ERATO セミナ 2010 - No. 14
Topological Quantum Computer のための量子回路設計問題

Byung-Soo Choi
梨花女子大学 研究教授

2010/8/20

概要

現在，注目されている「エラーに強い」量子計算のスキームである Topological Quantum Computer について，その動作原理を簡単に説明した後，その設計のために必要な最適化問題に関する定式化を行う．

Overview of Topological Cluster-State Quantum Computation on 2D Cluster-State

based on "***High-threshold universal quantum computation on the surface code***"

-Austin G. Fowler, Ashley M. Stephens, and Peter Groszkowski

-*Physical Review A* **80** 052312 (2009)

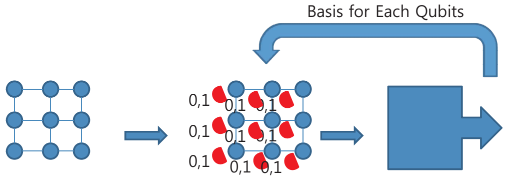
Note: All figures in these slides are reprinted from the above paper

Motivation of TCSQC Model

- **Needs an Extremely Efficient Scheme**
- FOR
  - **quantum error correction** and
  - **fault-tolerant quantum computation**
- WITHOUT
  - **Unphysical Demands on the Underlying Hardware**
  - **Excessive Time Overhead**
  - **Waste a Significant Amount of the Potential Performance Increase**

Model of Topological Cluster State Quantum Computation

Measurement Basis for Each Qubits



Initialization of Cluster State

Measurements of Cluster State

Classical Post-Processing For Computation and Error Correction

Properties of TCSQC

- Physical Constraints
  - 2D square lattice of physical qubits
  - Nearest-Neighbor Coupling
  - Less than 1% error rate for initialization, readout, memory, and gates

  => **the least challenging** set of physical requirements

- Moreover
  - log(n) time for quantum gate with two qubits having distance n, hence shorter time for long distance operation
  - Asymmetric error correction is possible
  - Dynamic error correction is possible

Several Experiments

- Superconductors
  - *EPL 85 50007 (2009)*
  - *Arxiv 0905.4839*
- Semiconductor nanophotonics
  - *IJQI* 8 295 (2010)

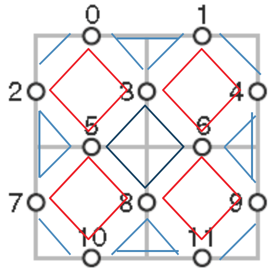
Stabilizer Formalism

- Express the quantum state as the unique simultaneous +1 eigenvector of the **commuting operators => Stabilizer**
  - **NOT State Vector**
  - **BUT a set of Commuting Operators**
- Any set of n **mutually commuting** and **independent operators** over n qubits has a **unique simultaneous** +1 eigenstate
- In TCSQC model, **stabilizers only with I, X, Z, Y matrices** are used
  - I=[(1,0),(0,1)], X=[(0,1),(1,0)], Z=[(1,0),(0,-1)], Y=[(0,-i),(i,0)]
- Unitary Operators and Measurements are formulated by Stabilizers

Measurements

- Mz measures the quantum state by |0> and |1> basis
  - Stabilizer form: [(1,0),(0,-1)]
- Mx measures the quantum state by |+> and |-> basis
  - Stabilizer form: [(0,1), (1,0)]

Surface Code Error Correction

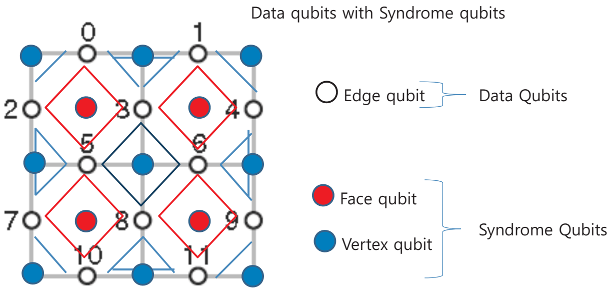


X_0X_2, $X_0X_1X_3$, X_1X_4, $X_2X_5X_7$, $X_3X_5X_6X_8$, $X_4X_6X_9$, X_7X_{10}, $X_8X_{10}X_{11}$, X_9X_{11},

$Z_0Z_2Z_3Z_5$, $Z_1Z_3Z_4Z_6$, $Z_5Z_7Z_8Z_{10}$, $Z_6Z_8Z_9Z_{11}$.

Figure 1 Basic layout of surface code data qubits

Surface Code Error Correction



Face qubit is for checking Z stabilizer
Vertex qubit is for checking X stabilizer

Surface Code Error Correction

- If no errors, the surface remains in the simultaneous +1 eigenstate of every stabilizer
- If errors, adjacent stabilizers become negative

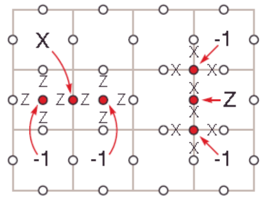


FIG. 2. (Color online) Effect of single bit flips (X) and phase flips (Z) on the surface code—adjacent face and vertex stabilizers are made negative, respectively.

Surface Code Error Correction

- How to measure **Z** and **X** stabilizers

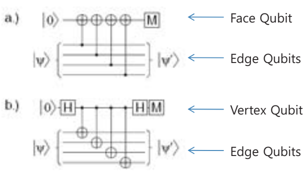


FIG. 23. Circuit showing how an additional syndrome qubit (top line of each figure) is used to measure (a) Z stabilizers and (b) X stabilizers.

J. Math. Phys. 43 4452

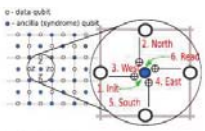


FIG. 24. (Color online) Syndrome measurement typically involves six gates: syndrome initialization, CNOT with the four surrounding data qubits (fewer on boundaries), and finally syndrome readout.

Surface Code Error Correction

- Two additional complications

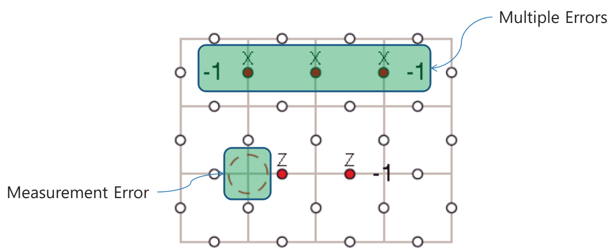


FIG. 3. (Color online) Surface code suffering from multiple errors (indicated by filled dots) and an incorrect syndrome measurement (indicated by dashed circle).

Surface Code Error Correction

- Have to keep track of every time the reported eigenvalue of **each stabilizer** changes (FIG 4a)
- Correction is delayed as long as possible and **pairs of flipped syndromes are then connected** by **paths in space and time or "matched" such that the total number of edges used is minimal** (FIG 4b)
  - Polynomial and Logarithmic Complexity Matching Algorithms have been designed
  - No Problem !!

Example for Z stabilizer information

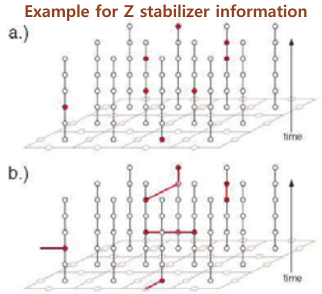


FIG. 4. (Color online) (a) Locations in space and time, indicated by filled dots, where and when the reported syndrome is different from that in the previous time step. Note that this is not a three-dimensional physical structure, just a three-dimensional classical data structure. (b) Optimal matching highly likely to lead to a significant reduction of the number of errors if bit-flips are applied to the spacelike edges.

Logical Qubits

- The Simplest Logical Qubit consists of a Single Face where we stop measuring the associated Z stabilizer
  - No further measuring of Z stabilizer
  - This introduces one new degree of freedom into the surface

Logical Qubits

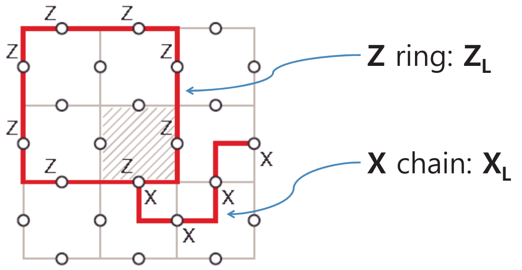
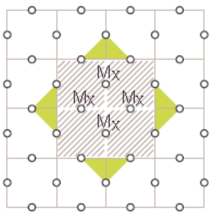


FIG. 6. (Color online) Surface code with one additional degree of freedom introduced by not enforcing the stabilizer associated with one face (shaded). This face or a region of such faces is called a smooth defect. The degree of freedom can be phase flipped by any ring of Z operators encircling the defect and bit flipped by any chain of X operators connecting the defect to a smooth boundary.

Logical Qubits

- Larger Smooth Defect



The qubit inside the defect play no further role in the computation unless the defect moves

FIG. 7. (Color online) Surface code with one degree of freedom introduced via the measurement M_X of four qubits in the X basis and removal of five stabilizers. Note that four new three term X stabilizers are created with not necessarily positive sign (indicated by shaded triangles).

Logical Qubits

- It is inconvenient to use a logical qubit with a logical operator that connects to a potentially distant boundary
- To avoid this problem, using a pair of defects to represent a single logical qubit

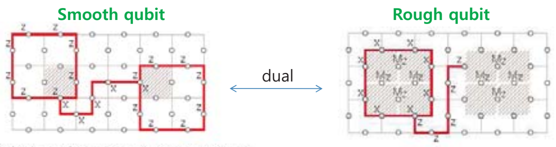


FIG. 8. (Color online) Smooth qubit comprised of two smooth defects. Z_L corresponds to any ring of Z operators around either defect. X_L corresponds to any chain of X operators connecting the two defects.

FIG. 10. (Color online) Initializing a rough qubit in the $|+_L\rangle$ state via Z basis measurements M_Z and ignoring stabilizers (shaded). X_L is any ring of X operators around either defect. Z_L is any chain of Z operators linking the two defects.

Logical Qubits

- Logical Measurement

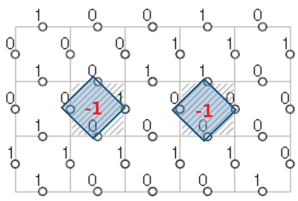


FIG. 11. Example of measurement of a smooth qubit in the Z_L basis in the absence of errors. Note that the measurements around every face have even parity whereas the parity of any path of measurements encircling either defect is odd. The figure thus corresponds to the measurement result $|1_L\rangle$.

Logical CNOT

- The effect of moving a smooth defect

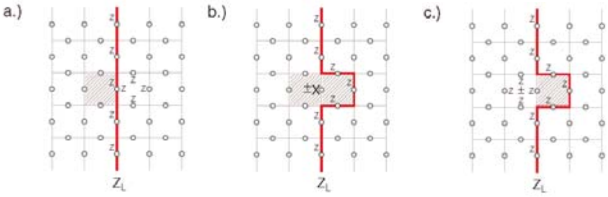


FIG. 12. (Color online) (a) Smooth defect and surface in the +1 eigenstate of Z_L. (b) After measuring the center qubit in the X basis, the shape of the Z_L operator is deformed. (c) Measuring and possibly correcting the indicated Z stabilizer using a bit flip on the center qubit completes the movement of the defect.

Deforms the **shape of Z_L stabilizers** passing nearby

Logical CNOT

- The effect of moving a smooth defect

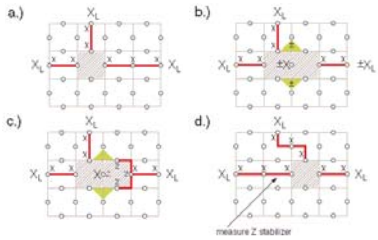


FIG. 13. (Color online) (a) Smooth defect and surface in the +1 eigenstate of X_L. (b) After measuring the center qubit in the X basis, it is possible that three term X stabilizers and X_L stabilizers with negative sign are created (potential locations indicated by shaded triangles). (c) All signs can be corrected by applying the appropriate single-qubit Z operators and chains of Z operators. (d) Measuring and possibly correcting the indicated Z stabilizer using a bit flip on the center qubit completes the movement of the defect.

Drags around **X_L stabilizers** attached to it

Logical CNOT

- Needs a larger defect

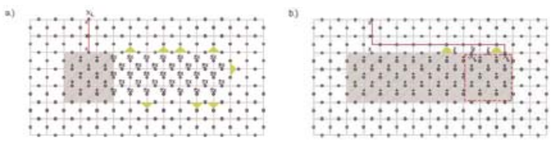


FIG. 14. (Color online) (a) Movement of a large smooth defect via many measurements in the X basis. Many pairs of three term X stabilizers with negative sign are likely to be created (indicated by shaded triangles). (b) After several rounds of error correction, it becomes exponentially unlikely that three term X stabilizers with negative sign remain that were generated in the measurement round. New chains of errors on the boundary can occur, but these will be corrected during normal error correction after the size of the defect is reduced to complete the movement.

Logical CNOT

- This defect movement procedure
  - deforms nearby Z_L stabilizers,
  - drags around X_L stabilizers attached to the defect, and
  - takes a total time that grows only logarithmically in the distance the defect is moved

Logical CNOT

- How to build a logical CNOT?
- These relationships can be combined to determine the action of CNOT on an arbitrary two-qubit stabilizer

$$\Lambda_{12}(I \otimes X)\Lambda_{12}^{\dagger} = I \otimes X, \tag{6}$$

$$\Lambda_{12}(X \otimes I)\Lambda_{12}^{\dagger} = X \otimes X, \tag{7}$$

$$\Lambda_{12}(I \otimes Z)\Lambda_{12}^{\dagger} = Z \otimes Z, \tag{8}$$

$$\Lambda_{12}(Z \otimes I)\Lambda_{12}^{\dagger} = Z \otimes I. \tag{9}$$

Logical CNOT

- CNOT for different type qubits such as (smooth, rough) or (rough, smooth) is easy by using Figures 15 and 16

Logical CNOT

For Eq (7)

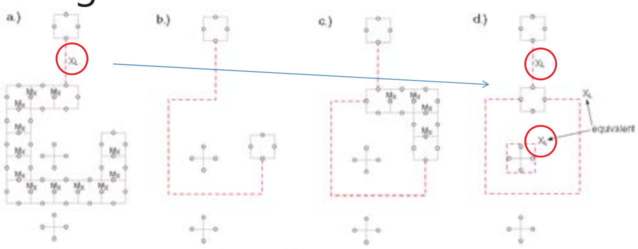


FIG. 15. (Color online) (a) Surface containing a smooth qubit in the +1 eigenstate of X_L and a rough qubit. The lower smooth defect has been braided around the upper rough defect using X measurements. Note that is not possible to complete the braiding in one step as a ring of X measurements corresponds to measurement of the rough qubit in the X_L basis. (b) Via correction of many Z stabilizers, the X_L operator is dragged around the upper rough defect. (c) Additional X measurements extend the defect back to its original position. (d) Further correction of Z stabilizers returns the defects to their original positions but the surface is now in the +1 eigenstate of both the smooth and rough X_L operators.

Logical CNOT

For Eq (8)

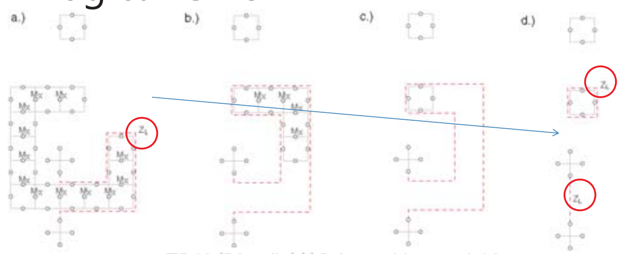
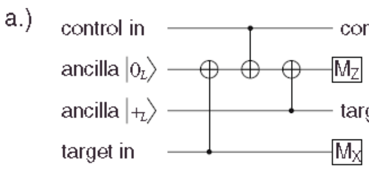


FIG. 16. (Color online) (a) Surface containing a smooth defect and a rough defect in the +1 eigenstate of Z_L. The lower smooth defect has been braided around the upper rough defect using X measurements, deforming the shape of the rough Z_L operator. (b) By first correcting many Z stabilizers and then performing further X measurements, the smooth defect can be extended back to its original position. (c) A final round of Z stabilizer correction returns the defects to their original configuration but with the state of the surface changed. (d) The Z_L operator shown in part (c) is equivalent to the tensor product of smooth and rough Z_L.

Logical CNOT

- Question: A CNOT between logical qubits of the same type



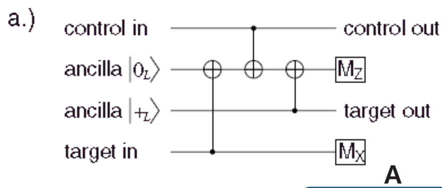
Smooth Qubit
Rough Qubit
Smooth Qubit
Smooth Qubit

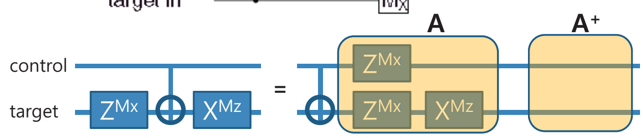
FIG. 17. (Color online) Smooth qubits are represented by black lines and rough qubits by lighter lines. (a) Circuit equivalent to Z^{M_X} on the target qubit followed by CNOT between the control and target qubit followed by X^{M_Z} on the target qubit. (b) Schematic representing the initialization, braiding, and measurement of defects in a surface code to implement (a). Time runs from left to right, and the surface code should be imagined oriented vertically and into and out of the page. (c) Simplified schematic equivalent to (b).

ALL CNOTs are for different type of qubits. Figures 15 and 16 can be used

Logical CNOT

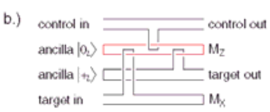
- Question: A CNOT between logical qubits of the same type



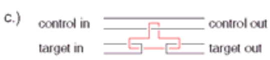


Logical CNOT

- Question: A CNOT between logical qubits of the same type



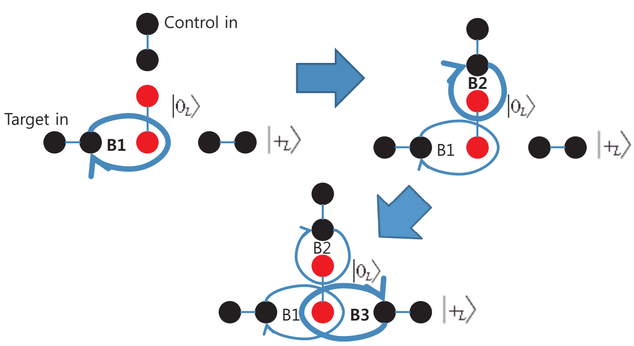
b) Shows the braiding patterns, where red line shows the rough qubit and block line shows the smooth qubit



c) Shows the simplified braiding pattern

Logical CNOT

- Braiding pattern



Universal Set of Logical Gates

- State Injection
- State Distillation
- Non-Clifford Gates

Creation of an Arbitrary Rough Qubit

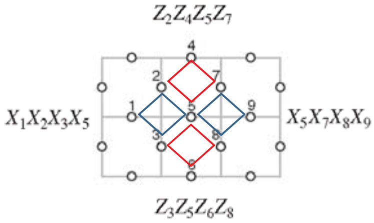
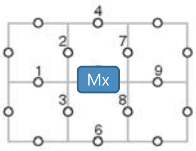


Figure 18. Surface code fragment

Creation of an Arbitrary Rough Qubit

- Measure Qubit 5 in the X basis



$$\begin{array}{c|ccccccccc} & 1 & 2 & 3 & 4 & 5 & 6 & 7 & 8 & 9 \\ \hline & X & X & X & & X & & & & \\ & & & & & X & & X & X & X \\ (-1)^{M_X} & & & & & X & & & & \\ & & Z & Z & Z & & Z & Z & Z & \end{array}$$

- If -1 eigenstate is obtained, apply either $Z_2Z_4Z_5Z_7$ or $Z_3Z_5Z_6Z_8$ to create the +1 eigenstate

$$\begin{array}{ccccccccc} 1 & 2 & 3 & 4 & 5 & 6 & 7 & 8 & 9 \\ \hline X & X & X & & & & & & \\ & & & & & & X & X & X \\ & & & & X & & & & \\ & Z & Z & Z & & Z & Z & Z & \end{array}$$

Creation of an Arbitrary Rough Qubit

- Hadamard transform and then unitarily rotate qubit 5 to the desired state

$$\alpha\left(\begin{array}{ccccccccc} 1 & 2 & 3 & 4 & 5 & 6 & 7 & 8 & 9 \\ \hline X & X & X & & & & & & \\ & & & & & & X & X & X \\ & & & & Z & & & & \\ & Z & Z & Z & & Z & Z & Z & \end{array}\right) + \beta\left(\begin{array}{ccccccccc} 1 & 2 & 3 & 4 & 5 & 6 & 7 & 8 & 9 \\ \hline X & X & X & & & & & & \\ & & & & & & X & X & X \\ & & & & -Z & & & & \\ & Z & Z & Z & & Z & Z & Z & \end{array}\right)$$

- Measure either $Z_2Z_4Z_5Z_7$ or $Z_3Z_5Z_6Z_8$

$$\alpha\left(\begin{array}{c|ccccccccc} & 1 & 2 & 3 & 4 & 5 & 6 & 7 & 8 & 9 \\ \hline & X & X & X & & & & X & X & X \\ & & & & & Z & & & & \\ (-1)^{M_Z} & & Z & & Z & Z & & Z & & \\ (-1)^{M_Z} & & & Z & & Z & Z & & Z & \end{array}\right) + \beta\left(\begin{array}{c|ccccccccc} & 1 & 2 & 3 & 4 & 5 & 6 & 7 & 8 & 9 \\ \hline & X & X & X & & & & X & X & X \\ & & & & & -Z & & & & \\ (-1)^{M_Z} & & Z & & Z & Z & & Z & & \\ (-1)^{M_Z} & & & Z & & Z & Z & & Z & \end{array}\right)$$

Creation of an Arbitrary Rough Qubit

- If the -1 eigenstate of $Z_2Z_4Z_5Z_7$ and $Z_3Z_5Z_6Z_8$ is obtained, apply X_5 and then either $X_1X_2X_3X_5$ or $X_5X_7X_8X_9$ to give the desired logical state

$$\alpha\left(\begin{array}{ccccccccc} 1 & 2 & 3 & 4 & 5 & 6 & 7 & 8 & 9 \\ \hline X & X & X & & & & X & X & X \\ & & & & Z & & & & \\ & Z & & Z & Z & & Z & & \\ & & Z & & Z & Z & & Z & \end{array}\right) + \beta\left(\begin{array}{ccccccccc} 1 & 2 & 3 & 4 & 5 & 6 & 7 & 8 & 9 \\ \hline X & X & X & & & & X & X & X \\ & & & & -Z & & & & \\ & Z & & Z & Z & & Z & & \\ & & Z & & Z & Z & & Z & \end{array}\right)$$

State Distillation

- Injection of two particular states

$$|Y\rangle = |0\rangle + i|1\rangle$$

$$|A\rangle = |0\rangle + e^{i\pi/4}|1\rangle$$

State Distillation

$$|Y\rangle = |0\rangle + i|1\rangle$$

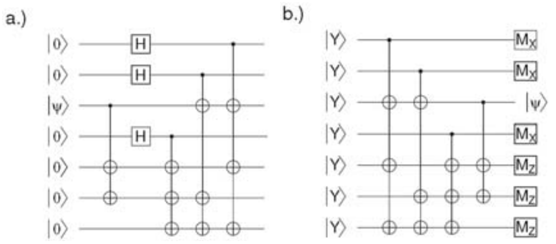


FIG. 19. (a) Encoding circuit for the seven-qubit Steane code. (b) Distillation circuit for the $|Y\rangle = |0\rangle + i|1\rangle$ state.

Repeat step b) multiple times to achieve the desired fidelity

State Distillation

$$|Y\rangle = |0\rangle + i|1\rangle$$

TABLE I. Possible measurement patterns after running the distillation circuit of Fig. 19(b) with perfect $|Y\rangle$ states and no gate errors.

$Pr(M)$	M_X	M_X	M_X	M_Z	M_Z	M_Z	$\lvert\Psi\rangle$
0.125	0	0	0	0	0	0	$Z\lvert Y\rangle$
0.125	0	0	1	1	1	1	$Z\lvert Y\rangle$
0.125	0	1	0	1	0	1	$\lvert Y\rangle$
0.125	0	1	1	0	1	0	$\lvert Y\rangle$
0.125	1	0	0	0	1	1	$\lvert Y\rangle$
0.125	1	0	1	1	0	0	$\lvert Y\rangle$
0.125	1	1	0	1	1	0	$Z\lvert Y\rangle$
0.125	1	1	1	0	0	1	$Z\lvert Y\rangle$

If perfec $|Y\rangle$ states are input, Table I summarizes the possible measurement patterns, their probabilities, and the output state

If less than perfect $|Y\rangle$states are input, other measurement patterns have nonzero probability. If a measurement pattern not listed in Table I is obtained, the distilled state is discarded

State Distillation

$$|A\rangle = |0\rangle + e^{i\pi/4}|1\rangle$$

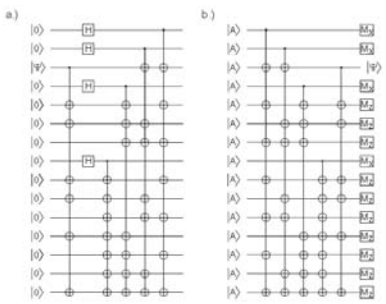


FIG. 20. (a) Encoding circuit for the 15-qubit Reed-Muller code. (b) Distillation circuit for the $|A\rangle = |0\rangle + e^{i\pi/4}|1\rangle$ state.

Non-Clifford gates

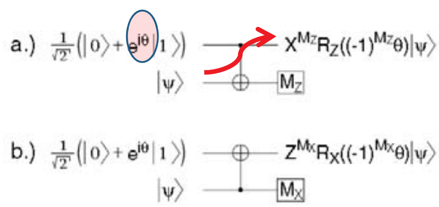


FIG. 21. (a) Circuit performing the single-qubit unitary $X^{M_Z}R_Z[(-1)^{M_Z}\theta]$ given an appropriate ancilla state. (b) Circuit performing the single-qubit unitary $Z^{M_X}R_X[(-1)^{M_X}\theta]$ given an appropriate ancilla state.

An Efficient Hadamard Gate

- One Way $\quad H \equiv R_Z(\pi/4)R_X(\pi/4)R_Z(\pi/4)$
- More Efficient Way
  - JPA:Math. Theor. 42 095302

An Efficient Hadamard Gate

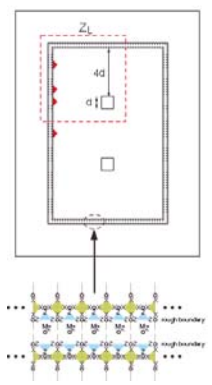


FIG. 22. (Color online) A smooth qubit isolated from a larger piece of surface code using a ring of Z measurements so that logical Hadamard can be applied by local transversal Hadamard gates. Shaded triangles in the upper part of the figure represent three term Z stabilizers of negative sign. See text for details.

Further Works

- Apply this model into other qubit technology
  - Quantum-dot qubit technology
  - In general, 2D like qubit layout technologies

Potential Problems

- The need of huge number of physical qubits and its maintenance would be the big problem
  - For each error correction period, it needs huge number of measurements and its feedback controls
  - In general, since the number of physical qubits is very large, the controlling of the whole physical qubits is very very hard !!!

Optimization of TCSQC Program

Motivation

- TCSQC model is heavily based on the **classical processing**
  - **Generation of measurement sequences** and **the analysis of measurement outputs** for fault tolerance quantum computation
  - **Generation of several sequences of braids** for executing algorithm
    - A braid is implemented by the measurement sequences
  - Generating measurement sequences and analyzing measurement outputs have to be done by classical processing

Program on TCSQC

- A quantum algorithm consists of **logical qubits** and **logical gates**

- In TCSQC Model
  - **A logical qubit** consists of **two defects**
    - **A defect** (or region) consists of **measured physical qubits**
  - **A logical gate** consists of **braids of defects**
    - **A braid** consists of **measurement sequence of defects** and **its neighbors**

- Hence, a **TCSQC program** consists of a **layout of defects (logical qubits) with a sequence of their braids** with some constraints

Constraints

- For error-tolerance, a minimum distance between defects have to be satisfied
  - It depends on the algorithm itself
  - It depends on the physical error rate
    - Usually the distance have to increase with the physical error rate

Constraints

- Overlapping between braids is not allowed
  - When the multiple braids overlap, multiple time slices must be used for each braid
  - If multiple braids have no overlap, they can be done simultaneously

What makes complex

- Most of gates can be directly implementable
  - Each gate has its braiding sequence which can be directly implementable on TCSQC model

- Some quantum gates (ex, Toffoli) have to use ancilla qubit
  - Ancilla qubit needs distillation protocol
  - Some teleportation protocol is necessary to implement the gate

In General

- From the quantum algorithm or circuit, we can derive **a set of defect pairs** and **their braids for logical gates**
- Then the general question is **how to make a layout of the defect pairs with braiding patterns with smaller overhead** (time and space)

Problem

- **Promise**: Given an ordering of sets of braids with qubits as

  T1: (q1,q2), (q3,q4)
  T2: (q3,q4), (q5,q6), (q10,q12)
  T3: (q1,q4), (q2,q6)
  ⋮
  Tm: (qi,qk), (qm,qn)

- **Problem**: Find **a Static Layout** of logical qubits qi (and hence the layout of defect pairs)
  – With smaller time or space

The overall time for Simple Model

- Each time slice consists of several braids
- The **max time for each time slice** infers from the **max distance between qubits** for one of braids
  – Time for braiding is related to the distance between source and target qubits
  – Hence, **the longest distance among qubit pairs** determines the max time for each time slice
- The overall time is the **SUM of max time for each slice**

Generalization 1: Considering Length of Braid

- Max Time for Each Time Slice
  – Each slice consists of several braids
  – Max Time should be the longest braid length, not the longest distance between qubits
    - Since the detour and the distance of braids has different properties
    - The time for braid is the length of detour, not the distance
  – Hence, each time slice must have a braid geometry which shows the min_max braid length
    - We have to find detours for all braids which shows the smallest of the longest detour distance

Generalization 2: Dynamic Layout

- We assumed to have a static layout of qubits at the initial time
- BUT, we can dynamically change the layout of qubits for each time slice
- Problem:
  – Have to consider SWAP operaions

Generalization 3: Put it all together

- By applying dynamic layout of qubits for each time slice, and optimizing the geometry of braids, the overall time can be globally minimized

Applying BDD

- Candidate Approach
  - Consider the simple case as 1D layout of qubits
  - How to reduce the computation time for this model by using BDD