Overview
Four Talks

- A Rosetta Stone for Quantum Computation
- Quantum Algorithms & Beyond
- Distributed Quantum Computing
- Topological quantum Computing and the Jones Polynomial
- A Quantum Computing Knot Theoretic Mystery -- Can be found on my webpage.

This work is supported by:

- The National Institute for Standards and Technology (NIST)
- The Mathematical Sciences Research Institute (MSRI).
- L-O-O-P Fund.
- The Institute of Scientific Interchange

First two papers Joint Work with


Lomonaco, Distributed quantum computing and decoherence, in preparation.

Dr. Anocha Yimsiriwattana
Distributed Quantum Computing

Why ???

Predicting the Future ???

- The first usable quantum computing devices will probably have small quantum memories -- squidgets
- Quantum repeater and quantum teleportation technology will be available, making it technically feasible to interconnect squidgets with EPR channels.
- Researchers will find that a network of EPR connected squidgets can more easily be protected from the effects of decoherence.

Ergo, the first scalable quantum computers will be distributive quantum systems that run (of course) distributed quantum algorithms.

Computational Penalty for Distributed Quantum Computing

Conversion using YL1 & YL2
### Computational Penalty

- A = a non-distributed Quantum Algorithm
- D(A) = a distributed Quantum Algorithm constructed from A using YL1 & YL2

\[ TC = \text{Time-Complexity, i.e., } \#\text{Steps executed} \]
\[ SC = \text{Space-Complexity, i.e., } \#\text{qubits} \]

### Conclusion

- For quantum algorithms exponentially faster than their corresponding classical algorithms (i.e., Shor's algorithm), the penalty is negligible.
- For quantum algorithms quadratically faster than their corresponding classical algorithms (i.e., Grover's algorithm), the penalty may or may not be too much. (?)

### Distributed Quantum Computing

**Why will a distributed quantum computer be more resistant to the effects of decoherence?**

It will provide a Divide & Conquer approach to the problem of decoherence.

### Decoherence

- A quantum system \( Q_{\text{sys}} \) simply does not want to be isolated, but instead wants to entangle with its environment \( Q_{\text{env}} \) (as well as with itself)
- The more a \( Q_{\text{sys}} \) entangles with its environment (and itself), the more (to one observing ONLY the QSys) does it appear to become
  - Noisy & classically random (i.e., loses coherence)
  - And hence, uncontrollable.
- By this process, \( Q_{\text{env}} \)-qubits appear to the observer to be degenerating into random classical bits

We call this phenomenon **Decoherence**
It will provide a Divide & Conquer approach to the problem of decoherence.

Why is a distributed quantum computer more resistant to the effects of decoherence?

A Mathematical Model for Decoherence

Let

\[ H_{\text{Sys}} \] = State space of \( Q_{\text{Sys}} \)
\[ H_{\text{Env}} \] = State space of \( Q_{\text{Env}} \)
\[ H_{\text{World}} = H_{\text{Sys}} \otimes H_{\text{Env}} \] = State space of the "world"

Finally, let

\[ U_{\text{World}} \] = Group of all unitary transformations on \( H_{\text{World}} \)

A Mathematical Model for Decoherence

For each \( j = 0, 1, 2, \ldots \), we think of decoherence as the application of a random element \( U \) of \( U_{\text{World}} \) to the state of the "world" at time \( t = j \Delta t \)

This is an unwanted event that is essentially out of our control.

But what are the characteristics of this probability distribution?

We postulate the following "rules of thumb":

Postulated "Rules of Thumb"

The principle of proximity (POP).

The greater the physical separation (distance) between system components \( Q_{\text{Sys1}} \) and \( Q_{\text{Sys2}} \), the less likely is it that a random \( U \) in \( U_{\text{World}} \) will affect both \( Q_{\text{Sys1}} \) and \( Q_{\text{Sys2}} \).

In other words,

The probability that a random \( U \) in \( U_{\text{World}} \) affects the two system components \( Q_{\text{Sys1}} \) and \( Q_{\text{Sys2}} \), decreases as the distance between these components increases.
Postulated "Rules of Thumb"

The principal of enchlophobia (POE).

Let $P_n$ be the probability that a random $U$ in $U_{n\text{sys}}$ will effect exactly $n$ system qubits. Then

$$P_n > P_{n+1}$$

A Divide & Conquer approach to decoherence

To overcome decoherence, we will distribute $Q_{\text{Sys}}$ into a quantum network made up of its three component subsystems $Q_{\text{Sys}_1}$, $Q_{\text{Sys}_2}$, and $Q_{\text{Sys}_3}$.

Section 2

Distributed Quantum Computing

- By a distributed quantum computer, we mean a network of quantum computers interconnected by quantum and classical channels
- The distributed computing paradigm provides an effective way to utilize a number of small quantum computers

Architecture

Each column represents a separate computer

quantum register

channel

qubits

quantum channels

classical channels

Each column represents a separate computer
**Observations**

* The CNOT gate and the set of all one-qubit gates are universal

* If we can implement a non-local CNOT, then a distributed version of any unitary transformation can be implemented

---

**Distributed Quantum Computing Primitives**

**Section 3**

The CNOT gate and the set of all one-qubit gates are universal, and if we can implement a non-local CNOT, then a distributed version of any unitary transformation can be implemented.

---

**The Application of Generalized GHZ States and Cat-Like States to Distributed Quantum Computing**

A Generalized GHZ state is a quantum state of the form:

\[ |00\ldots0\rangle + |11\ldots1\rangle / \sqrt{2} \]

A Cat-Like state is a quantum state of the form:

\[ \alpha|00\ldots0\rangle + \beta|11\ldots1\rangle \]

---

**Terminology**

**Key Idea**

Use Entanglement to Distribute Control

A generalized GHZ state can be used to create a “cat-like” state, which can in turn be used to distribute control.

---

**2 Qubit Entangling Gate**

\[ |00\rangle + |11\rangle / \sqrt{2} \]
Once EPR quantum channels are established, the generalized GHZ states and cat-like states can be created with only local operations.

Creating \( n \) qubit GHZ state from \( n-1 \) ebits using only local operations.

We now show how quantum entanglement can be used as a resource for remote control of other computing devices.
Factoring the Quantum Teleportation Circuit into Primitives

A Local Control-U Gate

A Non-Local Control-U Gate

Applying DQC Primitives

Section 4
A Local Control-U Gate

If gates share qubit 4, we can ...

Teleport the qubit to process

Teleport back

Establishing Entanglement

Sending two qubits => two ebits

Refreshing Entanglement

Non-local CNOT Gate

Quantum Teleportation Circuit

Third Fundamental Primitive for DQC
Quantum Teleportation Circuit

Quantum Distributed Computing Primitive Operations

- Cat-Creator
- Disentangler
- Reset
- Swap-Reset

A universal set of DQC primitives

These are the basic building blocks of distributed quantum algorithms

Fourth Fundamental Primitive for DQC

Section 5

A Distributed Quantum Fourier Transform

Quantum Fourier Transform

\[ \hat{U} = \frac{1}{\sqrt{2}} \begin{pmatrix} 1 & 1 \\ 1 & -1 \end{pmatrix} \]

\[ \begin{pmatrix} 1 \\ 0 \\ e^{2 \pi i/4} \end{pmatrix} \]

Distributed Quantum Fourier Transformation
Shor’s Quantum Factoring Algorithm reduces to the task of finding the order of the map:
\[ \mathbb{Z} \rightarrow \mathbb{Z}_N, \quad n \mapsto a^n \text{ mod } N \]

Order Finding Algorithm

1. Given \( N \) and \( a \in \mathbb{Z}_N \), find the order of \( a \). Let \( r \) be the order of \( a \).
2. Define \( M_a \) as follow: let \( x \in \mathbb{Z}_N \)
   \[ M_a : |x\rangle \rightarrow |ax \text{ mod } N\rangle \]

Order Finding \& Shor’s Factoring Alg.

1. Repeat to find different \( j \)
2. Use continued fraction algorithm to find the period \( r \)

Quantum Factoring reduces to Order Finding
Quantum Factoring Algorithm

- Quantum Factoring
- Order Finding
- Phase Estimation

A Proposed Experiment

- Implement the Deutsch’s Algorithm as a Distributive Quantum Algorithm

The End