# Fault-tolerant optical quantum computation with cluster states


**Michael A. Nielsen**

**School of Physical Sciences**

The University of Queensland

QiSci Quantum Information Science Initiative

Chris Dawson (UQ)
Henry Haselgrove (UQ)

# An optical quantum computer?

$|0\rangle \equiv$ horizontally polarized single photon

$|1\rangle \equiv$ vertically polarized single photon

Very long decoherence times

**Operations**

State preparation

Single-qubit gates

Measurement

Current technology adequate for basic experiments

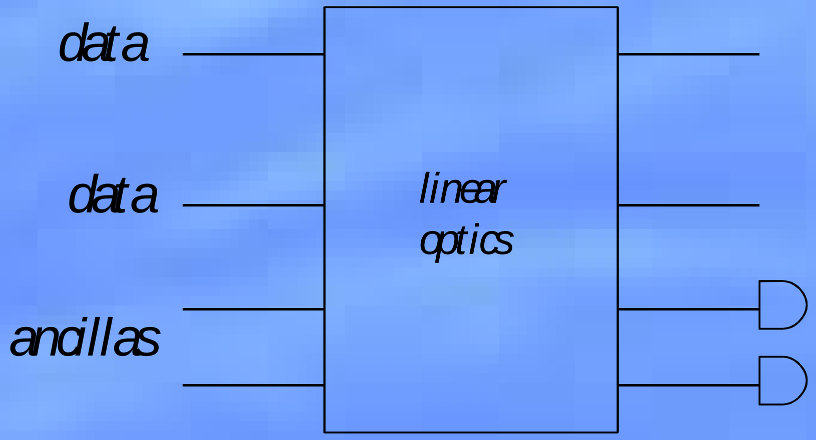
# An optical quantum computer?

## Can we do an entangling gate?

*CPHASE* $|x\rangle|y\rangle = (-1)^{xy}|x\rangle|y\rangle$

*Impossible with linear optics alone*

*Knill-Laflamme-Milburn (Nature, 2001) showed how to do this in a non-deterministic but heralded fashion.*

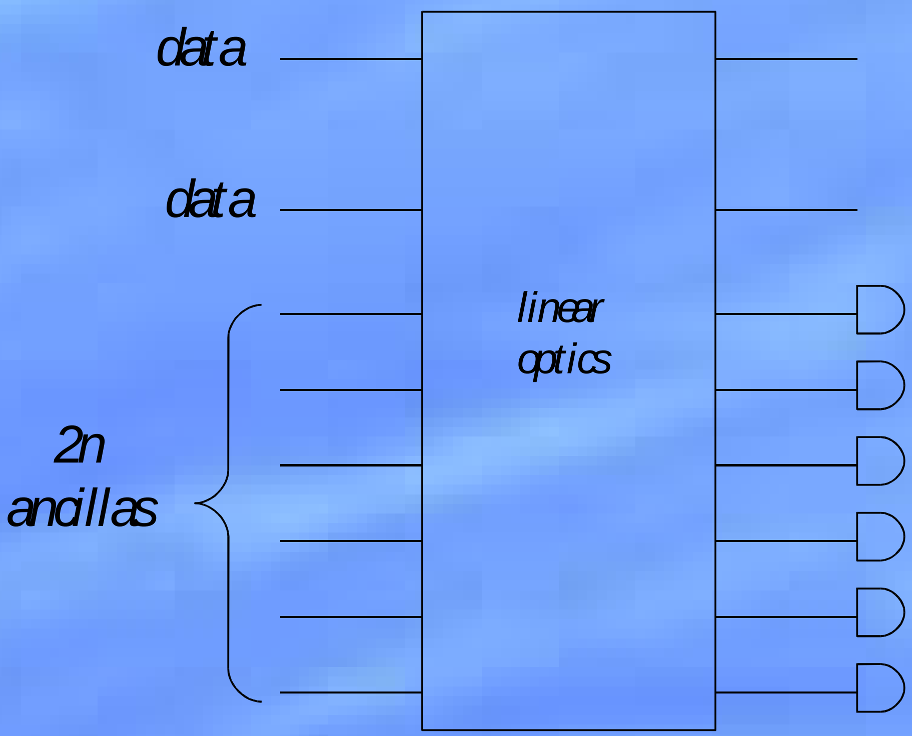



**Success:** *A single photon is measured at each port. Occurs with probability ¼.*

**Failure:** *Data measured in the computational basis.*

*KLM increase the probability of success using **two steps**:*

***Step 1***




***Success:*** *A single photon is measured at each port. Occurs with probability $(n/(n+1))^2$.*

***Failure:*** *Data measured in the computational basis.*

*Making n large increases the success probability, but makes doing the gate harder.*

***Sequential performance of the gate***

*Interact ancillas with data qubit 1.*

*Measure half the ancillas ! success probability n/(n+1).*

*Interact ancillas with data qubit 2.*

*Measure half the ancillas ! success probability n/(n+1).*

***Step 2 for increasing the probability of success:*** *Probability of success can be boosted closer to 1 using quantum error-correction.*

***Disadvantage:*** *Probability of success close to 1 requires $10^4$-$10^5$ optical elements to do a single entangling gate.*

***Is there a better way?***

*We'll use the n = 2 gate, which succeeds with probability $(2/3)^2$.*

*Higher values of n turn out not to be necessary.*

*No error-correction is required.*

*More like $10^2$ optical elements for a CPHASE gate.*

*The key is to combine the n = 2 KLM gate with the* ***one way quantum computer*** *or* ***cluster state*** *model of quantum computation (Raussendorf and Briegel, PRL 2001).*

# Overview of cluster-state computation

**Three steps:**

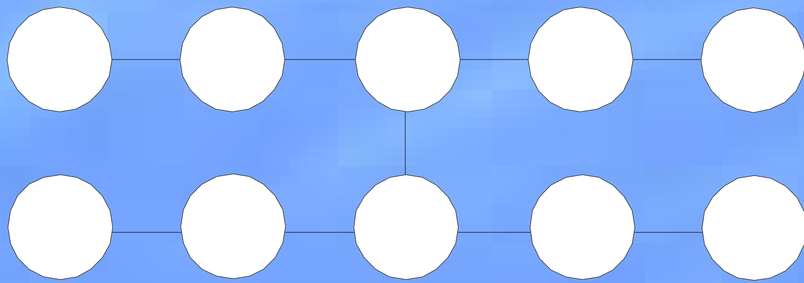
1. Prepare a many-qubit state, the cluster state.

2. Perform a sequence of adaptive single-qubit measurements on some subset of the cluster qubits.

3. The remaining qubits are the output of the computation.

These three steps can be used to simulate an arbitrary quantum circuit.

# Defining the cluster state



*Each node represents a qubit in the cluster*

*We* *define* *the cluster state as the result of the following two-stage preparation procedure:*

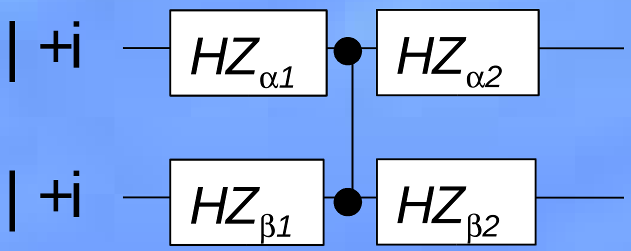
*1. Prepare each qubit in the state* $|+\rangle \equiv |0\rangle + |1\rangle$

*2. Apply a CPHASE gate between each pair of connected qubits.*

*It doesn't matter in which order the CPHASE gates are applied.*

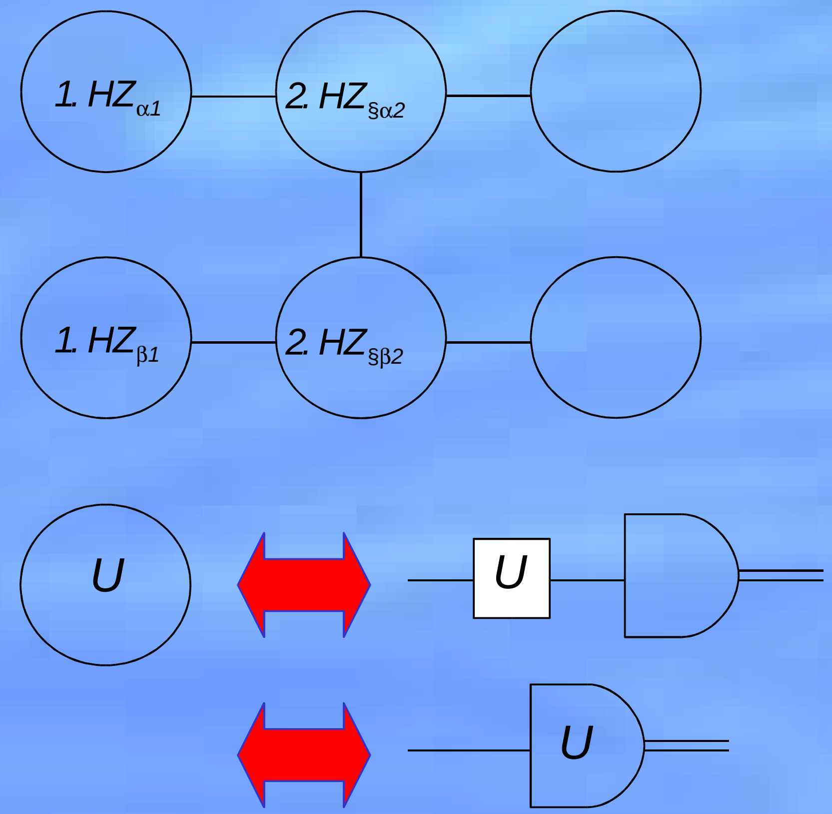
# Recipe to simulate a quantum circuit

## Circuit to be simulated

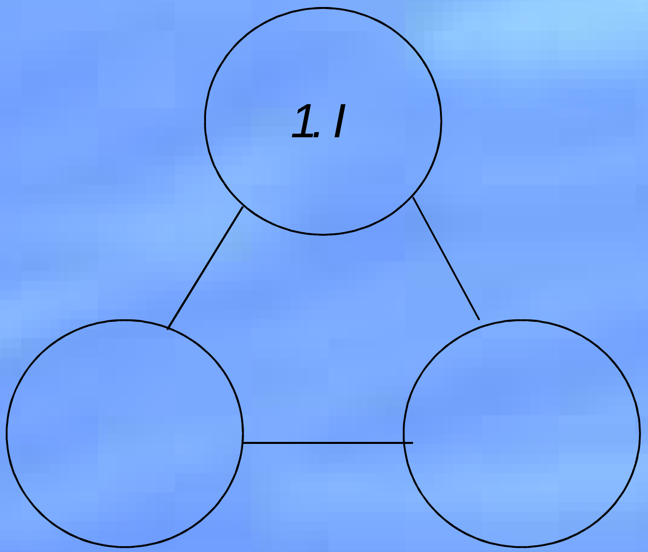



The gates $HZ_\alpha$ and CPHASE are universal.

## The cluster-state simulation



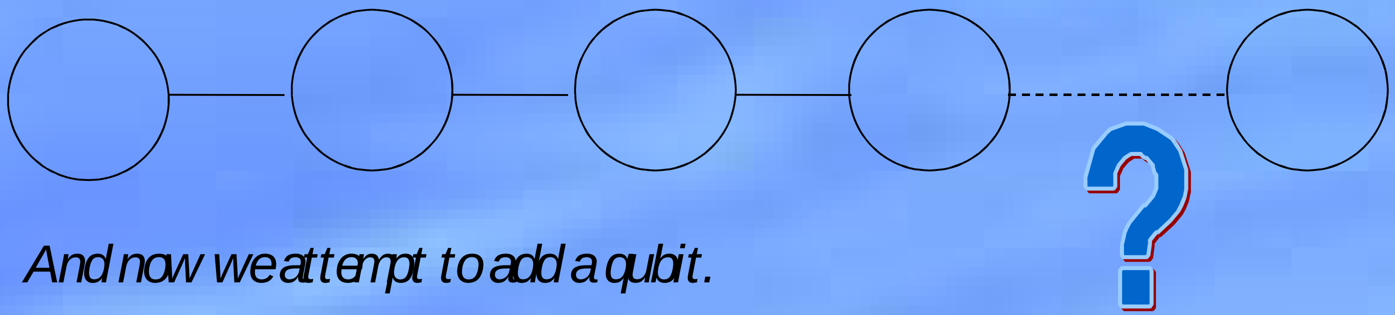
## *What happens if we measure a cluster qubit in the computational basis?*




*Measuring a cluster qubit in the computational basis simply removes it from the cluster.*

# Combining cluster-state computation with the KLM $(2/3)^2$ gate

*Suppose we've built up part of a cluster…*



*And now we attempt to add a qubit.*

***Success:*** *With probability 2/3 we add a qubit to the cluster.*

***Failure:*** *With probability 1/3 we lose a qubit from the cluster.*

*On average, we add 1/3 of a qubit to the cluster, per KLM gate.*

*Of course, it's not enough just to build up a linear cluster, we need a planar cluster.*

# Building general clusters using the KLM $(2/3)^2$ gate

*Can build up a general cluster using similar random walk ideas.*

***Result:*** *On average, we add 1/9 th of a qubit to the cluster per KLM gate performed.*

# Summary

*Basic KLM gate with success probability $(2/3)^2$ can be used to quickly build up clusters.*

➡ *Optical quantum computation (Nielsen, PRL, 2004).*

***Why this works***

1. *Failure mode of KLM gate is a computational basis measurement. Coincidentally, such a measurement simply deletes a cluster qubit.*

2. *Because the cluster is a **fixed** state, it's okay to lose a qubit, provided we can rebuild. In particular, losing quantum information is not a problem!*

# *What about noise?*

*A proposal for quantum computation should be able to tolerate a reasonable level of physical noise.*

- *in abstract circuit models, the techniques of* ***fault-tolerant quantum computation*** *enable a* ***threshold for quantum computation***

*For most proposals (e.g superconductors, KLM, ion trap,..).a physical threshold value follows from straightforward modifications of theoretical threshold constructions.*

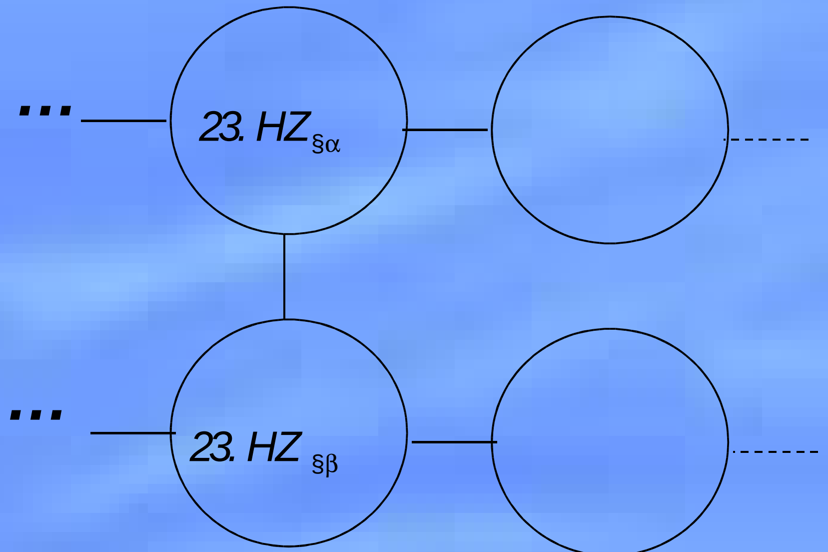
*With clusters, fault-tolerance is less obvious.*

- *In principle resolution found by Nielsen and Dawson (quant-ph, 2004). C.f. Raussendorf (thesis, 2003)*

# Two problems in proving a threshold

1. If we prepare too much of the cluster at once, some qubits will decohere.

**Possible solution**




Only add extra qubits slowly into the cluster: "just-in-time" preparation.

2. If we build the cluster up just-in-time, won't the stochastic nature of the KLM gate make erasing the cluster a possibility?

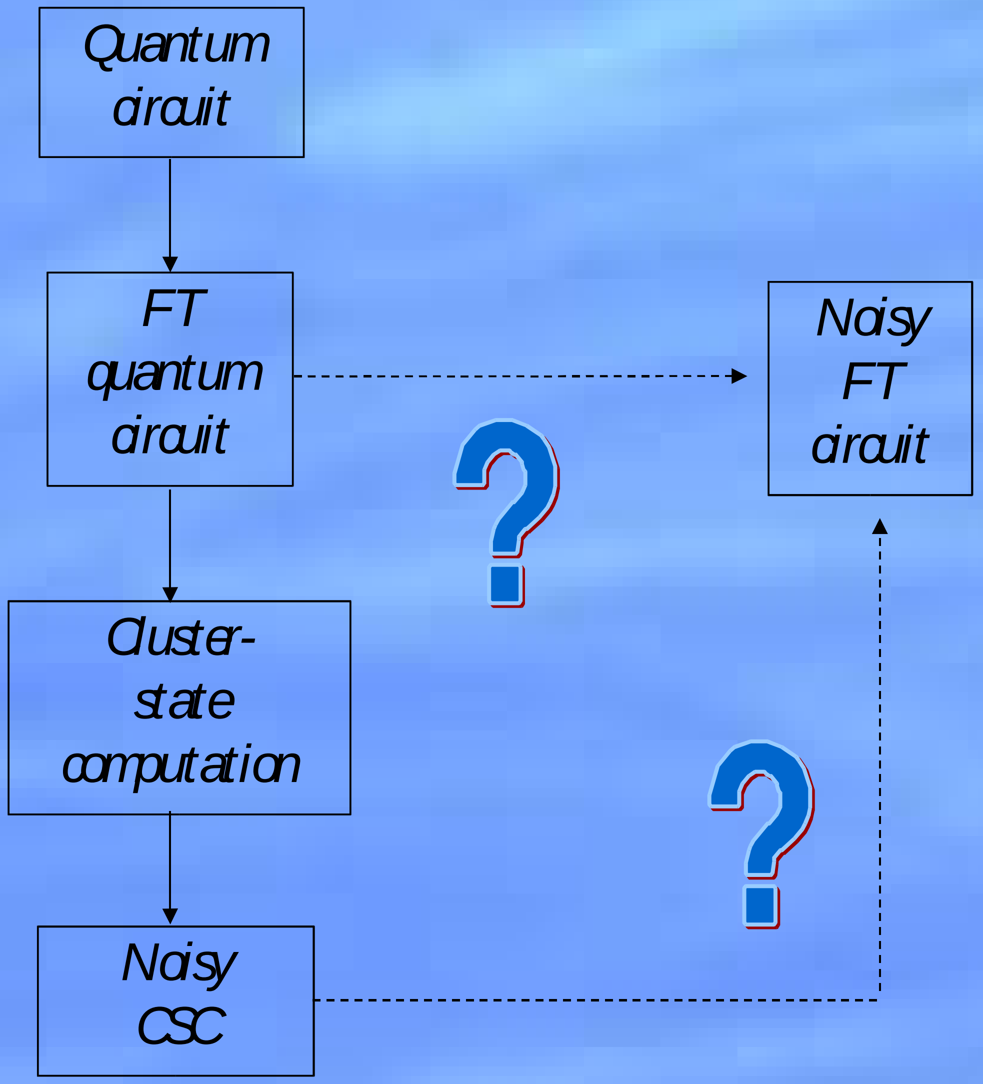
Yes, but this can be dealt with by building error-correction into the cluster.

# Basic idea

**Q:** *Is it possible to map noise in the CSC back to equivalent noise in the FT circuit?*

**Q:** *Is "physically reasonable" noise in the CSC mapped back to physically reasonable noise in the FT circuit?*

***Yes!*** *Such a mapping can be constructed. (Involved.)*



# What properties does the noise mapping have?

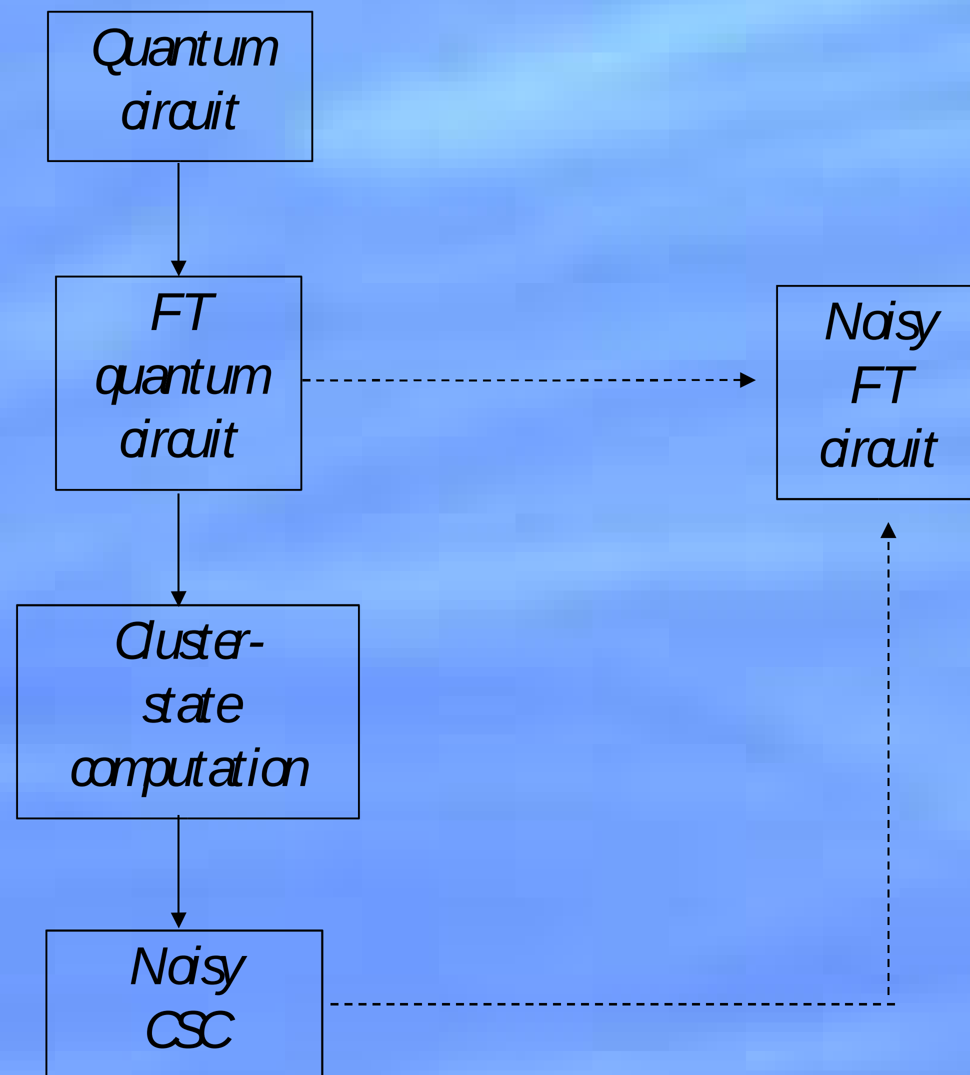
*Local, Markovian noise in CSC*

↓

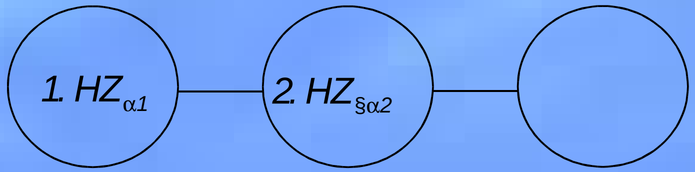
*Local, non-Markovian noise of about the same strength.*

*Terhal / Burkard (2004): There is a threshold for local non-Markovian noise in quantum circuits.*

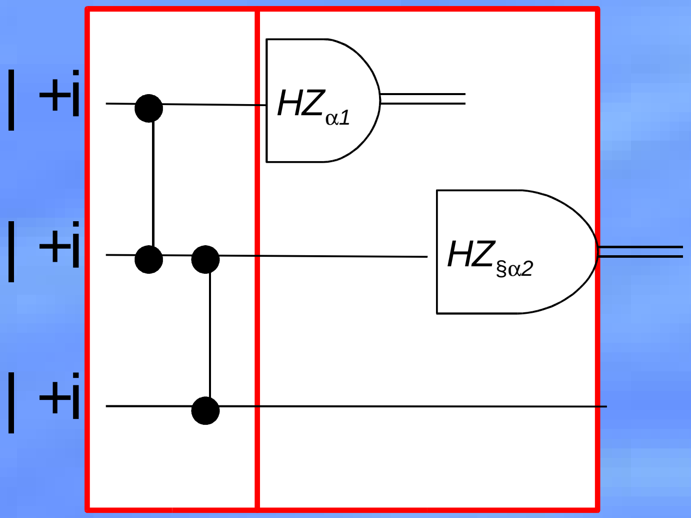
→ *There is a threshold for clusters!*



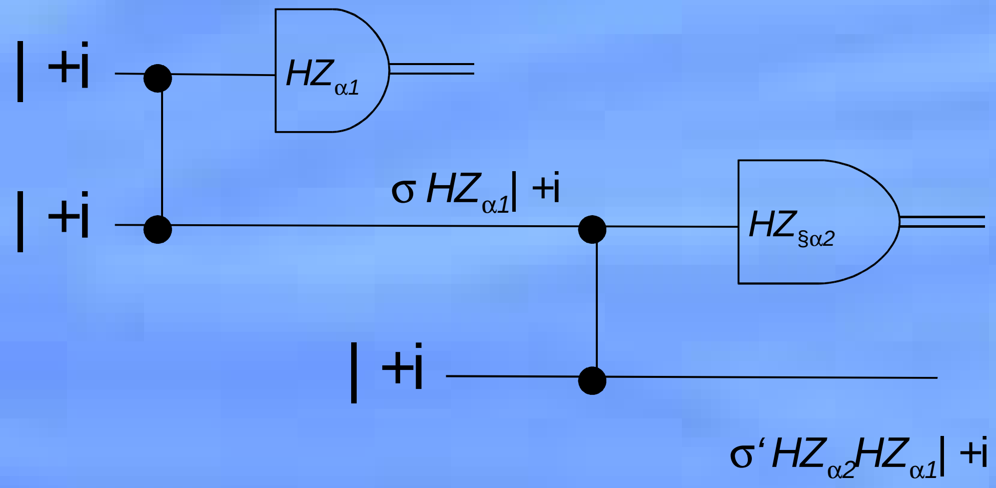
# Constructing the noise mapping for a toy example




*Standard implementation*



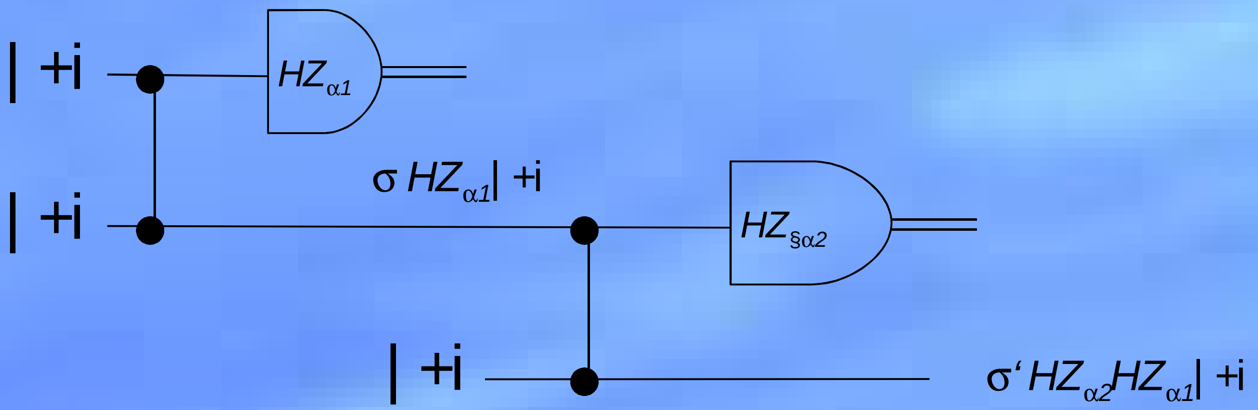

*Buffered implementation*



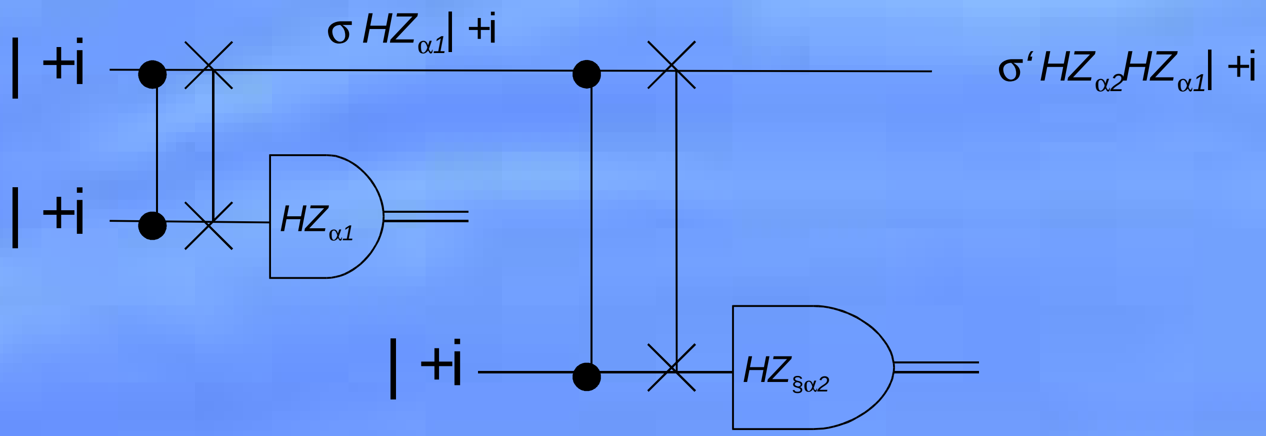

***Two points of view:*** *We'll be interested in both ideal circuit, and the real implementation of the circuit, with noisy elements.*

# Constructing the noise mapping for a toy example

*(Noisy) buffered cluster state computation*

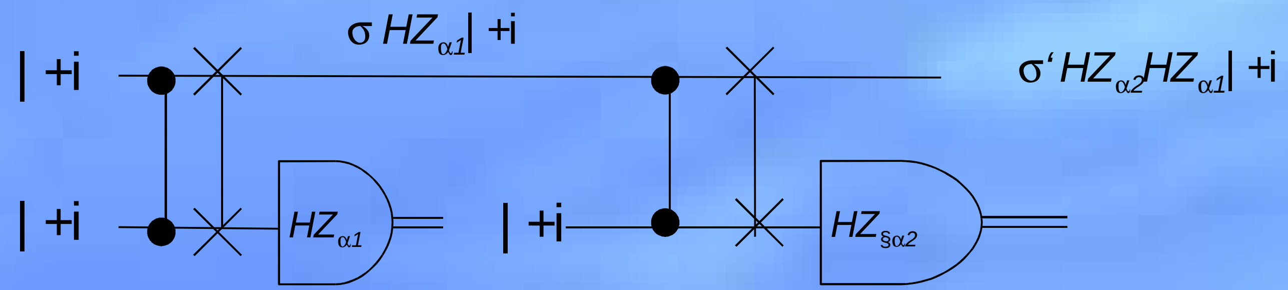



*Equivalent to:*

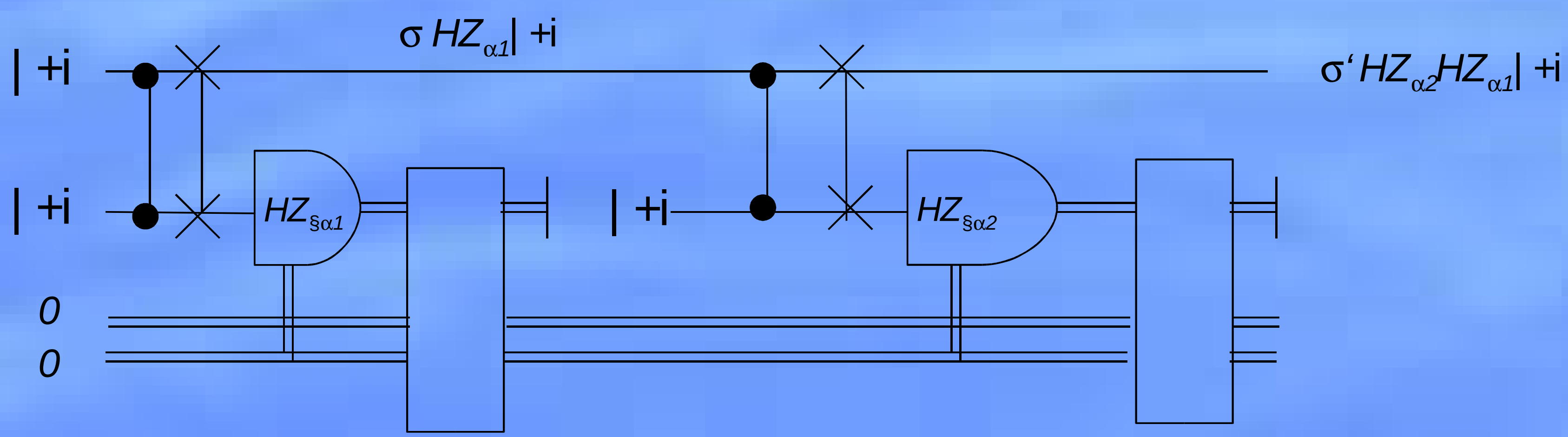


# Constructing the noise mapping for a toy example

*In a more compact form*

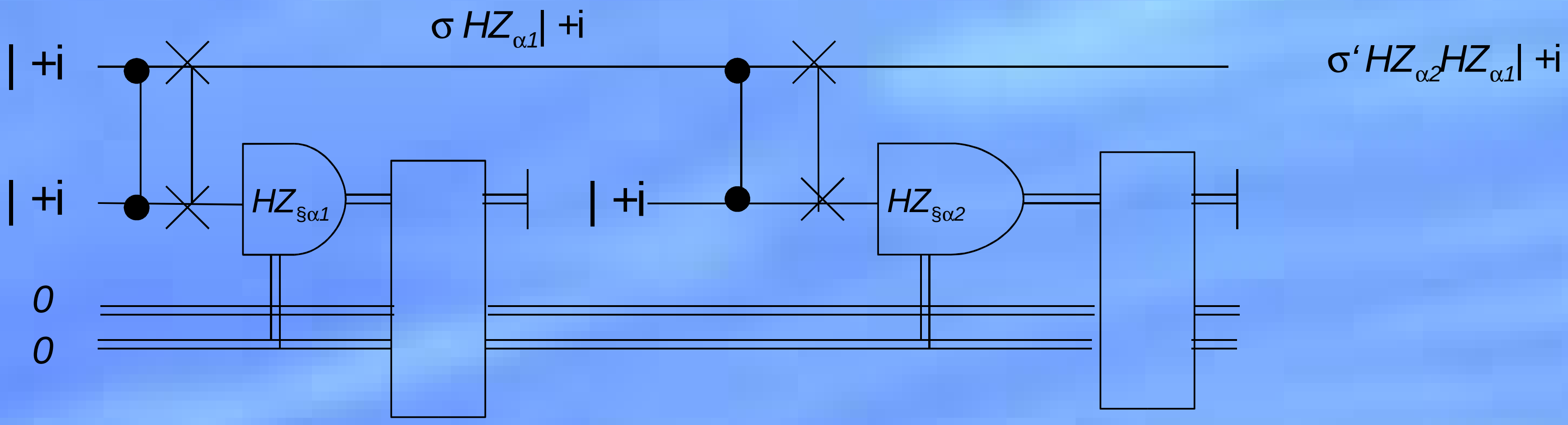



*Insert classical circuitry explicitly.*

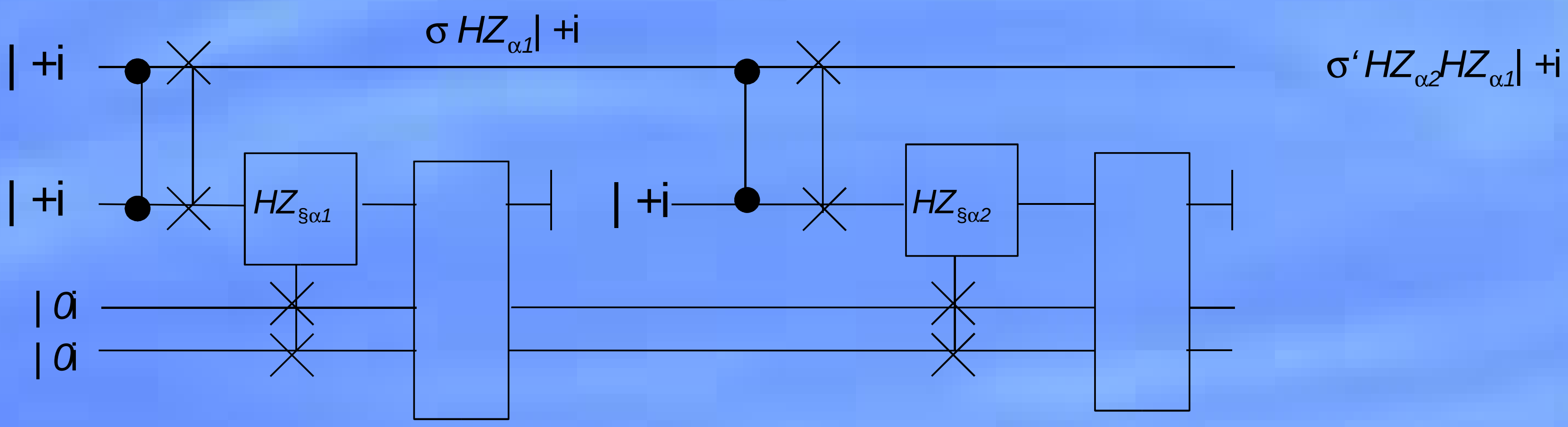


# Constructing the noise mapping for a toy example

*With classical circuitry*

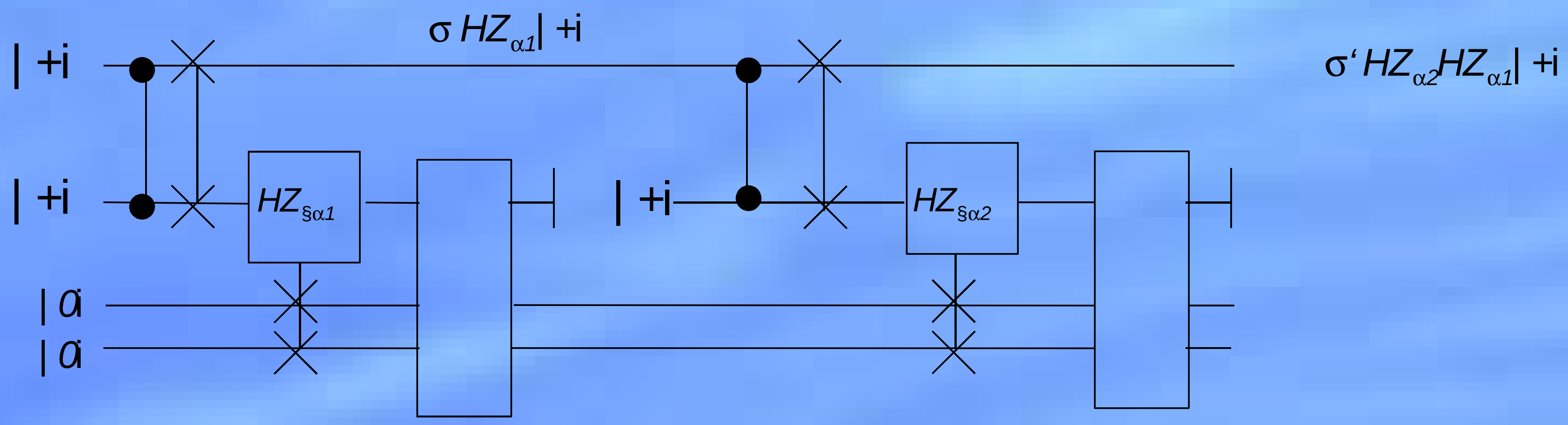



*Change classical to quantum*

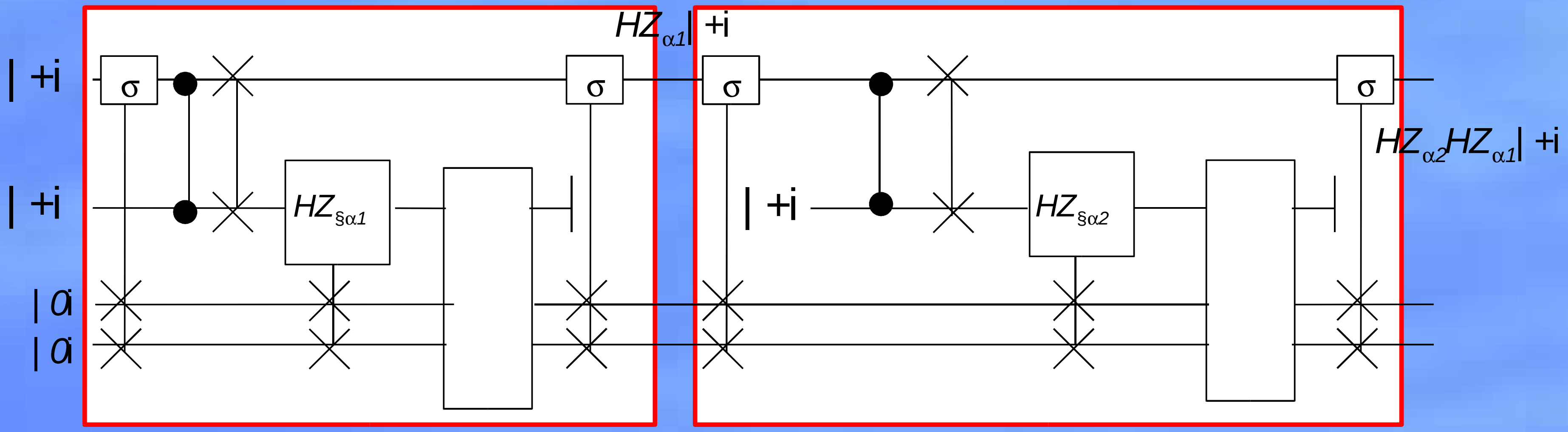


# Constructing the noise mapping for a toy example

*After changing classical to quantum*

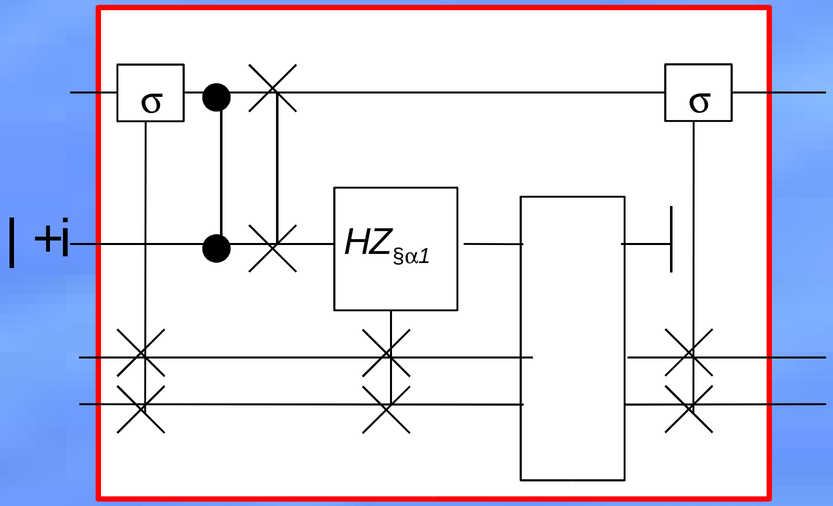



*Inserting the identity:*

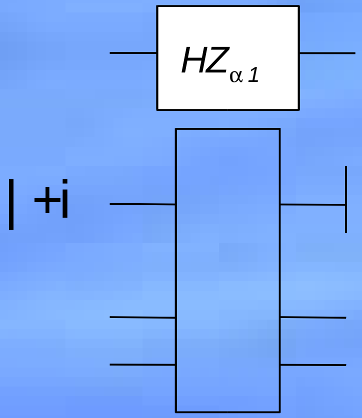


# Constructing the noise mapping for a toy example

*The cluster state computation is made up of repeating blocks of the form:*



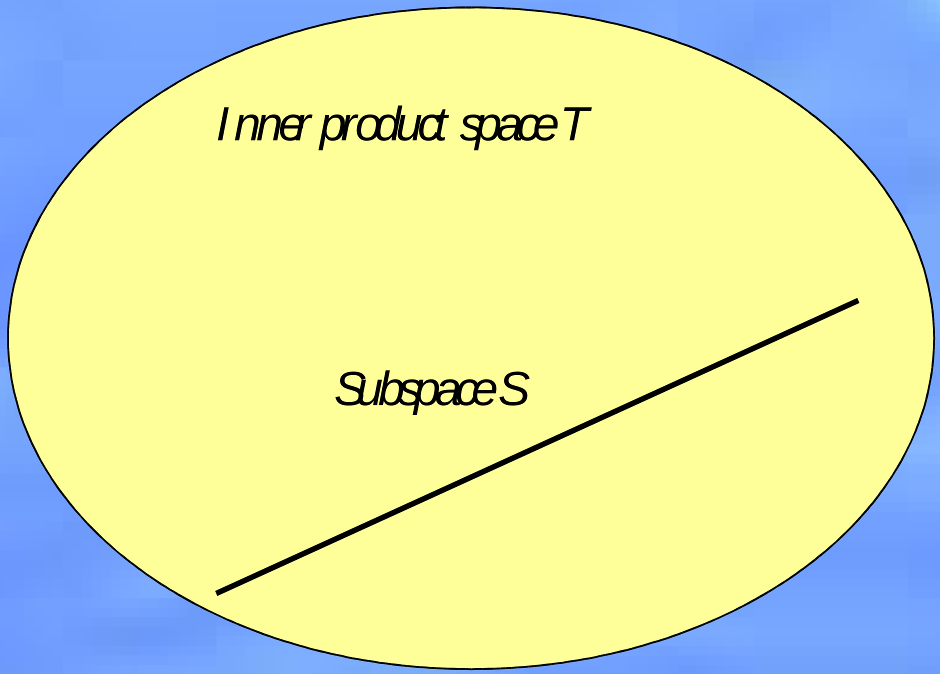

*When done perfectly this has the effect:*




*Intuitively, when some of the elements on the left are done imperfectly, we will get a noisy version of the gate on the right.*

# Constructing the noise mapping for a toy example

*The rigorous connection to the noise model of Terhal and Burkard may be made using the* ***unitary extension theorem****.*




*Let U and V be unitaries on T.*

*U and V may have quite different actions on T, but be quite similar on S.*

*E.g., $|U-V|$ may be large, while $|U_S - V_S|$ is small.*

*The unitary extension theorem guarantees the existence of a unitary W such that*

*(c)* $W_S = V_S$

*(d)* $|U - W| \cdot 2|U_S - V_S| = 2|U_S - V_S|$

# Constructing the noise mapping for a toy example

*Can extend the noise mappping to multiple-qubit cluster state computation using similar ideas.*

*The extension to optical cluster-state computation involves similar ideas, but also some additional ideas to cope with the occasional failure of the CPHASE.*

# *Conclusion*

*We are now doing numerical investigations of the threshold, basing our approach on Steane's threshold construction.*

*Goal is to see how (optical) cluster-state thresholds compare with standard constructions.*

*Best possible threshold?*

*Do simple proof-of-principle experiments.*

*Develop better sources and detectors.*