Stabilizers and Error Correcting Codes

Kuljit S. Virk

These notes describe the stabilizer formalism, and then apply that formalism to illustrate a few quantum error correction codes. I do this in the manner opposite to typical course or a textbook since the broadest conceptual framework is that for the stabilizers. I also find it to be a natural predecessor to the toric codes and the topological quantum computing. So I decided to think that way.

CONTENTS

I. PAULI GROUP

A. One qubit space

The first concept is that of a *Pauli Group* in a single qubit space. It is the group

$$\mathcal{G}_1 = \{\pm I, \pm X, \pm Y, \pm Z, \pm iI, \pm iX, \pm iY, \pm iZ\}\,.$$

I will rely as much as possible on using only the algebra of operators and not their representation. The algebra obeyed by these operators is as follows. First, the square of each operator is either a positive or a negative multiple of identity operator,

$$g^2 = \pm I,\ \ \forall g \in \mathcal{G}. \tag{1}$$

I will denote the set of positive multiples as $\mathcal{G}_1^+ = \pm\{I, X, Z, iY\}$, and the set of operators with the lower sign as $\mathcal{G}_1^- = \pm\{iI, Y, iX, iZ\}$. The operators g are thus their own inverses up to a multiplicative factor,

$$g = \pm g^{-1},$$

where the + sign is for the case where $g^2 = I$. Furthermore, the operators satisfy the relation

$$Y = ZX. \tag{2}$$

Since $Y \in \mathcal{G}_1^-$ while $X, Z \in \mathcal{G}_1^+$, it follows that,

$$\begin{aligned} X^{-1}Z^{-1} &= Y^{-1} = -Y \\ \Rightarrow XZ &= -Y. \end{aligned}$$

Therefore, X and Z anticommute, while their commutator is $2Y$,

$$\begin{aligned} ZX + XZ &= 0, \\ ZX - XZ &= 2Y. \end{aligned}$$

More relations follow straightforwadly. Thus to get the relation between X and Y,

$$\begin{aligned} YX &= Z \\ XY &= -Z \\ \Rightarrow XY + YX &= 0, \\ XY - YX &= 2Z. \end{aligned}$$

Given the relations above, we can also *generate* the group $\mathcal{G}_1$ by only the set $\{\pm X, \pm Z\}$. We will write the generator of the group as the elements $\langle g_i, \dots, g_l \rangle$, such that all elements of the group can be generated by *products* of the elements $g_1 \dots g_l$. We then define the notation $S = \langle g_i, \dots, g_l \rangle$ to mean that S is a subgroup of the Pauli group *generated* by the set $\langle g_i, \dots, g_l \rangle$.

B. Tensor product space

The Pauli group can be defined as operators acting on a Hilbert space of n-qubits, or a 2^n dimensional space as a tensor product

$$g = A_1 \otimes A_2 \otimes \cdots \otimes A_n, \tag{3}$$

where $A_j \in \mathcal{G}_1$ and A_j is acts on the Hilbert space of qubit j. Furthermore

$$[A_j, A_i] = 0, \ \forall j \neq i.$$

Note that the Pauli group $\mathcal{G}_n$ has size 2^{2n+1} where $2n$ arises from the fact that there is a choice of I, X, Y, Z at each qubit, so that there are $4^n = 2^{2n}$ combinations, and an extra 2 arises from the overall ± sign.

We have already understood that each A_j can be expressed as a product of the generators, $\pm X$ and $\pm Z$. It is easy to see that a generic operator in the n-qubit space can be specified with a binary string of length $2n$, $(\alpha_1, \dots \alpha_{2n})$, such that

$$\begin{aligned} g = M(\boldsymbol{\alpha}) &= \left(\prod_{j=1}^{n} Z_j^{\alpha_j}\right)\left(\prod_{j=1}^{n} X_j^{\alpha_{n+j}}\right), \ \alpha_j \in \{0,1\}, \qquad (4) \\ g^2 &= I. \end{aligned}$$

An extra bit can be used to cover the operators $g^2 = -I$, and we will ignore those as the discussion below does not need them. Note that all the products ZX occurs wherever Y occurs in the original form (3), where it is already in the form of Z to the left of X. Since the remaining operators commute, no overall sign arises. I am using $M(\boldsymbol{\alpha})$ to represent the above form of specifying the operator in the Pauli group. In fact, the multiplicative algebra of the Pauli group can be alternatively defined as an algebra of a $2n$ dimensional vector space of binary strings, $\boldsymbol{\alpha}$.

A product of two elements $M(\boldsymbol{\alpha})$ and $M(\boldsymbol{\alpha}')$ is given by

$$\begin{aligned} M(\boldsymbol{\alpha})M(\boldsymbol{\alpha}') &= (-1)^{\boldsymbol{\alpha}^T \Lambda_L \boldsymbol{\alpha}'} M(\boldsymbol{\alpha} + \boldsymbol{\alpha}'), \qquad (5) \\ \Lambda_L &= \begin{pmatrix} 0 & 0 \\ I & 0 \end{pmatrix}, \ \text{n-dim blocks} \qquad (6) \end{aligned}$$

where the matrix I is the identity matrix of dimension n and thus

$$\boldsymbol{\alpha}^T \Lambda \boldsymbol{\alpha}' = \sum_{j=1}^{n} \alpha_{n+j} \alpha_j .$$

This inner product counts the number of times X and Z collide in the two strings. Each such instance has X preceeding Z over the same Hilbert space and thus a minus sign is acquired when the operators are put into the canonical form (4). Similarly,

$$M(\boldsymbol{\alpha}')M(\boldsymbol{\alpha}) = (-1)^{\boldsymbol{\alpha}'^T \Lambda_L \boldsymbol{\alpha}} M(\boldsymbol{\alpha} + \boldsymbol{\alpha}'). \tag{7}$$

Now, we know that all elements of the Pauli group either commute or anti-commute. It follows form (5) and (7) that

$$(-1)^{\boldsymbol{\alpha} \odot \boldsymbol{\alpha}'} = \begin{cases} +1 & [M(\boldsymbol{\alpha}'), M(\boldsymbol{\alpha})] = 0 \\ -1 & \{M(\boldsymbol{\alpha}'), M(\boldsymbol{\alpha})\} = 0 \end{cases}, \tag{8}$$

$$\boldsymbol{\alpha} \odot \boldsymbol{\alpha}' = \boldsymbol{\alpha}^T \Lambda \boldsymbol{\alpha}', \tag{9}$$

$$\Lambda = \begin{pmatrix} 0 & I \\ I & 0 \end{pmatrix}. \tag{10}$$

Thus each $\boldsymbol{\alpha}$ corresponds to a unique operator in the Pauli group, and two operators commute if and only if

$$\boldsymbol{\alpha} \odot \boldsymbol{\alpha}' = 0 \mod 2,$$

and anticommute if and only if,

$$\boldsymbol{\alpha} \odot \boldsymbol{\alpha}' = 1 \mod 2.$$

In this notation, each generator g_l is fully specified with a certain $\boldsymbol{\alpha}_l$. We introduce a $m \times 2n$ matrix of binary entries in which each row represents one of the m generators and the generator g_l corresponds to row l.

$$G_n = \left[\begin{array}{ccc|ccc} \alpha_{11} & \cdots & \alpha_{1n} & \alpha_{1,n+1} & \cdots & \alpha_{1,2n} \\ & \vdots & & & & \\ \alpha_{m1} & \cdots & \alpha_{mn} & \alpha_{m,n+1} & \cdots & \alpha_{m,2n} \end{array} \right]. \tag{11}$$

This is called the *check matrix.*

C. Useful Propositions about Generators

1. Linear Independence

The generators of a group are specified as linearly independent operators, which means that no operator in the generator set can be expressed as a product of the other two. The equivalent statement for bit strings is,

Proposotion 1: Let $S = \langle g_1, \ldots, g_{n-k} \rangle$ be a subgroup of $\mathcal{G}_n$ *such that* $-I \notin S$, *i.e.* $g_j = M(\boldsymbol{\alpha}_j)$. Then g_l are linearly independent if and only if $\boldsymbol{\alpha}_l$ are linearly independent in the additive vector space.

Proof: Since $-I \notin S$, it follows that

$$g^2 = I \ \forall g \in S. \tag{12}$$

$$\Rightarrow g = g^{-1} \tag{13}$$

Furthermore, all $g_i \in S$ must commute with each other. This is so because all g_i in any sub-group of the Pauli group either commute or anti-commute. Suppose that there was a pair i, j that anti-commuted, then

$$g_i g_j = -g_j g_i \Rightarrow (g_i g_j)^2 = -g_i^2 g_j^2 .$$

By assumption of sub-group $g_i g_j \in S$ and thus by (12) $(g_i g_j)^2 = I$. Property (12) also implies that $g_i^2 = g_j^2 = I$, and we obtain a contradiction in the second equality. Thus all operators must commute. Their products thus generate an abelian group.

Now let us assume that $\boldsymbol{\alpha}_i$ are not linearly independent. Then there exists a set of *binary* coefficients c_i such that not all c_i are zero and,

$$\sum_{i=1}^{m} c_i \boldsymbol{\alpha}_i \;=\; 0.$$

Since the right hand side is a binary string representation for the identity matrix, in the operator notation, the above relation corresponds to the product of generators equal to identity,

$$\prod_{i=1}^{m} g_i^{c_i} \;=\; I.$$

Since not all c_i are zero, let us denote by j the index where $c_j = 1$. Then the above equation implies that,

$$\prod_{i \neq j} g_i^{c_i} \;=\; g_j^{-1} = g_j,$$

where the second equality follows from (13). But this contradicts the assumption that all generators in the group are linearly independent. Thus linear independence of the generators implies that all $\boldsymbol{\alpha}_i$ for the generator must also be linearly independent binary vectors.

2. Existence of anti-commutator

Proposition 2a: Let $S = \langle g_1, \ldots, g_{n-k} \rangle$ be a subgroup of $\mathcal{G}_n$ such that $-I \notin S$. Then for every $g_i \in S$, there exists a $g \in \mathcal{G}_n$ such that it satisfies the following two conditions,

$$\begin{aligned} g g_i &= -g_i g, \\ g g_j &= +g_j g, \;\; \forall j \neq i. \end{aligned}$$

Proof: Since S generates a sub-group of the Pauli group, for any given g_i we have a $2n$-dim binary string $\boldsymbol{\alpha}_i$. We define a vector $\boldsymbol{x}$ such that it satisfies

$$G_n \Lambda \boldsymbol{x} \;=\; \boldsymbol{e}_i, \tag{14}$$

where G_n is the check matrix introduced in (11) above and $\boldsymbol{e}_i$ is m-dim vector with 1 at index i and 0 everywhere else. The above equation has a solution because all rows of G_n are linearly independent. We let $g \in \mathcal{G}_n$ be the generator defined by the binary vector $\boldsymbol{x}$. Then, by definition (14),

$$\begin{aligned} \boldsymbol{\alpha}_i \odot \boldsymbol{x} &= 1, \\ \boldsymbol{\alpha}_j \odot \boldsymbol{x} &= 0, \;\; j \neq i. \end{aligned}$$

The last two equations show that g anti-commutes with g_i and commutes with all other generator elements.

This is a very useful result. It says that for every generator of a sub-group of the Pauli group that excludes $-I$, one can always find an element of the Pauli group that anti-commutes with that generator while commutes with every other. This fact is tremendously useful in defining the encoded qubit space below, and also the next proposition.

Proposition 2b: Let $g \in \mathcal{G}_n$ such that $g \neq \pm I$. Then g has as an equal number of $+1$ and -1 eigenvalues.

Proof: The proof follows since by Proposition 2a, we can always find another $h \in \mathcal{G}_n$ such that $gh = -hg$. Thus if a state $|\psi\rangle$ satisfies $g\,|\psi\rangle = |\psi\rangle$, then

$$gh\,|\psi\rangle \;=\; -hg\,|\psi\rangle = -h\,|\psi\rangle\,.$$

Thus the state $h\,|\psi\rangle$ is an eigenstate of g with eigenvalue -1. Thus there is a 1-1 correspondence between states with eigenvalues $+1$ and -1.

3. Dimension of simultaneous eigenspace

Proposition 3: Let $S = \langle g_1, \ldots, g_{n-k} \rangle$ be a subgroup of $\mathcal{G}_n$ *such that* $-I \notin S$. That is, S has $n-k$ generators. Let V_S be the sub-space of the n-qubit Hilbert space defined by the states,

$$V_S \;=\; \{\psi :\; g_l \psi = \psi,\; \forall g_l \in S\}\,. \tag{15}$$

Then, the dimension of V_S is 2^k.

Proof: We first note that since $g_l^2 = I$, the eigenvalues are ± 1. Every g_l thus splits the 2^n-dim Hilbert space into two 2^{n-1}-dim subspaces, one with eigenvalue $+1$ and the other with -1. Now take g_1. The set of states satisfying (15) lives in a dimension 2^{n-1} Hilbert space. Then take g_2. In order for both g_1 and g_2 to have eigenvalue $+1$, we must select states only from the $+1$ eigenspace of g_1. Thus the Hilbert space is now of dimension 2^{n-2}. When we reach the generator $n-k$, and since they are all linearly independent, the dimension of V_S is $2^{n-(n-k)} = 2^k$.

Note that by Proposition 3, there are a maximum of n generators for n-qubit space. This is clear since with n generators, the vector space V_S becomes one-dimensional ($2^{n-n} = 1$). One dimensional simultaneous eigenspace is possible, and there can be no smaller dimension and thus no more generators.

II. STABILIZERS, NORMALIZERS AND ENCODED QUBITS

A. Stabilizers and Normalizers

We can now define the concept of a stabilizer. It is actually already defined in (15), but we state it formally here. Let $S = \langle g_1, \ldots, g_m \rangle$ be a sub-group of the Pauli group with the property that $-I \in S$. Then define a vector space V_S as follows,

$$V_S \;=\; \{|\psi\rangle :\; g_l\,|\psi\rangle = |\psi\rangle\,,\; \forall g \in S\}\,.$$

The space V_S is said to be *stabilized* by the group S, and S is called the *stabilizer* of V_S. It is actually not too difficult to imagine how a state in V_S can be realized. A simple prescription is to pick any random state in the computational basis (so that it can be prepared in polynomial time), and define an equal superposition over the stabilizer group elements,

$$|\psi_S\rangle \;=\; \sum_{g \in S} g\,|\psi_0\rangle\,.$$

Note that this sum can be zero if the state ψ_0 splits the set S into an equal number of elements with eigenvalues $+1$ and -1. Therefore, this prescription requires choosing a good ψ_0 in the first place. We do not want to choose states that get annihilated under this sum.

Assuming $|\psi_S\rangle \neq 0$, we can easily see that S stabilizes $|\psi_S\rangle$, we do an explicit test with a generator $g_j \in \langle g_1, \ldots, g_m \rangle$,

$$g_j\,|\psi_S\rangle \;=\; \sum_{g \in \mathcal{G}_S} g_j g\,|\psi_0\rangle\,.$$

Since the group S is closed under multiplication, the summation of $g_j g$ is only a re-arrangement of the group elements. Thus we see that the state is stabilized, *i.e.*

$$g_j\,|\psi_S\rangle \;=\; \sum_{g' \in \mathcal{G}_S} g'\,|\psi_0\rangle = |\psi_S\rangle\,.$$

It is clear then that if the technology exists to apply the generators of the stabilizer, then the above operation can be continually applied to the state of the n-qubit system and it will remain within the vector space V_S. The rate of the refresh operation must be faster than any relevant decoherence times. The space V_S may not be known explicitly, but only the effect of operations by the Pauli group operators is needed to perform computations on it. We will now describe how this is accomplished. However, first we describe a larger sub-group than the stabilizer itself, defined as follows.

Definition: A *normalizer* of a sub-group S, denoted $S^\perp$ is the set of all operators in the Pauli group that commute with S. Thus in the operator notation,

$$S^\perp \; = \; \{g \in \mathcal{G}_n : \; gh = hg, \; \forall h \in S\} \, . \tag{16}$$

Note that if $-I \notin S$, then $S \subseteq S^\perp$ since in that case S is abelian and all its elements commute. However, $S^\perp$ is larger than S since it is possible to find operators that commute with all operators in S but have eigenvalue -1 on $|\psi\rangle \in V_S$. Discarding the negative Pauli operators, $n-k$ generators of S leave a total of $2n-(n-k) = n+k$ generators that commute with S. This follows simply from the fact that there are only $2n$ linearly independent binary strings of length $2n$. Therefore, $n-k$ binary strings of length $2n$ imply the existence of $n+k$ linearly indepdenent strings. Therefore the set $S^\perp$ has $n+k$ independent generators.

B. Encoded Qubits

Of the $n+k$ generators in the previous subsection, we can select $n-k$ to define S itself since $S \subseteq S^\perp$. This leaves $n+k-(n-k) = 2k$ generators. By Proposition 3, we can add up to k generators to S and make the generating set of size n. These k operators commute with each other by definition of their admissibility into S. The remaining k operators must anticommute with the chosen k. Thus the $2k$ operators can be split into two sets of size k. All operators within each set commute with each other, while anticommute with at least one operator from the other set.

We call the first set $\bar{Z}_1, \ldots \bar{Z}_k$. By definition of $S^\perp$, all these operators leave V_S invariant since $g\bar{Z}_j \, |\psi\rangle = \bar{Z}_j g \, |\psi\rangle = \bar{Z}_j \, |\psi\rangle$ for every $g \in S$. The operators $\bar{Z}_j^2 = I$ and thus have eigenvalues ± 1. Since the operators leave V_S invariant, both eigenspaces have eigenvalue $+1$ for $g \in S$.

The discussion in the previous section shows that $\bar{Z}_j^2$ have an equal number of positive and negative eigenvalues. We thus label the states of V_S as $|\bar{z}_1, \ldots, \bar{z}_k\rangle$ where $\bar{z}_j = \pm 1$ and,

$$\bar{Z}_j \, |\bar{z}_1, \ldots, \bar{z}_k\rangle \; = \; \bar{z}_j \, |\bar{z}_1, \ldots, \bar{z}_k\rangle \, .$$

We now define the remaining k operators as $\bar{X}_1, \ldots, \bar{X}_k$. We know from Proposition 2a that for every $\bar{Z}_j$ we can find an opearator $\bar{X}_j$ that anti-commutes with $\bar{Z}_j$ and commutes with the remainig set $S + \left\{\bar{Z}_i\right\}_{i \neq j}$. With this prescription, we find k operators $\bar{X}_i$ such that

$$\bar{Z}_j \bar{X}_i \; = \; (-1)^{\delta_{ij}} \, \bar{X}_i \bar{Z}_j .$$

Because of this property, we note that if there is a state $|\psi\rangle \in V_S$ such that $\bar{Z}_j \, |\psi\rangle = |\psi\rangle$, then we see that $\bar{X}_j$ flips the state into -1 eigenvalue state of $\bar{Z}_j$,

$$\bar{Z}_j \left[\bar{X}_j \, |\psi\rangle\right] \; = \; -\bar{X}_j \bar{Z}_j \, |\psi\rangle = -\bar{X}_j \, |\psi\rangle \, .$$

It is also clear that $\bar{Z}_j^2 = \bar{X}_j^2 = I$. The anti-commuting property defines $\bar{Y}_j = \bar{Z}_j \bar{X}_j$ so that

$$\begin{aligned} \bar{X}_j \bar{Y}_j + \bar{Y}_j \bar{X}_j \; &= \; -\bar{X}_j \bar{X}_j \bar{Z}_j + \bar{Z}_j \bar{X}_j \bar{X}_j = 0. \\ \bar{X}_j \bar{Y}_j - \bar{Y}_j \bar{X}_j \; &= \; -2\bar{Z}_j . \end{aligned}$$

Thus the operators $\bar{Z}_j, \bar{X}_j$ generate the algebra of the Pauli group and all computations that are encoded with Pauli matrices on 1 and 2-qubits can be defined via these encoded operators on a multi-qubit system with the state restriced to lie inside V_S.

C. Expressing Encoded Operators

In this section, we follow a recipe in Nielsen and Chuang's book to specify $\bar{Z}_j$ and $\bar{X}_j$ operators that act like k single qubit operators on a Hilbert space of dimension 2^k. We first write the check matrix in the following form, where G_Z and G_X are $(n-k) \times k$ matrices describing $n-k$ generators in 2^n-dim Hilbert space,

$$G \; = \; \left[\; G_Z \;\; G_X \; \right] .$$

Perform Gaussian elimination on G_x (this is a modulo 2 arithmetic), to obtain the form

$$G \;=\; \begin{array}{c} \left[\begin{array}{ccccc} B & C & | & I & A \\ D & E & | & 0 & 0 \end{array}\right] \\ \underbrace{\;\;}_{r}\underbrace{\;\;\;}_{n-r,}\underbrace{\;\;}_{r}\underbrace{\;\;\;}_{n-r} \end{array} \quad \begin{array}{l} \}r \\ \}n-k-r \end{array} .$$

Here r is the rank of G_X, Next we perform Gaussian elimination on E, and let the rank of that be $n-k-r-s$ for some $s \geq 0$.

$$G \;=\; \left[\begin{array}{cccccccc} B & C'' & C' & | & I & A_1 & A_2 \\ D' & I & E' & | & 0 & 0 & 0 \\ D'' & 0 & 0 & | & 0 & 0 & 0 \end{array}\right] \begin{array}{l} \}r \\ \}n-k-r-s \; . \\ \}s \end{array}$$

Here the number of columns are $r, n-k-r, k$ in the Z sector, and we have similarly split the X sector. If $D'' \neq 0$, then for some qubits, it will collide with the $r \times r$ identity matrix I in the X sector of the matrix (the first n columns). Since I has only one operator per generator, and D'' also has only r possibly non-zero locations, the last s generators will not commute with the first r. As a result $D'' = 0$, and we can thus set $s = 0$.

$$G \;=\; \left[\begin{array}{ccccccc} B & C'' & C' & | & I & A_1 & A_2 \\ D' & I & E' & | & 0 & 0 & 0 \end{array}\right] \begin{array}{l} \}r \\ \}n-k-r \end{array} .$$

The matrix block in the X sector has the overall shape of $n-k$ rows and n columns. Therefore, its rank is at most $n-k$. Similarly, the rank of the matrices B, C', C'' is at most r. We can choose to take combinations of rows of these three matrices to set one of them to zero. Note that any combinations of rows in G_X with another set of rows in G_Z is still a valid stabilizer since it is just another combination of tensor products of $2n$ operators. So we can choose rows only on the X sector and set $C'' = 0$, for example. Thus the following form is obtained (removing primes),

$$G \;=\; \left[\begin{array}{ccccccc} B & 0 & C & | & I & A_1 & A_2 \\ D & I & E & | & 0 & 0 & 0 \end{array}\right] \begin{array}{l} \}r \\ \}n-k-r \end{array} .$$

Let us now add k rows and the bottom of G, making it $n \times 2n$ matrix,

$$G \;=\; \left[\begin{array}{ccccccc} B & 0 & C & | & I & A_1 & A_2 \\ D & I & E & | & 0 & 0 & 0 \\ A_2^T & 0 & I & | & 0 & 0 & 0 \end{array}\right] \begin{array}{l} \}r \\ \}n-k-r \; . \\ \}k \end{array}$$

The matrix A_2^T is $k \times r$ matrix as needed, and the matrix I is $k \times r$. The operators corresponding to the last k rows obviously commute with those above them in the Z sector. The matrix A_2^T contains up to r different Z operators in tensor product, and it has k such products. To see if the tensor products of $(A_2^T, 0, I)$ commute with the operators in G_X, we have to take inner products between the n-dimensional rows of both matrices. A convenient way to write this is

$$\begin{aligned} Q &= I \times (A_2^{T)^T} + A_2 \times I \mod 2 \\ &= A_2 + A_2 \mod 2 \\ &= 0. \end{aligned}$$

This shows that the k new generators from the tensor products of Z operators commute with the stablizers. A similar construction can be performed on the X side, and the encoded operator set is formed.

D. Gottesman-Knill

I looked at some notes online for Gottesman Knill theorem, and a paper by Gottesman. I understand how this theorem comes about. First, we restrict to a state that is one of the computational basis states. This is important so that the state preparation is not exponential in complexity. The state is assumed to have associated with it a stabilizer. Thus the state is actually a vector space of states that transform trivially under the elements of the stabilizing group. This of course is nothing but an error-correction scheme.

Now, the logical states are defined by the action of the stabilizing group: $g\,|f\rangle = |f\rangle$.

When the state evolves by unitary matrix, U, then: $U\,|f\rangle = Ug\,|f\rangle = (UgU^{\dagger})U\,|f\rangle$.

Thus the evolved state is now described by the similarly evolved stabilizers of the initial state. The stabilizer group can be compactly defined by its generators, which are usually log(group size). If it happens that the unitary matrix U is a normalizer of the Pauli group, then it simply rearranges the group elements. No matrix outside of the Pauli group ever emerges in the evolution. Therefore, the state never maps to a state that is not described by 2n numbers giving the powers (0, or 1) of X and Z operators acting on the n qubits. Thus the evolution is captured within a fixed dimension. There are O(n^2) operations to update the operators of the stabilizer group.

The evolution matrices that are normalizer for the Pauli group contain Hadamard, CNOT and the pi/2 gate. These operators obviously are capable of generating some entaglement. Thus there is a class of entanglements that is easily handled with classical book-keeping of the rearrangement of stabilizer group due to the evolution as a function of time.

On the other hand, a Toffoli gate, or a pi/4 gate maps, X --> aX + bY, and it no longer belongs to the Pauli group. The state is not described by a product of elements of the Pauli group (is that obvious?). The presence of this gate in a -quantum circuit will quickly walk a state outside of the realm of classical description keeping track of 2n numbers.

III. ANCILLA, ENTANGLEMENT AND ENTROPY

[04/25]: In the morning, till 9 AM, I studied Preskill's notes on the error correction conditions. This time, it was natural to see how ancilla plays the necessary role of swapping entanglement of environment and qubit with the entanglement between ancilla and qubit so that the entropy rise in the environment is the same. I will follow this further and understand this more deeply.