Magic State Distillation

Juan Gonzalez De Mendoza

We show how a set of operations consisting of perfect Clifford gates plus access to a family of states on the Bloch sphere are sufficient for universal quantum computation. We also describe an algorithm to obtain these states (named "magic states") to arbitrary precision by feeding in noisy ancillas.

I. INTRODUCTION/PREFACE

This paper is meant to summarize in a slightly simpler manner, the findings of Bravyi and Kitaev in 2004 [1]. We go into more mathematical depth by filling in steps that the authors skipped, which we considered not to be trivial. We also are able to describe the algorithm in more detail by only focusing on T-Type magic states. Therefore, the reader may take for granted that anything not cited comes directly from the first reference, in order to avoid citing every paragraph.

II. COMPUTATIONAL MODEL AND CLASSICAL SIMULATION

We will start by defining a computational model with a set of elementary operations O. This set can be decomposed into two subsets, one containing error-free operations called O_{ideal} and a faulty operation O_{faulty}.

The set O_{ideal} is composed of the following operations:

- Creating the state $|0\rangle$
- Apply unitary gates from the Clifford group
- Measure the eigenvalues of a Pauli observable

The Clifford gates are assumed to be ideal due to Gottesman's work demonstrating how they can be implemented fault tolerantly on stabilizer codes [2]. These operations are also known to be efficiently simulated on a classical computer, thanks to the Gottesman-Knill Theorem. Additionally, simulating a fair coin toss is possible by using operations from O_{ideal}, for example by creating a $|0\rangle$ and measuring the Pauli X eigenvalues. These two statements together imply that tossing of a fair coin may also be simulated efficiently. These operations allow us to to apply Clifford gates depending on the outcomes of the coin, and thus obtain mixed states.

With all of this in mind, we can define a region in the Bloch sphere which can be efficiently simulated. This corresponds to the pure states located at the axis (±1,0,0), (0,±1,0), (0,0,±1)[1], and any convex combination of them. Described as:

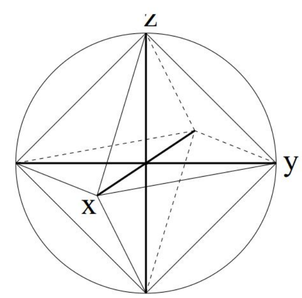

FIG. 1. An octahedron inscribed in the Bloch sphere which represents all states having 1-norm less than or equal to 1.

$\vec{a} = a_x(\pm 1,0,0) + a_y(0,\pm 1,0) + a_z(0,0,\pm 1) = (\pm a_x, \pm a_y, \pm a_z)$, where a_i represent probabilities, which bounds them as $0 \leq a_i \leq 1$. Thus, the one-norm of our Bloch vector must be less than one.

$$|\rho_x| + |\rho_y| + |\rho_z| \leq 1 \tag{1}$$

If we represent this geometrically, it corresponds to an octahedron inscribed inside the Bloch Sphere, as depicted in 1. Thus, the Gottesman-Knill theorem can be rephrased as follows: Let there be a quantum state contained within an octahedron inscribed in the Bloch sphere whose corners align with the Pauli eigenvectors, the evolution of said state through a circuit composed of Clifford gates can be efficiently simulated classically.

III. UNIVERSAL QUANTUM COMPUTATION

The Clifford group is known to not be sufficient for universal quantum computing. In contrast, the Clifford group in addition to any other 1-qubit gate not in the Clifford group has been shown to allow the approximation of any n-qubit gate to arbitrary precision.

With this in mind, we will now show how to implement a non-Clifford gate by starting in a state outside of the octahedron and only applying Clifford gates. Such states are called "Magic States".

Consider the T-Type magic state

$$|T\rangle\langle T| = \frac{1}{2}[\mathbb{1} + \frac{1}{\sqrt{3}}(\sigma_x + \sigma_y + \sigma_z)]. \tag{2}$$

---

[1] The Bloch vector representation comes from the fact that any two dimensional quantum state can be characterized by a vector $\vec{a}$ where $\rho = \frac{1}{2}(\mathbb{1} + \vec{a} \cdot \vec{\sigma})$

This $|T\rangle$ state is an eigenstate of the Clifford operator $\mathrm{T} = e^{\frac{i}{4}} SH$. Where

$$T = e^{\frac{i}{4}} \begin{matrix} 1 & 1 \\ i & -i \end{matrix} , S = \begin{matrix} 1 & 0 \\ 0 & i \end{matrix} , H = \frac{1}{\sqrt{2}} \begin{matrix} 1 & 1 \\ 1 & -1 \end{matrix} .$$

In the standard basis $|T\rangle = cos\beta |0\rangle + e^{\frac{i}{4}} sin\beta |1\rangle$ where $cos(2\beta) = \frac{1}{\sqrt{3}}$

Now Consider the state where we have two particles in the T state defined above $|TT\rangle = cos^2(\beta) |00\rangle + e^{\frac{i}{4}} sin(\beta)cos(\beta)[|01\rangle + |10\rangle] + e^{\frac{i}{2}} sin^2(\beta) |11\rangle$. Then we measure the ZZ observable, which is equivalent to measuring the parity of the bits. Thus, returning +1 for even parity, and -1 for odd parity. If the measurement yields -1, we start again, otherwise our resulting state is $|\Psi\rangle = cos(\frac{\pi}{12}) |00\rangle + isin(\frac{\pi}{12}) |11\rangle$. We then apply a CNOT_{01} to the state and discard the second qubit

$$|\Psi\rangle = cos(\frac{\pi}{12}) |0\rangle + isin(\frac{\pi}{12}) |1\rangle \tag{3}$$

Finally, we apply a Hadamard gate, and we obtain a state which we will label as

$$|A \quad_{\pi/6}\rangle = \frac{1}{\sqrt{2}} e^{\frac{i}{12}} (|0\rangle + e^{\quad i \frac{}{6}} |1\rangle) \tag{4}$$

With this state, we may now apply a non-Clifford gate to an arbitrary state $|\psi\rangle = \alpha |0\rangle + \beta |1\rangle$ by implementing the following algorithm:

1. Prepare the state $|\Psi_\theta\rangle = |\psi\rangle \otimes |A \quad_{\pi/6}\rangle$
2. Measure the observable ZZ, call the outcome z. The resulting state is $|\Psi_z\rangle = \delta_{z,+1}(\alpha |00\rangle + \beta e^{i\frac{}{6}} |11\rangle) + \delta_{z, \quad 1}(\alpha e^{i\frac{}{6}} |01\rangle + \beta |10\rangle)$
3. Apply a CNOT with the right qubit as the target and obtain $|\Psi_z\rangle = \delta_{z,+1}(\alpha |0\rangle + \beta e^{i\frac{}{6}} |1\rangle) \otimes |0\rangle + \delta_{z, \quad 1}(\alpha e^{i\frac{}{6}} |0\rangle + \beta |1\rangle) \otimes |1\rangle$
4. Discard the second qubit

Notice that we have now obtained one of two states depending on the measurement outcome z.

$$|\Psi_{out}\rangle = \alpha |0\rangle + e^{zi\frac{}{6}} \beta |1\rangle \tag{5}$$

If we perform this algorithm several times, the phases will add up on the exponent, with equal probability of contributing a positive or negative phase. The phase of the exponent after n iterations will follow a random walk distribution, which is guaranteed to reach every possible phase if ran enough times. Thus, with this construction we can evolve $\alpha |0\rangle + \beta |1\rangle \longrightarrow \alpha |0\rangle + e^{i\frac{}{6}} \beta |1\rangle$. Which is equivalent to applying the non-Clifford gate

$$\Lambda(\pi/6) = \begin{matrix} 1 & 0 \\ 0 & e^{i\pi/6} \end{matrix}$$

Thus, with Cliffords and access to a magic state we are able to implement non-Clifford gates, enabling universal quantum computation.

IV. THE MAGIC STATE FACTORY

There is plenty of literature describing how to obtain fault tolerant clifford operations, but in order to perform the algorithm described above we need access to a source of perfect magic states. The main idea presented in the Bravyi and Kitaev paper is a procedure used to obtain said magic states to arbitrary precision. This procedure was named "Magic State Distillation" by the authors, because it takes as input n states, and outputs one state that is closer to the desired magic state. Below we will describe the procedure tailored to obtain T-Type magic states, such that it is compatible with everything discussed in Section II.

Let us start by defining some notation that will simplify our discussion. We will take as input n identical copies of some state ρ_{in}. We can define our initial error, or the initial distance between the input and the desired magic state as

$$\epsilon = 1 - \langle T| \rho_{in} |T\rangle \tag{6}$$

If we substitute the Bloch vector expression with ρ_{in} we can write the error in terms of the vector components as such.

$$\epsilon = \frac{1}{2}[1 - \frac{1}{\sqrt{3}}(\rho_x + \rho_y + \rho_z)] \tag{7}$$

We will also define the eigenvectors of the T gate as follows

$$T |T_0\rangle = e^{i\pi/3} |T_0\rangle , \qquad T |T_1\rangle = e^{\quad i\pi/3} |T_1\rangle \tag{8}$$

These can be written as a density operator as follows

$$|T_{0,1}\rangle \langle T_{0,1}| = \frac{1}{2}[\mathbb{1} \pm \frac{1}{\sqrt{3}}(\sigma_x + \sigma_y + \sigma_z)] \tag{9}$$

We will also be defining the following set of stabilizers which describe a 5-qubit code which will be used to define a codespace and syndrome measurements.

$$\begin{aligned} S_1 &= \sigma_x \otimes \sigma_z \otimes \sigma_z \otimes \sigma_x \otimes \mathbb{1} \\ S_2 &= \mathbb{1} \otimes \sigma_x \otimes \sigma_z \otimes \sigma_z \otimes \sigma_x \\ S_3 &= \sigma_x \otimes \mathbb{1} \otimes \sigma_x \otimes \sigma_z \otimes \sigma_z \\ S_4 &= \sigma_z \otimes \sigma_x \otimes \mathbb{1} \otimes \sigma_x \otimes \sigma_z \end{aligned} \tag{10}$$

Because all of these stabilizers squared are equal to the identity, their eigenvalues must be -1 and +1. It can also be shown that the projectors to their eigenvectors can be described as

$$|\pm 1^{(\alpha)}\rangle \langle \pm 1^{(\alpha)}| = \frac{\mathbb{1} \pm S_\alpha}{2} \tag{11}$$

Thus, a measuring +1 on all four syndromes would collapse a state $|\psi\rangle$ to

$$\Pi = \frac{1}{16}\Pi_{\alpha=1}^{4}(\mathbb{1} + S_\alpha) \quad (12)$$

By definition, the codespace of a set of stabilizers corresponds to any state that is an eigenvector of all the elements of the set, with eigenvalue +1. This implies that the projector defined above, Π, is a projector onto the codespace.

A. T-Type Magic State Distillation Algorithm

1. Twirling

With equal probability we either: Do nothing; apply a T gate; apply a T^y. This represents the channel

$$D(\rho) = \frac{1}{3}(\rho + T\rho T^y + T^y\rho T) \quad (13)$$

Let us analyze what we obtain from applying this channel to our input.

$|T_0\rangle$ and $|T_1\rangle$ are orthogonal eigenvectors, thus they form an eigenbasis for $\mathcal{H}^2$. If we represent our input in this basis we obtain

$$\rho = a|T_0\rangle\langle T_0| + b|T_1\rangle\langle T_1| + c|T_0\rangle\langle T_1| + d|T_1\rangle\langle T_0| \quad (14)$$

Substitute this into equation 13, and use the fact that quantum channels are linear

$$\begin{aligned} D(\rho) &= aD(|T_0\rangle\langle T_0|) + bD(|T_1\rangle\langle T_1|) \\ &+ cD(|T_0\rangle\langle T_1|) + dD(|T_1\rangle\langle T_0|) \end{aligned} \quad (15)$$

Note that

$$\begin{aligned} D(|T_0\rangle\langle T_1|) &= \tfrac{1}{3}[|T_0\rangle\langle T_1| \\ &+ T|T_0\rangle\langle T_1|T^y + T^y|T_0\rangle\langle T_1|T] \end{aligned} \quad (16)$$

$$\begin{aligned} D(|T_0\rangle\langle T_1|) &= \tfrac{1}{3}[1 + e^{i2\pi/3} + e^{\ \ i2\pi/3}]|T_0\rangle\langle T_1| \\ &= 0 \\ \therefore D(|T_0\rangle\langle T_1|) &= 0 = D(|T_1\rangle\langle T_0|) \end{aligned} \quad (17)$$

From this result we conclude that only the diagonal terms are non-zero after applying the quantum channel. Now we utilize the error symbol ϵ we defined to describe our state as

$$D(\rho) = (1-\epsilon)|T_0\rangle\langle T_0| + \epsilon|T_1\rangle\langle T_1| \quad (18)$$

At this point we have obtained a probabilistic mixture of the magic state we want, and a pure state in an orthogonal direction. We repeat this procedure with a total of 5 states from the initial n copies.

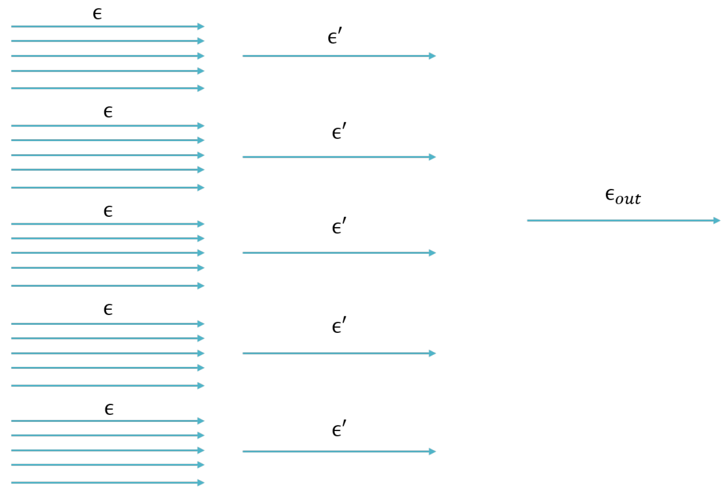

FIG. 2. Example of the application of the distillation scheme recursively, starting with 25 copies of a state with error . These are separated into groups of 5 and each group is distilled separately. Then, the output of each group is grouped together and distilled again. If the initial error is below the accepted threshold ($< \ _0$) then we expect $< \ ' < \ _{out}$[3]

2. Syndrome Measurements

The combined state of the 5 qubits can be described as

$$\rho_{input} = \rho^{\ \ 5} = \text{X}\ \epsilon^{jxj}(1-\epsilon)^{5\ \ jxj}|T_x\rangle\langle T_x| \quad (19)$$

Where x represents a 5 bit string, such that

$$|T_x\rangle = |T_{x1}\rangle \otimes |T_{x2}\rangle \otimes |T_{x3}\rangle \otimes |T_{x4}\rangle \otimes |T_{x5}\rangle, \quad (20)$$

and the sum goes over all possible values of x.

We will now claim that if $\overline{\sigma_\alpha} = \sigma_\alpha{}^{5}$, then $\overline{T} = T^{\ \ 5}$. To prove this, first notice that the relationship between these operators are

$$T\sigma_x T^y = \sigma_z, \qquad T\sigma_y T^y = \sigma_x, \qquad T\sigma_z T^y = \sigma_y \quad (21)$$

From this one can calculate the commutator and find that $[\Pi, T] = 0$. From equation 8, we can conclude that applying a logical T operator onto $|T_x\rangle$ will result in the same vector but multiplied by $e^{i\pi/3}$ for every 0 in x, and multiplied by $e^{\ \ i\pi/3}$ for every 1 in x. Thus we define the hamming weight of x as $|x|$, which by definition contains the number of 1's in x. On the other hand, because x contains 5 bits, $|x|$ - 5 is equal to the number of 0's in x. Using this we obtain

$$\overline{T}|T_x\rangle = e^{\ \ ijxj\pi/3}e^{i(5\ \ jxj)\pi/3}|T_x\rangle = e^{i\pi(5\ \ 2jxj)/3}|T_x\rangle \quad (22)$$

Let us now consider a state

$$|\Phi\rangle = \sqrt{6}\Pi\,|T_{00000}\rangle \tag{23}$$

In the appendix of [1], they show the magnitude of this vector to be non-zero. Thus, $|\Phi\rangle$ lies in our codespace. By using eq. 22 and the fact that $[\Pi, T] = 0$, we obtain

$$\begin{aligned}\hat{T}\,|\Phi\rangle = \sqrt{6}\hat{T}\Pi\,|T_{00000}\rangle = \sqrt{6}\Pi\hat{T}\,|T_{00000}\rangle \\ = e^{5i\pi/3}\Pi\,|T_{00000}\rangle = e^{-i\pi/3}\Pi\,|T_{00000}\rangle = e^{-i\pi/3}\,|\Phi\rangle\end{aligned} \tag{24}$$

There are two important things to notice from this equation. First, is that when we apply $\hat{T}$ to $|\Phi\rangle$ we obtain a vector proportional to $|\Phi\rangle$. Thus showing that $|\Phi\rangle$ is an eigenvector of $\hat{T}$. Second, the eigenvalue is the same as $|T_1\rangle$'s eigenvector in eq. 8. From this we conclude that $|\Phi\rangle$ is equivalent to the logical $|T_1\rangle$ vector

$$|T_1^L\rangle = \sqrt{6}\Pi\,|T_{00000}\rangle \tag{25}$$

Similarly

$$|T_0^L\rangle = \sqrt{6}\Pi\,|T_{11111}\rangle \tag{26}$$

By following the same procedure and using the fact that our codespace can only have two eigenvalues due to its dimensions, Kitaev and Bravyi computed the projections of all $|T_x\rangle$ which were summarized in the following piecewise defined equation

$$\hat{T}\,|T_x\rangle = \begin{cases} 6^{-1/2}\,|T_1^L\rangle & \text{if } |x| = 0, \\ 0 & \text{if } |x| = 1, \\ (5/6)^{-1/2}\,|T_0^L\rangle & \text{if } |x| = 2, \\ (5/6)^{-1/2}\,|T_1^L\rangle & \text{if } |x| = 3, \\ 0 & \text{if } |x| = 4, \\ 6^{-1/2}\,|T_0^L\rangle & \text{if } |x| = 5, \end{cases} \tag{27}$$

With this, we have the necessary mathematical relationships to describe states in terms of our codespace.

The next step in our algorithm is to measure the stabilizers. If any of the four eigenvalues is -1, we discard our 5 copies and start over. On the other hand, if we measure (1,1,1,1), the resulting state will be a projection onto the codespace.

$$\rho_{cs} = \Pi\rho_{input}\Pi \tag{28}$$

$$\rho_{cs} = \sum_x \epsilon^{|x|}(1-\epsilon)^{5-|x|}\Pi\,|T_x\rangle\langle T_x|\,\Pi \tag{29}$$

Plug in the projected operators described in eq.27

$$\begin{aligned}\rho_{cs} = [\tfrac{\epsilon^5+5\epsilon^2(1-\epsilon)^3}{6}]\,|T_0^L\rangle\langle T_0^L| \\ +[\tfrac{(1-\epsilon)^5+5\epsilon^3(1-\epsilon)^2}{6}]\,|T_1^L\rangle\langle T_1^L|\end{aligned} \tag{30}$$

If we normalize the state we obtain

$$\begin{aligned}\rho_{cs}^{'} = [\tfrac{\epsilon^5+5\epsilon^2(1-\epsilon)^3}{a}]\,|T_0^L\rangle\langle T_0^L| \\ +[\tfrac{(1-\epsilon)^5+5\epsilon^3(1-\epsilon)^2}{a}]\,|T_1^L\rangle\langle T_1^L|\end{aligned} \tag{31}$$

where $a = \epsilon^5 + 5\epsilon^2(1-\epsilon)^3 + (1-\epsilon)^5 + 5\epsilon^3(1-\epsilon)^2$

3. Decode

Finally, we take our state out of the codespace by decoding it. The stabilizer framework guarantees the existence of a mapping V such that

$$V\,|\overline{\psi}\rangle = |\psi\rangle \otimes |0,0,0,0\rangle \tag{32}$$

Where $|\overline{\psi}\rangle$ is any codeword in the codespace stabilized by 10.

Apply this to our output state

$$\begin{aligned}V\rho_{cs}^{'}V^{\dagger} = [\tfrac{\epsilon^5+5\epsilon^2(1-\epsilon)^3}{a}]\,|T_0\rangle\langle T_0| \\ +[\tfrac{(1-\epsilon)^5+5\epsilon^3(1-\epsilon)^2}{a}]\,|T_1\rangle\langle T_1|\end{aligned} \tag{33}$$

Observe that this state is closer to $|T_1\rangle$, while we are trying to distill T_0. We now apply $\sigma_y H$, which effectively flips $|T_0\rangle \longrightarrow |T_1\rangle$

We have effectively finished the distillation algorithm. Therefore whatever probability accompanies the state $|T_0\rangle$ corresponds to $1 - \epsilon_{out}$

$$\epsilon_{out} = \frac{\epsilon^5 + 5\epsilon^2(1-\epsilon)^3}{\epsilon^5 + 5\epsilon^2(1-\epsilon)^3 + (1-\epsilon)^5 + 5\epsilon^3(1-\epsilon)^2} \tag{34}$$

It is important to point out that the output error is not necessarily smaller than the input error. Trivially, in order for this algorithm to be of any use, we must obtain a smaller error than we started with ($\epsilon_{out} \leq \epsilon$). We obtain the lower bound by setting $\epsilon_{out}(\epsilon_0) = \epsilon_0$, and find that there is only one solution to this equation at $\epsilon_0 \approx .173$. Giving an error threshold which our n copies must lie below in order for the distillation scheme to be successful. Otherwise, the state becomes closer to the maximally mixed state after distillation.

This algorithm can be recursively implemented as shown in 2 to achieve arbitrary precision, given that the initial input error is below the threshold ($\epsilon < \epsilon_0$). To find the error after k layers of distillation subroutines, one must plug in the output error given in eq. 34 as input error. By doing this we find that

$$\epsilon_{out} \approx \frac{1}{5}(5\epsilon)^{2^k} \tag{35}$$