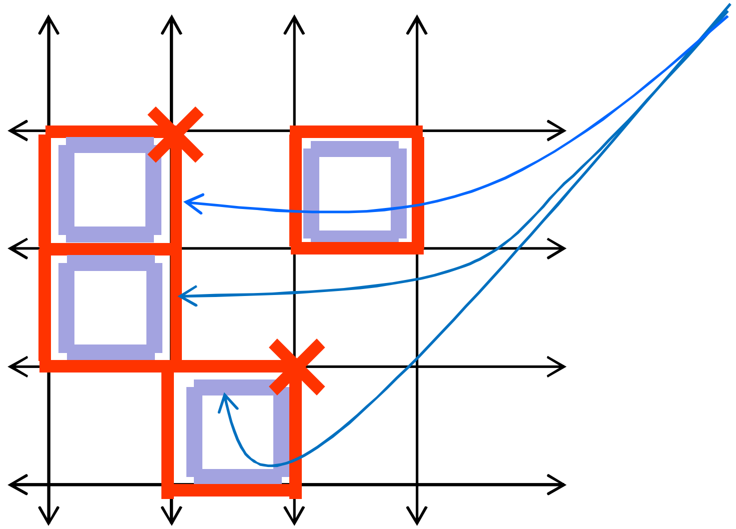
$$|\psi\rangle = \sum |\text{configs}\rangle$$

Sum is over all configs that can be obtained by flipping plaquettes

Another Way to Think About Errors

Start with reference state (all up)

actual "string" position is uncertain



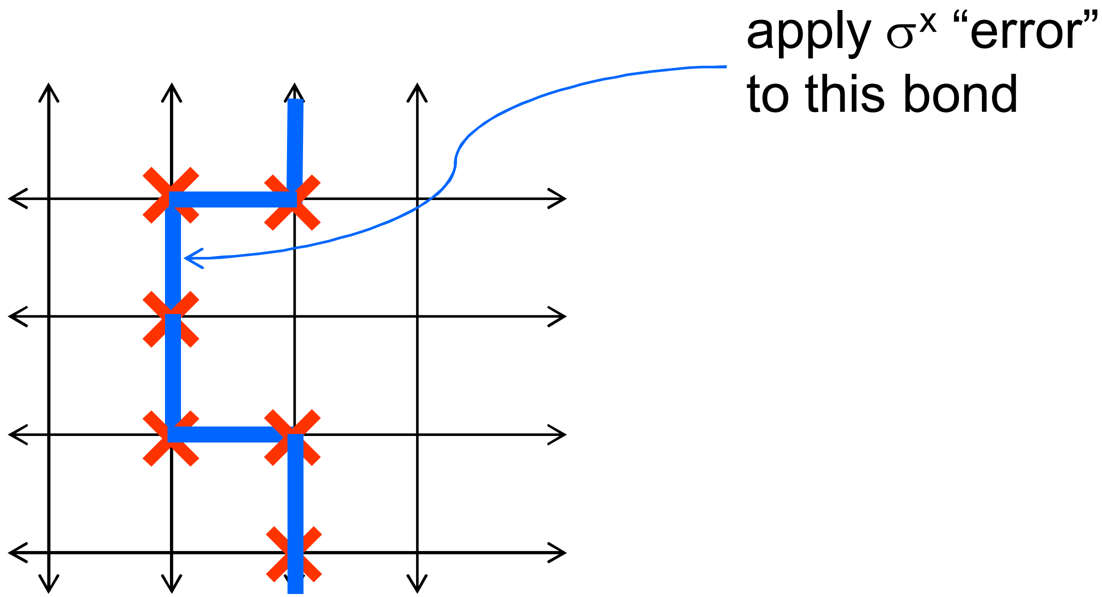
apply σ^x "error" to these bond

$$|\psi\rangle = \sum |\mathrm{configs}\rangle$$

Sum is over all configs that can be obtained by flipping plaquettes

How well is the code space protected?

Creating an “error”

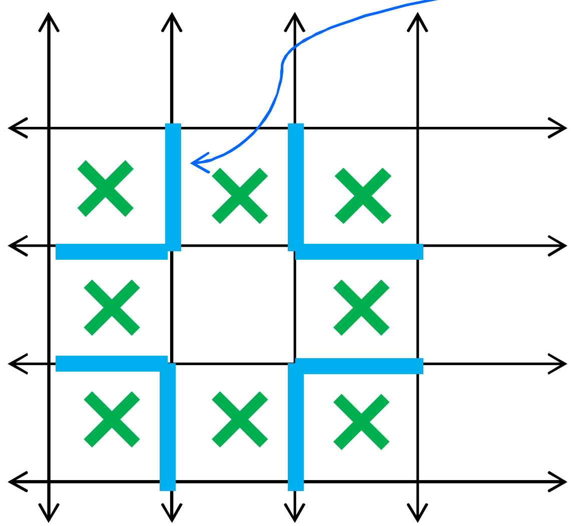


Blue loop around handle = $\prod_{\substack{loop\\around\\handle}} \sigma_x$ = flips code sector!

What about σ^z errors?

Error Correction

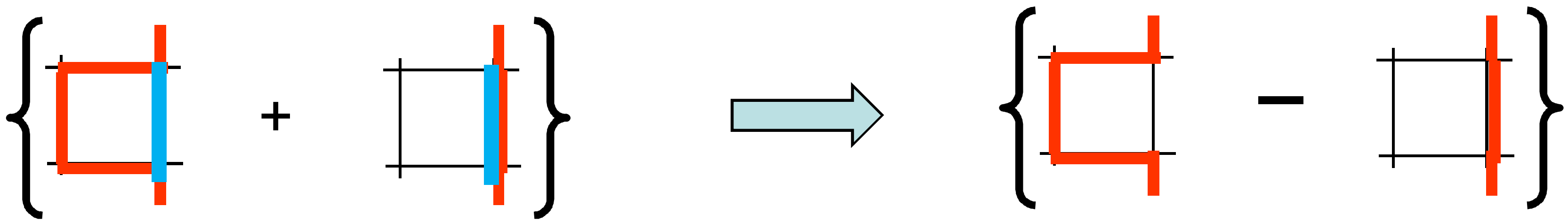
Creating an "error"



apply σ^z "error" to this bond

anticommutes with neighboring plaquette operators

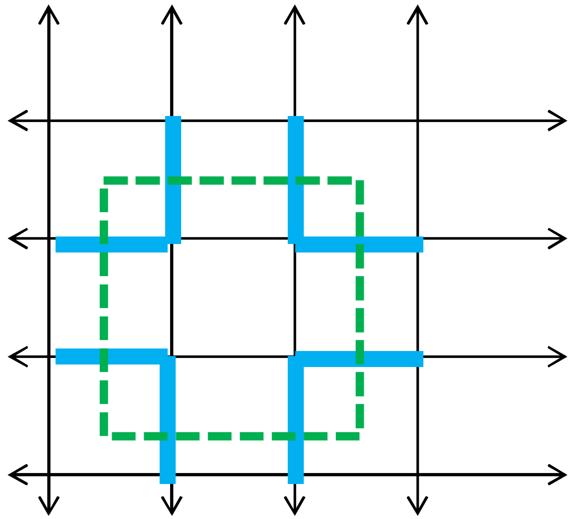
creates two plaquette "defects" P_α = -1 instead of +1



What about σ^z errors?

Error Correction

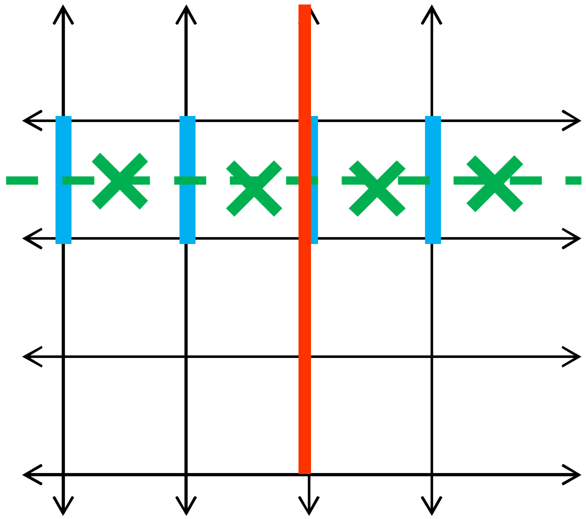
Creating an "error"



Closed dual loop = $\prod_{dual\ loop} \sigma_z = \prod_{\alpha\ enclosed} V_\alpha$ = +1 in code space

Plaquette Defects Around a Handle

Creating an "error"



dual loop around handle $= \prod_{dual\,loop} \sigma_z$ measures parity of spins = cutting dual loop (differs in different sectors).

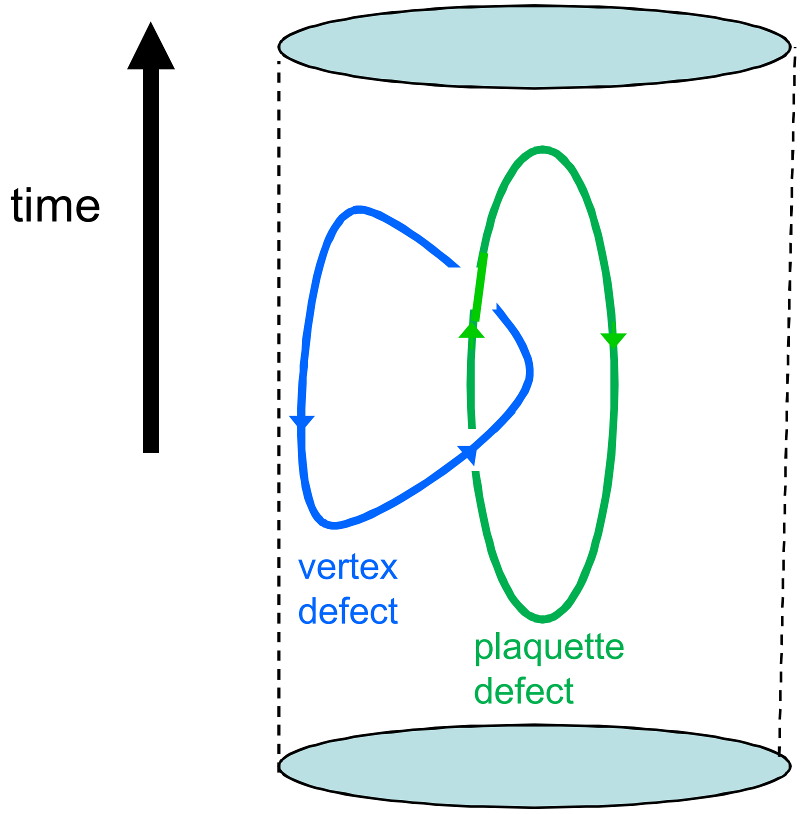
All error correctable (as long as demon isn't too fast)

Considered σ^x
Considered σ^z

$$\sigma^y = i\,\sigma^z\,\sigma^x$$

Any operation on one bit is a combination of these

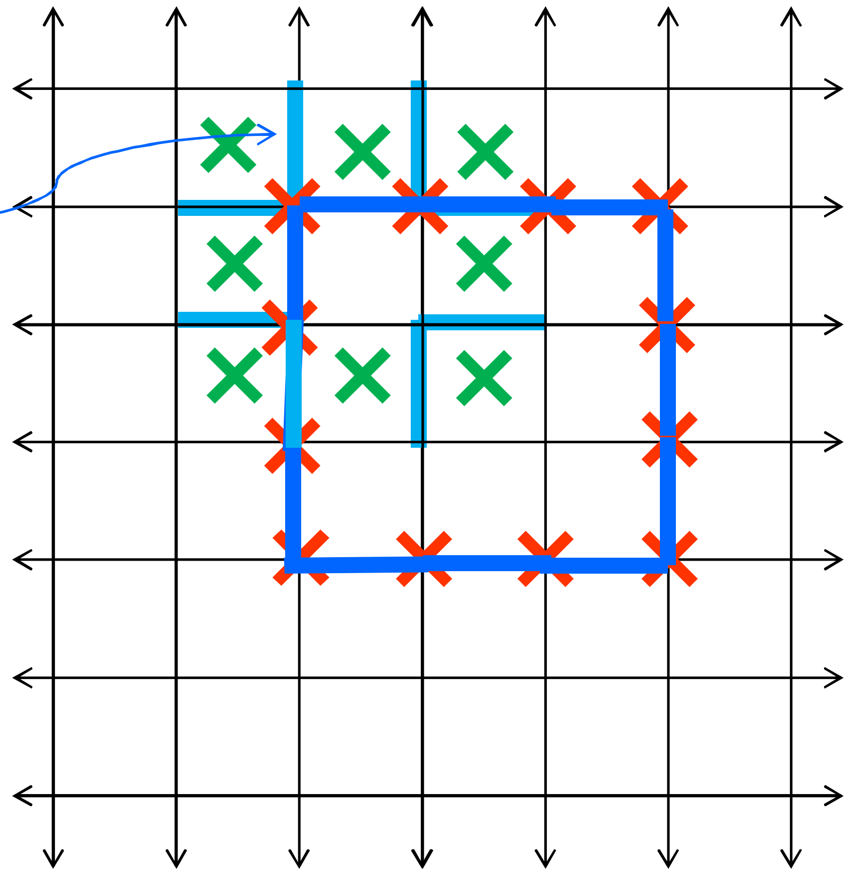
Braiding Defects?



World lines
in 2+1 D

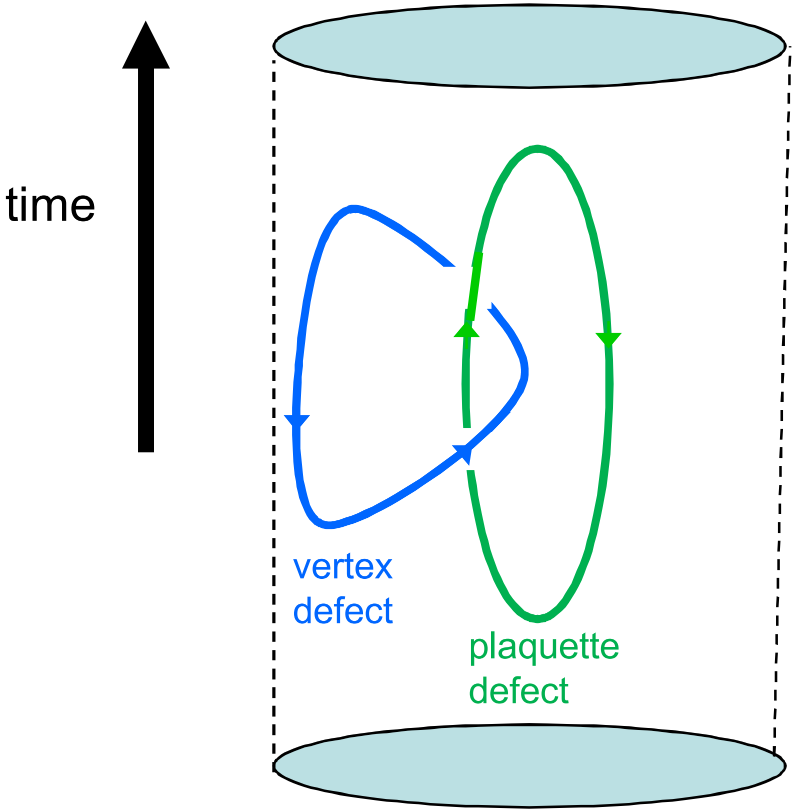
Braiding Defects

apply σ^z to make / move plaquette defects



Dark blue loop = $\prod_{loop} \sigma_x = \prod_{\beta\, enclosed} P_\beta$ = -1 due to enclosed defect

Braiding Defects?



$= \quad -1$

OK, but why should we care?

Quantum codes based on the Toric code
are about the most robust and practical codes known!

But what does this have to do with:

Particles / Condensed Matter / Anyons:

Let there be a Hamiltonian

$$H = -\sum_{\alpha} V_\alpha - \sum_{\beta} P_\beta$$

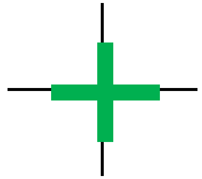
Ground state manifold
is code space…
(Error space is excitations)

Multiple ground states on torus that cannot be changed or distinguished by a local operations

"Topological Order" (Wen)

Rule 1) Specify all V_α=1

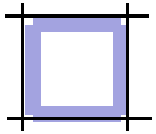
Rule 2) Specify all P_β=1



= Vertex operator

$$V_\alpha = \prod_{i \in vertex\ \alpha} \sigma_i^z = \prod_{i \in vertex\ \alpha} Z_i$$

= +1 if an even # of up spins
= -1 if an odd # of up spins



= Plaquette operator

$$P_\beta = \prod_{i \in plaquette\ \beta} \sigma_i^x = \prod_{i \in plauqette\ \beta} X_i$$

= flip spins around a plaquette

Quantum Error Correcting Code

$\Updownarrow$

Topological Ordered Matter

Ground States = Code Space

Working Definition of Topological Order

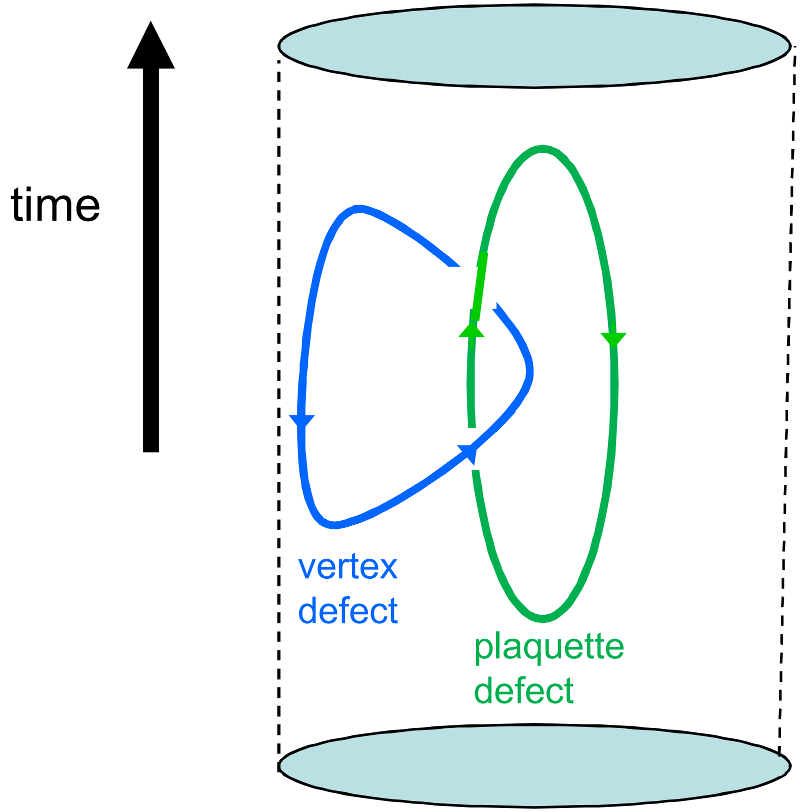
1. Ground state degeneracy on a torus

2. Matrix element between ground states:

$$\langle \psi_i | \text{any local operator} | \psi_j \rangle = C\delta_{ij} \quad + \text{small}$$

This always implies exotic braiding statistics for particles

Braiding Defects?



$= \quad -1 \quad (!!!)$

Let there be a Hamiltonian

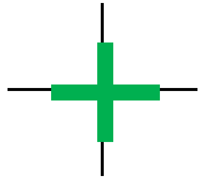
$$H = -\sum_{\alpha} V_\alpha - \sum_{\beta} P_\beta$$

Ground state manifold
is code space
(Error

Multiple ground states on torus that cannot be changed or distinguished by a local operations

Rule 1) Specify all V_α=1

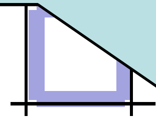
pecify all P_β=1

 = Vertex operator

$$V_\alpha = \prod_{i \in vertex\, \alpha} \sigma_i^z = \prod_{i \in vertex\, \alpha} Z_i$$

= +1 if an even # of up spins

ns

 tte operator

$$P_\beta = \prod_{i \in plaquette\, \beta} \quad \prod_{i \in plaquette\, \beta} X_i$$

pins around a plaquette

Pretty strange looking Hamiltonian !

But this exactly solvable H tells us about an entire phase of matter

Topological Robustness

What if we perturb the Hamiltonian a bit? Does anything change?

$$H = H_{toric\,code} + \lambda \delta H$$ example $$\delta H = \sum_i \sigma_i^x$$

Ground state wavefunction:

$$|\tilde{\psi}\rangle = |\psi\rangle + (G\,\delta H)|\psi\rangle + (G\,\delta H)^2|\psi\rangle + \ldots$$

$$G(E) = \sum_{n\,\mathrm{excited}} \frac{\lambda|n\rangle\langle n|}{E - E_n^0}$$

Each term is smaller by (λ/Gap)

$$E = \text{ diagonalize } \langle\tilde{\psi}_i|H_{toric} + \lambda\delta H|\tilde{\psi}_j\rangle$$

Matrix proportional to unity up to order $(\lambda/\mathrm{Gap})^{\text{Length of torus}}$

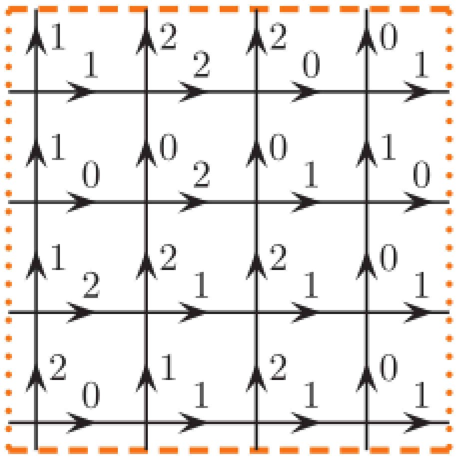
Ground state degeneracy is topologically robust!

(For large system it fails only when there is a phase transition)

More interesting models like the toric code

(each exactly solvable case describes a phase of matter)

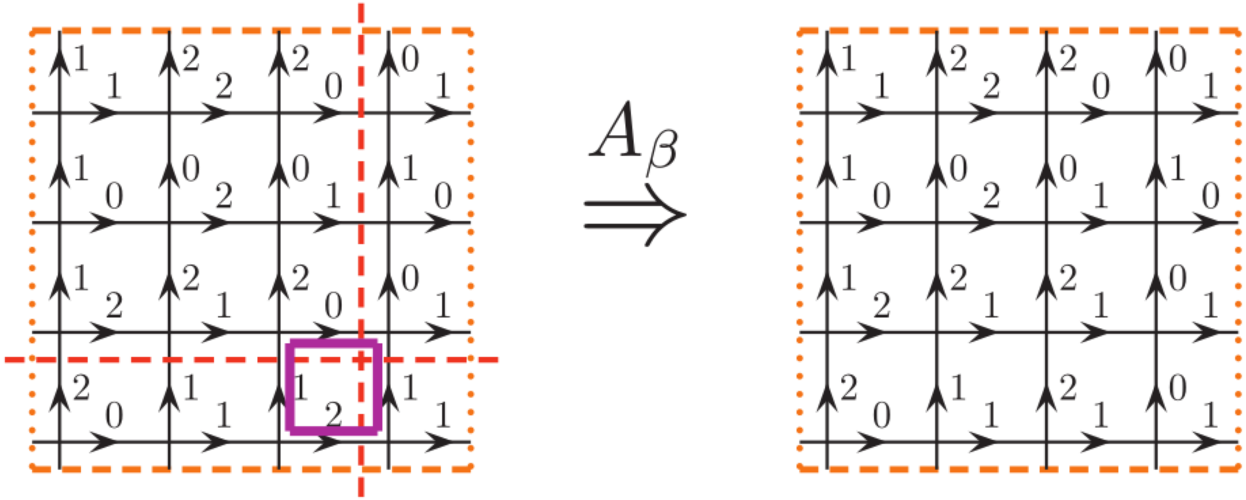
Z3 toric code: Use qutrits not qubits



Vertex Rule: Arrows add to zero mod 3

(addition is modulo 3 i.e., group Z3)

Z3 toric code: Use qutrits not qubits



Plaquette Operator: Increment clockwise,
decrement counterclockwise

Conserves “flux” mod 3.

Z3 toric code: Use qutrits not qubits

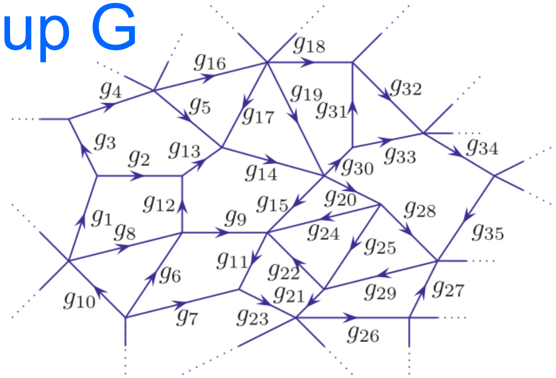
Spectrum:

Z_3 “charges” (plaquette defects)
Z_3 “fluxes” (vertex defects)

Braiding charge around flux gives phase $e^{2\pi i/3}$

“Dyon” (combination of flux and charge) is an anyon!

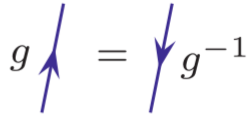
Kitaev ('97) Quantum Double Model for Group G



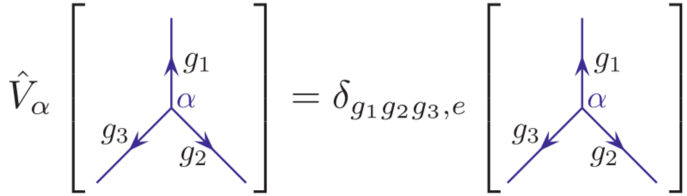
Choose any lattice (even irregular) with arrows

Edges labeled with elements of discrete group G

Reversing arrow inverts label

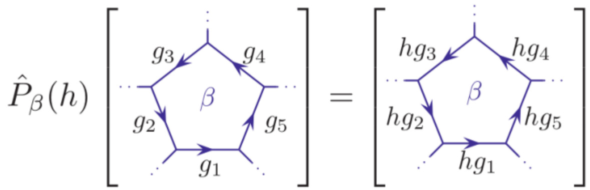


Vertex operator
enforce group multiplication



Plaquette operator
enforce group multiplication

$$\hat{P}_\beta = \frac{1}{|G|} \sum_{h \in G} \hat{P}_\beta(h)$$

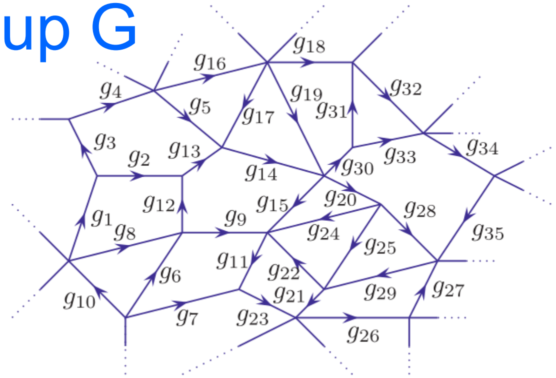


Nonabelian group gives nonabelian Anyons!

--- mostly equivalent to discrete lattice gauge theory!

Advert: Upcoming work with Jean-Noel Fuchs, Julien Vidal, Anna Ritz Zwilling, Benoit Doucot

Kitaev ('97) Quantum Double Model for Group G



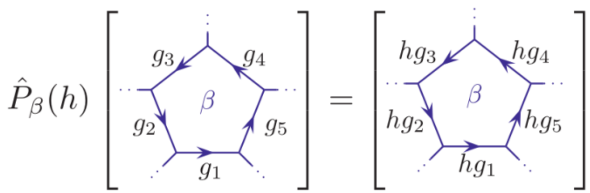
Degenerate Hilbert space from Fusion of Reps:

$$R_1 \otimes R_2 = R_3 \oplus R_4 \oplus \ldots$$

Character of Rep R

Projector to excited plaquette $\quad \hat{P}^R_\beta = \frac{d_R}{|G|} \sum_{h \in G} \chi^R(h) \hat{P}_\beta(h)$

Plaquette operator
enforce group multiplication $\quad \hat{P}_\beta = \frac{1}{|G|} \sum_{h \in G} \hat{P}_\beta(h)$



Nonabelian group gives nonabelian Anyons!

--- mostly equivalent to discrete lattice gauge theory!

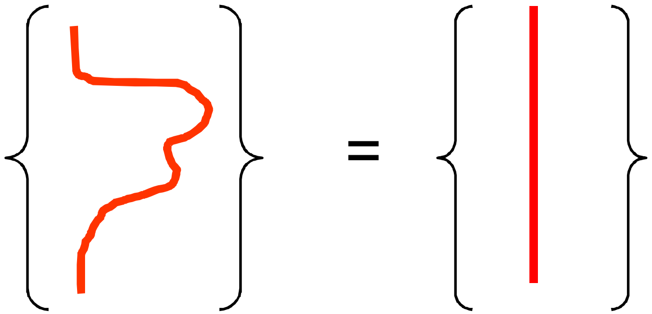
Advert: Upcoming work with Jean-Noel Fuchs, Julien Vidal, Anna Ritz Zwilling, Benoit Doucot

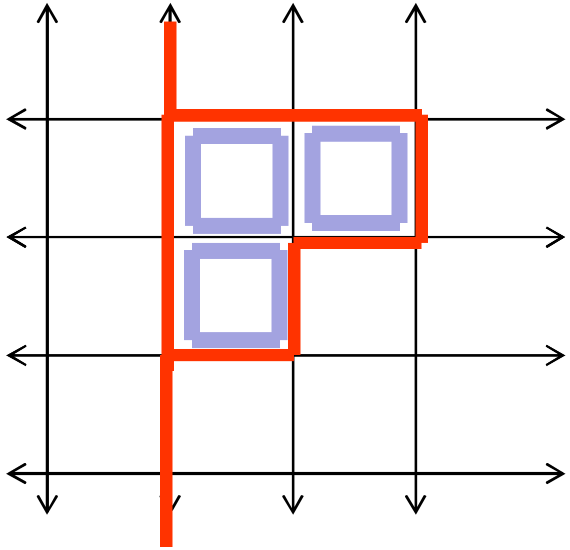
Abstracting the Toric Code

$$|\psi\rangle = \sum |\text{loop config}\rangle$$

Sum is over all loop configs that can be obtained by flipping plaquettes

1. Isotopy



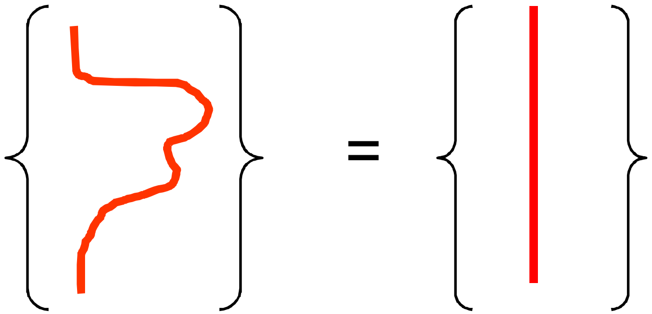


Abstracting the Toric Code

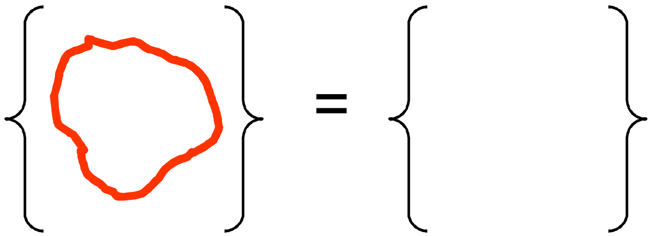
$$|\psi\rangle = \sum |\text{loop config}\rangle$$

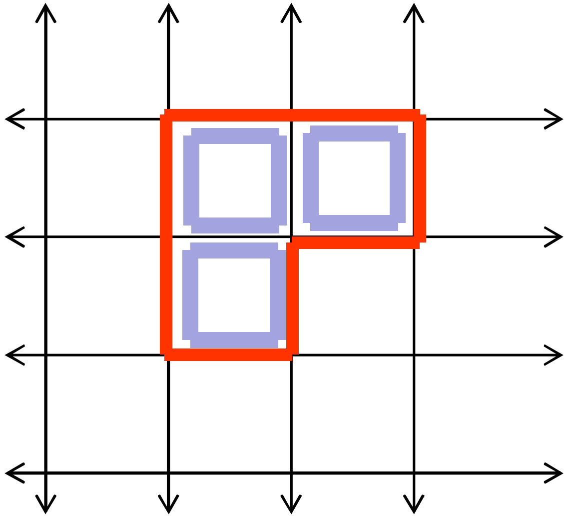
Sum is over all loop configs that can be obtained by flipping plaquettes

1. Isotopy



2. Loop Removal



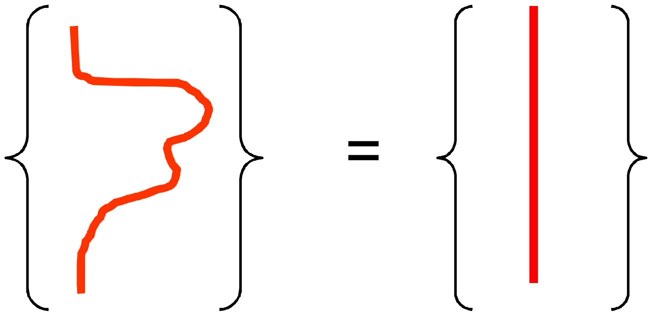


Abstracting the Toric Code

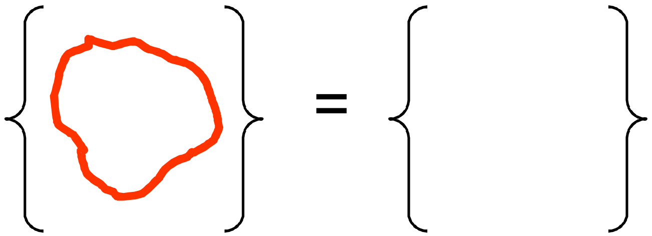
$$|\psi\rangle = \sum |\text{loop config}\rangle$$

Sum is over all loop configs that can be obtained by flipping plaquettes

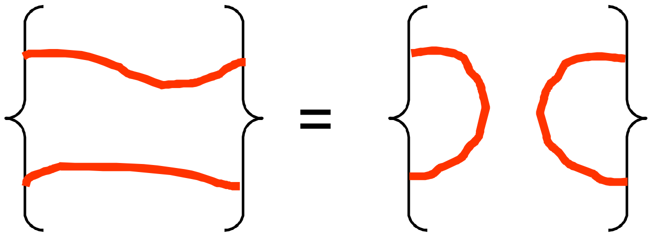
1. Isotopy

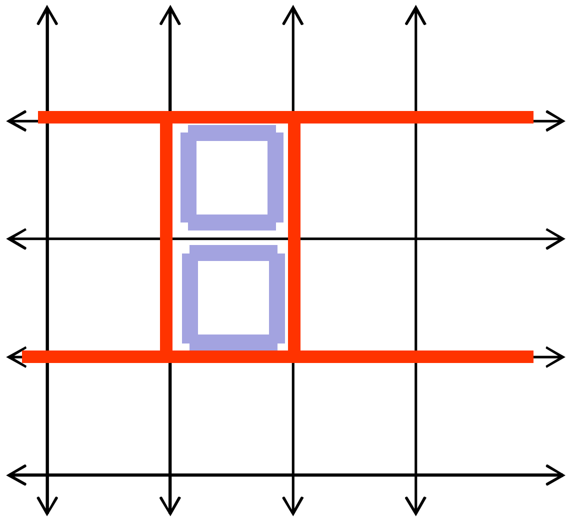


2. Loop Removal



3. Surgery



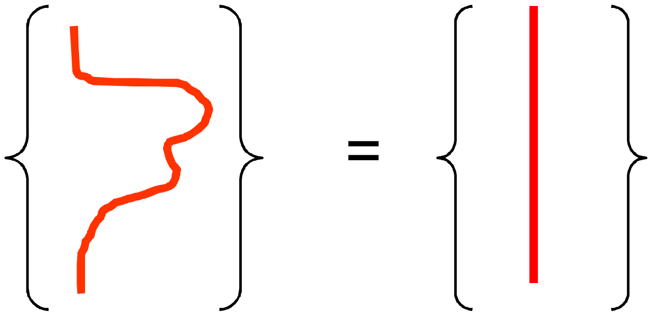


Abstracting the Toric Code

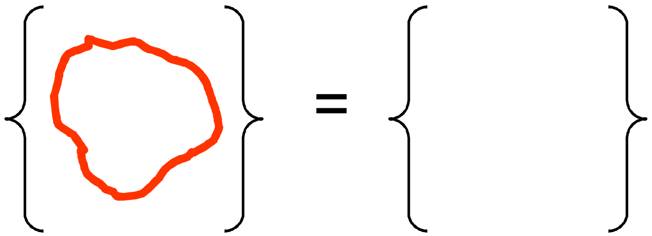
$$|\psi\rangle = \sum |\text{loop config}\rangle$$

Sum is over all loop configs that can be obtained by ~~flipping plaquettes~~ rules 1-3

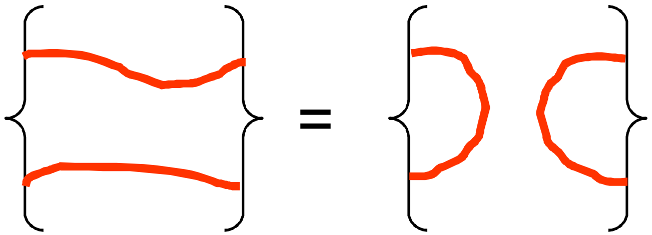
1. Isotopy

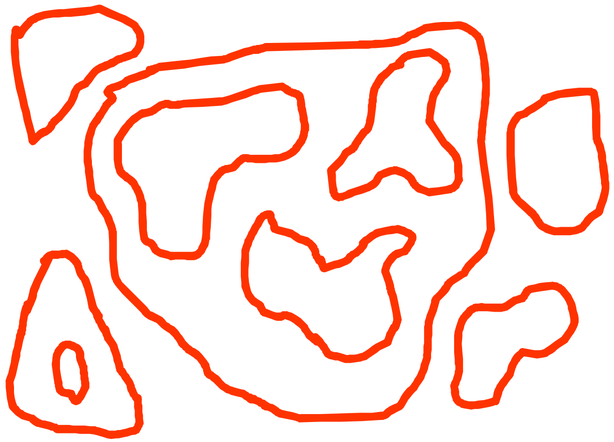


2. Loop Removal



3. Surgery





Continuum Loop Gas

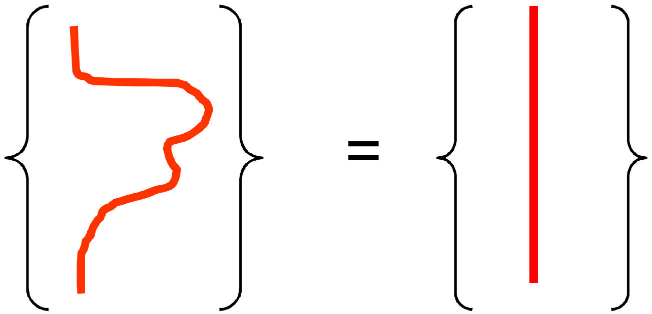
Modifying the game:

By adding phases to plaquette operator (messy)

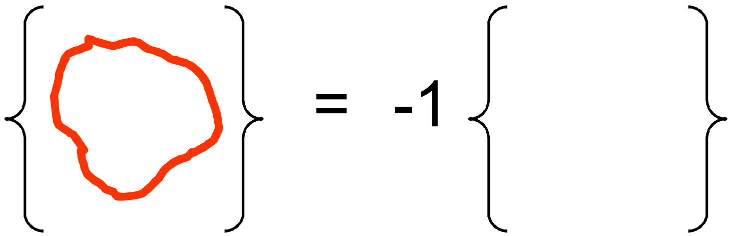
$$|\psi\rangle = \sum (-1)^{\#\, loops} |\text{loop config}\rangle$$

$\theta = \pi/2$
mid way between
bose and fermi

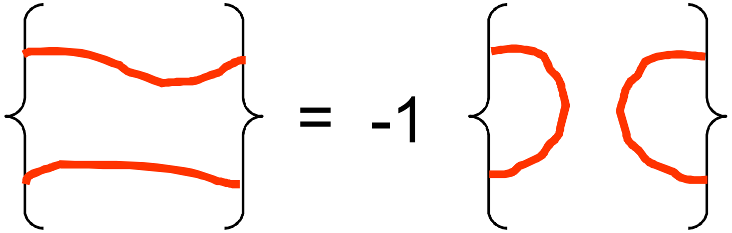
1. Isotopy



2. Loop Removal

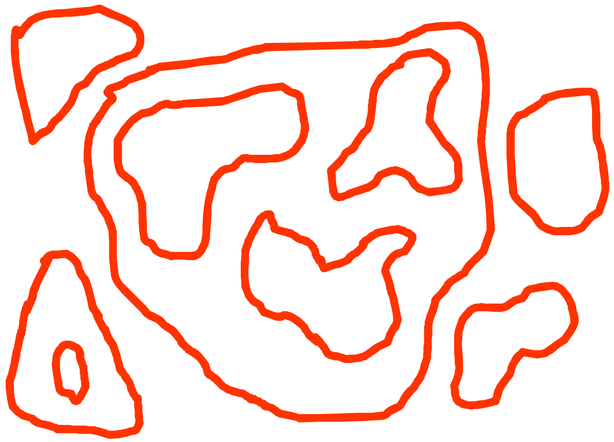


3. Surgery



Doubled Semion Model

Levin and Wen, Phys. Rev. B71, 045110 (2005)
Freedman et al, Annals of Physics 310 428 (2004)



Continuum Loop Gas

Same ground state degeneracy
Defects get different braiding phases

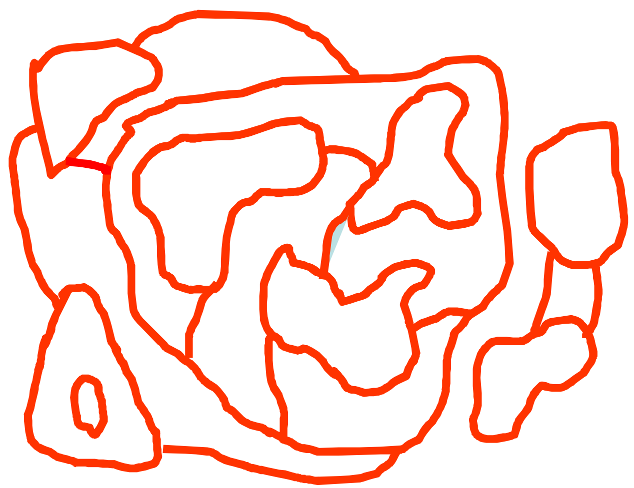
How much can we generalize?

Levin and Wen, Phys. Rev. B71, 045110 (2005)

How much can we generalize?

Levin and Wen, Phys. Rev. B71, 045110 (2005)

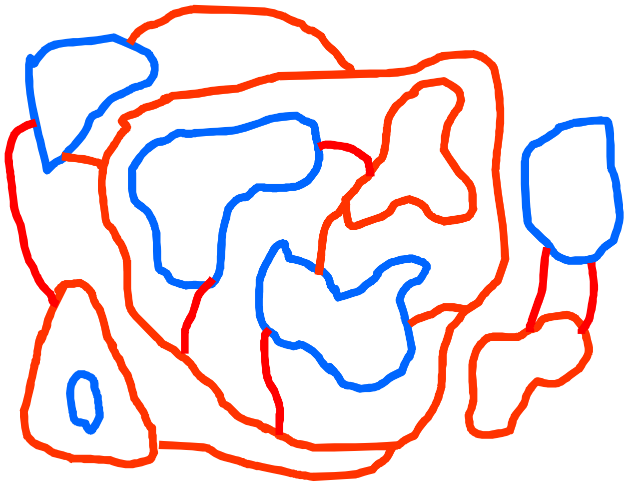
1. Allow branching loops (endpoints are defects)



How much can we generalize?

Levin and Wen, Phys. Rev. B71, 045110 (2005)

1. Allow branching loops (endpoints are defects)
2. Allow multiple “color” edges (color branching rules)

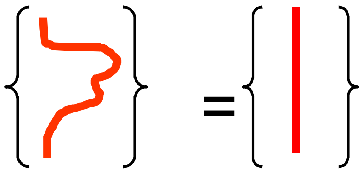


How much can we generalize?

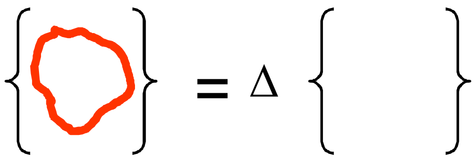
Levin and Wen, Phys. Rev. B71, 045110 (2005)

1. Allow branching loops (endpoints are defects)
2. Allow multiple “color” edges (color branching rules)
3. $$\psi = \sum w(\text{config})\,|\text{config}\rangle$$

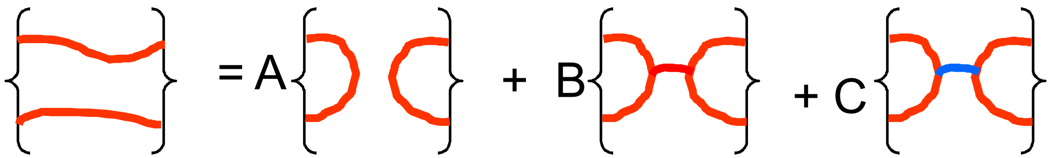
Isotopy



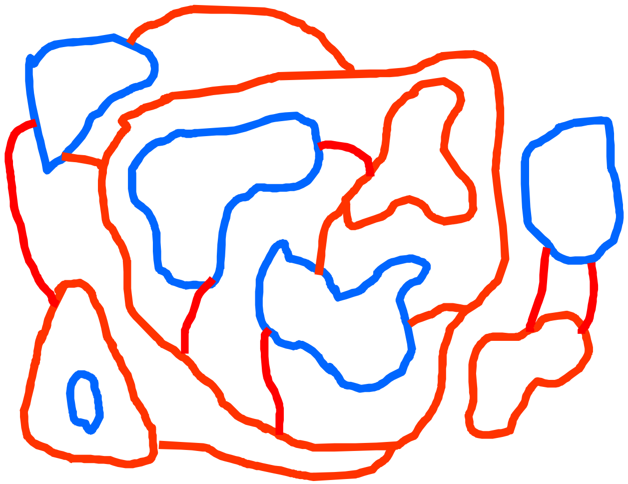
Loop Removal



Surgery



Consistent diagram rules
= Fusion Category

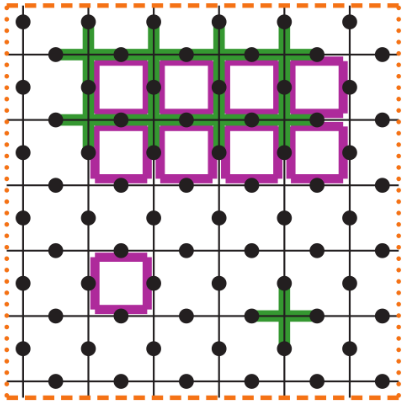


Generate all Drinfeld Center TQFTs

Back to the Toric Code

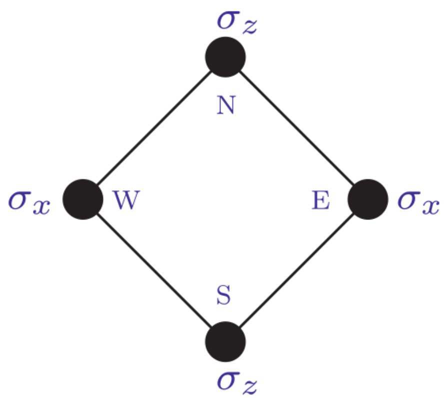
Anyon permuting symmetry

(V and P look similar)



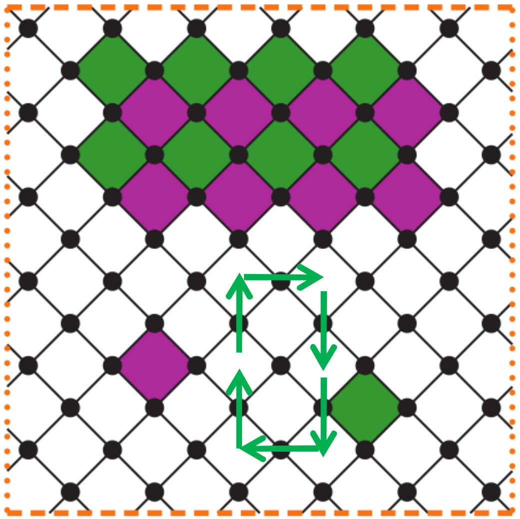
Rotate spins (X to Z,   Z to –X) on east and west corners of green

Every plaquette looks the same

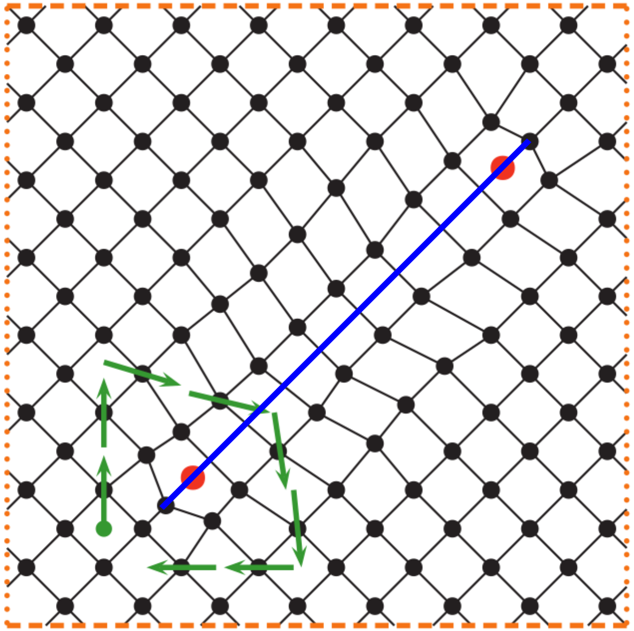


Wen 2003

Symmetry Defect (Bombin 2010):



Dislocations



Vertex defect comes back
as a plaquette defect!

Adding pair of dislocations removes M plaquettes and M -1 spins.

Removes one constraint – introduces one logical qubit
split between two dislocations

Roushan Lecture!

THE END

Steven H. Simon

UNIVERSITY OF OXFORD

Oxford Physics

Thank you for Listening