Creating Cluster States

Josh Kettlewell, Josh Waddington, Dr Pieter Kok

Department of Physics & Astronomy, University of Sheffield

The University Of Sheffield.

Abstract

Cluster states, first introduced by Raussendorf and Briegel[1] and initially conceived for optical lattices and linear optical computing, are essential entanglement resources which can be used for quantum computing through the use of linear operators. They are created via entangling operations on qubits with an arbitrary probability of success. As there is a lower limit placed upon the lifetime of qubits, and thus cluster states, we can represent a threshold for the qubit decay time in various dimension cluster states[2].

The One Way Model

Clusters are formed by entangling multiple qubits using CZ operations. These may then be used to process information.

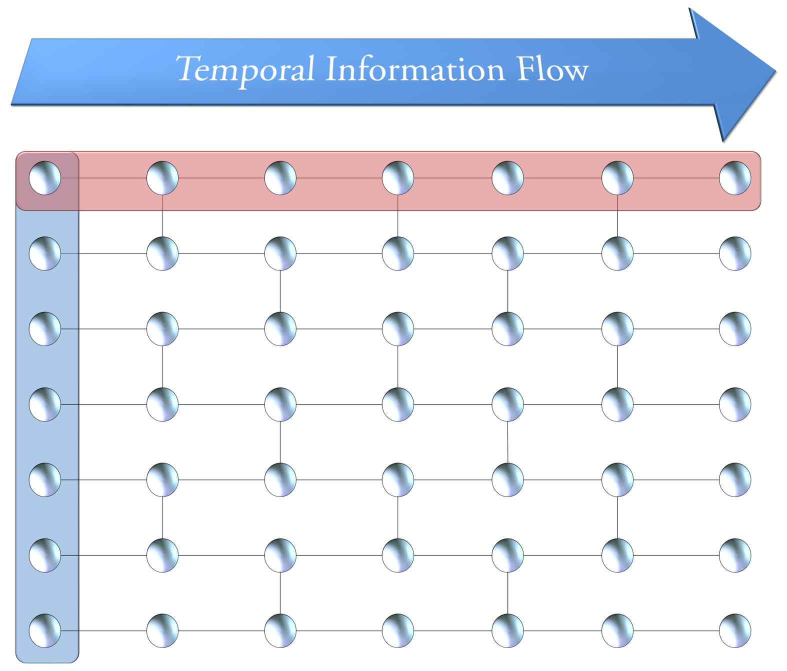

Figure 1: Information flow in a universal cluster state[3]. Circles present ancilla qubits in state $|+\rangle$ and lines represent entanglement

Here the red and blue bands represent a single logical and a single-qubit measurement respectively. The information is *pushed* to the right by the measurement which takes place from the top to the bottom of the leftmost column

Measurement Based Quantum Computing

In order to apply an arbitrary rotation $r(\theta)$ to a qubit eigenvalue, we require measurement;

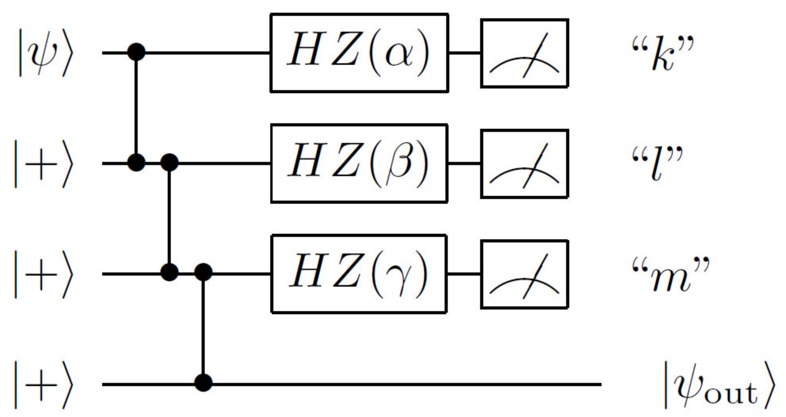

Figure 2: One way model[4]

where $|\psi_{out}\rangle = (X^m HZ(\gamma))(X^l HZ(\beta))(X^k HZ(\alpha))\,|\psi\rangle$. X^k, X^l and X^m are dependent on the realised values and commuting them through the Pauli gates and Hadamards will remove them. This leads to an adjustment of the measurement bases contingent to the previous measurement outcomes resulting in a temporal direction of information flow.

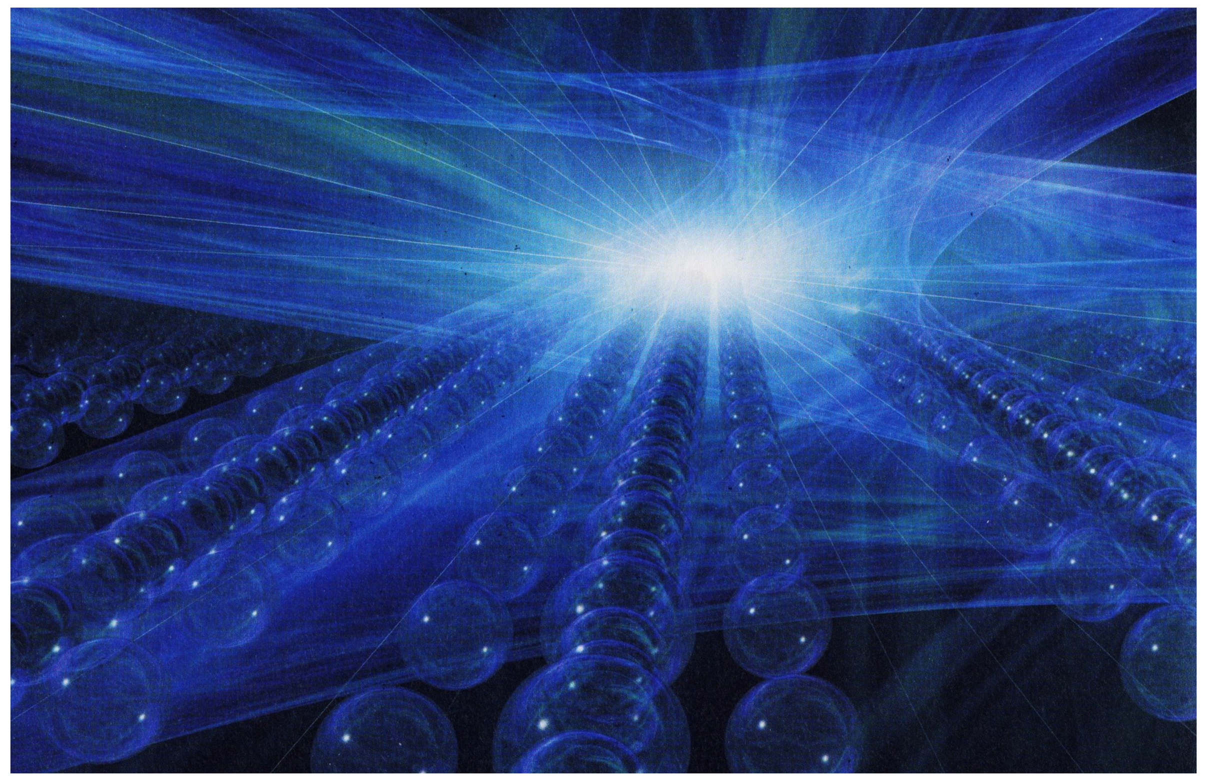

Figure 3: © New Scientist, Vol 2544, 25.03.2006.

Just-in-time computing

The qubits on the far left are measured to pass information in the computation, meanwhile mini-clusters of length m are added in order to maintain the cluster at a consistent size. N is the "buffer size" which, if depleted, will cause the computation to crash. ΔN is the variance of the buffer zone and M is the number of logical qubits used in the computation.

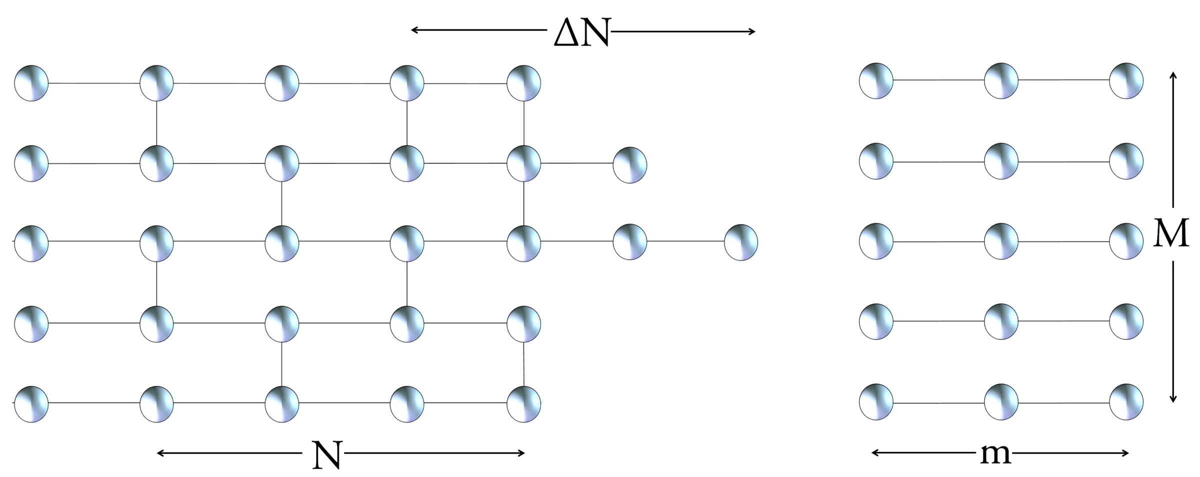

Figure 4: *Just-in-time* computing[5]

Creating a minicluster of size m takes, on average, a time τ, which is dependent on additional factors such as the total pool of qubits. This pool is finite and unsuccessful entanglements will require qubits to be reinitialised. The lifetime of the qubit at the time of measurement is thus

$$\tau_{average} = \tau + (\langle N\rangle + m)\Delta t. \tag{1}$$

2 Dimensional Cluster States

Entangling Procedure	c_1	c_2
Type-I fusion	2	2
Type-II fusion	3	1
Double-heralding fusion	1	1
Repeat-until-success	1	0
Broker-client model	0	0

Table 1: The entangling procedures[6] and there respective values of c_1 and c_2

There are several entangling gates, all creating different entanglement links in the cluster states. A failed entanglement will require us to re-purify the cluster by measuring a subset of qubits, whereas a successful entanglement will either create a *cherry* (a qubit which attaches to a single qubit on the cluster chain), or add a minicluster to the cluster. c_1 is the number of qubits that do not add to the buffer in a successful entanglement, and c_2 is the number of qubits that are removed from the cluster in the case of an unsuccessful entanglement. Thus the *rate of growth* is

$$R = p(m - c_1) - (1-p)c_2 - 1, \tag{2}$$

where p is the probability of a successful entanglement. For zero growth we require $R = 0$.

$\tau_{average}$ must be much less than T_2, the age of the leftmost qubit, so

$$T_2 = \alpha(\tau + (\langle N\rangle + m)\Delta t), \tag{3}$$

where α is the *fault tolerance factor*, with $\alpha \geq 1$ and determined by the fault tolerance threshold of a given quantum computer. The value of R is probabilistic, thus the buffer must be sufficiently long as not to disappear in a run of unsuccessful entanglements. The *buffer factor* β is given by

$$\langle N\rangle = \beta\Delta N. \tag{4}$$

Fluctuation size can be calculated using a random walk argument;

$$\langle \Delta N^2\rangle = \frac{(1+c_2)^2}{p} - c_2(2-c_2). \tag{5}$$

Buffer factor β is dependent on the number of logical qubits and the logarithm of the computation time. Thus

$$\begin{aligned} T_2 &= \alpha(\tau + \beta\Delta N + m) \\ &= \alpha(\tau + \beta\sqrt{\frac{4-p}{p} + \frac{2}{p}}) \end{aligned} \tag{6}$$

for Double-Heralding gates.

Conclusions

Figure 5 shows, for a conservative estimate of τ, the threshold of T_2 for the broker-client model (lower curve) and double heralding (upper curve) entanglement methods. Shaded areas define where qubit decay forbids quantum computation utilising cluster states created via entangling gates with p probability of success. No lower limit of success probability p exists.

This model assumes qubits are cheap, to the extent that it is simple to create a large number of qubits, and that arbitrarily entangling two qubits can be achieved quickly. The optimal entangling gate and procedure is something that the project has yet to conclude.

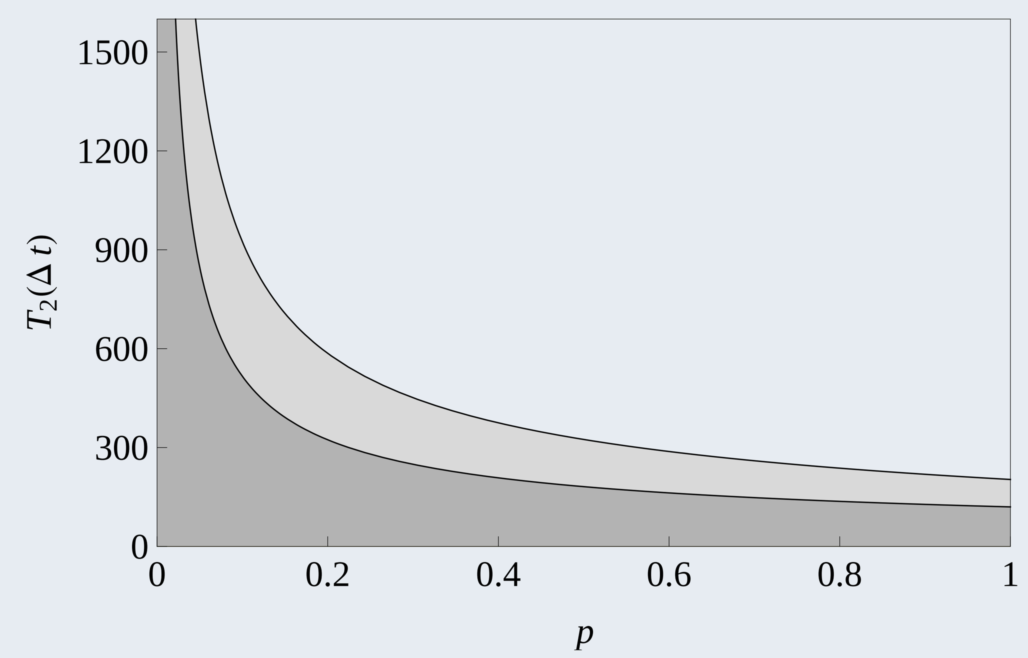

Figure 5: T_2 v.s. p for $\alpha = \beta = 10$ and $\tau = \frac{1}{p}$

Continuing the Project

Further simulations are being conducted to verify this result with varying qubit pools and entanglement strategies for mini-cluster production as well as the creation on the higher dimensional clusters. We will also compare current experimental values of T_2 times to our results.

References

1. R. Raussendorf and H.J. Briegel, Phys. Rev. Lett. 86, 5188 (2001).
2. S.D Barrett and Pieter Kok, Phys. Rev. A 71, R060310 (2005).
3. Pieter Kok, arXiv:0705.4193 (2009).
4. R. Raussendorf and H.J. Briegel, Phys. Rev. A 68, 22312 (2003).
5. Yuan Liang Lim, Sean D. Barrett, Almut Beige, Pieter Kok, Leong Chuan Kwek, Phys. Rev. A 73, 012304 (2006), arXiv:quant-ph/0508218
6. Yuan Liang Lim, Sean D. Barrett, Almut Beige, Pieter Kok, Phys. Rev. Lett. 95,30505 (2005)