CSE 291 / Math 277A - Quantum Complexity Theory	November 3, 2022
Lecture 12	
Lecturer: Daniel Grier	*Scribe: Laura Shea*

1 Warmup

Question 1. *Which of the following are valid stabilizer groups for some 2-qubit state?*

$$(\mathbf{1})\begin{pmatrix} I\otimes I \\ X\otimes I \\ Z\otimes I \\ I\otimes Y\end{pmatrix} \qquad (\mathbf{2})\begin{pmatrix} I\otimes I \\ X\otimes X \\ Y\otimes Y \\ Z\otimes Z\end{pmatrix} \qquad (\mathbf{3})\begin{pmatrix} I\otimes I \\ X\otimes X \\ Y\otimes Y \\ -Z\otimes Z\end{pmatrix}$$

Both **(1)** and **(2)** are not valid stabilizer groups. For **(1)**, if we multiply $(X\otimes I)(Z\otimes I) = (XZ)\otimes I = (-iY)\otimes I$, we get a result $(-iY)\otimes I$ that is not in **(1)**, so it is not a group. For a different justification, we could use the fact that if P, Q are stabilizers for some state, then P, Q commute. Since the Pauli elements X and Z anti-commute, $X\otimes I, Z\otimes I$ cannot be in the same stabilizer group.

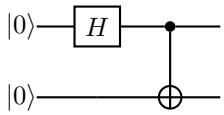
In **(2)**, we have both the elements $Z\otimes Z$ and $(X\otimes X)(Y\otimes Y) = (XY)\otimes(XY) = (iZ)\otimes(iZ) = -Z\otimes Z$. Since $Z^2 = I$, we also have $-I\otimes I$ in the group, but $-I\otimes I$ clearly does not stabilize any state. Thus **(2)** is not a stabilizer group.

(3) is a valid stabilizer group because all of the multiplications do work out. All of the elements squared yield $I\otimes I$, and any element times $I\otimes I$ remains in the group. We then have six nontrivial multiplications to consider: $(X\otimes X)(Y\otimes Y) = -Z\otimes Z = (Y\otimes Y)(X\otimes X)$, $(X\otimes X)(-Z\otimes Z) = Y\otimes Y = (-Z\otimes Z)(X\otimes X)$, and $(Y\otimes Y)(-Z\otimes Z) = X\otimes X = (-Z\otimes Z)(Y\otimes Y)$. In general, a set is a stabilizer group for some n-qubit state if it contains 2^n elements that all commute, and doesn't contain $-I$.

Question 2. *What is the stabilizer group for the 2-qubit cat state:* $\frac{|00\rangle+|11\rangle}{\sqrt{2}}$ *?*

(From now on we will drop imlicit tensor products from our notation. For example, the elements of **(3)** above would be written as II, XX, YY, and $-ZZ$.)

Recall that the following circuit yields the 2-qubit cat state:



We will determine the stabilizer group of the 2-qubit cat state by starting with the stabilizer group of $|00\rangle$, then applying the conjugations associated to H and CNOT. We will need the following three facts. First, the stabilizer group of $|00\rangle$ is

$$\begin{pmatrix} II \\ IZ \\ ZI \\ ZZ\end{pmatrix}.$$

Second, when we apply a Hadamard gate on the ith qubit, we conjugate the ith Pauli matrix in each stabilizer. Specifically, for Pauli P we have $P \mapsto HPH^\dagger$, which is given by $\{X\mapsto Z, Y\mapsto -Y, Z\mapsto X\}$. Third, when we apply CNOT from the ith qubit to the j qubit, we conjugate the ith and jth Pauli terms in the stabilizer. The conjugation map $P\otimes Q \mapsto \text{CNOT}(P\otimes Q)\text{CNOT}^\dagger$, where P is on the control and Q is on the target, is given by $\{XI\mapsto XX, IX\mapsto IX, ZI\mapsto ZI, IZ\mapsto ZZ\}$. (The remaining CNOT rules can be determined from these four, but these will be enough for our example.)

Now we are ready to compute the stabilizer group of the 2-qubit cat state. We start with the stabilizer group of $|00\rangle$ then apply H to the first qubit and I to the second qubit:

$$\begin{pmatrix} II \\ IZ \\ ZI \\ ZZ \end{pmatrix} \xrightarrow{H\otimes \mathrm{I}} \begin{pmatrix} II \\ IZ \\ XI \\ XZ \end{pmatrix},$$

then apply CNOT with the first qubit as the control and the second as the target:

$$\begin{pmatrix} II \\ IZ \\ XI \\ XZ \end{pmatrix} \xrightarrow{\mathrm{CNOT}} \begin{pmatrix} II \\ ZZ \\ XX \\ -YY \end{pmatrix}.$$

Note that in this last step, in addition to the CNOT rules listed above, we have used $XZ = (XI)(IZ) \mapsto (XX)(ZZ) = -YY$. In fact, the generators of the stabilizer group for the 2-qubit cat state are only ZZ and XX; we can get II and $-YY$ from those two.

We have answered Question 2, but is there a general way to compute the stabilizer group for an arbitrary state? The answer is that it depends on the representation of the state. If we know a circuit that yields the state, then we can use the above method to derive the stabilizer groups starting from the all-zero states.

2 Constant-depth circuits

We now consider the question: Are constant-depth quantum circuits better than constant-depth classical circuits? This is motivated by the observation that, while we have mostly considered BQP until now, it is still far from the model of quantum computation today. In particular, today's quantum computers are lossy. There is noise associated with every operation and, as we saw in Homework 1, the errors add up linearly. If we apply 100 gates, each with error 0.01, can we still do useful computations? The answer is that we can perform quantum error correction by building redundancy into circuits. This results in needing on the order of 1000 physical qubits to implement one logical qubit, while today's quantum computers have on the order of 100 physical qubits. Thus it is reasonable to consider arbitrarily large quantum circuits, restricted to constant depth. We might also hope to more easily prove statements in this model.

2.1 NC^0 and QNC^0

Definition 3 (NC^0)**.** *The class of languages L such that there exists a uniform family of constant-depth classical circuits $C_n\colon \{0,1\}^n \to \{0,1\}$ built from* AND*,* OR*, and* NOT *gates with bounded fanin where $x \in L$ if and only if $C_n(x) = 1$.*

Definition 4 (QNC^0)**.** *The class of languages L such that there exists a uniform family of constant-depth quantum circuits $\{Q_n\}_{n=1}^{\infty}$ built from a universal family of 1- and 2-qubit gates where if $x \in L$ the probability of measuring 1 on the first qubit of $Q_n |x\, 0 \cdots 0\rangle$ is $\geq 2/3$ (where x is the input and the rightmost 0's are ancillas), and if $x \notin L$ the probability of measuring 1 on the first qubit is $\leq 1/3$.*

Given these two definitions for constant-depth decisional circuits, we can now ask if there is a language in QNC^0 that is not in NC^0. The answer is no!

To explain why any language in QNC^0 is also in NC^0, we will use a light-cone argument. Suppose we have a quantum circuit Q with depth at most a constant d, and we only measure the first qubit at the end. There is a light cone of input qubits that could possibly affect the first output qubit, and this light cone also has depth at most d. Then at most 2^d input qubits can affect our final measurement — where 2^d is another constant. (Note that the base 2 is coming from the definition of QNC^0 being based on 1- and 2-qubit gates only.) Now we can simply classically simulate the constant-size computation over these 2^d qubits.

Thus NC^0 and QNC^0 do not give us a separation. To obtain a separation, we will need to further modify our question. The key point above is that the behavior of a constant-depth quantum circuit is completely simulatable if we only measure the first output qubit in decision problems; however, we can try considering relation problems instead, which may require simulating the outcomes of measuring multiple output qubits.

2.2 Relational FNC^0 and FQNC^0

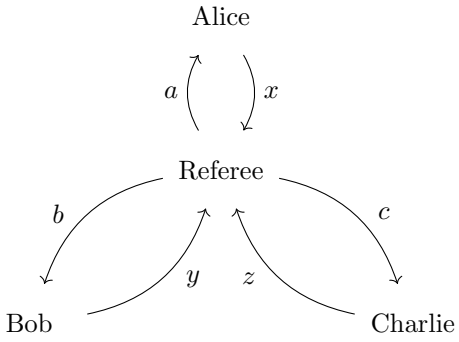
Relational NC^0 and QNC^0 are also called FNC^0 and FQNC^0, respectively. Recall that for relation problems, for every input x, there is a set S_x such that on input x we want to output something in S_x. Concretely, FNC^0 is the class of relations $R \subseteq X \times Y$ where $C_n(x)$ outputs y such that $(x, y) \in R$. (The remainder of Definition 3 is unchanged.) Similarly, FQNC^0 is a class of relations $R \subseteq X \times Y$ where $Q_n |x\, 0 \cdots 0\rangle$ outputs y such that $(x, y) \in R$, for the measurement probabilities as in Definition 4. In particular, for FQNC^0, we need to consider all output qubits, not only the first.

Our new question is whether there is a relation in FQNC^0 that is not in FNC^0, and the answer turns out to be yes! It was proved in 2017 (and published in 2018) by Bravyi, Gosset, and König in [BGK18]. One subtlety in our question is that we need to consider the size of subsets S_x required to achieve a separation. If $|S_x| = 1$, meaning that the relation maps $x \mapsto f(x)$ for a function f, the light-cone argument still holds because we can classically simulate measuring the output qubits in parallel. However, what if we enforce a constraint like parity on the output qubits? Then such a simulation won't work so easily; this is in fact a starting point for the next problem we define.

3 The GHZ game

We will not quite prove the separation result in [BGK18] until next class. In this lecture, we will define the GHZ game and bound the advantage of any classical strategy. In the next class, we will give a quantum strategy with a strictly higher advantage. Then, we will see how the advantage in playing the game can be "leveraged" to show advantage of constant-depth quantum circuits.

The GHZ game is a quantum nonlocal game with a referee and three players. The referee sends bit a to Alice, b to Bob, and c to Charlie. (Bits a, b, c are chosen uniformly at random.) Alice sends back bit x, Bob sends back y, and Charlie sends back z. Alice, Bob, and Charlie can agree on a strategy ahead of time but cannot communicate during the game.



We are promised that $a \oplus b \oplus c = 0$ and we win the game if $x \oplus y \oplus z = a \vee b \vee c$. In the following theorem, we bound the advantage of classical players in the GHZ game.

Theorem 5. *For classical Alice, Bob, and Charlie, the probability of winning the GHZ game is* 0.75.

Proof. We first show how to achieve winning probability 0.75, then prove it is the maximum. Note that the promise $a \oplus b \oplus c = 0$ means there are only four possible scenarios for the game:

a	b	c	$a \vee b \vee c$
0	0	0	0
1	1	0	1
0	1	1	1
1	0	1	1

Our strategy is simple: Alice always outputs 1, while Bob and Charlie always output 0. Then $x \oplus y \oplus z = 1 \oplus 0 \oplus 0 = 1$, which is equal to $a \vee b \vee c$ in three out of the four game scenarios. Since the referee chooses a, b, c uniformly at random, this means we win with probability 0.75.

To prove that this is the best possible strategy, suppose that Alice, Bob, and Charlie are classical and deterministic. Write that $A(a) \to x$, $B(b) \to y$, and $C(c) \to z$. If they are correct in all four game scenarios, then we have

$$\begin{aligned} A(0) \oplus B(0) \oplus C(0) &= 0 \\ A(1) \oplus B(1) \oplus C(0) &= 1 \\ A(0) \oplus B(1) \oplus C(1) &= 1 \\ A(1) \oplus B(0) \oplus C(1) &= 1 \, . \end{aligned}$$

However there is a contradiction in these four equations. If we add them all together (via $\oplus$), the lefthand terms all cancel and we end up with $0 = 0 \oplus 1 \oplus 1 \oplus 1$, a contradiction. Thus it is impossible for determininstic Alice, Bob, and Charlie to be correct in all four game scenarios. However, they can be correct in three, as shown above.

What about nondeterministic (or randomized) strategies? If Alice, Bob, and Charlie are randomized, their strategy will only be a convex combination of deterministic strategies. (This is because the input distribution over a, b, c is fixed as uniform, and no communication is allowed during the game.) Concretely, suppose there are n deterministic strategies $D_1, \ldots, D_n$ for Alice, Bob, and Charlie. If they are randomized, they execute strategy D_i with probability p_i, where $\sum_i p_i = 1$. We have already shown that for any fixed D_i, the probability of winning is at most 0.75. In the randomized case the probability of winning is now $\sum_i p_i \Pr[\text{win on } D_i] \leq \sum_i p_i (0.75) \leq 0.75$. Thus for all classical strategies, deterministic or randomized, the best possible advantage in the GHZ game is 0.75. □

References

[BGK18] Sergey Bravyi, David Gosset, and Robert König. Quantum advantage with shallow circuits. *Science*, 362(6412):308–311, October 2018.