Quantum Information Science Notes

Andy Lutomirski

Tuesday, March 4, 2008

Contents

1 Alternative QC models

The previous circuit model of QC involved a sequence of unitary gates. Classical computation models include:

1. Circuits (i.e. AND, OR, NOT) – not reversible
2. Turing machines
3. Memory-based computers (close to what we actually use)
4. Wolfram's cellular automata
5. Strange things such as Conway's fraction-based machine.

These are mostly equivalent up to polynomial factors.

In QC, there are bunch of such things as well.

1. Quantum Turing machines (QTM)
2. Quantum RAM (QRAM) – iffy to implement
3. Adiabatic QC
4. Approximating Jones polynomials at 5th root of unity (i.e. $p\left(e^{\frac{2i\pi}{5}}\right)$) (BCQ-complete, but the polynomial order is huge).
5. Topological QFT (TQFT) – computing with anyons (to be discussed in the next two lectures)
6. Probably more

BQP is the class of problems solvable on a QC with bounded error in polynomial time. Scott Aaronson will teach more about that later.

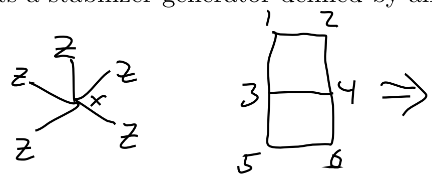
1.1 Measurement-based computation

Otherwise known as one-way QC or cluster state QC.

We start with a bunch of qubits and we measure them one at a time in bases determined by program and/or previous measurements. (Question: how powerful is the classical computer? It turns out that if you're willing to restrict yourself to Clifford states you get a Clifford QC and can do it entirely in parallel. But how much post-processing is needed and are the classical problems parallelizable?)

1.1.1 Cluster states

A cluster state is a stabilizer state defined on any graph G. There is qubit on each vertex and each vertex gets a stabilizer generator defined by an X on that vertex and a Z on adjacent vertices.

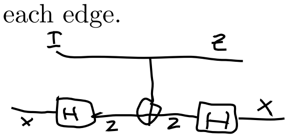


```
1: XZZI
2: ZXIX
3: ZIXZ
4: IZZX
5: IIZI
6: IIIZ
```

They commute because for two non-adjacent vertices, there can't be any overlap except for two X's in the same place, and, for two adjacent vertices, there are two XZ pairs which contribute $(-1)^2$. They are independent because each has an X in one unique spot.

To generate these states, we measure all the generators of the stabilizer group. Then apply Z to each vertex with the wrong sign. The single-vertex Z operators commute with the rest of the vertices, so all of the generator operators are forced simultaneously into the $+1$ eigenstate.

Alternatively, start in the state $|+\rangle^{\otimes|G|}$ (eigenstate of X) and apply $\begin{pmatrix} 1 & & & \\ & 1 & & \\ & & 1 & \\ & & & -1 \end{pmatrix}$ (controlled-Z) to each edge.



This Clifford group circuit works because $|0?\rangle \to |0?\rangle$ trivially, $|10\rangle \to |1\rangle(|0\rangle + |1\rangle) \to |10\rangle$, and $|11\rangle$ is the same except that the CNOT generates a sign change. Applied to a stabilizer $IIIIXI$ (with the X at a vertex), it turns the adjacent vertices into Z's. These all compute because they are all diagonal.

Yet another way is based on

$$\begin{pmatrix} 1 & & & \\ & 1 & & \\ & & 1 & \\ & & & -1 \end{pmatrix} = \exp\left[i\pi \begin{pmatrix} 0 & & & \\ & 0 & & \\ & & 0 & \\ & & & 1 \end{pmatrix}\right]$$

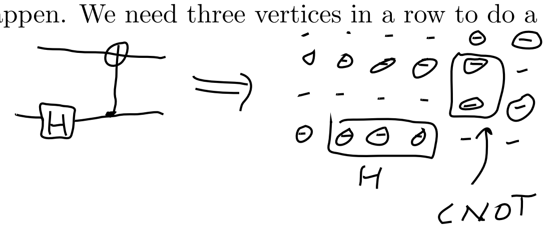
and using the appropriate Hamiltonian to apply these operators.

If we measure a cluster state vertex in the Z basis, then we anticommute with exactly one stabilizer generator (the one belonging to that vertex) and we end up with a cluster state for the graph minus that vertex.

1.1.2 Computing with cluster states

We assume that all qubits start in $|0\rangle$.

The frist step is to lay out "qubit paths" such that the paths only touch where we want operations to happen. We need three vertices in a row to do a one-qubit gate.



(We're drawing these on lattices, but that's not a requirement. It just happens to be handy because we might be able to build these on optical lattices.)

The next step is to measure all the qubits outside the logical path in the Z basis. All that's left now is a cluster state on the qubit paths.

If we were to start with the left side of a wire in $|\psi\rangle$ and the rest of the wire vertices in $|+\rangle$ (this is *not* a cluster state) and apply CZ to each edge (in any order), then we end up with what we call the "logical state $|\psi\rangle$" or $|\psi_L\rangle$. (This is not how we'll actually generate $|\psi_L\rangle$.

To "move the logical state" means to move the position of the $|\psi\rangle$ (prior to the CZ's) one to the right and the remove the previous position from the cluster by measuring it. We measure the vertex with the $|\psi\rangle$ in the X basis. This only fails to commute with the single closest CZ, so we can consider only the two-qubit case. Algebraically:

$$\begin{aligned} (\alpha|0\rangle + \beta|1\rangle)(|0\rangle + |1\rangle) \quad & \underset{CZ}{\rightarrow} \quad (\alpha|00\rangle + \beta|10\rangle + \alpha|01\rangle - \beta|11\rangle) \\ & \underset{\text{measure } X_1}{\rightarrow} \quad \begin{cases} (\alpha+\beta)\,|0\rangle + (\alpha-\beta)\,|1\rangle, & \text{if } 0 \\ (\alpha-\beta)\,|0\rangle + (\alpha+\beta)\,|1\rangle, & \text{if } 1 \end{cases} \end{aligned}$$

Now, if you get a + outcome, then you've applied H. If you get a - outcome, then you've applied XH. We can't undo the X, but we can keep track of it and, in the future, apply GX instead of G. We end up accumulating a Pauli error, but we can keep track of it.