Measurement based quantum computation

Yuan Zhan
Department of Physics, University of Colorado Boulder
yuan.zhan@colorado.edu
Mar 21, 2020; revised April 6, 2020

1 Gate array model

In order to better understand the principle of measurement-based quantum computation (MBQC), let's review the building blocks in gate array model first, which is the basic framework for many quantum computation researches nowadays, say, trapped ion QC, superconducting QC, etc.. Gate array model is sort of the direct quantum generalization of the classical computer structure, and is Boolean logic based. It involves unitary operations as computational steps to process information, and only at the end to make measurements and read out the results by converting quantum information into classical one [1].

The basic idea of the gate array model is that universal quantum computation can be performed using some basic single-qubit and two-qubit gates. Here are several commonly used ones. Later we will map the corresponding measurement-based operations to these basic gates in order to prove that it is possible to perform universal quantum computation via a measurement-based way.

1.1 Single- and two-qubit gates

A random single-qubit gate can be decomposed into four Pauli operators, which serve as the complete basis in the "single-qubit-gate space". The Pauli operators are

$$I = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}, \quad X = \begin{bmatrix} 0 & 1 \\ 1 & 0 \end{bmatrix}, \quad Y = \begin{bmatrix} 0 & -i \\ i & 0 \end{bmatrix}, \quad Z = \begin{bmatrix} 1 & 0 \\ 0 & -1 \end{bmatrix}. \tag{1}$$

Commonly encountered two-qubit gates include, for example, the controlled-NOT gate (or CX gate), controlled-phase gate (or CZ gate), Hadamard gate (or H gate). More explicitly, we have

$$CX\left|i\right\rangle\left|j\right\rangle = \left|i\right\rangle\left|i\oplus j\right\rangle,\ i,j=0,1, \text{ and } \oplus \text{ denotes addition modulo 2, i.e. } \begin{cases} CX\left|0\right\rangle\left|0\right\rangle = \left|0\right\rangle\left|0\right\rangle, \\ CX\left|0\right\rangle\left|1\right\rangle = \left|0\right\rangle\left|1\right\rangle, \\ CX\left|1\right\rangle\left|0\right\rangle = \left|1\right\rangle\left|1\right\rangle, \\ CX\left|1\right\rangle\left|1\right\rangle = \left|1\right\rangle\left|0\right\rangle. \end{cases} \tag{2}$$

We call $\left|i\right\rangle$ the control bit. Also,

$$CZ\left|i\right\rangle\left|j\right\rangle = (-1)^{ij}\left|i\right\rangle\left|j\right\rangle,\ i,j=0,1, \text{ i.e. } \begin{cases} CZ\left|0\right\rangle\left|0\right\rangle = \left|0\right\rangle\left|0\right\rangle, \\ CZ\left|0\right\rangle\left|1\right\rangle = \left|0\right\rangle\left|1\right\rangle, \\ CZ\left|1\right\rangle\left|0\right\rangle = \left|1\right\rangle\left|0\right\rangle, \\ CZ\left|1\right\rangle\left|1\right\rangle = -\left|1\right\rangle\left|1\right\rangle. \end{cases} \tag{3}$$

We can see that CZ gate is symmetrical in the two input qubits, and only if both inputs are $\left|1\right\rangle$ will the output two-qubit state pick up a phase shift of π. We can regard both input qubits as the "control" bit, and the phase shift is why we call it "controlled-phase" gate. Besides, the reason that we call the above two gates CX and CZ is that they can also be expressed in the following form. Here, we use 1 to denote the control bit which will not change after the gate, and the latter single-qubit operation applies to the controlled bit subspace, which is denoted by 2.

$$\begin{aligned} CX &= \left|0\right\rangle_1\left\langle 0\right| \otimes \mathbb{1}_2 + \left|1\right\rangle_1\left\langle 1\right| \otimes X_2, \\ CZ &= \left|0\right\rangle_1\left\langle 0\right| \otimes \mathbb{1}_2 + \left|1\right\rangle_1\left\langle 1\right| \otimes Z_2. \end{aligned} \tag{4}$$

Another two-qubit gate we usually use in quantum information processing is Hadamard gate. We first define two states, which are eigenstates of the single-qubit X gate, with eigenvalues 1 and -1 respectively,

$$|\pm\rangle = \frac{1}{\sqrt{2}}(|0\rangle \pm |1\rangle). \tag{5}$$

The Hadamard gate written in matrix form with $|0\rangle$ and $|1\rangle$ as basis reads

$$H = \frac{1}{\sqrt{2}}\begin{bmatrix} 1 & 1 \\ 1 & -1 \end{bmatrix}. \tag{6}$$

Then we can easily verify that

$$\begin{aligned} H|0\rangle &= |+\rangle, \quad H|1\rangle = |-\rangle, \\ H|+\rangle &= |0\rangle, \quad H|-\rangle = |1\rangle. \end{aligned} \tag{7}$$

1.2 Bell basis states

We are also interested in the maximally entangled two-qubits states, which are Bell basis states

$$\begin{aligned} |B_{00}\rangle &= \frac{1}{\sqrt{2}}(|00\rangle + |11\rangle) = I \otimes I\,|B_{00}\rangle, \\ |B_{01}\rangle &= \frac{1}{\sqrt{2}}(|01\rangle + |10\rangle) = X \otimes I\,|B_{00}\rangle, \\ |B_{10}\rangle &= \frac{1}{\sqrt{2}}(|00\rangle - |11\rangle) = Z \otimes I\,|B_{00}\rangle, \\ |B_{11}\rangle &= \frac{1}{\sqrt{2}}(|01\rangle - |10\rangle) = iY \otimes I\,|B_{00}\rangle. \end{aligned} \tag{8}$$

We can verify this easily, for example, for the second formula, we have

$$\begin{aligned} &|00\rangle = |0\rangle \otimes |0\rangle = \begin{bmatrix} 1 \\ 0 \\ 0 \\ 0 \end{bmatrix}, \quad |01\rangle = |0\rangle \otimes |1\rangle = \begin{bmatrix} 0 \\ 1 \\ 0 \\ 0 \end{bmatrix}, \quad |10\rangle = |1\rangle \otimes |0\rangle = \begin{bmatrix} 0 \\ 0 \\ 1 \\ 0 \end{bmatrix}, \quad |11\rangle = |1\rangle \otimes |1\rangle = \begin{bmatrix} 0 \\ 0 \\ 0 \\ 1 \end{bmatrix}, \\ &\text{and } X \otimes I = \begin{bmatrix} 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \\ 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \end{bmatrix}, \\ \Rightarrow \quad &X \otimes I\,|B_{00}\rangle = \frac{1}{\sqrt{2}}\begin{bmatrix} 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \\ 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \end{bmatrix}\begin{bmatrix} 1 \\ 0 \\ 0 \\ 1 \end{bmatrix} = \frac{1}{\sqrt{2}}\begin{bmatrix} 0 \\ 1 \\ 1 \\ 0 \end{bmatrix} = |B_{01}\rangle. \end{aligned} \tag{9}$$

Also, we notice that

$$iY = \begin{bmatrix} 0 & 1 \\ -1 & 0 \end{bmatrix} = ZX, \tag{10}$$

so, the four Bell basis states can be expressed as

$$|B_{ij}\rangle = Z^i X^j \otimes I\,|B_{00}\rangle. \tag{11}$$

Besides, in the context of cluster states, in order to entangle two qubits, people usually first prepare the two qubits in states $|+\rangle \otimes |+\rangle$, then apply a CZ gate to them, and get

$$\begin{aligned}
CZ|+\rangle|+\rangle &= CZ\frac{1}{2}(|0\rangle+|1\rangle)(|0\rangle+|1\rangle) = CZ\frac{1}{2}(|0\rangle|0\rangle+|0\rangle|1\rangle+|1\rangle|0\rangle+|1\rangle|1\rangle)\\
&= \frac{1}{2}(|0\rangle|0\rangle+|0\rangle|1\rangle+|1\rangle|0\rangle-|1\rangle|1\rangle)\\
&= \boxed{\frac{1}{\sqrt{2}}(|0\rangle|+\rangle+|1\rangle|-\rangle)}\\
&= \boxed{\frac{1}{\sqrt{2}}(|+\rangle|0\rangle+|-\rangle|1\rangle)}.
\end{aligned} \tag{12}$$

The fact that the boxed two forms are equal can be understood in the sense that a CZ gate is symmetrical in the two input qubits.

2 Teleportation quantum computation (TQC)

An alternative protocol for universal quantum computation is measurement-based. Gottesman and Chuang [2] proposed an idea of teleporting quantum gates, and Raussendorf and Briegel [3] proposed the idea of "one-way quantum computer" based on cluster states. It is rather surprising, at least at first thought, that quantum computation could be measurement-based, since measurements are irreversibly destructive, and will involve much loss of potential information that a quantum state carries. This protocol is of great interest and has no classical analogues. Besides, this measurement-based way has different computer architecture and offers different strategies for loss tolerance, which will be discussed in latter sections.

2.1 Single-qubit gates

Let's first consider the teleportation quantum computation (TQC). The standard quantum teleportation is shown in Fig 1.

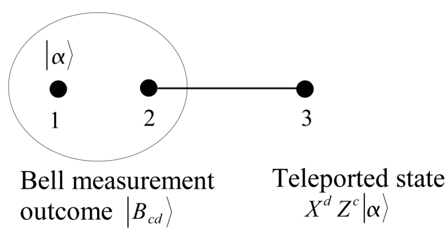


Figure 1: The standard quantum teleportation.

Our goal is to teleport the unknown state of qubit 1 to qubit 3. The game is that initially, qubit 1 is in state $|\alpha\rangle = a|0\rangle + b|1\rangle$, and qubits 2 and 3 are in a Bell basis state $|B_{00}\rangle_{23}$ (other two-qubit maximally entangled states would also work). Then we perform a Bell measurement on qubits 1 and 2, and the measurement result is $|B_{cd}\rangle_{12}$, then qubit 3 will be in state $X^d Z^c|\alpha\rangle$. This can be easily verified from our definition above. We have

$$\begin{aligned}
|00\rangle &= \frac{1}{\sqrt{2}}(|B_{00}\rangle+|B_{10}\rangle), \quad |11\rangle = \frac{1}{\sqrt{2}}(|B_{00}\rangle-|B_{10}\rangle),\\
|01\rangle &= \frac{1}{\sqrt{2}}(|B_{01}\rangle+|B_{11}\rangle), \quad |10\rangle = \frac{1}{\sqrt{2}}(|B_{01}\rangle-|B_{11}\rangle).
\end{aligned} \tag{13}$$

So for $|\alpha\rangle = a|0\rangle + b|1\rangle$, then the three qubit state can be expressed as

$$\begin{aligned}
|\psi\rangle &= \frac{1}{\sqrt{2}}(a|000\rangle + a|011\rangle + b|100\rangle + b|111\rangle)_{123}\\
&= \frac{1}{2}[a(|B_{00}\rangle + |B_{10}\rangle)_{12}|0\rangle_3 + b(|B_{01}\rangle - |B_{11}\rangle)_{12}|0\rangle_3 + a(|B_{01}\rangle + |B_{11}\rangle)_{12}|1\rangle_3 + b(|B_{00}\rangle - |B_{10}\rangle)_{12}|1\rangle_3]\\
&= \frac{1}{2}[|B_{00}\rangle_{12}(a|0\rangle + b|1\rangle)_3 + |B_{01}\rangle_{12}(b|0\rangle + a|1\rangle)_3 + |B_{10}\rangle_{12}(a|0\rangle - b|1\rangle)_3 + |B_{11}\rangle_{12}(-b|0\rangle + a|1\rangle)_3]\\
&= \frac{1}{2}[|B_{00}\rangle_{12}|\alpha\rangle_3 + |B_{01}\rangle_{12}(X|\alpha\rangle)_3 + |B_{10}\rangle_{12}(Z|\alpha\rangle)_3 + |B_{11}\rangle_{12}(XZ|\alpha\rangle)_3]\\
&= \frac{1}{2}|B_{cd}\rangle_{12}(X^d Z^c|\alpha\rangle)_3.
\end{aligned} \tag{14}$$

We note that if we define the so-called two-qubit "maximally entangled" state, as they called it in [1], to be $|B_{00}\rangle$, then the projection of the three-qubit state onto this maximally entangled state for qubits 1 and 2 would be $\frac{1}{2}|\alpha\rangle_3$, as shown in Eq. 14 by choosing $c = d = 0$. We can define a rotated maximally entangled state in 2-dimension as

$$|B(U)_{00}\rangle = U^\dagger \otimes I\,|B_{00}\rangle\,, \tag{15}$$

then other three Bell basis states, $|B_{cd}\rangle$, can be regarded as the rotated maximally entangled state by choosing $U^\dagger = Z^c X^d$, as we can see from Eq. 11. Then what we can tell from Eq. 14 is that the projection onto $|B(U)_{00}\rangle$ state results in the state of qubit 3 to be $\frac{1}{2}U|\alpha\rangle$. This statement also works for other three rotated Bell basis states, i.e. by defining $|B(U)_{cd}\rangle = U^\dagger \otimes I\,|B_{cd}\rangle$, we get the result of qubit 3 state projected onto rotated Bell basis state $|B(U)_{cd}\rangle$ to be $X^d Z^c U|\alpha\rangle$. So if we do the Bell measurement of qubits 1 and 2 under rotated Bell basis, $\mathcal{B}(U) = \{|B(U)_{cd}\rangle\}$, then we will get the teleported state for qubit 3 to be $X^d Z^c U|\alpha\rangle$.

The example above is given for a 2-dimension case, since our basis is $\{|0\rangle, |1\rangle\}$. More generally, for any dimension of d, we can prove the similar lemma. If we define the d-dimensional two-node maximally entangled state to be

$$|\phi\rangle = \frac{1}{\sqrt{d}}\sum_{i=0}^{d-1}|i\rangle|i\rangle\,, \tag{16}$$

which is just $|B\rangle_{00}$ for $d = 2$, then the projection of three-node state $|\alpha\rangle_1|\phi\rangle_{23}$ onto $|\phi\rangle_{12}$ results in state $\frac{1}{d}|\alpha\rangle_3$ at node 3, as shown in Fig. 2. The proof is as follows. We can decompose the initial state of node 1 into the d-dimention basis, $|\alpha\rangle = \sum_{j=0}^{d-1} a_j|j\rangle$, then the projection onto the maximally entangled state of nodes 1 and 2 is

$${}_{21}\left(\frac{1}{\sqrt{d}}\sum_i \langle i|\langle i|\right)\left(\sum_j a_j|j\rangle_1 \sum_k \frac{1}{\sqrt{d}}|k\rangle|k\rangle_{23}\right) = \frac{1}{d}\sum_{ijk} a_j\delta_{ij}\delta_{ik}|k\rangle_3 = \frac{1}{d}\sum_k a_k|k\rangle_3 = \frac{1}{d}|\alpha\rangle_3\,. \tag{17}$$

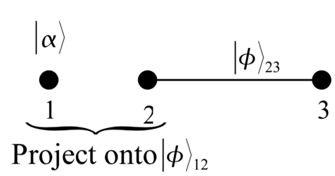


Figure 2: The general case for dimension d.

Now, if we rotate the maximally entangled state in d-dimension, $|\phi(U)\rangle = U^\dagger \otimes I\,|\phi\rangle$, where U and I are unitary and unit operations in d-dimension, then the projection of $|\alpha\rangle_1|\phi\rangle_{23}$ onto the rotated state $|\phi(U)\rangle_{12}$

results in the state $\frac{1}{d}U|\alpha\rangle_3$ at node 3. Note that initially nodes 2 and 3 are still in the original maximally entangled state $|\phi\rangle_{23}$. The proof is as follows.

$$\begin{aligned}
\text{rotated max entangled state }\ |\phi(U)\rangle = U^\dagger \otimes I \frac{1}{\sqrt{d}} \sum_i |i\rangle|i\rangle &= \frac{1}{\sqrt{d}} \sum_i |U^\dagger i\rangle|i\rangle \\
\xrightarrow{\text{projection}} \quad & \frac{1}{d}\left(\sum_i \langle i|\langle U^\dagger i|\right)\left(\sum_{jk} a_j |j\rangle|k\rangle|k\rangle\right) \\
=& \frac{1}{d}\sum_{ijk} a_j \delta_{jk} \langle i|Uj\rangle|k\rangle \\
=& \frac{1}{d}\sum_{jk} a_j |k\rangle\langle k|Uj\rangle \\
=& \frac{1}{d}\sum_{j} a_j |Uj\rangle \\
=& \frac{U}{d}|\alpha\rangle_3 .
\end{aligned} \tag{18}$$

The 2-dimensional case has been discussed above. So back to our 2D case, we have the standard Bell basis states $\{|B_{cd}\rangle\}$ for measurement. **If we want to apply an arbitrary unitary single-qubit operation, U, on the initial state $|\alpha\rangle$, what we need to do is just to apply a joint measurement on qubits 1 and 2 under the rotated Bell basis states, $\mathcal{B}(U) = \{|B(U)_{cd}\rangle\}$, then the teleported state at qubit 3 would be $X^d Z^c(U|\alpha\rangle)$. Since we know what c, d are from our measurement result, we can easily do the correction on qubit 3 to get the desired state $U|\alpha\rangle$.** This means that we can use a way totally based on measurements to simulate a random single-qubit operation.

2.2 Two-qubit gates

The next question is what about two-qubit gates, since we need both single- and two-qubit gates for universal quantum computation? The answer is obvious and straightforward by considering the 4-dimensional case in the general discussion above. We know two-qubit gates are also unitary, so by letting $d = 4$, we can realize an arbitrary unitary two-qubit gates by measurements, although they would be not simply 2D Bell measurements but high-dimensional Bell measurements. Let's take the CZ gate for example.

Still, we have three nodes, but this time each node contains two qubits so that we need to use a two-qubit state to describe every node. Node 1 is initially in a two-qubit state $|\alpha\rangle$ (for example, $|\alpha\rangle = |+\rangle \otimes |+\rangle$), and we want to apply a CZ gate to these two qubits and get an output state $CZ|\alpha\rangle$ (for example, $CZ|\alpha\rangle = \frac{1}{\sqrt{2}}(|0\rangle|+\rangle + |1\rangle|-\rangle)$, just like the boxed one in Eq. 12). We can choose our maximally entangled state, in which nodes 2 and 3 are initially, as $|\phi\rangle = |B_{00}\rangle|B_{00}\rangle$. In 4-dimension, we can construct an orthonormal set of Bell basis states from $|\phi\rangle$, $\{U_{ij}^\dagger \otimes I|\phi\rangle\}$, where $U_{ij} = P_i \otimes P_j$, and $P_{i,j}$ are standard Pauli operators. Then after measuring nodes 1 and 2 under this set of basis, we can teleport the two-qubit state carried by node 1 to node 3, and end up with node 3 in state $U_{ij} = P_i \otimes P_j |\alpha\rangle$.

So similar to the single-qubit gate case, we can also construct a set of rotated orthonormal 4-dimensional Bell basis states, $\{(U_{ij}CZ)^\dagger \otimes I|\phi\rangle\}$, then by measuring nodes 1 and 2 under this set of rotated basis, node 3 will end up in state $U_{ij}CZ|\alpha\rangle = (P_i \otimes P_j)CZ|\alpha\rangle$. So we have realized the two-qubit CZ gate via measurements only.

Since $U_{ij} = P_i \otimes P_j$ has 16 possible combinations in total, this means our Bell measurement is 16-dimensional. We can reduce this to a 8-dimensional measurement [4], as shown by Fig. 3. Here, to be consistent with [4], we use $|H\rangle = \frac{1}{2}(|00\rangle + |01\rangle + |10\rangle - |11\rangle)$ as the two-qubit maximally entangled state. We prepare qubit pairs $\{3,4\}, \{5,7\}, \{6,8\}$ to be in $|H\rangle$ initially, and qubit pair $\{1,2\}$ is out input, which is in state $|\alpha\rangle = a|00\rangle + b|01\rangle + c|10\rangle + d|11\rangle$. Then we perform 3-qubit Bell measurements on pairs

$\{1,3,5\}$ and $\{2,4,6\}$ under basis

$$\begin{aligned}&\{X^i \otimes X^j \otimes I(|000\rangle \pm |111\rangle)\}\\ =&\{|000\rangle \pm |111\rangle, |001\rangle \pm |110\rangle, |010\rangle \pm |101\rangle, |100\rangle \pm |011\rangle\},\ \text{which is 8-dimensional.}\end{aligned} \tag{19}$$

This will leave qubit pair $\{7,8\}$ ending up with $|\psi\rangle_{78} = (H \otimes H)CZ|\alpha\rangle$, up to an unitary factor of Pauli operations.

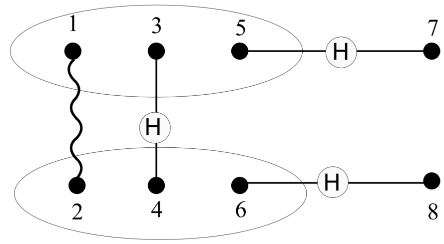


Figure 3: CZ gate realization by teleportation with 8-dimensional measurements.

We can verify this via simple calculation for the case that both 3-qubit Bell measurement results are $|000\rangle + |111\rangle$. Then, we start with 8-qubit state

$$\begin{aligned}|\Psi\rangle &= |\alpha\rangle_{12}|H\rangle_{34}|H\rangle_{57}|H\rangle_{68}\\ &= \frac{1}{8}(a|00\rangle + b|01\rangle + c|10\rangle + d|11\rangle)_{12}(|00\rangle + |01\rangle + |10\rangle - |11\rangle)_{34}\\ &\qquad (|00\rangle + |01\rangle + |10\rangle - |11\rangle)_{57}(|00\rangle + |01\rangle + |10\rangle - |11\rangle)_{68},\end{aligned} \tag{20}$$

then for the measurement results known, we only care about terms with pairs $\{1,3,5\}$ and $\{2,4,6\}$ in states $|000\rangle$ and $|111\rangle$. These relevant terms are

$$\begin{aligned}&|000\rangle_{135}|000\rangle_{246}\left[\frac{1}{8}a(|00\rangle + |01\rangle + |10\rangle + |11\rangle)_{78}\right]\\ +&|000\rangle_{135}|111\rangle_{246}\left[\frac{1}{8}b(|00\rangle - |01\rangle + |10\rangle - |11\rangle)_{78}\right]\\ +&|111\rangle_{135}|000\rangle_{246}\left[\frac{1}{8}c(|00\rangle + |01\rangle - |10\rangle - |11\rangle)_{78}\right]\\ +&|111\rangle_{135}|111\rangle_{246}\left[-\frac{1}{8}d(|00\rangle - |01\rangle - |10\rangle + |11\rangle)_{78}\right].\end{aligned} \tag{21}$$

For a 4-dimensional vector, we have

$$\begin{aligned}\vec{r} &= A\hat{a} + B\hat{b} + C\hat{c} + D\hat{d}\\ &= \frac{A+B+C+D}{2}\frac{\hat{a}+\hat{b}+\hat{c}+\hat{d}}{2} + \frac{A+B-C-D}{2}\frac{\hat{a}+\hat{b}-\hat{c}-\hat{d}}{2}\\ &\quad + \frac{A-B+C-D}{2}\frac{\hat{a}-\hat{b}+\hat{c}-\hat{d}}{2} + \frac{A-B-C+D}{2}\frac{\hat{a}-\hat{b}-\hat{c}+\hat{d}}{2},\end{aligned} \tag{22}$$

so for $(|000\rangle + |111\rangle)_{135}(|000\rangle + |111\rangle)_{246}$ case, we have (not normalized)

$$\begin{aligned}&|\psi\rangle_{78} = (a+b+c-d)|00\rangle + (a-b+c+d)|01\rangle + (a+b-c+d)|10\rangle + (a-b-c-d)|11\rangle,\\ &\text{while } (H\otimes H)CZ|\alpha\rangle = \begin{bmatrix}1 & 1 & 1 & 1\\ 1 & -1 & 1 & -1\\ 1 & 1 & -1 & -1\\ 1 & -1 & -1 & 1\end{bmatrix}\begin{bmatrix}1 & 0 & 0 & 0\\ 0 & 1 & 0 & 0\\ 0 & 0 & 1 & 0\\ 0 & 0 & 0 & -1\end{bmatrix}\begin{bmatrix}a\\ b\\ c\\ d\end{bmatrix} = \begin{bmatrix}a+b+c-d\\ a-b+c+d\\ a+b-c+d\\ a-b-c-d\end{bmatrix} = |\psi\rangle_{78}.\end{aligned} \tag{23}$$

The cases with other possible measurement results can also be easily verified using the same way.

In summary so far (which I just copy from [1]), if we have a pool of maximally entangled states we can apply any unitary gate U to any (multi-qubit) input state $|\psi\rangle$ by measurements alone. A significant annoyance is that we do not get the exact desired result $U|\psi\rangle$ but instead get $PU|\psi\rangle$ where P is some Pauli operation (on each qubit) depending on the measurement outcome. This is the residue of the randomness of quantum measurement outcomes in our computational formalism.

2.3 Adaptive measurements

A natural question to ask is that will this extra Pauli operations cause any issues when performing a universal quantum computation? For example, if we need to sequentially perform single-qubit operations $U_3U_2U_1|\psi\rangle$, where $|\psi\rangle$ is the input state, then the actual result we would get following the discussion above is $P_3U_3P_2U_2P_1U_1|\psi\rangle$, where $P_{1,2,3}$ are Pauli operators which depend on measurement outcomes for each step. This is not the net unitary single-qubit gate we want, $U = U_3U_2U_1$. To fix this issue, we remember that any 3D rotation can be expressed in the form of Euler angle, i.e.,

$$\begin{aligned}
&U = R_x(\zeta)R_z(\eta)R_x(\xi),\ \text{where } \xi,\eta,\zeta \text{ are Euler angles, and} \\
&R_x(\theta), R_z(\theta) \text{ are rotations about } x \text{ and } z \text{ axes by an angle } \theta, \\
&R_x(\theta) = e^{-i\theta X} = \begin{bmatrix} \cos\theta & -i\sin\theta \\ -i\sin\theta & \cos\theta \end{bmatrix}, \quad R_z(\theta) = e^{-i\theta Z} = \begin{bmatrix} e^{-i\theta} & 0 \\ 0 & e^{i\theta} \end{bmatrix}, \\
&\text{from the fact that } e^{-i\hat{\sigma}\cdot\hat{n}\theta} = \mathbb{1}\cos\theta - i\hat{\sigma}\cdot\hat{n}\sin\theta,\ \text{where } \hat{n} \text{ is the rotation axis.}
\end{aligned} \tag{24}$$

Equivalently, we can also express a random 3D rotation as

$$\begin{aligned}
&U = W(0)W(\theta_1)W(\theta_2)W(\theta_3),\ \text{for some angles } \theta_{1,2,3}, \\
&\text{where } W(\theta) = \frac{1}{\sqrt{2}}\begin{bmatrix} 1 & e^{i\theta} \\ 1 & -e^{i\theta} \end{bmatrix} = HP(\theta),\ H \text{ is the Hadamard gate, } P(\theta) = \begin{bmatrix} 1 & 0 \\ 0 & e^{i\theta} \end{bmatrix}.
\end{aligned} \tag{25}$$

Single-qubit gates, together with CZ gate, consist of a complete set for universal quantum computation, so it is equivalent to take sets $\{CZ, R_x(\theta), R_z(\phi),\ \text{for all } \theta \text{ and } \phi\}$, and $\{CZ, W(\theta),\ \text{for all } \theta\}$.

Then we know from before that the extra Pauli operations are just combinations of X and Z, and if we express any rotation under the above two formalism, we have

$$\begin{aligned}
R_x(\theta)X &= XR_x(\theta), \quad R_z(\theta)Z = ZR_z(\theta), \\
R_x(\theta)Z &= ZR_x(-\theta), \quad R_z(\theta)X = XR_z(-\theta), \\
\text{and } W(\theta)X &= e^{i\theta}ZW(-\theta), \quad W(\theta)Z = XW(\theta).
\end{aligned} \tag{26}$$

Then for example, if we want to do $\cdots R_z(\beta)R_x(\alpha)|\psi\rangle$, then the real outcome would be

$$\begin{aligned}
&\cdots R_z(\beta)R_x(\alpha)|\psi\rangle \\
\Rightarrow\quad &\cdots X^cZ^dR_z(\beta)X^aZ^bR_x(\alpha)|\psi\rangle, \quad a,b,c,d = 0,1 \\
= &\cdots X^cZ^dX^aR_z((-1)^a\beta)Z^bR_x(\alpha)|\psi\rangle \\
= &\cdots X^cZ^dX^aZ^bR_z((-1)^a\beta)R_x(\alpha)|\psi\rangle.
\end{aligned} \tag{27}$$

We notice that after commuting, the second gate $R_z(\beta)$ is replaced by $R_z((-1)^a\beta)$. However, the first gate is always performed before the second gate, i.e., before performing the second, we have known the measurement results a, b. So we can choose the second gate to be $R_z((-1)^a\beta)$, then

$$\begin{aligned}
&\cdots R_z(\beta)R_x(\alpha)|\psi\rangle \\
\Rightarrow\quad &\cdots X^cZ^dR_z((-1)^a\beta)X^aZ^bR_x(\alpha)|\psi\rangle \\
= &\cdots X^cZ^dX^aZ^bR_z(\beta)R_x(\alpha)|\psi\rangle.
\end{aligned} \tag{28}$$

Another point to notice is that although single-qubit gates X and Z don't commute, their anticommutation is zero, i.e., $[X, Z] \neq 0, \{X, Z\} = 0$. Then if we swap the positions of them, this will just cause a overall global phase of π, which doesn't matter, so we can rewrite our result above as

$$\cdots X^{a+c} Z^{b+d} R_z(\beta) R_x(\alpha) |\psi\rangle . \tag{29}$$

So, continuing in this way by using adaptive measurements (choosing the rotation angles based on former measurement results), if our input state is a multi-qubit state, we get

$$\cdots X_2^{m_2} Z_2^{n_2} X_1^{m_1} Z_1^{n_1} (\text{the correct wanted } U) |\psi\rangle , \tag{30}$$

where U is the direct product of single-qubit operations, and X_i, Z_i means single-qubit gates on the i-th qubit, and m_i, n_i are accumulations of single-qubit-measurement outcomes for the i-th qubit.

Actually, more generally, we know $UX = X(XUX), UZ = Z(ZUZ)$, so when swapping positions of Pauli operations and single-qubit unitary operation U, we just change U to XUX or ZUZ, then $(XUX)X = XU, (ZUZ)Z = ZU$. One may ask if so, why bother to decompose U into three or even four operations? Well, that's because by doing that, the change of U would be just simple signs on the angles [1].

Then what about two-qubit gates, such as a CZ gate? We want to find out what would happen if we swap the CZ gate with Pauli operations. We can express a CZ gate in the matrix form, $CZ = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & -1 \end{bmatrix}$. It turns out that we have the following propagation relations,

$$\begin{aligned}
&\text{(i) } CZ(Z \otimes I) = (Z \otimes I)CZ, \text{ since } Z \otimes I \text{ is also diagonalized,} \\
&\text{(ii) } CZ(X \otimes I) = (X \otimes Z)CZ, \\
&\qquad \text{since } CZ(X \otimes I) = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & -1 \end{bmatrix} \begin{bmatrix} 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \\ 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \end{bmatrix} = \begin{bmatrix} 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \\ 1 & 0 & 0 & 0 \\ 0 & -1 & 0 & 0 \end{bmatrix} \\
&\qquad = \begin{bmatrix} 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & -1 \\ 1 & 0 & 0 & 0 \\ 0 & -1 & 0 & 0 \end{bmatrix} \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & -1 \end{bmatrix} = (X \otimes Z)CZ, \\
&\text{(iii) } CZ(X \otimes Z) = (X \otimes I)CZ, \text{ by taking Hermitian conjugate for both sides of (ii).}
\end{aligned} \tag{31}$$

And we recall a CZ gate is symmetric in both inputs, so similar relations work if X and/or Z are applied on the second qubit. So after swapping positions, CZ gates remain the same, and what are possibly added in front are just Pauli operations.

Here we come to the last step to complete a round of universal quantum computation–reading out by measuring the teleported state (maybe consisting of many qubits) in Z-basis. The question is will all the Pauli operations, $X_i^{m_i} Z_i^{m_i}$, in front affect the measurement result, since based on the principle of MBQC, we don't apply single-qubit gates to correct these Pauli operations? The answer is they don't cause any problems, and what we need to do is reinterpretation of our measurement results. For example, if our final state for one qubit is $U|\psi\rangle = a|0\rangle + b|1\rangle$, then $Z_i^{n_i} U |\psi\rangle_i = \begin{bmatrix} 1 & 0 \\ 0 & -1 \end{bmatrix} \begin{bmatrix} a \\ b \end{bmatrix} = \begin{bmatrix} a \\ (-1)^{n_i} b \end{bmatrix}$, so it has no effect on the measurement result. $X_i^{2n+1} U |\psi\rangle = a|1\rangle + b|0\rangle$, and $X_i^{2n} U |\psi\rangle_i = a|0\rangle + b|1\rangle$, so we just need to reinterpret the measurement outcome k_i, which is 0 or 1, as $k_i \oplus m_i$, where m_i is the corresponding X Pauli exponent.

An interesting fact is that these final Z measurements are never adaptive, which means they never depend on former measurement results. Also, they won't affect any of the former measurements either, so we can actually perform them first before any of the other measurements, and after all the measurements are done, we come back to reinterpret these results!

3 The one-way quantum computer (1WQC)

It it not an easy task in reality to implement joint measurements, so let's introduce another measurement-based model which only needs local measurements, with the expense of generating an initially highly-entangled state for universal quantum computation, the so called "the one-way quantum computer" or "cluster state model" [3, 5].

The one-way quantum computer starts with a pool of qubits prepared in state $|+\rangle$, then a 2-dimensional square grid is constructed by applying CZ gates to every nearest-neighbor pair of qubits that are connected by a line, as shown in Fig. 4. Since these CZ gates commute with each other, they can be applied in parallel. The resulting 2D grid is a highly-entangled state, called a cluster state. The reason we need 2D is that although 1D cluster state, which is a string, is enough for single-qubit gates, we need 2D to perform CZ gates, which enables the universal quantum computation.

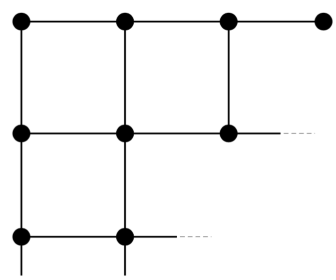


Figure 4: A 2D cluster state for the one-way quantum computer.

The idea of the one-way quantum computer is that **any quantum gate array can be implemented as a pattern of single-qubit measurements on a large 2-dimensional cluster state**. The reason that we call this model "one-way" quantum computer is that the pure cluster state we start with is irreversibly degraded as the computation proceeds. Similar to TQC, the output state differs from the desired one by some Pauli operators, which are also called bi-product operators in 1WQC literature.

3.1 Single-qubit gates

The single-qubit measurements could be under the basis $M_z = \{|0\rangle, |1\rangle\}$ or $M(\theta) = \{|0\rangle \pm e^{i\theta}|1\rangle\}$, where the latter is a combination of M_x and M_y, and for $\theta = 0$, it corresponds to a measurement under X basis. The measurement outcomes are always labeled as 0 or 1.

Ex.1 Let's first consider the following example as shown in Fig. 5.

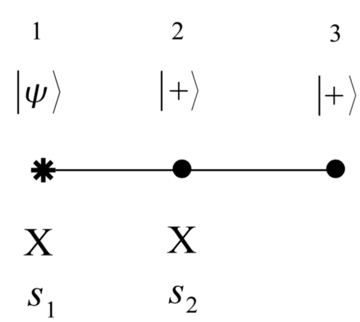


Figure 5: An analogue of teleportation in a 1D cluster state model.

We start with input state $|\psi\rangle = a|0\rangle + b|1\rangle$ in a line with two $|+\rangle$ states, where $|0\rangle, |1\rangle$ are basis of Z.

Each pair of nearest-neighbor qubits are entangled via CZ gates. Then we successively (from left to right) measure qubits 1,2 under X basis, with outcomes s_1 and s_2, where $s_{1,2} = 0, 1$, then qubit 3 will end up in state $X^{s_2} Z^{s_1} |\psi\rangle$. The whole process can be expressed as $M_{x,2} M_{x,1} CZ_{23} CZ_{12} |\psi\rangle_1 |+\rangle_2 |+\rangle_3$. This can be easily verified, as below

$$
\begin{aligned}
|\Psi\rangle_{123} &= CZ_{12} CZ_{23} |\psi\rangle_1 |+\rangle_2 |+\rangle_3 \\
&= \frac{1}{2} CZ_{12} CZ_{23} [a(|000\rangle + |001\rangle + |010\rangle + |011\rangle) + b(|100\rangle + |101\rangle + |110\rangle + |111\rangle)]_{123} \\
&= \frac{1}{2} [|0\rangle_1 a(|00\rangle + |01\rangle + |10\rangle - |11\rangle)_{23} + |1\rangle_1 b(|00\rangle + |01\rangle - |10\rangle + |11\rangle)_{23}] \\
&= \frac{1}{2\sqrt{2}} \{|+\rangle_1 [(a+b)(|00\rangle + |01\rangle) + (a-b)(|10\rangle - |11\rangle)]_{23} + |-\rangle_1 [(a-b)(|00\rangle + |01\rangle) + (a+b)(|10\rangle - |11\rangle)]_{23}\} \\
&= \frac{1}{2} [|{+}{+}\rangle_{12} (a|0\rangle + b|1\rangle)_3 + |{+}{-}\rangle_{12} (b|0\rangle + a|1\rangle)_3 + |{-}{+}\rangle_{12} (a|0\rangle - b|1\rangle)_3 + |{-}{-}\rangle_{12} (-b|0\rangle + a|1\rangle)_3] \\
&= \frac{1}{2} [|{+}{+}\rangle_{12} |\psi\rangle_3 + |{+}{-}\rangle_{12} X|\psi\rangle_3 + |{-}{+}\rangle_{12} Z|\psi\rangle_3 + |{-}{-}\rangle_{12} XZ|\psi\rangle_3].
\end{aligned}
\tag{32}
$$

Note that these two X measurements can actually be done in parallel. We can also think this process as

- input state qubit 1 in $|\psi\rangle$, and prepare qubits 2 and 3 in state $|+\rangle$;
- entangle qubits 1 and 2, and $|\Psi\rangle_{12} = \frac{1}{\sqrt{2}} \{|+\rangle_1 [(a+b)|0\rangle + (a-b)|1\rangle]_2 + |-\rangle_1 [(a-b)|0\rangle + (a+b)|1\rangle]_2\} = |+\rangle_1 (H|\psi\rangle_2) + |-\rangle_1 (XH|\psi\rangle_2)$;
- measure qubit 1 under X basis, with outcome s_1, which leaves qubit 2 in state $X^{s_1} H |\psi\rangle_2$;
- entangle qubits 2 and 3, then measure qubit 2 under X basis with outcome s_2, leaving qubit 3 in state $(X^{s_2} H)(X^{s_1} H) |\psi\rangle_3$. By choosing $\theta = 0$ in Eq. 26, we have $HX = ZH$, so $(X^{s_2} H)(X^{s_1} H) |\psi\rangle_3 = X^{s_2} Z^{s_1} |\psi\rangle$.

So we change the order to $M_{x,2} CZ_{23} M_{x,1} CZ_{12} |\psi\rangle_1 |+\rangle_2 |+\rangle_3$. This is totally doable, since the CZ gate between qubits 2 and 3 has nothing to do with the measurement on qubit 1, so they definitely commute. So this example tells us the process that prepares a large entangled network and then do all the measurements is equivalent to step-by-step, i.e. entangle-measure-entangle, approach.

Ex.2 Let's now turn to another example. We consider a single-step process for a measurement under the general basis, $M(\theta) = \{|0\rangle \pm e^{i\theta} |1\rangle\}$, as shown in Fig. 6.

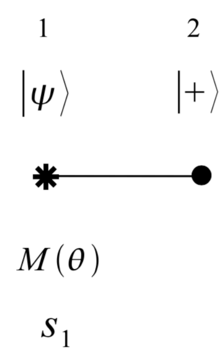


Figure 6: A single-step process for a measurement under the general basis $M(\theta)$.

We know for a 2D vector, which can be expressed as $\vec{r} = a\hat{i} + b\hat{j}$, if we replace the basis with $\hat{i} \pm e^{i\theta}\hat{j}$, then $\vec{r} = a\hat{i} + b\hat{j} = \frac{a+e^{-i\theta}b}{\sqrt{2}}\frac{\hat{i}+e^{i\theta}\hat{j}}{\sqrt{2}} + \frac{a-e^{-i\theta}b}{\sqrt{2}}\frac{\hat{i}-e^{i\theta}\hat{j}}{\sqrt{2}}$. So for input $|\psi\rangle = a|0\rangle + b|1\rangle$, we have

$$\begin{aligned} |\Psi\rangle_{12} &= CZ_{12}|\psi\rangle_1 |+\rangle_2 \\ &= \frac{1}{\sqrt{2}}[|0\rangle_1 a(|0\rangle + |1\rangle)_2 + |1\rangle_1 b(|0\rangle - |1\rangle)_2] \\ &= \frac{1}{2}\{|+(\theta)\rangle_1 [(a+e^{i\theta}b)|0\rangle + (a-e^{i\theta}b)|1\rangle]_2 + |-(\theta)\rangle_1 [(a-e^{i\theta}b)|0\rangle + (a+e^{i\theta}b)|1\rangle]_2\} \\ &= \frac{1}{\sqrt{2}}[|+(\theta)\rangle_1 (W(-\theta)|\psi\rangle_2) + |-(\theta)\rangle_1 (XW(-\theta)|\psi\rangle_2)]. \end{aligned} \tag{33}$$

So after measurement with outcome s_1, qubit 2 ends up in state $X^{s_1}W(-\theta)|\psi\rangle$. Then any 1-dimensional pattern can be viewed as the sequential application of this single-step process, and with the decomposition Eq. 25 and propagation relation Eq. 26, we can realize any single-qubit by repeating this unit.

Ex.3 As the application of the two examples above, let's consider the third example of measurement pattern for implementing a single-qubit gate [5]. Still, we start with input state $|\psi\rangle$ in a line with four $|+\rangle$ states. Each pair of nearest-neighbor qubits are entangled via CZ gates, as shown in Fig. 7. The unitary single-qubit operation U can be decomposed into Euler angles ξ, η, ζ, then we successively (from left to right) measure qubits 1,2,3,4 adaptively under basis $M(0), M(-\xi(-1)^{s_1}), M(-\eta(-1)^{s_2}), M(-\zeta(-1)^{s_1+s_3})$, respectively, where $s_{1,2,3,4} = 0, 1$ are measurement outcomes on qubits 1,2,3,4, and qubit 5, which serves as the output qubit, ends up in state $X^{s_2+s_4}Z^{s_1+s_3}U|\psi\rangle$. The minus sign in the measurement basis come from the fact that they are adaptive measurements and the propagation relations.

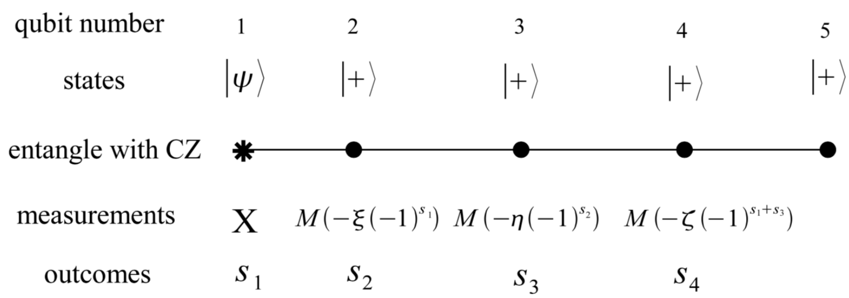


Figure 7: A single-qubit gate U applied to input state $|\psi\rangle$ using the cluster state model.

Furthermore, the input state $|\psi\rangle$ can also be regarded as $|+\rangle$ applied by a unitary operator U, i.e. $|\psi\rangle = U|+\rangle$. So actually, after we prepare a large 2D cluster state, all the steps following are just measurements, including the input process.

3.2 Two-qubti gates (CZ gate)

The last missing block is the CZ gate. An explicit measurement patter for a CZ gate is shown in Fig. 8, but not unique. I cannot find an elegant way to prove it (maybe using stablizer formalism would help?), but we can have a sanity check for a simplest case. Assume the two input qubits are initially both in state $|+\rangle$ and not entangled with each other, then all the Z-measurements just detach those nodes from the graph, leaving the graph to be a 1D chain. Then another fact is that when we apply two X-measurements for neighbor qubits, it will also detach these two from the graph, and connect the rest [6]. Finally, we will see that the graph ends up with the two output qubits entangled with each other, which is exactly the result we want, $CZ|+\rangle|+\rangle$. This can be expressed using a graph, as in Fig. 9.

So, by now, we have known that we can perform universal quantum computation, which can be decomposed into basis $\{CZ, W(\theta),$ for all $\theta\}$, by firstly preparing a large 2D cluster state, and then performing only local single-qubit measurements. The basis under which the measurements are depend on the operations that we wanna realize, and also former measurement outcomes, but should be either M_z or $M(\theta)$.

The final result may differ from the desired one by some bi-product operators, which also depend on former measurement results and can be corrected by reinterpretation, just as we discussed at the end of the section of TQC.

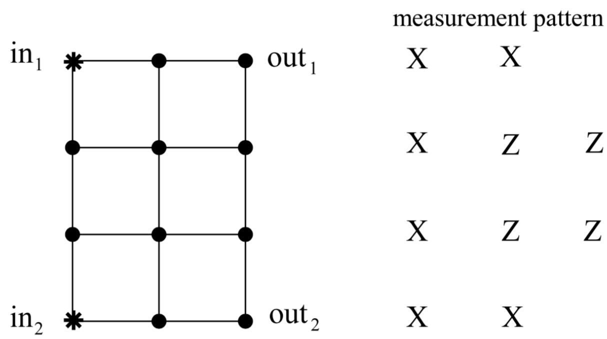


Figure 8: A measurement pattern for a CZ gate.

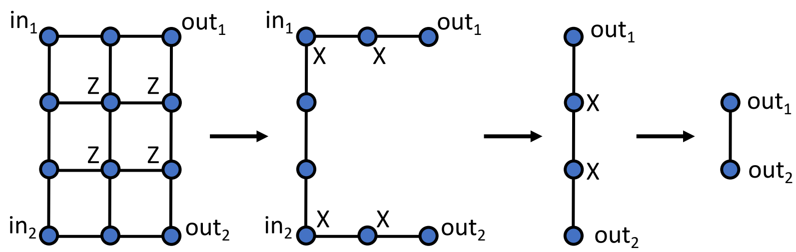


Figure 9: A CZ gate entangles two input qubits which are both initially in state $|+\rangle$.

4 Loss tolerance in one-way quantum computation

The one-way quantum computer requires a large highly-entangled cluster state, so the prerequisite for implementing this measurement-based model is to generate this pure 2D cluster state. However, especially when using photons as qubits, there could be photon loss during transmission when the measurement pattern proceeds. This may cause the whole process to fail and we have to start over. So quantum computation architectures can only be considered viable if they are demonstrably fault tolerant. There are some error correction mechanism, but the threshold of the error on one gate is just around 0.01% using standard fault tolerance techniques to this new cluster state model. M. Varnava, D. E. Browne, and T. Rudolph [6] proposed a way that can tolerate up to 50% qubit loss, which is remarkable, while with the expense of a more-complicated cluster state structure by introducing tree states.

One of the basic ideas in this loss tolerance mechanism is the **indirect measurement**, which relies on the correlation between nodes in a cluster state. Let's consider a simplest example to see how this works. Suppose we have two qubits initially prepared in state $|+\rangle$ and then entangled via a CZ gate, then the two-qubit state would be $|\psi\rangle_{12} = \frac{1}{\sqrt{2}}(|+\rangle_1 |0\rangle_2 + |-\rangle_1 |1\rangle_2)$. Then assume we need to perform a Z-measurement on qubit 2, but it has been lost at this moment. Then we can perform a X-measurement on qubit 1 instead, if the outcome is 0 (1), then it means that equivalently the outcome of Z-measurement on qubit 2 would have been 0 (1) as well. So we can still perform the required Z-measurement, in an indirect way, even though qubit 2 is no longer available. This analysis is called "counterfactual reasoning" in [6], which lies at the heart of this loss tolerance scheme.

We use tree states to realize this loss tolerance. A tree state is a highly-entangled cluster state, which can be specified by a branching parameter set. Fig. 10 is an example of a depth $d = 3$, number of arms is

$k = 2$ which is identical for all levels, with branching parameters $\{b_0 = b_1 = b_2 = 2\}$.

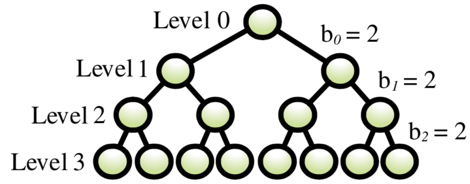


Figure 10: A tree state with branching parameters $\{b_0 = b_1 = b_2 = 2\}$.

4.1 Indirect measurements

Let's firstly discuss some properties of cluster states, as listed below and shown in Fig. 11.

- A Z-measurement on a certain qubit detaches it from the cluster, along with all the entanglement relations of it. This is discussed in my former notes [7]. The qubits originally connected to the measured one may need reinterpretation of the basis depending on the measurement outcome.

- Two adjacent X-measurements are performed on qubits 2 and 3 in Fig. 11(b), which removes these two qubits and bond their neighbors. This can be verified as below.

$$
\begin{aligned}
|\Psi\rangle_{12345} &= CZ_{34} |+\rangle_4 (|0++{+}\rangle + |1-{-}{-}\rangle)_{2135} \quad \text{neglecting normalization coefficient} \\
&= CZ_{34} |+\rangle_4 (|0+0+\rangle + |0+1+\rangle + |1-0-\rangle - |1-1-\rangle)_{2135} \\
&= (|{+}00++\rangle + |{+}01-+\rangle + |{-}10+-\rangle - |{-}11--\rangle)_{12345} \\
&= |00\rangle_{23} |+++\rangle_{145} + |01\rangle_{23} |+-+\rangle_{145} + |10\rangle_{23} |-+-\rangle_{145} - |11\rangle |---\rangle_{145} \\
&= |{+}{+}\rangle_{23} (|+++\rangle + |+-+\rangle + |-+-\rangle - |---\rangle)_{145} \\
&\quad + |{+}{-}\rangle_{23} (|+++\rangle - |+-+\rangle + |-+-\rangle + |---\rangle)_{145} \\
&\quad + |{-}{+}\rangle_{23} (|+++\rangle + |+-+\rangle - |-+-\rangle + |---\rangle)_{145} \\
&\quad + |{-}{-}\rangle_{23} (|+++\rangle - |+-+\rangle - |-+-\rangle - |---\rangle)_{145} \\
&= |{+}{+}\rangle_{23} (|{+}0{+}\rangle + |{-}1{-}\rangle)_{145} \\
&\quad + |{+}{-}\rangle_{23} (|{+}1{+}\rangle + |{-}0{-}\rangle)_{145} \\
&\quad + |{-}{+}\rangle_{23} (|{+}0{+}\rangle - |{-}1{-}\rangle)_{145} \\
&\quad + |{-}{-}\rangle_{23} (|{+}1{+}\rangle - |{-}0{-}\rangle)_{145}
\end{aligned}
\tag{34}
$$

  Also, we need reinterpretation if the measurement outcomes are not 00. For example, if we got 10 for qubits 2 and 3, then we need to reinterpret the basis for qubit 4, $|0\rangle \to |0\rangle, |1\rangle \to -|1\rangle$. After reinterpretation, the resulting state of qubits 1,4,5 is $(|{+}0{+}\rangle + |{-}1{-}\rangle)_{145}$, as we expected.

- An indirect Z-measurement can be performed as shown in Fig. 11(c). Here, qubit 2 is measured under X basis, and qubits 1,3,4 are all those connected to 2. We measure qubits 1 and 4 under Z basis, resulting in an indirect Z-measurement on qubit 3. Qubits 1,3,4 could have other connections. We can verify this easily. Firstly, all the three measurements commute, since they are applied to different qubits. The Z-measurements on qubits 1 and 4 detach them, and of course all their connections, from the graph. Then qubits 2 and 3 are left in state $|\psi\rangle_{23} = |{+}0\cdots\rangle_{23\ldots} + |{-}1\rangle_{23\ldots}$, where $\cdots$ means other connections of qubit 3. So it is clear that this indirect Z-measurement works, and it also detaches qubit 3 from the rest part of graph which is denoted as $\cdots$ in Fig. 11(c).

- With these properties in mind, let's discuss how we perform an indirect Z-measurement in a tree. As shown in Fig. 11(d), which is part of a tree with the branching parameter being 2 for the shown level. An indirect Z-measurement will be performed if the three measurements circles in **either** orange **or** green are successful. With a larger branching parameter, as long as there is one branch successfully realizing the desired measurement patter, the indirect Z-measurement on the root qubit will be done.

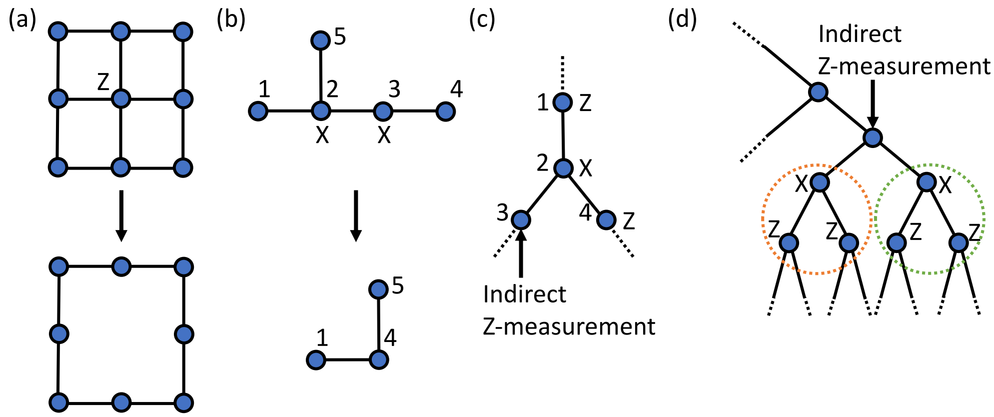


Figure 11: Three properties of cluster states. (a) A Z-measurement detaches a qubit from the cluster and erases its entanglement relations. (b) Two adjacent X-measurements remove these two qubits and bond between their neighbors. (c) An indirect Z-measurement is realized by performing a X-measurement on the qubit it connects and Z-measurements on all other qubits that connected to the X-measured qubit.

4.2 Loss-tolerant single-qubit measurements in a cluster state

Back to our discussion of measurement-based quantum computation. From the discussion in Sec. 3, we recall that we only need single-qubit measurements under either Z basis ($M_z = \{|0\rangle, |1\rangle\}$) or a combination of X and Y basis ($M(\theta) = \{|0\rangle \pm e^{i\theta} |1\rangle\}$) to perform universal quantum computation. But what if the qubit which needs to be measured is lost before the measurement? We can fix this by attaching a tree structure to this qubit, then even if this qubit is lost, it is still possible to equivalently measure it. This is the key idea in [6], and the loss tolerance, which means the probability of loss for a qubit during transimission, can be up to 50%.

Let's discuss how this tree structure works. For a Z-measurement, it is easy, and has been discussed in Fig. 11(d). What we need to do is to attach several branches to the qubit that needs to be measured. Then even if the original qubit is no longer there, we can still realize the required Z-measurement in an indirect way, as long as there exists one branch that allows us to do so.

What about $M(\theta)$? Let's turn to the language in [6], $M(\theta)$ is just the measurement under the basis of single-qubit operator $A(\alpha) = \cos\alpha X + \sin\alpha Y$. This can be performed loss-tolerantly as shown in Fig. 12.

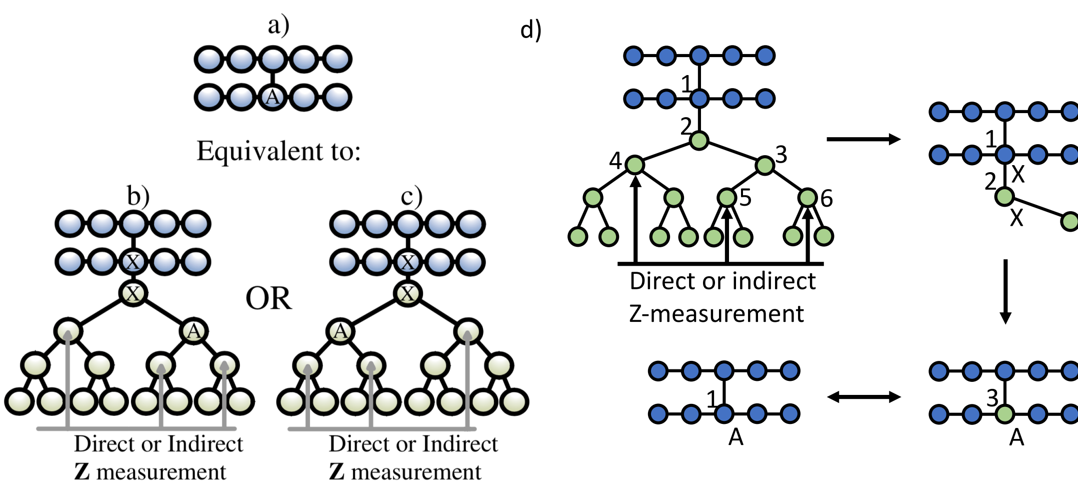


Figure 12: A single-qubit local measurement under A basis, as shown in (a), can be performed equivalently and loss-tolerantly either through (b) or (c). The detailed steps for (b) are shown in (d).

Here, if we want to A-measure qubit 1, as denoted in Fig. 12(d), we can either perform the measurement pattern as Fig. 12(b) or (c). The three Z-measurements can be either direct or indirect using the subtrees below these nodes. Take Fig. 12(b) for example, the detailed steps are

- perform the three Z-measurements, directly or indirectly, on qubits 4,5,6, which detach them from the graph;
- perform to X-measurements on qubits 1 and 2, which detach them from the graph and at the same time bond qubit 3 to the position where qubit 1 originally is;
- perform single-qubit A-measurement on qubit 3, which is equivalently an A-measurement on qubit 1 at the first beginning.

One may ask (I would), this still needs a X-measurement on qubit 1, and if we are allowed to do so, it means that qubit 1 is not lost, then why don't we directly perform the A-measurement on qubit 1? Or to say, how does these two equivalent measurement patterns solve the problem of the loss of qubit 1? The good news is that we can perform the two X-measurements on qubits 1 and 2 in advance as part of the preparing process when qubit 1 is still there, way before we come to this A-measurement step. So what we need to do when the A-measurement is required is just the three Z-measurements and an A-measurement on the qubits that have not been lost yet, plus Z-measurements on those subtrees, which we don't need, for detaching. For example, if we want to perform the pattern in Fig. 12(b), and just find that qubit 3 is lost (it doesn't matter if qubits 1 and 2 are still there, because we have performed the pair of X-measurements way earlier, so they have accomplished their missions), then we turn to the chance to perform the pattern in Fig. 12(d). Although qubit 3 is already lost, it is sitll somewhere in the universe and entangled with our graph, so we must perform an indirect Z-measurement to detach it from the graph. Then we just need to complete the rest two Z-measurements and also the A-measurement on qubit 4. Only if these two patterns both fail, due to the loss, do we fail to perform the desired single-qubit A-measurement. With a larger branching parameter, we will have a higher probability of success.

4.3 Effective loss rate and the optimized tree structure

The next question is, what is the optimized tree structure that can help us achieve the maximum probability of success of the A-measurement? And what is this maximum probability of success that is possible? We start our analysis with a simple example, and then extend to the general case. Here we have several assumptions:

- the probability of loss for a single qubit, denoted as ε_0, is fixed for all the qubits in the cluster;
- the pair of X-measurements is assumed to be successful already;
- as long as a qubit to be measured is not lost, we always have a probability of 1 to perform the measurement;
- if we want to perform a Z-measurement on a qubit, and it is not lost, we always choose the direct way, and it is always successful.

For the example given in Fig. 12, where the tree has a depth of $d = 3$, and branching parameters $b_0 = b_1 = b_2 = 2$, the conditions of success are:

- level 1 has at least one node that is not lost, and we are able to detach all other qubits in this level with Z-measurements, directly for those not lost or indirectly for those lost;
- we are able to detach all the child nodes of the A-measured level-1 node with Z-measurements, directly for those not lost or indirectly for those lost.

We define the probabilities of successfully performing an indirect Z-measurement on any lost qubit in level-i as R_i. Then for the example above, the probability of success can be calculated as

$$\begin{aligned}
&P(\text{successfully detach a lost level-1 qubit}) = R_1\\
&P(\text{successfully detach a level-2 qubit})\\
=&1 - P(\text{fail to detach a level-2 qubit})\\
=&1 - P(\text{a level-2 qubit is lost}) \times P(\text{the indirect } Z\text{-measurement on a level-2 qubit fails})\\
=&1 - \varepsilon_0(1 - R_2)\\
=&1 - \varepsilon_0 + \varepsilon_0 R_2,
\end{aligned} \tag{35}$$

$$\begin{aligned}
\Rightarrow P = &[P(\text{both level-1 nodes are not lost})\\
&+ P(\text{one of the level-1 nodes is lost}) \times P(\text{successfully detach a lost level-1 qubit})]\\
&\times [P(\text{successfully detach a level-2 qubit})]^2\\
= &[(1-\varepsilon_0)^2 + C_2^1(1-\varepsilon_0)\varepsilon_0 R_1](1-\varepsilon_0+\varepsilon_0 R_2)^2\\
= &[(1-\varepsilon_0)^2(\varepsilon_0 R_1)^0 + C_2^1(1-\varepsilon_0)(\varepsilon_0 R_1) + C_2^2(1-\varepsilon_0)^0(\varepsilon_0 R_1)^2 - C_2^2(1-\varepsilon_0)^0(\varepsilon_0 R_1)^2](1-\varepsilon_0+\varepsilon_0 R_2)^2\\
= &[(1-\varepsilon_0+\varepsilon_0 R_1)^2 - (\varepsilon_0 R_1)^2](1-\varepsilon_0+\varepsilon_0 R_2)^2.
\end{aligned} \tag{36}$$

For the general case where the tree has a depth of $d = m+1$, and branching parameters $\{b_0, b_1, \cdots, b_m\}$, the probability of success can be calculated as

$$\begin{aligned}
P &= \left[(1-\varepsilon_0)^{b_0} + C_{b_0}^1 \varepsilon_0 (1-\varepsilon_0)^{b_0-1} R_1 + C_{b_0}^2 \varepsilon_0^2 (1-\varepsilon_0)^{b_0-2} R_1^2 + \cdots\right](1-\varepsilon_0+\varepsilon_0 R_2)^{b_1}\\
&= \left[\sum_{n=0}^{b_0-1} C_{b_0}^n (\varepsilon_0 R_1)^n (1-\varepsilon_0)^{b_0-n}\right](1-\varepsilon_0+\varepsilon_0 R_2)^{b_1}\\
&= \left[\sum_{n=0}^{b_0} C_{b_0}^n (\varepsilon_0 R_1)^n (1-\varepsilon_0)^{b_0-n} - (\varepsilon_0 R_1)^{b_0}\right](1-\varepsilon_0+\varepsilon_0 R_2)^{b_1}\\
&= \left[(1-\varepsilon_0+\varepsilon_0 R_1)^{b_0} - (\varepsilon_0 R_1)^{b_0}\right](1-\varepsilon_0+\varepsilon_0 R_2)^{b_1},
\end{aligned} \tag{37}$$

which is formula (1) in [6]. The probability of successfully performing an indirect Z-measurement on any lost qubit in level-i is

$$\begin{aligned}
R_i &= 1 - P(\text{failure of an indirect } Z\text{-measurement on level-}i)\\
&= 1 - [P(\text{failure of a set of } X, Z\text{-measurements on level-}(i+1))]^{b_i}\\
&= 1 - [1 - P(\text{success of a set of } X, Z\text{-measurements on level-}(i+1))]^{b_i}\\
&= 1 - [1 - P(\text{success of a } X\text{-measurement}) \times P(\text{success of a direct or indirect } Z\text{-measurement})^{b_{i+1}}]^{b_i}\\
&= 1 - [1 - (1-\varepsilon_0)(1-\varepsilon_0+\varepsilon_0 R_{i+2})^{b_{i+1}}]^{b_i}, \quad \mathrm{i \le m}
\end{aligned} \tag{38}$$

and $R_{m+1} = R_{m+2} = 0$, $b_{m+1} = 0$. This is formula (2) in [6].

We then define $\varepsilon_{\text{eff}} = 1 - P$ to represent the overall effective loss rate for the tree-encoded logical cluster qubit. In order to achieve our goal of loss tolerance, ε_{eff} should be lower than ε_0. Then the next question is, for a given number of total qubits, Q, in a tree and single-qubit loss rate ε_0, how can we design the tree structure to minimize ε_{eff}? This question is actually kind of equivalent to the question in [6], which is, for fixed ε_0 and ε_{eff}, what is the minimum number of total qubits in the tree? Let's explore our version following.

We can start with an example of total number of qubits, $Q = 15$, which is actually the case shown in Fig. 12. There are several choices of branching parameters that give a total number of qubits of $Q = 15$. We know the relation between Q and branching parameters is given by

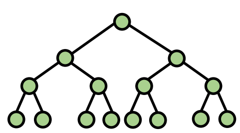
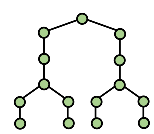
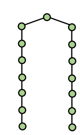
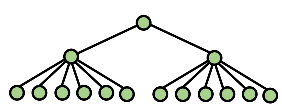
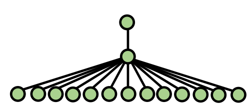
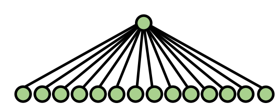
$$\begin{aligned}
Q &= 1 + b_0 + b_0 b_1 + b_0 b_1 b_2 + \cdots\\
&= 1 + b_0\{1 + b_1[1 + b_2(1 + \cdots)]\}
\end{aligned} \tag{39}$$

$$
\begin{aligned}
\Rightarrow \quad & b_0 \text{ is a factor of } Q-1 \\
& b_1 \text{ is a factor of } \frac{Q-1}{b_0} - 1 \\
& b_2 \text{ is a factor of } \frac{\frac{Q-1}{b_0} - 1}{b_1} - 1 \\
& \cdots .
\end{aligned}
\tag{40}
$$

We can write a program to calculate all possible branching parameter sets for a fixed Q, and the corresponding effective loss rates, in order to find out which tree structure has the best loss tolerance. The code has been attached in the Appendix. Considering with a large Q, the recursive algorithm might be slow, I used C^{++} to do so.

For the case with $Q = 15$, it turns out there are 53 possibilities of branching parameters in total, and the branching parameters for the three smallest and largest effective loss rates and corresponding tree structures are listed in Table 1.

Table 1: Different branching parameters and corresponding ε_{eff} for a tree with total number of qubits $Q = 15$, and single-qubit loss rate $\varepsilon_0 = 0.2$. The upper line shows three best trees, and the lower line shows three worst.

branching parameter	$b_0 \sim b_2 = 2, 2, 2$	$b_0 \sim b_4 = 2, 1, 1, 2, 1$	$b_0 \sim b_6 = 2, 1, 1, 1, 1, 1, 1$
depth	3	5	7
ε_{eff}	0.13029	0.157366	0.159056
tree structure			
branching parameters	$b_0 \sim b_1 = 2, 6$	$b_0 \sim b_1 = 1, 13$	$b_0 = 14$
depth	2	2	1
ε_{eff}	0.748347	0.95602	0.95602
tree structure			

We can see that thin-tall trees are better choices than wide-short trees. Actually, for the case with depth $d = 1$ or $d = 2$ but $b_0 = 1$, it is impossible to get an effective loss rate smaller than single-qubit loss rate, since for these two extreme cases,

$$
\begin{aligned}
\varepsilon_{\text{eff}} &= 1 - (1 - \varepsilon_0)^Q \\
&\geq 1 - (1 - \varepsilon_0) \\
&= \varepsilon_0 .
\end{aligned}
\tag{41}
$$

So these two cases are the worst ones, and they are equally worst.

One may make this guess–is the binary tree always the best choice? The answer is, it is not, but a binary tree usually has a not-bad performance, although not best. For example, for $Q = 31$, the best tree has branching parameters $b_0 \sim b_3 = 2, 2, 3, 1$ with the optimized effective loss rate $\varepsilon_{\text{eff, min}}(Q = 31) = 0.0846487$, while the binary tree ($b_0 \sim b_3 = 2, 2, 2, 2$) ranks 5-th out of 438 total possible cases, with $\varepsilon_{\text{eff, binary}}(Q = 31) = 0.142407$, which is even worse than the $Q = 15$ case (this is interesting). For $Q = 63$, the best tree has branching parameters $b_0 \sim b_4 = 2, 3, 3, 1, 1$ with the optimized effective loss rate $\varepsilon_{\text{eff, min}}(Q = 63) = 0.0620743$, while the binary tree ($b_0 \sim b_4 = 2, 2, 2, 2, 2$) ranks 3-rd out of 4856 total possible cases, with $\varepsilon_{\text{eff, binary}}(Q = 63) = 0.0809607$. The two best cases are shown in Fig. 13.

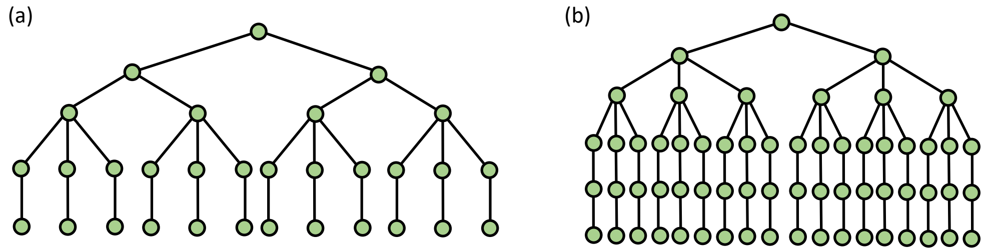


Figure 13: The best tree for (a) $Q = 31$ with $\varepsilon_{\text{eff, min}}(Q = 31) = 0.0846487$, and (b) $Q = 63$, with $\varepsilon_{\text{eff, min}}(Q = 63) = 0.0620743$. The single-qubit loss rate $\varepsilon_0 = 0.2$.

We notice that with the total number of qubit Q increasing, the best result for effective loss rate decreases. We can try more values of Q, and different single-qubit loss rates. The best effective loss rates are plotted in Fig. 14. Here, we use the same axes scales as in [6], i.e., log vs loglog. However, here we choose single-qubit loss rates to be $\varepsilon_0 = 0.1, 0.2, 0.3, 0.4$, since the case with $\varepsilon_0 = 0.49$, we cannot get ε_{eff} better than 0.49 for just about 100 total qubits.

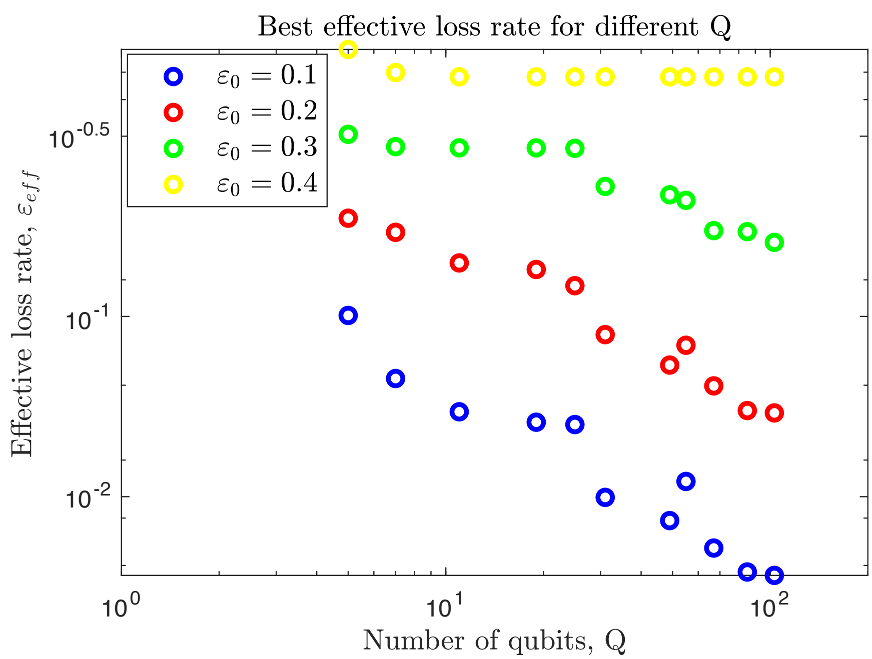


Figure 14: The best effective loss rate ε_{eff} versus total number of qubits Q, for single-qubit loss rate $\varepsilon_0 =$ 0.1 (blue) , 0.2 (red) , 0.3 (green) , 0.4 (yellow) , respectively.

Our result is similar to the result in [6]. Unfortunately, due to limit of the computing power of my laptop, I am not able to go further to see what will happen with $Q \sim 10^3$ and even larger, since with $Q = 127$, there are already 88041 possible choices of tree structures! Another point worth to note is that, it is not definitely the case that we can achieve better ε_{eff} with a larger Q. We can see this by comparing the 7-th and 8-th points in Fig. 14, where the 8-th point is higher than the 7-th for $\varepsilon_0 = 0.1$ and 0.2. This is because for some values of Q, we cannot construct a set of branching parameters that give a good loss-tolerant tree structure, in the limit that the branching parameters must be integers. But the trend is the case that with larger Q, we are able to achieve lower ε_{eff}.

We fixed the total number of qubits above in order to compare the performance of a thin-tall tree and a fat-short tree under the same given overhead Q. We can also do the same calculation as they did in [6], since usually, we may care more about if we want to achieve a goal of effective loss rate, what is the minimum total number of qubits we need? This version of numerical simulation gives the result as shown in Fig. 15, which basically agrees with the result in [6]. Again, due to the limited computing power, I just tried all values of Q from $5 \sim 103$, and dropped the ones whose best ε_{eff} are even worse than the results we can achieve with fewer qubits.

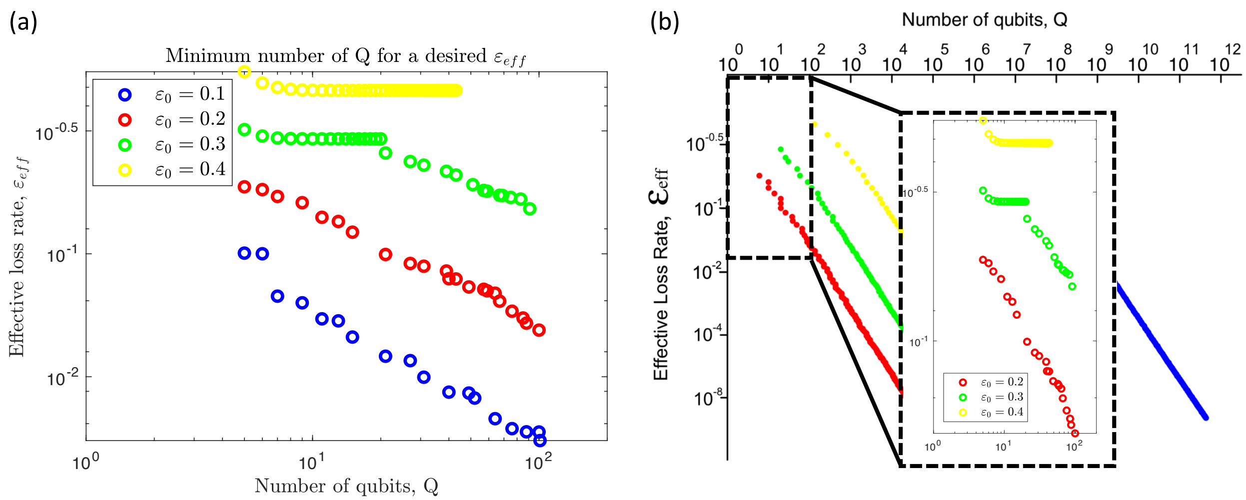


Figure 15: The minimum total number of qubits needed to achieve a desired effective loss rate. (a) Our numerical simulation result for $\varepsilon_0 = 0.1$ (blue) , 0.2 (red) , 0.3 (green) , 0.4 (yellow) . (b) The result in [6] and the comparison with ours for $\varepsilon_0 = 0.2$ (red) , 0.3 (green) , 0.4 (yellow) .

Note that, when we say a "thin-tall" tree, it may look not that "thin" and "tall" as it sounds. This is because of **the property of the tree structure itself–the number of qubits increases exponentially with the depth**, especially for a large depth. For example, we have a binary tree with total number of qubits $Q = 127$, then the bottom most layer has 64 qubits, even more than half of the total number! So except the case where the branching parameter equals to 1, the number of qubits in the bottom of the tree is still large, even for a "thin-tall" tree. So the word "thin-tall" just means that comparing with an even "fatter" tree, for example a $b_0 = 126$ tree with $Q = 127$ still, the binary tree is relatively thin and tall.

So how to understand this conclusion intuitively, that a relatively thin-tall tree has a better performance than a fat-wide tree? At first sight, one may think that with more branches at level 1, that is, a larger b_0, there must be a higher probability to have one branch with which we can successfully perform the A-measurement, so that we'll have a higher probability of success. This is not true, since for all the other branches, we can not just ignore them, we need to detach them from the tree. This means that with more branches at level 1, there are more branches that we need to detach. So as long as there is one branch lost, and the indirect Z-measurement also fails, the whole process fails. This explains why a large b_0, on the contrary, decreases the probability of success. Therefore, a better choice may be that, we just have 2 or 3 child nodes for each parent node, and make sure we have a higher probability of success to perform the direct or indirect Z-measurements to detach the parent node, when we need to. **Of course, there exist some trade-off considerations here, and we also need to take into account the limit of total number of qubits, single-qubit loss rate, etc.. There is NOT a certain law behind, and this conclusion is just kind of a rule of thumb. We cannot just say a tree with a certain structure must be the best choice without considering all the factors**. For example, with $Q = 103$ and $\varepsilon = 0.2$, the best tree is $b_0 \sim b_3 = 3, 3, 5, 1$ with $\varepsilon_{\text{eff}} = 0.0355506$, and there are 90 qubits in the two bottom most layers. While the tree with branching parameters $b_0 \sim b_5 = 2, 2, 3, 1, 2, 2$, which may sound like a promising candidate, just ranks 10-th out of 35394, with $\varepsilon_{\text{eff}} = 0.0734095$, which is not so bad, but maybe also not good enough.

On the other hand, just for a single A-measurement, we may not want to spend too many resources on it, which means the total number of qubits in a tree may not be that large. For example, we may just budget 15 qubits for a single A-measurement, then the binary tree would be a good choice, as we discussed before.

In summary, we can achieve a very small value of effective loss rate with a large number of

qubits to construct the tree, which could be much lower than the single-qubit loss rate. Also, for a certain value of Q, usually a choice of branching parameters that $b_0 = 2$ and $b_i = 3$ or 2 or 1 for $i > 0$ would be a good one.

5 Loss tolerance in one-way quantum repeater

Tree states can also be useful as quantum repeaters [8]. When a tree state, which carries the quantum information, arrives at the next repeater station, it will be measured in a certain pattern to retrieve the message, which will be re-encoded into the next tree. This measuring process is irreversible, so we call this model the "one-way quantum repeater". Similar to the one-way quantum computation case, encoding the message in trees significantly increases the loss tolerance during transmission, and through a chain of repeater stations, we can transmit this information to a distant receiver with a pretty high probability of success. This loss-tolerant transmission of quantum information is essential in quantum networking.

The question is, the sender has a qubit in an unknown state $|\phi\rangle = \alpha|0\rangle + \beta|1\rangle$, and he/she wants to send it to the receiver who is thousands of kilometers away. For this kind of long-range and high-rate transmission, photons are the most promising carriers of the information with no doubt. However, due to the photon loss during transmission, we need multiple repeater stations in between to refresh the information again and again to make sure the right message is sent to the receiver. But even we do this, the loss between repeater stations still affects the success probability of information transmission, so we need other error-correction mechanism to help us, and using tree states is a great idea.

5.1 Using single photons as "couriers"

Encoding. Let's consider the simplest case. For the sender and receiver, matter qubits are usually the ideal memory for the quantum information, but for transmission purpose, photons are the ideal carriers. So we must be able to encode quantum information from a matter qubit to a photonic qubit, or from a photonic qubit to a matter qubit. This is usually done using a CZ gate between a matter qubit and a photonic qubit, plus a Bell measurement. Suppose we have a matter qubit at the sending station, which is in state $|\psi\rangle_{s_1} = |\phi\rangle = \alpha|0\rangle + \beta|1\rangle$. We can prepare another matter qubit, which is entangled with a photonic qubit in an entangled state $\frac{1}{\sqrt{2}}(|0\rangle_{s_2}|+\rangle_p + |1\rangle_{s_2}|-\rangle_p)$. Then we perform a Bell measurement on the two spins, which encodes the message into the photonic qubit, up to Pauli corrections and a Hadamard gate, as shown below (neglecting the normalization factors). This is actually a similar operation to the one in Fig. 1.

$$\begin{aligned}
|\Psi\rangle &= (\alpha|0\rangle + \beta|1\rangle)_{s_1}(|0\rangle_{s_2}|+\rangle_p + |1\rangle_{s_2}|-\rangle_p) \\
&= \alpha|00\rangle_{s_1s_2}|+\rangle_p + \alpha|01\rangle_{s_1s_2}|-\rangle_p + \beta|10\rangle_{s_1s_2}|+\rangle_p + \beta|11\rangle_{s_1s_2}|-\rangle_p \\
&= (|00\rangle + |11\rangle)_{s_1s_2}(\alpha|+\rangle_p + \beta|-\rangle_p) + (|00\rangle - |11\rangle)_{s_1s_2}(\alpha|+\rangle_p - \beta|-\rangle_p) \\
&\quad + (|01\rangle + |10\rangle)_{s_1s_2}(\alpha|-\rangle_p + \beta|+\rangle_p) + (|01\rangle - |10\rangle)_{s_1s_2}(\alpha|-\rangle_p - \beta|+\rangle_p) \\
&= |B_{00}\rangle_{s_1s_2}(\alpha|+\rangle_p + \beta|-\rangle_p) + |B_{10}\rangle_{s_1s_2}(\alpha|+\rangle_p - \beta|-\rangle_p) \\
&\quad + |B_{01}\rangle_{s_1s_2}(\alpha|-\rangle_p + \beta|+\rangle_p) + |B_{11}\rangle_{s_1s_2}(\alpha|-\rangle_p - \beta|+\rangle_p), \quad \text{according to Eq. 8.}
\end{aligned} \tag{42}$$

So after the Bell measurement between two spins, the photon will be in state $|\psi\rangle_p = HX^dZ^c(\alpha|0\rangle + \beta|1\rangle)$, where $c, d = 0, 1$ are the outcome of Bell measurement. So far we have completed the simplified version of the "encoding" step at the sending station in [8].

Re-encoding. Next step is "re-encoding" at repeater stations. Of course in our simplified version, there is no reason to do this, since we only have one photon, and once it is lost, there is no way to retrieve it unless we request the sender to sent it again (but this is not in the range of our discussion of the one-way quantum repeater). But let's just do the similar thing as in [8], in order to help understand the more practical cases. So our "courier" photon arrives at a repeater station carrying one qubit of quantum information, we then want to pass this information to another photon, and send this another photon to the next station, either a repeater or the receiver. Similar to the encoding step, we may need to prepare a spin entangled with a photon, and then perform a Bell measurement on the arriving photon and the spin. However, later we will use a tree state whose root qubit is a spin, and once the Bell measurement fails due to photon loss, we need prepare the whole tree again. So instead of a direct Bell measurement between the arriving photon and the root spin, we can firstly transfer the information to an auxiliary spin, then perform a Bell measurement between the two spins, as shown in Fig. 16.

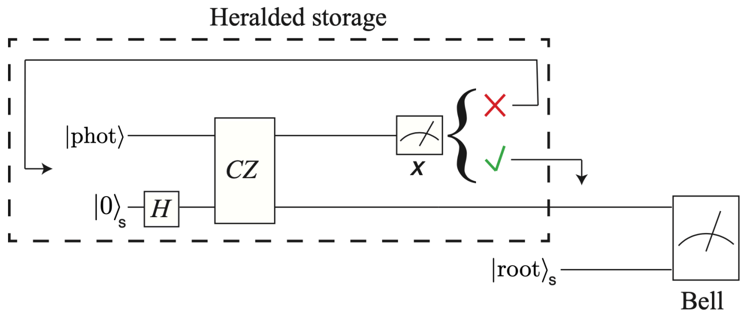


Figure 16: The re-encoding process can be done through three steps–firstly entangle the photon and the spin; then X-measure the photon, leaving the spin in state $|\phi\rangle = \alpha|0\rangle + \beta 1$ up to a Hadamard correction; finally perform a Bell measurement on two spins, so that transfer the information to the next photon.

Transferring information from a photon to a spin is easy. We firstly prepare the spin in state $|+\rangle = \frac{1}{\sqrt{2}}(|0\rangle + |1\rangle)_{a.s.}$ (a.s. means auxiliary spin), and the photon is in state $|\psi\rangle_p = |\phi\rangle = (\alpha|0\rangle + \beta|1\rangle)_p$. Then we apply a controlled-phase gate to entangle them, followed by a X-measurement on the photon. Then

$$\begin{aligned}|\Psi\rangle &= CZ_{p,a.s.}(\alpha|0\rangle + \beta|1\rangle)_p(|0\rangle + |1\rangle)_{a.s.} \\ &= (\alpha|00\rangle + \alpha|01\rangle + \beta|10\rangle - \beta|11\rangle)_{p,a.s.} \\ &= |0\rangle_p \alpha|+\rangle_{a.s.} + |1\rangle_p \beta|-\rangle_{a.s.} \\ &= |+\rangle_p (\alpha|+\rangle + \beta|-\rangle)_{a.s.} + |-\rangle_p (\alpha|+\rangle - \beta|-\rangle).\end{aligned} \tag{43}$$

Still, up a Hadamard gate, the information is transferred to the spin after X-measuring the photon. The CZ gate between a photon and a spin can be implemented via a single-sided cavity coupled to the spin [9]. Next we perform a Bell measurement on the two spins, which is exactly the same process as the encoding step, leaving the photon carrying the information.

Decoding. The last step is to transfer the information to the receiver, considered as a receiving spin, at the receiving station. One may have noticed, this is exactly what we did in step one at any repeater station–entangling the photon with the spin, followed by a X-measurement on the photon. By now, we successfully transmit a message qubit from the sender to the receiver.

5.2 Encoding a qubit into tree states

Now here comes the question, what if the photon gets lost during transmission? This will result in the failure of our whole process and then we have to start over again. One of the solutions is to encode this qubit into tree states. Let's take a tree with $b_0 = 2$ (only level-1 nodes) for example. The root of the tree is, again, a spin qubit. The encoding, re-encoding, and decoding steps are shown in Fig. 17.

Encoding. The tree is prepared in state $|\psi\rangle_{\text{tree}} = (|0++\rangle + |1--\rangle)_{123}$, where 1 represents the root spin qubit, and 2, 3 represent the two level-1 photons. Another spin, denoted by 0, is the memory storing the information that needs to be transmitted, which is in state $|\psi\rangle_0 = (\alpha|0\rangle + \beta|1\rangle)_0$.

$$\begin{aligned}|\Psi\rangle &= |\psi\rangle_0 |\psi\rangle_{\text{tree}} \\ &= (\alpha|0\rangle + \beta|1\rangle)_0(|0++\rangle + |1--\rangle)_{123} \\ &= |B_{00}\rangle_{01}(\alpha|++\rangle + \beta|--\rangle)_{23} + |B_{10}\rangle_{01}(\alpha|++\rangle - \beta|--\rangle)_{23} \\ &\quad + |B_{01}\rangle_{01}(\alpha|--\rangle + \beta|++\rangle)_{23} + |B_{11}\rangle_{01}(\alpha|--\rangle - \beta|++\rangle)_{23}.\end{aligned} \tag{44}$$

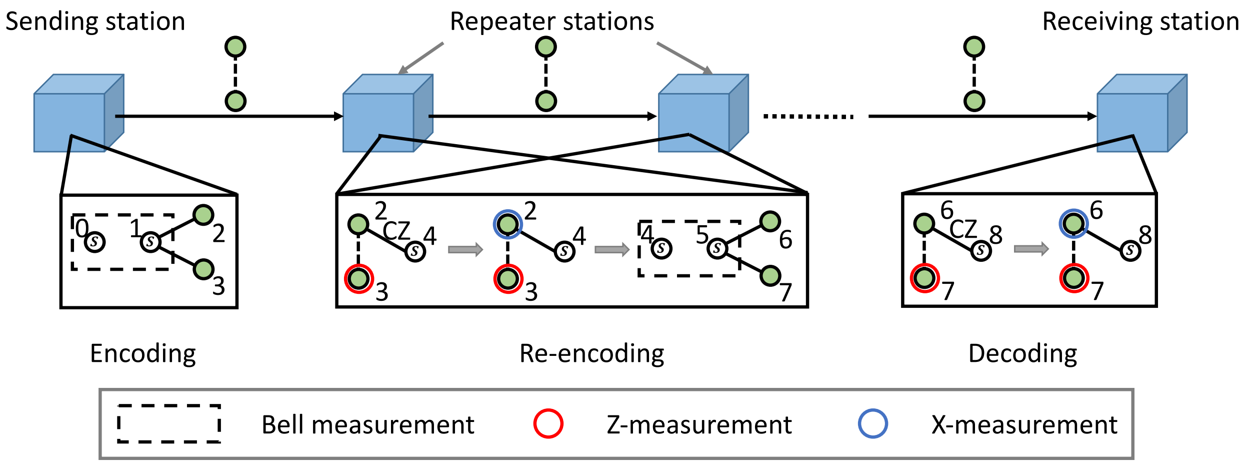


Figure 17: The one-way quantum repeater when encoding a qubit into a tree state with only two photonic qubits in level-1.

We then perform a Bell measurement on the two spins, and after correction depending on the measurement outcome, we get the two photons 2 and 3 in state

$$\begin{aligned} |\psi\rangle_{23} &= (\alpha\,|{+}{+}\rangle + \beta\,|{-}{-}\rangle)_{23} \\ &= |0\rangle_2\,(\alpha\,|+\rangle + \beta\,|-\rangle)_3 + |1\rangle_2\,(\alpha\,|+\rangle - \beta\,|-\rangle)_3. \end{aligned} \tag{45}$$

We can see that qubits 2 and 3 are entangled, but in a slightly different way compared with those we have seen before. We use a connecting dashed line to represent this kind of entanglement in the graph. Then these two qubits are sent to the repeater station.

Re-encoding. As we discussed before, at the repeater station, we firstly transfer the message to an auxiliary spin, and this spin again transfer the message to a new tree via a Bell measurement. In this case, we have two photonic qubits, if not lost, so we must use a Z-measurement to detach one. Actually this is why we need a tree instead of a single photon. If we got one of the level-1 qubits lost, we can use an indirect Z-measurement to detach it if there are level-2 photons (which we don't have here, but an indirect Z-measurement has the same function as a direct one, so we just use this simplified case to help understand, and make our lives easier).

For example, we choose qubit 3 to be the "lost" one, and we use a Z-measurement to detach it. Since the Z-measurement on qubit 3 commutes with the CZ gate between qubit 2 and the auxiliary spin, it doesn't matter which one we perform first. So this gives us

$$\begin{aligned} |\Psi\rangle_{234} =& [|0\rangle_2\,(\alpha\,|+\rangle + \beta\,|-\rangle)_3 + |1\rangle_2\,(\alpha\,|+\rangle - \beta\,|-\rangle)_3]\,|+\rangle_4 \\ \xrightarrow{M_Z \text{ on } 2}& \;\; (\alpha\,|+\rangle + \beta\,|-\rangle)_3\,|+\rangle_4 \\ \xrightarrow{CZ_{34}}& \;\; \alpha(|+\rangle\,|0\rangle + |-\rangle\,|1\rangle)_{34} + \beta(|-\rangle\,|0\rangle + |+\rangle\,|1\rangle)_{34} \\ =& \,|+\rangle_3\,(\alpha\,|0\rangle + \beta\,|1\rangle)_4 + |-\rangle_3\,(\alpha\,|1\rangle + \beta\,|0\rangle)_4. \end{aligned} \tag{46}$$

So after we X-measure qubit 2, and do the correction based on the measurement result, we get the auxiliary spin in state $|\psi\rangle_4 = (\alpha\,|0\rangle + \beta\,|1\rangle)_4$. Then we are back to the encoding step, so we just need to do a Bell measurement on the auxiliary spin and the root spin of the new tree, and transmit the two level-1 photons 6 and 7 to the next station.

Decoding. The decoding step is easy. It is just the first half of the re-encoding step, where the auxiliary spin is replaced by the receiver spin. So we have successfully transmitted the message from the sender to the receiver.

5.3 Repeater performance

We now analyze the performance of this "one-way quantum repeater". According to [8], we quantify the performance in terms of the maximum quantum bit rate for a given distance between the sender and receiver.

5.3.1 Transmission success probability

Let's firstly analyze the transmission success probability, which means the probability of successfully transmitting a qubit from the sender to the receiver. Similar to before, we assume as long as a photonic qubit is not lost, all the operations (including Bell measurements, X or Z measurements, CZ gates) relevant to this photonic qubit are definitely successful. Also, we assume the generations of tree states are all successful, or to say, they have been prepared in advance. So under these assumptions, the encoding step is definitely successful, and the re-encoding and decoding steps have equal successful probabilities. The only reason that may cause the failure is photon loss.

Then suppose we have m repeater stations between the sending and receiving stations, and the probability of successfully transmitting the message from one station to the next one is η_e. This gives the transmission success probability

$$P_{\text{trans}} = \eta_e^{m+1}. \tag{47}$$

To calculate η_e, we recall our analysis in Section. 4. Suppose the probability of successfully transmitting a single photonic qubit between two repeater stations (which means it is not lost during transmission) is η, and the detection efficiency of the photon detectors is η_d, then a photonic qubit is considered not lost only if it is transmitted to the station, and detected by the detector. We use $\mu = 1 - \eta\eta_d$ to represent the single-qubit loss, which is the analogy to ε_0 in Section. 4.

So, the message is successfully transmitted from one station to the next one, only if one of the level-1 branches is not lost so that we can transfer the message it carries to the auxiliary spin, and all other branches can be detached via either direct (if not lost) or indirect (if lost) Z-measurements. So similar to Eq. 37 and Eq. 38 [6], for a tree with branching parameters $\{b_0, b_1, \cdots, b_d\}$, we have

$$\begin{aligned} &\eta_e = [(1-\mu+\mu R_1)^{b_0} - (\mu R_1)^{b_0}](1-\mu+\mu R_2)^{b_1}, \\ \text{where} \quad &R_k = 1 - [1-(1-\mu)(1-\mu+\mu R_{k+2})^{b_{k+1}}]^{b_k}, \quad k < d \\ &R_d = R_{d+1} = 0, \quad b_d = 0. \end{aligned} \tag{48}$$

Here, again, R_k represents the probability of successfully performing an indirect Z-measurement on a level-k qubit.

Furthermore, to link our calculation to the total transmission distance, we have

$$\eta = \exp(-L_0/L_{\text{att}}), \tag{49}$$

where L_0 is the distance between two adjacent stations, and $L_{\text{att}} = 20$km is the attenuation length of the optical fiber for telecom frequency.

Appendices

Best effective loss rate that can be achieved for a fixed total number of qubits.

```
//
//  main.cpp
//  lossTole
//
//  Created by YZhan on 3/27/20.
//  Copyright l' 2020 YuanZhan. All rights reserved.
//

#include <iostream>
#include <math.h>
using namespace std;
#define Q 15    // total number of qubits
#define NN 1000 // estimation of number of possibilities

int b[NN][Q] = {0}; // branching parameters
int d[NN] = {0};   // depth of tree
int Index = 1;
void Factorize(int avaiNum, int parentLevPara, int level);

int main() {
    // all possible branching parameters for Q=15
    int rootPara = 1;   // only one root
    double ep = 0.2;   // single-qubit loss rate
    double effLossRate[NN] = {0};    // effective loss rates
    Factorize(Q, rootPara, 1);
    Index--;
    cout<< "Number of possible branching parameter sets is "<< Index<< " for a total number of qubits ...
         Q="<< Q<< endl;

    // calculate effective loss rate for this new set
    for(int i = 1; i ≤ Index; ++i){     // for the i-th set, calculate epsilon_eff
        double R[Q + 1] = {0};  // success prob. of indirect Z-measurement in i-th level
        for(int j = d[i] - 1; j > 0; --j){
            R[j] = 1 - pow(1 - (1 - ep)*pow(1 - ep + ep*R[j + 2], b[i][j + 2]), b[i][j + 1]);
        }
        effLossRate[i] = 1 - (pow(1 - ep + ep*R[1], b[i][1]) - pow(ep*R[1], b[i][1]))*pow(1 - ep + ...
             ep*R[2], b[i][2]);
    }

    // sort effective loss rates
    int oriIndex[NN] = {0};   // keep track of original index
    for(int i = 1; i ≤ Index; ++i){
        oriIndex[i] = i;
    }
    for(int i = 1; i < Index; ++i){ // bubble sort
        for(int j = 1; j < Index - i; ++j){
            if(effLossRate[j] > effLossRate[j + 1]){
                double temp = effLossRate[j];
                effLossRate[j] = effLossRate[j + 1];
                effLossRate[j + 1] = temp;
                int tempIndex = oriIndex[j];
                oriIndex[j] = oriIndex[j + 1];
                oriIndex[j + 1] = tempIndex;
            }
        }
    }

    // output the three smallest and largest cases
    cout<< endl<< "The top 10 best structures:"<< endl;
    for(int i = 1; i ≤ 10; ++i){
        cout<< endl;
        cout<< "The "<< i<< "-th"<< " minimum effective loss rate is "<< effLossRate[i]<< ", with ...
             branching parameters:"<< endl;
        for(int j = 1; j ≤ d[oriIndex[i]]; ++j){
            cout<< b[oriIndex[i]][j]<< " ";
        }
        cout<< endl;
    }
    cout<< endl<< "The top 3 worst structures:"<< endl;
    for(int i = 1; i ≤ 3; ++i){
        cout<< endl;
        cout<< "The "<< i<< "-th"<< " maximum effective loss rate is "<< effLossRate[Index + 1 - i]<< ...
             ", with branching parameters:"<< endl;
        for(int j = 1; j ≤ d[oriIndex[Index + 1 - i]]; ++j){
```

```
            cout<< b[oriIndex[Index + 1 - i]][j]<< " ";
        }
        cout<< endl;
    }
    return 0;
}

void Factorize(int avaiNum, int parentLevPara, int level) {
    int Inte;   // the integer that needs factorization
    Inte = avaiNum/parentLevPara - 1;
    // cout<< Inte<< endl;
    for(int i = 1; i ≤ Inte; ++i){
        if(Inte%i == 0){
            if(i == Inte){  // we got a new set
                b[Index][level] = i;
                d[Index] = level;   // record the depth
                b[Index++][level + 1] = 0;  // mark the end
            }
            else{
                for(int j = Index; j < NN; ++j){   // cover all parent nodes
                    b[j][level] = i;
                }
                Factorize(Inte, i, level + 1);
            }
        }
    }
}
```

Minimum total number of qubits needed for a desired effective loss rate.

```
//
//  main.cpp
//  lossTole2
//
//  Created by YZhan on 3/30/20.
//  Copyright 1’ 2020 YuanZhan. All rights reserved.
//

#include <iostream>
#include <math.h>
using namespace std;
#define Q 103    // max test total number of qubits
#define NN 80000 // estimation of number of possibilities

int b[NN][Q] = {0}; // branching parameters
int d[NN] = {0};   // depth of tree
int Index = 1;
void Factorize(int avaiNum, int parentLevPara, int level);

int main() {
    // all possible branching parameters for Q=5-103
    double currentMin = 0.5;
    int currentIndex = 0;
    double resultEpEff[Q] = {0}; // record all best results
    int resultQ[Q] = {0}; // record corresponding Q
    for(int testQ = 5; testQ ≤ Q; ++testQ){
        // clear all arrays
        Index = 1;
        for(int i = 1; i < NN; ++i){
            if(d[i] == 0){  // all clear
                break;
            }
            d[i] = 0;
            for(int j = 1; j < Q; ++j){
                if(b[i][j] == 0){   // have reached the tail
                    break;
                }
                b[i][j] = 0;
            }
        }
        int rootPara = 1;   // only one root
        double ep = 0.2;   // single-qubit loss rate
        double effLossRate[NN] = {0};    // effective loss rates
        Factorize(testQ, rootPara, 1);
        Index--;
        cout<< "Number of possible branching parameter sets is "<< Index<< " for a total number of ...
```

```
             qubits Q="<< testQ<< endl;

        // calculate effective loss rate for this new set
        for(int i = 1; i ≤ Index; ++i){    // for the i-th set, calculate epsilon_eff
            double R[Q + 1] = {0};  // success prob. of indirect Z-measurement in i-th level
            for(int j = d[i] - 1; j > 0; --j){
                R[j] = 1 - pow(1 - (1 - ep)*pow(1 - ep + ep*R[j + 2], b[i][j + 2]), b[i][j + 1]);
            }
            effLossRate[i] = 1 - (pow(1 - ep + ep*R[1], b[i][1]) - pow(ep*R[1], b[i][1]))*pow(1 - ep + ...
                 ep*R[2], b[i][2]);
        }

        // sort effective loss rates
        int oriIndex[NN] = {0};   // keep track of original index
        for(int i = 1; i ≤ Index; ++i){
            oriIndex[i] = i;
        }
        for(int i = 1; i < Index; ++i){ // bubble sort
            for(int j = 1; j < Index - i; ++j){
                if(effLossRate[j] > effLossRate[j + 1]){
                    double temp = effLossRate[j];
                    effLossRate[j] = effLossRate[j + 1];
                    effLossRate[j + 1] = temp;
                    int tempIndex = oriIndex[j];
                    oriIndex[j] = oriIndex[j + 1];
                    oriIndex[j + 1] = tempIndex;
                }
            }
        }

        // record all best results
        if(effLossRate[1] < currentMin){    // only a better result worth recording
            resultEpEff[currentIndex] = effLossRate[1];
            currentMin = effLossRate[1];
            resultQ[currentIndex++] = testQ;
        }
    }

    // output results
    cout<< "There are "<< currentIndex<< " results worth recording"<< endl;
    cout<< "For fixed effective loss rates, the minimum Q needed:"<< endl<< "Q:"<< endl;
    for(int i = 0; i ≤ currentIndex; ++i){
        cout<< resultQ[i]<< " ";
    }
    cout<< endl<< "Corresponding effective loss rates:"<< endl;
    for(int i = 0; i ≤ currentIndex; ++i){
        cout<< resultEpEff[i]<< " ";
    }
    cout<< endl;

    return 0;
}

void Factorize(int avaiNum, int parentLevPara, int level) {
    int Inte;   // the integer that needs factorization
    Inte = avaiNum/parentLevPara - 1;
    // cout<< Inte<< endl;
    for(int i = 1; i ≤ Inte; ++i){
        if(Inte%i == 0){
            if(i == Inte){  // we got a new set
                b[Index][level] = i;
                d[Index] = level;   // record the depth
                b[Index++][level + 1] = 0;  // mark the end
            }
            else{
                for(int j = Index; j < NN; ++j){   // cover all parent nodes
                    b[j][level] = i;
                }
                Factorize(Inte, i, level + 1);
            }
        }
    }
}
```

References

[1] Richard Jozsa, *An introduction to measurement based quantum computation*, arXiv:quant-ph/0508124v2 (2005).

[2] Daniel Gottesman and Issac L. Chuang, *Demonstrating the viability of universal quantum computation using teleportation and single-qubit operations*, Nature **402**, 390-393 (1999).

[3] Robert Raussendorf and Hans J. Briegel, *A one-way quantum computer*, Phys. Rev. Lett. **86**, 5188 (2001).

[4] F. Verstraete and J. I. Cirac, *Valence-bond states for quantum computation*, Phys. Rev. A **70**, 060302(R) (2004).

[5] Robert Raussendorf, Daniel E. Browne, and Hans J. Briegel, *Measurement-based quantum computation on cluster states*, Phys. Rev. A **68**, 022312 (2003).

[6] Michael Varnava, Daniel E. Browne, and Terry Rudolph, *Loss tolerance in one-way quantum computation via counterfactual error correction*, Phys. Rev. Lett. **97**, 120501 (2006).

[7] "Generation of photonic repeater graph states - part I" note.

[8] Johannes Borregaard, Hannes Pichler, Tim Schöder, Mikhail D. Lukin, Peter Lodahl, and Anders S. Sørensen, *One-way quantum repeater based on near-deterministic photon-emitter interfaces*, arXiv:1907.05101 [quant-ph] (2019).

[9] L.-M. Duan and H. J. Kimble, *Scalable Photonic Quantum Computation through Cavity-Assisted Interactions*, Phys. Rev. Lett. **92**, 127902 (2004).