MATH 3952 (Undergraduate Seminar): Quantum Information Theory **Spring 2024**

Lecture 7, Talk 2: March 18, 2024

Organizer: Patrick Lei; Presenter: Sam Kim *Scribe: Mark Chen*

Chapter 7: Stabilizers (Ctd.)

1 Normal Subgroups

Definition 1 (Normal Subgroup)**.** *Let $H \leq G$ be groups. $H \trianglelefteq G$ (said H is a normal subgroup of G) if H is invariant under conjugation by all elements of G, i.e. $ghg^{-1} \in H, \forall g \in G, h \in H$.*

Definition 2 (Left Cosets (right is defined analogously))**.** *A left coset of H is $gH = \{gh : h \in H\}$ for some fixed $g \in G$.*

Proposition 1. *Let $g_1, g_2 \in G$. Then, g_1H and g_2H are either **identical** or **disjoint**.*

Proof. Proof is given in algebra I, and can be quite a tedious one. The idea is, suppose g_1H and g_2H are not disjoint, say they share some common elements, then you can show that it will lead to the two being identical. ■

Proposition 2. *If $H \trianglelefteq G$ also, then $gH = Hg$.*

Proof. Normal subgroup means $ghg^{-1} \in H \forall h \in H, g \in G$, then $gHg^{-1} \subseteq H \implies (gHg^{-1})\, g \subseteq Hg \implies \boxed{gH \subseteq Hg}$. On the other hand, $g^{-1}hg \in H$, because $g^{-1} \in G$, so $h \in gHg^{-1} \implies H \subseteq gHg^{-1} \implies \boxed{Hg \subseteq gH}$. By two-way containments, we have $gH = Hg$, indeed. ■

Definition 3 (Quotient Groups)**.** *With proposition 2, we can define the quotient group G/H, which consists of cosets with the operation defined by*

$$g_1H \cdot g_2H = (g_1g_2)H$$

[due to left and right cosets' equality].

Theorem 1 (Lagrange's Theorem)**.** *Let $H \leq G$, and denote the number of cosets of G given by H by $|G : H|$, we have*

$$|G| = |G : H| \cdot |H|.$$

Proposition 3. *Given an arbitrary subgroup $H \leq G$, we can construct a larger subgroup $K \leq G$ in which H is normal, i.e.*

$$H \trianglelefteq K \leq G.$$

Definition 4 (Normalizer)**.** *As pointed out in 3, normal supergroups, K, exist between H and G. The largest such K in G is called the normalizer and is denoted*

$$N_G(H).$$

Proposition 4. *It is NOT TRUE that $H \trianglelefteq K, K \trianglelefteq G \implies H \trianglelefteq G$.*

2 Pauli Normalizers

As we will see next, Pauli stabilizers are not normal subgroups of $\mathcal{P}_n$, so we instead study their normalizers. Firstly, let's see that, after choosing a stabilizer $\mathcal{S}$, there are two subgroups of $\mathcal{P}_n$ that pop up:

Definition 5 (Centralizer)**.** *The centralizer is the set of all elements in $\mathcal{P}_n$ that commutes with every element of $\mathcal{S}$:*

$$Z_G(\mathcal{S}) = \{g \in \mathcal{P}_n : gsg^{-1} = s, \forall s \in \mathcal{S}\}$$

Notice how this is similar to the normalizer but more rigid (though generally distinct):

$$N_G(\mathcal{S}) = \{g \in \mathcal{P}_n : sgs^{-1} \in S, \forall s \in \mathcal{S}\}$$

Proposition 5. *In the context of Pauli groups, $Z_G(\mathcal{S}) = N_G(\mathcal{S})$.*

Proof. Notice that, by the definition of $N_G(\mathcal{S})$ ($gsg^{-1} = s'$), it becomes $Z_G(\mathcal{S})$ only when $s' = s$ for all $s \in \mathcal{S}$ as well. But, any two elements of the Pauli group either commute (same or with $\mathbb{1}$) or anticommute (otherwise), so $s' = s$ or $s' = -s$. But, if $s \in \mathcal{S}$, it is a stabilizer, so it cannot be the case that $s' = -s$ (exercise to show). So, $s' = s$, which means that $Z_G(\mathcal{S}) = N_G(\mathcal{S})$. ■

Proposition 6. *We have two normal subgroups:*

- $\mathcal{S} \trianglelefteq N(\mathcal{S})$.

  Proof. By definition. ■

- $N(\mathcal{S}) \trianglelefteq \mathcal{P}_n$.

  Proof. Let $g \in \mathcal{P}_n$ be arbitrary. Then, consider gng^{-1} for some $n \in N(\mathcal{S})$:

  $$(gng^{-1})s = s(gng^{-1}),$$

  because s either commute with both g and g^{-1} or anticommute with both, in which case the two minus signs would cancel out anyways. Therefore,

  $$s^{-1}(gng^{-1})s = gng^{-1} \implies gng^{-1} \in N(\mathcal{S}).$$

  ■

- *We can also show that S is not a normal subgroup of $\mathcal{P}_n$.*

  Proof. Pick any $g \in \mathcal{P}_n$ that anticommute with some $s \in \mathcal{S}$, such that $gsg^{-1} = -s$. Since we already know that $s \in \mathcal{S} \implies -s \notin \mathcal{S}$, so $gsg^{-1} \notin \mathcal{S}$, which shows what we wanted to show. ■

Proposition 7. *Since we have shown that $\mathcal{S} \trianglelefteq N(\mathcal{S}) \trianglelefteq \mathcal{P}_n$, we have two quotient groups: $\underline{\mathcal{P}_n/N(\mathcal{S})}$ and $\underline{N(\mathcal{S})/\mathcal{S}}$.*

Remark 1. *The way how Pauli stabilizer slices its normalizer into cosets, and its normalizer in turn slices the Pauli group into cosets is very useful for **quantum error correction** and **fault tolerance**.*

Here, we explain how it works:

- *The stabilizer will partition the Hilbert space of n qubits into subspaces, then the one that is fixed by the stabilizer will be chosen as a **codespace**.*
- *All operators in the normalizer will then become **logical operators** on the codespace.*
- *The cosets of the normalizer in $\mathcal{P}_n$ will group together operators that describe errors of a similar type (those with the same **error syndrome**).*
- *It will be a useful fact to know that $\mathcal{P}_n/N(\mathcal{S})$ is abelian (exercise).*

Proposition 8. *We know that* $|\mathcal{P}_n| = 4^{n+1}$ *and that* $|\mathcal{S}| = 2^r$*. Then, we have:*

- $|N(\mathcal{S})| = \boxed{4 \cdot (4^n/2^r)}$.

  Proof. By definition, normalizer consists of all the operators that commute with all the generators of the stabilizer. There are

  $$4 \cdot (4^n/2)$$

  that commute with the first generator, half of which also commute with the second generator, a further half of which also commute with the third generator, and so on. So, considering all r generators, we have

  $$|N(\mathcal{S})| = 4 \cdot (4^n/2^r)$$

  ■

- $|\mathcal{P}_n/N(\mathcal{S})| = \frac{4^{n+1}}{4 \cdot (4^n/2^r)} = \boxed{2^r}$.

  Proof. This and next by Lagrange's theorem. ■

- $|N(\mathcal{S})/\mathcal{S}| = \frac{4 \cdot (4^n/2^r)}{2^r} = \boxed{4^{n-r+1}}$.

3 Clifford Walks on Stabilizer States

We have so far seen two ways of defining stabilizer states of n qubits. Now, we introduce the third.

We can describe the n-qubit stabilizer states as the states that are reachable from the $|0\rangle^{\otimes n}$ state using only the CNOT gate, the H gate, and the phase gate $S = \begin{bmatrix} 1 & 0 \\ 0 & i \end{bmatrix}$.

Remark 2. *Using these three gates tend to let us end up in a discrete state, never anything in between them.*

Proposition 9. *Circuits composed of only* CNOT*,* H *and* $S = P_{\pi/2}$ *are in effect unitaries that map stabilizer states to stabilizer states.*

Definition 6 (Clifford group)**.** *The n-qubit **Clifford group** $\mathcal{C}_n$ is the group generated by these three unitaries (CNOT, H and $S = P_{\pi/2}$), and it happens to be exactly the normalizer of the n-qubit Pauli group inside the group of all $(2^n \times 2^n)$ unitary matrices:*

$$\mathcal{C}_n = \{U \in U(2^n) \mid UPU^\dagger \in \mathcal{P}_n, \forall P \in \mathcal{P}_n\} =: N_{U(2^n)}(\mathcal{P}_n).$$

Proposition 10. *Suppose we have some vector space V stabilized by the group $\mathcal{S}$, and we apply some unitary operation U. For $|\psi\rangle$ as an arbitrary element of V, and $\forall S \in \mathcal{S}$,*

$$U|\psi\rangle = US|\psi\rangle = US(U^\dagger U)|\psi\rangle = (USU^\dagger)U|\psi\rangle,$$

so $U|\psi\rangle$ is stabilized by $USU^\dagger$.

Proposition 11. *From this, we can deduce that the vector space:*

$$UV := \{U|\psi\rangle \mid |\psi\rangle \in V\}$$

is stabilized by the group

$$USU^\dagger := \{USU^\dagger \mid S \in \mathcal{S}\}.$$

Proposition 12. *Furthermore, if $G_1, \ldots, G_r$ generate $\mathcal{S}$, then $UG_1U^\dagger, \ldots, UG_rU^\dagger$ generate $U\mathcal{S}U^\dagger$, so to compute the change in the stabilizer we need only compute how it affects the generators of the stabilizer. But, the Clifford group is generated only by three elements, so we can just work out how these gates act by conjugation on the Pauli group:*

- *We already know how H works under conjugation:*

$$HXH = Z, HZH = X, HYH = i(HXH)(HZH) = iZX = -Y.$$

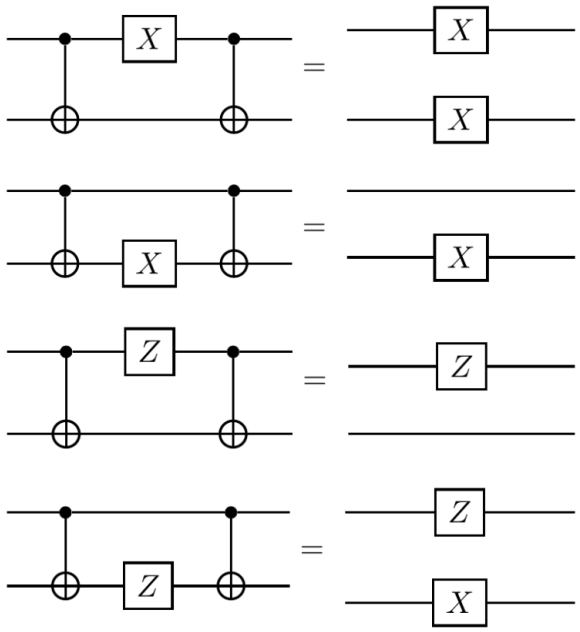
- *We can use a briefer notation that describes the conjugation map of the gate S:*

$$\begin{bmatrix} X \mapsto Y \\ Y \mapsto -X \\ Z \mapsto Z \end{bmatrix}.$$

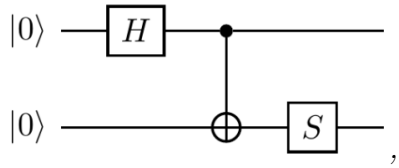
- *... ad of the gate CNOT:*

$$\begin{bmatrix} \mathbb{1}X \mapsto \mathbb{1}X \\ X\mathbb{1} \mapsto XX \\ \mathbb{1}Y \mapsto ZY \\ Y\mathbb{1} \mapsto YX \\ \mathbb{1}Z \mapsto ZZ \\ Z\mathbb{1} \mapsto Z\mathbb{1} \end{bmatrix}.$$

From these, we can express them as circuit identities, e.g.,



Example 1. *To see how these rules work in practice, here is a simple stabilizer circuit:*



through which we embark on a Clifford walk between 2-qubit stabilizer states:

$$\begin{aligned} |00\rangle &\overset{H\otimes\mathbb{1}}{\mapsto} \frac{1}{\sqrt{2}}(|0\rangle + |1\rangle)|0\rangle \\ &\overset{\textsc{cnot}}{\mapsto} \frac{1}{\sqrt{2}}(|00\rangle + |11\rangle) \\ &\overset{\mathbb{1}\otimes S}{\mapsto} \frac{1}{\sqrt{2}}(|00\rangle + i|11\rangle), \end{aligned}$$

which can be described in terms of stabilizer generators:

$$\begin{vmatrix} Z & \mathbf{1} \\ \mathbf{1} & Z \end{vmatrix} \overset{H\otimes\mathbf{1}}{\longmapsto} \begin{vmatrix} X & \mathbf{1} \\ \mathbf{1} & Z \end{vmatrix} \overset{\textsc{c-not}}{\longmapsto} \begin{vmatrix} X & X \\ Z & Z \end{vmatrix} \overset{\mathbf{1}\otimes S}{\longmapsto} \begin{vmatrix} X & Y \\ Z & Z \end{vmatrix}.$$

Here, the first column corresponds to the first qubit, and the second column to the second qubit, which is how the gate actions make sense, based on the mapping specified in proposition 12.

Theorem 2 (**Gottesman-Knill Theorem**). *Gottesman-Knill Theorem states that stabilizer circuits with only gates from the normalizer of the qubit Pauli group, known as the Clifford group, can be perfectly simulated in polynomial time on a probabilistic classical computer. Note that:*

- *We can also efficiently compute the expectation values of any physical observables by examining the updated list of stabilizers.*
- *Computing a list of amplitudes would not be efficient, since there are exponentially many of them.*

Remark 3. *Because stabilizer circuits can be classically simulated, they necessarily do not capture the full power of quantum computation. Specifically, fully universal quantum computation requires at least one non-Clifford gate, such as the* $T = \begin{bmatrix} 1 & 0 \\ 0 & e^{-i\pi/4} \end{bmatrix}$ *gate. With it, we can create circuits that will take us from any initial state, such as ours in this case* $|0\rangle^{\otimes n}$*, to arbitrarily close to any other state in the n-qubiti Hilbert space.*

Remark 4. *Stabilizer circuits, regardless of its limitations, is central to QC, because of its role in* ***quantum error correction*** *and* ***fault-tolerant computation****. Almost all of the quantum error correcting codes are stabilizer codes, and are presented using the stabilizer formalism.*