The Complexity of Entanglement

CSC2429/MAT1751 offered at the University of Toronto, Fall 2020

Instructor: Henry Yuen

Abstract

Contents

Chapter 1

Introduction to the class

Scribes: Abhinav Anand, Deepanshu Kush

There are a lot of exciting things going on in quantum computing and quantum information processing. The field is growing really rapidly and it's impossible to have a single course that covers all of the really cool cutting edge topics. Quantum computing and quantum information is bridging so many different areas of science and engineering: from algorithms to communications to metrology to fundamental physics to complexity to pure mathematics.

This class is going to focus on cutting edge topics in quantum information that relate to the following theme: *Complexity of Entanglement*. This theme is deeply conceptual: it seeks to understand entanglement — this bizarre phenomenon in nature where the state of two particles is intimately intertwined — from the perspective of computer science and information processing. One could name a lot of weird aspects of quantum physics: uncertainty principle, superposition, wave-particle duality, etc. But Erwin Schrödinger, one of the founders of modern Quantum Mechanics, was quick to single out entanglement as perhaps the most important weird feature of quantum mechanics. He made this comment in 1935:

I would not call entanglement one but rather ***the*** *characteristic trait of quantum mechanics, the one that enforces its entire departure from classical lines of thought.*

Despite this, entanglement as a phenomenon was not studied very much in mainstream physics. For many years, the only people who were seriously thinking about entanglement were those thinking about foundational questions in quantum physics, that was basically borderline philosophy. But this didn't really impact the mainstream work that was focused on discovering new exotic subatomic particles and figuring out the Standard Model of particle physics.

Fast forward to the 1980s and early 1990s. Information processing and quantum physics collided. Entanglement takes center stage, because now people realize that

1. Entanglement is a barrier: because of entanglement, the state of n particles requires at least 2^n amplitudes to describe. This exponential complexity prevents quantum physics from being easily simulatable on a classical computer.
2. Entanglement is a useful information processing resource: it can be used to create perfectly secret keys between two distant parties. It can be used to teleport quantum information.

You need to generate highly entangled states on a quantum computer in order to solve hard problems.

3. Entanglement is a source of exotic quantum phenomenon in nature: superconductivity, superfluidity, Bose-Einstein condensates.

Since then, the computational and information processing lens on entanglement has produced a steady river of fascinating questions. Here's a selection:

1. How complicated can it get to describe minimum energy states of quantum systems? Can physical systems occurring in nature be described with a small number of parameters? If such small descriptions exist, can they be efficiently found on a computer?
2. Entanglement can be a useful computational and information processing resource. Can *unentanglement* be useful?
3. Can complex entanglement persist at "room temperature"? Is there a quantum analogue of the *Probabilistically Checkable Proofs (PCP) Theorem* from classical complexity theory?
4. Is it possible to efficiently verify the result of a quantum computer using only classical resources?
5. What is the complexity of optimizing over quantum correlations? Are there physical phenomena that cannot even be approximately modeled using finite-dimensional systems? What can quantum complexity theory tell us about long-standing open problems in functional analysis and operator algebras?

These questions probably seem wildly unconnected from each other. In this class, we'll see how they all relate to each other, and how the theme of Complexity of Entanglement has given us ways of answering some of these questions.

1.1 A quantum information theory refresher

Here's an extremely brief refresher of some quantum information theory facts that we'll need today. For a more comprehensive review, please consult something like Nielsen and Chuang [**nc**].

Quantum States: Consider some physical system that can be in d different, mutually exclusive classical states. We call these states $|1\rangle, |2\rangle, \ldots, |d\rangle$. Roughly, by a *classical* state we mean a state in which the system can be found if we observe it. A d-dimensional *quantum* state looks as follows:

$$|\psi\rangle = \sum_{i=1}^{d} \alpha_i \, |i\rangle = \begin{pmatrix} \alpha_1 \\ \vdots \\ \alpha_d \end{pmatrix}.$$

The state when measured in the standard basis $\{|1\rangle, |2\rangle, \ldots, |d\rangle\}$, collapses to $|i\rangle$ with probability $|\alpha_i|^2$. A qubit lives in a 2 dimensional Hilbert space. So, when $d = 2$, the quantum state describes the state of a qubit. To visualize the state of a single qubit, one can use the Bloch sphere representation as shown in Figure 1.1.

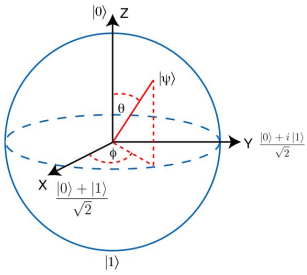


Figure 1.1: A figure of the Bloch sphere. Any point on the unit sphere corresponds to a state of the qubit, and the antipodal points correspond to orthogonal vectors. The standard basis $\{|0\rangle, |1\rangle\}$ are on the north and south pole of the sphere. Image taken from [**bloch**].

Measurement in Different Bases: Let $B = \{|v_1\rangle, |v_2\rangle, \ldots, |v_d\rangle\}$ denote an orthonormal basis for $\mathbb{C}^d$. When we measure in basis B, we associate $|v_i\rangle$ with outcome i. The probability of obtaining outcome i when measuring $|\psi\rangle$ in basis B is

$$|\langle v_i|\psi\rangle|^2.$$

In other words, it's the squared overlap between $|\psi\rangle$ and $|v_i\rangle$. After obtaining outcome i, the *post-measurement* state of the system is in $|v_i\rangle$. In other words, if we measure it again, it is going to be outcome i with certainty.

More generally, we do not have to project onto single basis vectors. We can group different basis vectors together into a single outcome. This gives rise to a general *projective measurement*: A k-outcome projective measurement consists of projections $\{P_1, \ldots, P_k\}$ acting on $\mathbb{C}^d$ such that $P_1 + \cdots + P_k = I$, the projectors are orthogonal ($P_i P_j = 0$ if $i \neq j$). The probability of obtaining outcome i is

$$\| P_i |\psi\rangle \|^2 = \langle\psi| P_i |\psi\rangle .$$

And the post-measurement state then becomes

$$\frac{P_i |\psi\rangle}{\| P_i |\psi\rangle \|} .$$

Observables: One can conveniently describe measurements using *observables*. An observable A corresponds to a physical quantity that can be measured, and is associated with a Hermitian operator $\hat{A}$. Given a d-dimensional observable A, one can diagonalize it, and it can then be written as follows:

$$A = \sum_{i=1}^{k} \lambda_i P_i$$

where $\lambda_1 < \lambda_2 < \cdots < \lambda_k$ are distinct real numbers, and $P_1, \ldots, P_k$ are orthogonal projection operators. The λ_i's are the eigenvalues of A and the set $\{P_i\}$ specifies the projective measurements. The eigenvalue λ_i can be interpreted as a "weight" for the i-th measurement outcome when measuring the observable. Let us dive into more detail by looking at the expression for the *expected value* of an observable A with respect to a state $|\psi\rangle$, which can be written as follows:

$$\langle\psi| A |\psi\rangle = \sum_{i=1}^{k} \lambda_i \langle\psi| P_i |\psi\rangle .$$

The term $\langle\psi| P_i |\psi\rangle$ is the probability of the outcome i, so the expected value can be interpreted as the expected weight you obtain if you measure $|\psi\rangle$ using the $\{P_i\}$'s.

Let us consider the $d = 2$ (qubit) scenario. Given an orthonormal basis $B = \{|v_0\rangle, |v_1\rangle\}$ for $\mathbb{C}^2$, and an observable A written as,

$$A = P_0 - P_1$$

where $P_0 = |v_0\rangle\langle v_0|$ is the projection onto $|v_0\rangle$, and $P_1 = |v_1\rangle\langle v_1|$ is the projection onto $|v_1\rangle$. Thus $|v_0\rangle$ is the $+1$ eigenvector of A, and $|v_1\rangle$ of -1 eigenvector of A.

So using the form of A from above we can write the expectation values with respect to $|\psi\rangle$ as:

$$\begin{aligned}\langle\psi| A |\psi\rangle &= \langle\psi| A_0 |\psi\rangle - \langle\psi| A_1 |\psi\rangle \\ &= \Pr[\text{outcome } 0] - \Pr[\text{outcome } 1] \\ &= 2\Pr[\text{outcome } 0] - 1 \\ &= 1 - 2\Pr[\text{outcome } 1].\end{aligned}$$

Oftentimes, nice measurement bases yield observables with nice algebraic properties.

Pauli Matrices: These are a set of particularly nice qubit observables, defined below:

$$I = \begin{pmatrix}1 & 0\\ 0 & 1\end{pmatrix}, \qquad X = \begin{pmatrix}0 & 1\\ 1 & 0\end{pmatrix}, \qquad Y = \begin{pmatrix}0 & -i\\ i & 0\end{pmatrix}, \qquad Z = \begin{pmatrix}1 & 0\\ 0 & -1\end{pmatrix}.$$

They satisfy the following nice properties:

1. A is *Unitary* i.e., $AA^\dagger = I$ for all $A \in \{I, X, Y, Z\}$.
2. A is *Hermitian* i.e., $A = A^\dagger$ for all $A \in \{I, X, Y, Z\}$.
3. In particular, being Unitary implies that their eigenvalues are roots of unity. Being Hermitian implies that their eigenvalues are real. Thus, their *eigenvalues are* ± 1.
4. A is *involutory* i.e., $A^2 = I$ for all $A \in \{I, X, Y, Z\}$.
5. X, Y, Z pairwise *anti-commute* i.e.,

$$XY = -YX, \qquad YZ = -ZY, \qquad ZX = -XZ.$$

6. *Projective Measurement:* Each of the Pauli matrices corresponds to a certain projective measurement, described below:
   (a) I corresponds to projection onto $\begin{pmatrix}1\\0\end{pmatrix}, \begin{pmatrix}0\\1\end{pmatrix}$ (or any other orthonormal basis). Note also that I has just the one eigenspace, corresponding to the $+1$ eigenvalue.
   (b) Z corresponds to projection onto $|0\rangle \coloneqq \begin{pmatrix}1\\0\end{pmatrix}$ (with eigenvalue $+1$) and $|1\rangle \coloneqq \begin{pmatrix}0\\1\end{pmatrix}$ (with eigenvalue -1).
   (c) X corresponds to projection onto $|+\rangle \coloneqq \frac{1}{\sqrt{2}}\begin{pmatrix}1\\1\end{pmatrix}$ (with eigenvalue $+1$) and $|-\rangle \coloneqq \frac{1}{\sqrt{2}}\begin{pmatrix}1\\-1\end{pmatrix}$ (with eigenvalue -1).

(d) Y corresponds to projection onto $\frac{1}{\sqrt{2}}\begin{pmatrix}1\\ i\end{pmatrix}$ (with eigenvalue $+1$) and $\frac{1}{\sqrt{2}}\begin{pmatrix}1\\ -i\end{pmatrix}$ (with eigenvalue -1).

The most important ones we will consider are I, X, and Z.

Heisenberg Uncertainty Principle: This principle arises in physics and chemistry, and states that it is impossible to determine the position and the momentum of a particle at the same time. In quantum information theory terminology, this boils down to the following statement: it is not possible for a qubit $|\psi\rangle$ to be determined using both the standard basis and the plus-minus basis. In other words, if we measure $|\psi\rangle$ in the standard basis and get a deterministic outcome, then we cannot get a deterministic outcome when measuring $|\psi\rangle$ in the plus-minus basis, and vice-versa. We say that these two bases are incompatible.

This incompatability arises directly from the fact that the observables Z, X don't commute.

On the contrary, if you have two observables A, B that *do* commute, then you can simultaneously measure them i.e., it is possible to simultaneously have compatible outcomes for both observables.

1.2 The EPR Paradox, Bell's Theorem, and the CHSH game

EPR Paradox: In 1935, Einstein, Podolsky and Rosen published a paper titled "Can Quantum-Mechanical Description of Physical Reality be considered Complete?", where they argued that quantum mechanics provides an incomplete description of nature and speculated that there should be a better theory which can describe all this perfectly. The thought experiment that led them to write this paper can be described as follows:

1. They considered a situation where two participants, Alice and Bob, who each share a qubit of the two qubit entangled state $|\psi\rangle = \frac{1}{\sqrt{2}}(|00\rangle + |11\rangle)$, and are unable to communicate to each other because they are at the opposite end of the galaxy millions of light years away.

2. If Alice decides to measure her qubit in the standard basis $\{|0\rangle, |1\rangle\}$ (i.e. measure the Z observable), she measures the state $|0\rangle$ or $|1\rangle$ with equal probability $\frac{1}{2}$, which collapses the overall state of the system to $|00\rangle$ or $|11\rangle$ correspondingly.

3. On the other hand, if Alice decides to measure her qubit in the basis $\{|+\rangle, |-\rangle\}$ (i.e. measure the X observable), she measures the state $|+\rangle$ or $|-\rangle$ with equal probability $\frac{1}{2}$, which collapses the overall state to $|++\rangle$ or $|--\rangle$ correspondingly.

4. They concluded that no matter what basis Alice choses to measure her qubit in, the state of Bob's qubit collapses to the same state as the outcome of her measurement. Thus, based on the local hidden variable theory, EPR thought this was really weird, as this meant that even if the two qubits are million light years apart, Alice could tell what the outcome of Bob's qubit would be by just performing a measurement of her qubit, even without exchange of any information of the measurement basis.

5. EPR concluded that this violated the Heisenberg uncertainty principle as this meant that one can tell the outcome of Bob's qubit in both the $\{|0\rangle, |1\rangle\}$ and $\{|+\rangle, |-\rangle\}$ basis at the

same time, based on the measurement Alice made on her qubit. Thus, they concluded that something must be wrong with quantum mechanics, and they posited that there should a better classical theory out there which gives a complete description of the physical reality.

Bell's Theorem: In 1964, this was established by John Stewart Bell leading to a way to reconcile the principles of quantum mechanics with the apparent paradox put forth by EPR. Informally, it asserts the following:

*EPR Phenomenon cannot be explained by any **local**, classical theory.*

In other words, our world is inherently non-local ([**scho**]). "Non-local" here means that there exist interactions between events that are too far apart in space and too close together in time for the events to be connected even by signals moving at the speed of light. This conclusion is very surprising, since non-locality is normally taken to be prohibited by the theory of relativity.

CHSH Game: Named after John Clauser, Michael Horne, Abner Shimony, and Richard Holt, it provides an experimental framework for supporting Bell's theorem. We provide a description of the CHSH game, which is a hypothetical Bell test experiment. The players of this game can use either classical strategies (corresponding to local hidden variable theories) or quantum strategies, which involve measurements of a shared entangled bit ([**logan**]). It will be shown that quantum strategies allow a greater winning probability than classical strategies. Here is the game setup:

A referee chooses bits $x, y \in \{0, 1\}$ independently and uniformly at random, and then sends x to the player Alice and y to the player Bob, who cannot communicate with each other. Alice answers by sending a bit a back to the referee and similarly, Bob answers with bit b, as shown in Figure 1.2. They win the game if their answers satisfy $a + b \equiv xy \pmod 2$.

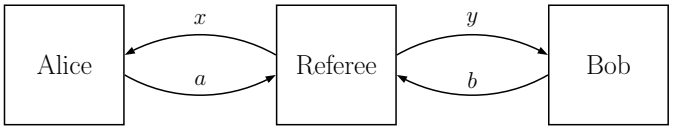


Figure 1.2: Setup for the CHSH Game

How well can they do? If they use a classical strategy (i.e., local hidden variable theory), then we claim that their maximum probability of winning is 3/4.

Let us first see the claim for the case when they use *deterministic* strategies. In other words, Alice's response a is a (deterministic) function $f(x)$ of the bit she receives and similarly Bob's response $b = g(y)$. Taking cases on the values of the bits x, y, winning corresponds to the following:

- When $x = 0$ and $y = 0$, players win if $f(0) + g(0) \equiv 0 \pmod 2$.
- When $x = 0$ and $y = 1$, players win if $f(0) + g(1) \equiv 0 \pmod 2$.
- When $x = 1$ and $y = 0$, players win if $f(1) + g(0) \equiv 0 \pmod 2$.
- When $x = 1$ and $y = 1$, players win if $f(1) + g(1) \equiv 1 \pmod 2$.

But for any fixed functions f, g, no more than three of the four equations above can hold simultaneously because adding them all up immediately yields a parity contradiction. Thus, if Alice and Bob use deterministic strategies, their chance of winning is at most 75%. So, even using *randomized* strategies does not improve this bound in expectation.

Bibliography

[1] Nielsen, Michael and Chuang, Isaac. *Quantum Information & Quantum Computation.*

[2] Quantum Inspire. *Bloch Sphere.* URL: https://www.quantum-inspire.com/kbase/bloch-sphere/.

[3] Scholarpedia. *Bell's Theorem.* URL: http://www.scholarpedia.org/article/Bell%27s_theorem#Bell.27s_theorem.

[4] Logan Meredith. *The CHSH game as a Bell test thought experiment*, 2017. URL: https://www.sas.rochester.edu/pas/assets/pdf/undergraduate/The_CHSH_game_as_a_Bell_test_thought_experiment.pdf.

Chapter 2

Introduction to Quantum Hamiltonian Complexity

Scribes: Ariel Kelman, Gary Tom

2.1 Finishing up the CHSH game

In the previous lecture, we discussed the Clauser-Horne-Shimony-Holt (CHSH) game, which provides a modern experimental formulation of Bell's Theorem. Bell's Theorem says that the consequences of entanglement in quantum mechanics are incompatible with the classical interpretation of local variables, as described in the Einstein-Podolsky-Rosen paradox.

The setup of the game is as follows. Alice and Bob are arbitrarily far away, and can only communicate with a referee in the middle. The referee will generate two independent uniformly random numbers $x, y \in \{0, 1\}$ which are sent to Alice and Bob respectively. Based on their received numbers, Alice and Bob will generate a reply, and return it to the referee as bits a and b, respectively. Alice and Bob win the CHSH game if the following condition is satisfied,

$$a \oplus b = x \wedge y, \tag{2.1}$$

which can also be phrased as $a + b = xy \mod 2$. Note that $\oplus$ is the exclusive-or (XOR) logical operator, and $\wedge$ is the and (AND) operator.

Deterministic strategy: If Alice and Bob use fixed functions to generate their reply, such that $a = f(x)$ and $b = g(y)$, their chance of winning is at most 75%. This was shown last lecture.

Probabilistic strategy: Alice and Bob may consider introducing an element of randomness in their deterministic functions. In this strategy, Alice and Bob can use a shared random variable r, which they presumably generated before being separated. The game proceeds as above, but Alice and Bob will incorporate the variable r in their reply functions, such that they have $a = f(x, r)$ and $b = g(y, r)$. This type of strategy is also allowed by classical physics, so we consider such randomized strategies to still be classical.

We suppose that Alice and Bob pick the best probabilistic strategy for winning the CHSH game,

such that

$$Pr_{x,y,r}[f(x,r) \oplus g(y,r) = x \wedge y] > 0.75. \tag{2.2}$$

Then there is guaranteed to be some fixed random variable r^*, such that we would have

$$f(x,r^*) \oplus g(y,r^*) = x \wedge y, \tag{2.3}$$

with probability > 0.75. But this is analogous to the deterministic strategy above, which cannot exceed a win probability of 0.75. And so, the best probabilistic strategy they can choose will have a maximum winning probability of 75%, by the same arguments as before. Without loss of generality, all local and classical strategies are deterministic, and will only allow for a maximum winning probability of 75%.

Quantum strategy: With this strategy, Alice and Bob will share an EPR pair $\frac{1}{\sqrt{2}}(|00\rangle + |11\rangle)$. Prior to the game, Alice agrees to make measurements with operators A_0, A_1 for $x = 0$ and 1 respectively. And Bob does the same with operators B_0, B_1 for $y = 0$ and 1 respectively. The operators are as follows

$$A_0 = Z \qquad A_1 = X, \tag{2.4}$$

$$B_0 = \frac{1}{\sqrt{2}}(X+Z) \qquad B_1 = \frac{1}{\sqrt{2}}(Z-X). \tag{2.5}$$

These operators are *observables* with eigenvalues ± 1, which Alice and Bob will use to give their reply. On getting result $+1$, their reply will be a or $b = 0$. For -1, their reply will be a or $b = 1$. Let's consider the winning probability of such a strategy. We will utilize a formula derived in Lecture 1, where for two-outcome observables

$$\langle\psi| A |\psi\rangle = 1 - 2\Pr(\text{outcome } (-1)) = 2\Pr(\text{outcome } (+1)) - 1 \tag{2.6}$$

For $x = 0, y = 0$: Alice and Bob need to have output $a = b$ in order to win. For Alice, she makes a measurement with A_0,

$$\langle EPR| A_0 |EPR\rangle = \left(\frac{1}{\sqrt{2}}\langle 00| + \langle 11|\right) Z \left(\frac{1}{\sqrt{2}}|00\rangle + |11\rangle\right) = \left(\frac{1}{2}(+1) + \frac{1}{2}(-1)\right). \tag{2.7}$$

She will obtain with $1/2$ probability $a = 0$ and collapse the the state to $|0\rangle$ for Bob. She also has $1/2$ probability of $a = 1$, collapsing the state to $|1\rangle$ for Bob.

- If Alice measures $+1$, or $a = 0$, then Bob then makes a measurement B_0 on $|0\rangle$. From the expectation value of Bob's measurement, we can get the winning probability

$$\langle 0| B_0 |0\rangle = 2\Pr(b = 0|a = 0) - 1 = \frac{1}{\sqrt{2}}\langle 0|(X+Z)|0\rangle = \frac{1}{\sqrt{2}}, \tag{2.8}$$

$$\Rightarrow \quad \Pr(b = 0|a = 0) = \frac{1}{2} + \frac{1}{2\sqrt{2}} \approx 0.854... \tag{2.9}$$

- If Alice measures -1 or $a = 1$, then Bob measures B_0 (because he got question $y = 0$) on $|1\rangle$, and the winning probability

$$\langle 1| B_0 |1\rangle = 1 - 2\Pr(b = 1|a = 1) = \frac{1}{\sqrt{2}}\langle 1|(X+Z)|1\rangle = -\frac{1}{\sqrt{2}} \tag{2.10}$$

$$\Rightarrow \Pr(b = 1|a = 1) \approx 0.854... \tag{2.11}$$

Doing the same for all other combinations of x and y will yield the same winning probability. The quantum strategy allows Alice and Bob to win with a probability of 85.4...%, exceeding the win chance of classical strategy. Therefore, quantum entanglement cannot be modelled by any local classical theory, demonstrating Bell's Theorem.

Over many years, the CHSH game has been experimentally demonstrated many times, and it provides an operational interpretation of entanglement: entanglement is the phenomenon that allows you to win the CHSH game with higher probability than what is allowed classically.

2.2 Introduction to Hamiltonian Complexity

Hamiltonian complexity is the study of quantum constraint satisfaction problems (QCSP). In physics, a physical system is modelled first as a Hamiltonian, an observable comprised of the forces and energies in a physical system. The goal is to then *solve* the Hamiltonian in order to calculate properties and make predictions of the physical system. In computer science, Hamiltonians can be viewed as the quantum analogues of constraint satisfaction problems.

2.3 Classical Constraint Satisfaction Problems

Classical constraint satisfaction problems (CSPs) are mathematical problems with a set of objects whose states must satisfy a set of constraints. Here are two examples:

1. **k-local CSP**

   A particular instance of k-local CSP is a collection of Boolean variables $x_1, \ldots, x_n$ that satisfies a set of constraints $C_1, \ldots, C_m$. Each clause C_i is a function of at most k variables. The goal is to find an assignment of variables that satisfies as many of the constraints as possible.

   For example, the 3SAT (satisfiability) problem involves constraints of the three boolean variables, such as $C_i = x_1 \vee \neg x_2 \vee x_3$. That is, each constraint term is a *disjunction* of $k = 3$ variables.

   For more general CSPs the constraints can be different. A *satisfying assignment* to a CSP is a setting of boolean values to the variables that satisfies *all* of the constraints simultaneously.

2. **Max-Cut**

   An instance of Max-Cut consists of a graph $G = (V, E)$ with vertices V and edges E. The goal is to *cut* the vertices into a left and right side, such that we maximize the number of edges crossing over. An example is given in Figure 2.1.

   Let us rephrase this into a satisfiability problem. We have have Boolean variables $\{x_v, \ldots, x_n\}$, one for each vertex $v \in V$, which specifies which side the vertex is on (left or right). There is a constraint C_e for every edge $e \in E$: each edge is between two vertices, $e = (u, v)$, and we require that $x_u \neq x_v$. The Max-Cut problem is to find assignment of $\{x_v\}$ that maximizes the number of satisfied constraints.

   The Max-Cut problem can also be described as a physical magnetic system. Consider a graph G that is a 2D grid with magnets on the vertices. The magnets can either point *up* or *down*,

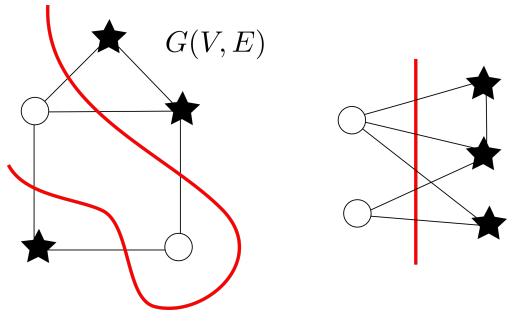


Figure 2.1: (*left*) An instance of Max-Cut on a graph $G(V, E)$ with 5 vertices and 6 edges. (*right*) The vertices are partitioned into left (circles) and right (stars) sides. The number of edges cut is 5 out of 6. There exists more than one way to achieve Max-Cut.

analogous to the left and right partitions. The most favourable condition would be the anti-parallel alignment of neighbouring magnets, which is analogous to a cut edge, since opposite poles of the magnets attract each other. Finding the optimal configuration of this system is a Max-Cut problem (Figure 2.2).

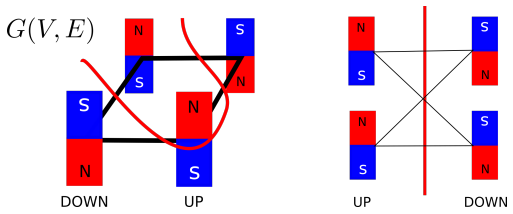


Figure 2.2: (*left*) Max-Cut instance of 2x2 grid G of magnets. All neighbouring magnets are anti-aligned. (*right*) The up and down magnets partition the vertices into left and right.

2.3.1 NP completeness

CSPs are, in general, difficult problems to computationally solve. Complexity classes provide a unifying framework in complexity theory that describes CSPs and their difficulty; of these classes, P and NP are particularly relevant.

Decision problems can be given by a **language** L, and are problems where we have yes or no answers: $L = L_{yes} \cup L_{no}$. For example, finding out whether a number is prime. In this case, we have $L_{yes} = \{x : x \text{ is prime}\}$, and $L_{no} = \{x : x \text{ is not prime}\}$. P and NP are two classes of such decision problems.

The complexity class P, which stands for **Polynomial Time** consists of decision problems that are solvable in polynomial time. A language $L \in P$ if there exists a deterministic algorithm A such that it

1. Runs in time poly($|x|$) where $|x|$ is the number of bits to describe x,
2. If $x \in L_{yes}$, then $A(x) = 1$, and
3. If $x \in L_{no}$, then $A(x) = 0$.

The complexity class NP, which stands for **Non-deterministic Polynomial Time** contains decision problems that have an algorithm that can verify the solution to a problem in polynomial time. A language $L \in \mathsf{NP}$ if there exists a deterministic algorithm $A(x, y)$, where y is a proof, such that

1. A runs in time poly($|x|$) where $|x|$ is the number of bits to describe x,
2. If $x \in L_{yes}$, then *there exists* a string y such that $A(x, y) = 1$, and
3. If $x \in L_{no}$, then *for all* strings y we have $A(x, y) = 0$.

In NP problems, A is trying to decide if $x \in L_{yes}$, but it isn't able to determine this outright. Instead it hopes that a *proof* y will "fall down from the sky", proving to A that $x \in L_{yes}$. The algorithm A then checks (in polynomial time) whether this proof is indeed correct.

If x is a YES instance, then there is such a proof that falls from the sky. If x is a NO instance, then no proof will convince A that x is a YES instance. As such, $A(x, y)$ acts as a *verifier*, rather than a solver.

All CSPs are NP problems. Consider the previous examples:

1. In the 3SAT problem, consider some instance φ, which consists of variables and the constraints on those variables. A proof y could be the satisfying assignments, which can be checked in polynomial time. But if φ is a NO instance, there is no proof that can convince you that it's satisfiable.
2. The Max-Cut problem is not a decision problem itself, but we can formulate it as one. Let's define "Max-Cut-0.90" as the decision problem where:

   (a) L_{yes} are all graphs G that can be partitioned with 90% of all edges crossing the cut, and
   (b) L_{no} are all graphs that cannot be partitioned in such a way.

   The proof y would be a partition of a graph in G. To check, you would just have to count the number edges that are cut in the partition.

2.3.2 The Cook-Levin Theorem

The famous Cook-Levin theorem (named after Steven Cook who is a professor emeritus here at UofT and Leonid Levin who was in the Soviet Union at the time) says that 3SAT is NP-complete (and this is true for most CSPs). That is, CSPs are generally *complete* for the class NP. In other words, not only are CSPs in NP, but they are as hard as any other problem in NP. More formally, given a decision problem $L \in \mathsf{NP}$, there is an efficient way to transform instances x of L into instances φ_x of a CSP such as SAT: if $x \in L_{yes}$ then φ_x is a satisfiable SAT instance, otherwise φ_x is unsatisfiable.

Proof Sketch

We want to show that any instance x of a problem L can be transformed into an instance $\varphi_x \in$ 3SAT efficiently (i.e. in polynomical time) such that if $x \in L_{yes}$ then φ_x is a satisfiable SAT instance, otherwise φ_x is unsatisfiable (note the fact that any problem in SAT can be reduced to 3SAT in turn, so it does not matter if the SAT instance we construct is in 3SAT).

Starting with our NP problem $L = L_{yes} \cup L_{no}$, by definition we know that there's a polynomial-time algorithm A that can verify proofs for $x \in L_{yes}$. We want to know whether there's a proof y such that $A(x, y) = 1$ since if there is such a proof, then $x \in L_{yes}$.

Consider the following mental image. The algorithm A is running on a Turing machine - an idealized model of computation with a head that's moving around on a 1-dimensional memory tape, reading/writing to the tape one symbol at a time. You have an instance x of the language L, and from the sky comes down a purported proof y. You plug x and y into the program A, and you run the program to see if it accepts (i.e. outputs 1) or rejects (output 0). Let's suppose that after $T = n^c$ time steps the Turing machine stops and accepts.

Now suppose you'd like to convince a friend that $x \in L_{yes}$, but can't transport the Turing Machine to show them. All that is available is a description of the algorithm/Turing Machine A and the input x. Consider taking a snapshot of the Turning Machine at each time step, where each snapshot captures the entire configuration of the machine: the contents of the tape, location of the head, state of the machine, etc. This entire sequence of snapshots $S_0, S_1, ...S_T$ will be used to prove that there is some y such that $A(x, y) = 1$ and therefore $x \in L_{yes}$. Note that since A runs in polynomial time, all the snapshots (both individually and collectively) take up only polynomial space. Taken together, we'll call all the snapshots the *computational tableau of A on input (x, y)*.

Now your friend can verify that that this sequence of snapshots represents the execution of $A(x, y)$ and really does output 1. This can be done by checking the following three conditions:

1. **Starts OK**: check that in S_0, the Turing machine is properly initialized, with x written as the first input on the memory tape (there are no constraints on the second input - if some assignment can be found to the final SAT formula, that assignment will determine the proof y), and the scratch space of the algorithm (i.e. the remainder of the infinitely long tape) is set to 0.
2. **Evolves OK**: check that for every consecutive pair of snapshots S_t and S_{t+1}, the snapshots are consistent with each other. That is, check that if the state of the algorithm A was as described in snapshot S_t, then the snapshot S_{t+1} really would be the correct next state of the computation.
3. **Ends OK**: finally, check that the final snapshot S_T indicates that the final state of the algorithm is that it accepted and outputted 1. The description of the Turing Machine will have designated part of the Turing machine memory to show what its output is.

If these conditions are satisfied, then there is some y such that $A(x, y) = 1$, showing that $x \in L_{yes}$.

All of these conditions on the snapshots can be expressed as an instance of φ_x of 3SAT. As mentioned above, the tableau is polynomially sized: since A runs in time $T = n^c$, each snapshot is bounded in memory size by some polynomial n^c (since A runs in time $T = n^c$). Construct a 2D grid of variables $\{z_{t,i}\}$ where t indicates the time step, and i indicates the i-th bit of the Turing machine state and memory. Thus there are $O(T \cdot n^c)$ total variables, and these are going to be the variables of φ_x. This model is shown in figure 2.3.

Each of the three conditions (Starts OK, Evolves OK, Ends OK) can be broken down into a collection of clauses on the $\{z_{t,i}\}$ variables. Furthermore, these clauses only have to involve 3 variables at time, so φ_x is a 3SAT formula. There's going to be a lot of clauses, but at the end of the day there will only be poly(n)-many clauses, each with only 3 variables.

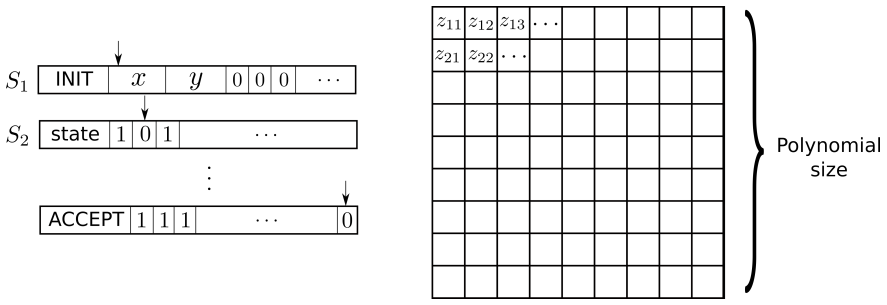


Figure 2.3: A computation tableau. On the left is a series of snapshots, capturing the state of the machine, tape and head at each step. On the right, this is transformed into a large grid of (boolean) variables from which a 3SAT instance is constructed.

The crucial point is that a satisfying assignment of variables $z = \{z_{t,i}\}$ for the 3SAT formula will essentially be a computational tableau (see figure 2.3) that proves that there is a y for which $A(x, y) = 1$, and therefore $x \in L_{yes}$. On the other hand if $x \in L_{no}$, then there is no way to satisfy all of these constraints.

One key insight from the Cook-Levin theorem is that it relies on the fact that computation is fundamentally *local*: to perform any computation, you only need to change a few bits at a time in some systematic manner. This is because when constructing the constraints on $\{z_{t,i}\}$, each constraint $z_{t',i'}$ only relies on the variables immediately surrounding it on the tableau. That is what allows the Starts OK, Evolves OK, and Ends OK checks to be definable via local clauses.

2.4 Quantum Constraint Satisfaction Problems and QMA

The following table shows a classical-quantum dictionary which highlights the correspondence between concepts from computer science and physics that will be fleshed out in this section.

Classical	Quantum
Constraint Satisfaction Problem (CSP)	Hamiltonian
Variables	Qubits
Constraints	Hamiltonian terms
Solution quality	Energy
Optimal solution	Ground state
P	BQP
NP	QMA
Cook-Levin SAT formula	Feynman-Kitaev Hamiltonian

2.4.1 k-local Hamiltonians

A k-local Hamiltonian is the analogue of a k-local CSP (defined above). Such a Hamiltonian

- Acts on n qubits.

- Consists of m *Hamiltonian terms* $H_1, \ldots, H_m$ where each H_i is a Hermitian matrix defined over $(\mathbb{C}^2)^{\otimes n}$. Each H_i can be written in the form

$$H_i = h_i \otimes \underbrace{I \otimes I \otimes \cdots \otimes I}_{n-k'}$$

where h_i is a Hermitian matrix defined on $(\mathbb{C}^2)^{\otimes k'}$. Conceptually, each H_i acts non-trivially on k' qubits (with $k' \leq k$), and acts on the remaining $n - k'$ qubits with identity. As in the table, each Hamiltonian term H_i corresponds to a contraint in a CSP.

Note: This notation can be misleading, because it seems to suggest that the Hamiltonian term H_i acts nontrivially on the *first* k-qubits (if we were to arrange the qubits on a line). However, h_i may act on some other subset of k qubits, and so the reader will have to look for where this is specified. For example sometimes for a two-local Hamiltonian, we will write $H_{i,j}$ to indicate a Hamiltonian term that acts nontrivially on qubits i and j, where the qubits have some numbering. Sometimes we may write something like $H_i = (h_i)|_S \otimes I_{[n]\setminus S}$ to indicate that h_i acts on a subset of qubits S.

To get a full constraint satisfaction problem from these Hamiltonian terms, consider the full Hamiltonian

$$H = H_1 + H_2 + \cdots + H_m.$$

Since we're adding up a bunch of Hermitian matrices together, H is a Hermitian matrix acting on n qubits. Since H is Hermitian, by the Spectral Theorem we can diagonalize it:

$$H = \sum_i \lambda_i |\psi_i\rangle\langle\psi_i|$$

where $\lambda_1 \leq \lambda_2 \leq \cdots \leq \lambda_N$ (for $N = 2^n$) are real eigenvalues and $|\psi_i\rangle$ is the i-th eigenvector of H with eigenvalue λ_i.

A k-local Hamiltonian acting on n-qubits assigns every state $|\varphi\rangle \in (\mathbb{C}^2)^{\otimes n}$ an *energy*, which is given by

$$\langle\varphi| H |\varphi\rangle = \sum_i \lambda_i \langle\varphi| \cdot |\psi_i\rangle\langle\psi_i| \cdot |\varphi\rangle = \sum_i \lambda_i |\langle\varphi|\psi\rangle|^2$$

which is a real number. Going back to our dictionary, if we think of a Hamiltonian H as a CSP and a state $|\varphi\rangle$ as assignment to the variables, the energy of the state λ_i is like how many clauses are violated by the assignment (though it needn't be an integer value).

The state $|\varphi\rangle$ that *minimizes* the energy of H is called a *ground state* of the Hamiltonian H. It's the analogue of having an optimal assignment (one that violates the fewest constraints).

It's a standard fact in linear algebra that a ground state of a Hamiltonian is equivalently an eigenvector with the smallest eigenvalue. So the eigenvalue λ_1 is the minimum energy of H. There may be multiple eigenvectors with the same eigenvalue (this is the case when $\lambda_1 = \lambda_2 = \cdots$), we say then that H has a *degenerate ground space*, which is the space of states spanned by all ground states.

2.4.2 Local Hamiltonians in Physics

When modelling a physical system, a local Hamiltonian describes a system that has only local interactions. For instance, consider a lattice shown in figure 2.4.

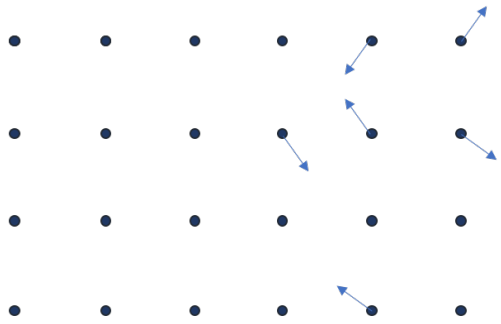


Figure 2.4: A lattice (e.g. of atoms). Considering the system to model the spin of each atom, with the arrows indicating the direction of spin, there will be an energy penalty (i.e. constraint violation) for misalignment of neighboring pairs. Similarly, one could use this picture to model macroscopic bar magnets, for which the energy penalty occurs when neighboring pairs *do* align: such a model is more similar to max-cut, as neighboring lattice-points need to have opposite orientation as in figure 2.2 above.

Each particle feels some individual force (e.g. if the entire setup is in some external potential), together with forces from all neighboring particles. The more distant interactions between non-neighboring particles are generally negligible, giving rise to a *local* Hamiltonian.

Often in such systems, physicists are interest in how the system evolves with time (its dynamics - in quantum mechanics this is given by the Schrodinger equation), as well as with the properties of low-energy or ground states (for instance, magnets are modelled as such a system of spins, and the low energy state might correspond to all aligning along one axis).

2.4.3 Examples of Local Hamiltonians

(a) Max-Cut - the Classical Ising Model

Let $G = (V, E)$ be a graph with n vertices. Consider n qubits, one for each vertex. Let Z_u be the Pauli Z operator acting on the u-th qubit. Then the Hamiltonian corresponding to Max-Cut is

$$H = \sum_{e=(u,v)\in E} Z_u \otimes Z_v$$

where we leave off the identity matrices to make the notation more understandable. The claim is that the ground energy of this Hamiltonian is achieved by state $|x\rangle$ where $x \in \{0,1\}^n$ is an optimal solution to the Max-Cut problem.

To see this, consider an individual term $Z_u \otimes Z_v$. The Z_u is a Pauli Z matrix $\begin{pmatrix} 1 & 0 \\ 0 & -1 \end{pmatrix}$, but it will be more useful to write it another way, more algebraically:

$$Z_u = |0\rangle\langle 0| - |1\rangle\langle 1|.$$

and similarly for Z_v (this representation may be familiar as a density operator). Taking the tensor

product of Z_u with Z_v, we have

$$\begin{aligned} Z_u \otimes Z_v &= \big(|0\rangle\langle 0|_u - |1\rangle\langle 1|_u\big) \otimes \big(|0\rangle\langle 0|_v - |1\rangle\langle 1|_v\big) \\ &= |0\rangle\langle 0|_u \otimes |0\rangle\langle 0|_v - |0\rangle\langle 0|_u \otimes |1\rangle\langle 1|_v - |1\rangle\langle 1|_u \otimes |0\rangle\langle 0|_v + |1\rangle\langle 1|_u \otimes |1\rangle\langle 1|_v \\ &= |00\rangle\langle 00|_{uv} - |01\rangle\langle 01|_{uv} - |10\rangle\langle 10|_{uv} + |11\rangle\langle 11|_{uv} \end{aligned}$$

where in the last line, we just regrouped the bra's and ket's to make the notation a little more compact.

We see that $Z_u \otimes Z_v$, written in this way, is already diagonalized, and we can read off what the eigenvalues and eigenvectors are: the $+1$ eigenvectors are when the u and v qubits are in the state $|0,0\rangle$ or $|1,1\rangle$ and the -1 eigenvectors are when the u and v qubits are in the state $|1,0\rangle$ or $|0,1\rangle$.

Therefore, the minimum energy states of qubits u and v, at least with respect to the Hamiltonian term $Z_u \otimes Z_v$, are the ones where u and v have opposite classical values assigned to them (corresponding to eigenvalue -1). This is the precisely the constraint that we want u and v to be on opposite sides of the cut. If the qubits u and v have matching values, then $Z_u \otimes Z_v$ assigns a $+1$ energy penalty.

We can rewrite $Z_u \otimes Z_v$ one more time:

$$Z_u \otimes Z_v = \sum_{x_u, x_v \in \{0,1\}} \epsilon(x_u, x_v) |x_u x_v\rangle\langle x_u x_v|$$

where $\epsilon(x_u, x_v)$ denotes $+1$ if $x_u = x_v$, and -1 otherwise ($x_i \in \{0,1\}$ is the state of the i^{th} qubit). Note that we're still leaving out all the other qubits in the state description (in the ket) and the identity operators on them, but that's about to change. Let's plug this back into our Hamiltonian H:

$$\begin{aligned} H &= \sum_{(u,v)\in E} \sum_{x_u, x_v \in \{0,1\}} \epsilon(x_u, x_v) |...x_u...x_v...\rangle\langle ...x_u...x_v...| \\ &= \sum_{x_u, x_v \in \{0,1\}} \left(\sum_{(u,v)\in E} \epsilon(x_u, x_v) \right) |...x_u...x_v...\rangle\langle ...x_u...x_v...| \\ &= \sum_{x \in \{0,1\}^n} \left(\sum_{(u,v)\in E} \epsilon(x_u, x_v) \right) |x\rangle\langle x| \end{aligned}$$

where the "..." signifies all the other qubits, which are then absorbed into the ket on the final line. Now, for each x, the quantity $\sum_{(u,v)\in E} \epsilon(x_u, x_v)$ is a number and in fact it's an integer. It's the sum of $+1$ for every constraint $e = (u, v)$ that's violated and -1 for every constraint that's satisfied. If $c(x)$ denotes the number of satisfied constraints (a number between 0 and m), then

$$\sum_{(u,v)\in E} \epsilon(x_u, x_v) = m - 2c(x)$$

(recall that H is the sum of m terms of the form $Z_u \otimes Z_v$), so we can finally express H as

$$H = \sum_{x \in \{0,1\}^*} \big(m - 2c(x)\big) |x\rangle\langle x|. \tag{2.12}$$

Note that this is a diagonalization of H, where the vector $|x\rangle$ is an eigenvector with eigenvalue $m - 2c(x)$. Here we're explicitly including all the other qubits - x includes the states for all of them. Thus the state $|x\rangle$ that has minimum eigenvalue – or, equivalently, the maximum number of satisfied constraints – is a ground state of H. Furthermore, $|x\rangle$ is a classical state that indicates how to partition the vertices.

It's important to note that it's possible that there are several different ground states that give the same ground state energy - i.e. there may be multiple $|x\rangle$ all with the same maximum eignevalue. Each would correspond to a solution to the max-cut problem. Thus, the ground-space (the space spanned by the states with maximum eigenvalue) is spanned by optimal solutions to the max-cut problem, since each ground state eigenvector of H is corresponds to one such solution.

(b) The Quantum Ising Model

We'll just mention the Quantum Ising Model, and discuss it further next week. Consider n qubits arranged on a ring. Then the Transverse Field Ising Model is the family of Hamiltonians composed of terms with the form:

$$H(g) = -\sum_i Z_i \otimes Z_{i+1} - g\sum_i X_i$$

where g is a real number, Z_i and Z_{i+1} indicate the Pauli Z matrix acting on qubits i and $i+1$ (we consider $N+1$ to be the same as 1), and X_i is the Pauli X matrix acting on the i-th qubit. So for every value g there is a corresponding Hamiltonian with different properties.

Chapter 3

QMA and QMA-completeness

Scribes: Alexandre Choquette, Hao Zhang

3.1 The Classical-Quantum Dictionary

As seen previously, there is a correspondence between concepts of computer science and the language of physicist. These can be summarized in the following table:

Classical	**Quantum**
Constraint Satisfaction Problem (CSP)	Hamiltonian
Variables	Qubits
Constraints	Hamiltonian terms
Solution quality	Energy
Optimal solution	Ground state
P	BQP
NP	QMA
Cook-Levin SAT formula	Feynman-Kitaev Hamiltonian

Because of their analogy with k-local CSPs, we introduced k-local Hamiltonians and described some examples. The first one was the classical Ising model which was analogue to Max-Cut. This model is said to be *classical* since the Hamiltonian is diagonal in the computational basis. We now describe the quantum version of the Ising model.

(b) The quantum Ising model

Consider N qubits (or spins) arranged on a ring, where we associate qubit $N+1$ with qubit 1. The transverse field Ising model describes a nearest-neighbor magnetic dipole interaction along the Z axis when all spins are subject to a transverse magnetic field along the X axis. The family of such Hamiltonians is given by:

$$H(g) = -\sum_{i=1}^{N} Z_i \otimes Z_{i+1} - g\sum_{i=1}^{N} X_i$$

where g is a real parameter corresponding to the strength of the transverse field, Z_i and Z_{i+1}

indicate the Pauli Z matrix acting on qubits i and $i+1$, and X_i is the Pauli X matrix acting on the i-th qubit.

For every value g there is a corresponding Hamiltonian with properties that can vary vastly. For example, if $g = 0$ we find the classical Ising model, which is the same as the Max Cut Hamiltonian, but on a ring graph. The ground state of $H(g = 0)$ is therefore a simple unentangled basis state, or bit string. Similarly, when $g \gg 1$, the transverse field part of H dominates and the ground state is again classical and given by $|+\rangle^{\otimes N}$, or all spins aligned with the magnetic field. However, for $g \neq 0$, the system truly becomes quantum, because the Hamiltonian terms don't necessarily commute anymore, *e.g.* $[Z_i \otimes Z_{i+1}, X_i \otimes I_{i+1}] \neq 0$. The Hamiltonian matrix is therefore not diagonal in the computational basis. The ground states of these Hamiltonians will in general exhibit quantum entanglement and all sorts of interesting phases of matter.

(c) The quantum Heisenberg model

A generalization of the quantum Ising model is the Heisenberg model which includes magnetic dipole interactions along the X, Y and Z axes. Notably, it models the behavior of quantum magnetism in atomic systems. It's general Hamiltonian may be written as

$$H(J_x, J_y, J_z, g) = J_x \sum_i X_i X_{i+1} + J_y \sum_i Y_i Y_{i+1} + J_z \sum_i Z_i Z_{i+1} + g \sum_i X_i,$$

where J_x, J_y, J_z and g are real parameters and X_i, Y_i, and Z_i are Pauli matrices acting on qubit i.

Again, the ground state properties of $H(J_x, J_y, J_z, g)$ depend largely on the values of the parameters. As in the case of the quantum Ising model, setting $J_x = J_y = 0$ and $g = 0$ gives you the classical Max-Cut problem since the resulting Hamiltonian is diagonal. However, taking $J_x = J_y = J_z$ and $g = 0$, we find a non-diagonal Hamiltonian which we call "quantum Max-Cut".

In this case, the ground state becomes a non-trivial entangled state. To gain some intuition on how this arises, let's first focus on any pair of adjacent qubits. Along each edge $(i, i+1)$, the state of qubits i and $i+1$ that would get the lowest energy is the so-called singlet state

$$|\psi_-\rangle = \frac{1}{\sqrt{2}} (|01\rangle - |10\rangle)$$

which is one of the four Bell states. We can relate $|\psi_-\rangle$ to the EPR pair $|\phi_+\rangle = \frac{1}{\sqrt{2}} (|00\rangle + |11\rangle)$ by applying the Pauli X then Pauli Z operator on the first qubit. As the EPR pair, $|\psi_-\rangle$ is a maximally entangled state.

However, while each local term wants two consecutive qubits to be maximally entangled, this is impossible to satisfy globally: there is no quantum state where the state of qubit i and $i+1$ is maximally entangled, and the state of qubit $i+1$ and $i+2$ is also maximally entangled. This is a phenomenon called *monogamy of entanglement* that says that entanglement is not something that can be freely shared, unlike classical correlations (this is morally related to the no-cloning principle). Thus the ground state of the Heisenberg model gets really interesting because it's trying to satisfy all these local demands as best as possible, but it won't be able to do so perfectly.

It is worth noting that the 1D Ising model and Heisenberg model are rare examples of Hamiltonians that can be solved exactly using the Bethe ansatz. How exactly this is achieved is beyond the scope of this course and requires a lot beautiful mathematical physics – that's really a task for a condensed matter theory course.

3.2 Quantum complexity classes

We'll now take a step back and think about the problem more abstractly, and consider the task of solving *general* local Hamiltonians. Let's define the following decision problem, called k-LOCAL-HAM$_{a,b}$ where $a < b$ are real numbers. The instances of k-LOCAL-HAM$_{a,b}$ are going to be all k-local Hamiltonians $H = H_1 + \cdots + H_m$ such that the operator norm of each H_i is at most 1 (this is to ensure consistent normalization), and in the YES case the ground energy $\lambda_{\min}$ of H is at most a (i.e. the Hamiltonian's ground energy is "low"), and in the NO case the ground energy of H is at least b (the ground energy is "high"). We ignore all Hamiltonians that don't fall into either category. Formally, we write

YES instance	$\lambda_{\min}(H) \leq a$
NO instance	$\lambda_{\min}(H) \geq b.$

How is a local Hamiltonian presented? This is relevant to Problem 3 on the problem set. Well, there's a lot of flexibility, but here's a convenient way to describe local Hamiltonians:

1. the number of qubits n,
2. the number of terms m,
3. for each term $i = 1, \ldots, m$, we write what subset $S_i \subseteq [n]$ of k qubits the i-th term H_i is going to non-trivially act on, and then a description of the $2^k \times 2^k$ matrix h_i such that $H_i = h_i \otimes I^{\otimes n-k}$.

If k is small and $m = \text{poly}(n)$, then this presentation takes only $\text{poly}(n)$ bits to describe.

So we've just defined a decision problem that models an important task in physics: try to figure out the ground state energies of different Hamiltonians. Of course, this is not the only thing one does – you'd also like to figure out what the ground state is, what properties it has, and so on, but as a start you first want to figure out the minimum eigenvalue.

We now turn to the complexity of quantum decision problems. We proceed in analogy with the classical case. For instance, just like classical CSPs have NP-completeness, the quantum analogue for quantum CSPs is QMA-completeness, as we will see later. Let's first investigate the quantum analogue of P, which is BQP.

3.2.1 BQP

The quantum analogue of P is BQP (which stands for "Bounded-Error Quantum Polynomial Time"), which is the class of decision problems that can be decided by polynomial-time quantum algorithms with bounded error. That is, L is in BQP if there exists a polynomial time quantum algorithm A such that, if $x \in L_{yes}$, then $A(x) = 1$ with probability at least $2/3$, and if $x \in L_{no}$, then $A(x) = 1$ with probability at most $1/3$. The constants $2/3$ and $1/3$ are arbitrary, they can be set to any two distinct constants, say .99 and .01 – the gap between these numbers reflect the confidence you have in the output of the algorithm.

Quantum algorithms What do we mean by quantum algorithm? Well, there is something called a quantum Turing machine, but its definition is incredibly unwieldly and basically nobody uses it. Instead, people like to think about quantum circuits. So when we say polynomial-time quantum algorithm A, we really are referring to an infinite family of quantum circuits $\{A_n\}$ that are indexed by integers $n \in \mathbb{N}$. The circuits A_n have size at most $\text{poly}(n)$, and when the input is some n-bit string x, we run the circuit A_n on input $|x\rangle$ and some ancilla qubits $|0\rangle$. The output bit of the circuit is obtained by measuring the first qubit at the end of the circuit in the standard basis. An example of such circuit is shown on figure 3.1. So basically there's a different circuit for each input length. We also insist that the infinite array of circuits $\{A_n\}$ are *uniformly generated*, meaning that there's a *classical* Turing machine M that, when given input n, outputs the description of the n-th circuit A_n. This is to ensure that all of the circuits A_n are all "related to each other," and it's unlike having a completely different algorithm for each input length. If you're not so familiar with this distinction of uniform-vs-non-uniform family of circuits, don't worry about it, it won't be so essential to what we're talking about.

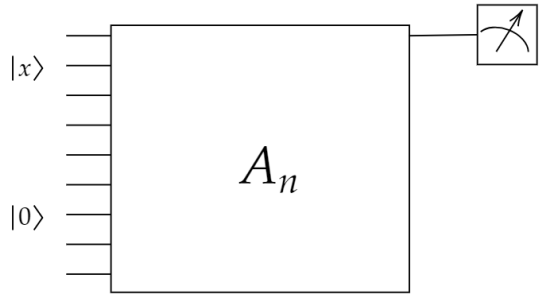


Figure 3.1: BQP Verifier Circuit A_n acting on the input $|x\rangle$ and some ancilla qubits initialized in $|0\rangle$.

3.2.2 QMA

The quantum analogue of NP, called QMA (which stands for "Quantum Merlin-Arthur"), is naturally defined as the following: a decision problem L is in QMA if there exists a family of polynomial-sized *verifier circuits* $\{A_n\}$ such that if $x \in L_{yes}$, then there exists a polynomial-sized *proof state* $|\psi\rangle$ where $\Pr[A_n \text{ accepts } |x\rangle \otimes |\psi\rangle] \geq 2/3$, where $n = |x|$, the number of qubits needed to encode $|x\rangle$, and if $x \in L_{no}$, then *for all* proof states $|\psi\rangle$ we have $\Pr[A_n \text{ accepts } |x\rangle \otimes |\psi\rangle] \leq 1/3$.

In other words, to determine whether x is a YES or NO instance, the verifier circuit gets a proof "from the sky" to help it determine which is the case – with the twist that the *proof* is now a quantum state!

The name "Quantum Merlin-Arthur" is based on medieval folklore about King Arthur and the Knights of the Round Table. King Arthur gets advice from the all-powerful wizard Merlin, but doesn't necessarily trust everything Merlin says. So Arthur has to *verify* what Merlin tells him. We think of Arthur as being a polynomial-time verifier, and Merlin as a *prover* who can solve any (computational) problem he likes. Merlin tries to convince Arthur of some statement X by sending Arthur a proof, which Arthur then checks.

In Quantum Merlin-Arthur, Arthur is a polynomial-time *quantum verifier*, and Merlin can send

quantum proofs to Arthur to verify.

3.2.3 A quantum notion of proofs

Quantum proofs define a really intriguing model of "mathematical proof". The traditional notion of a mathematical proof is that there's a formal statement X, which may or may not be true. Maybe X is something like "There are infinitely many primes" or "$\mathsf{P} \neq \mathsf{NP}$". And someone tries to convince you that X is true by giving you a table of text π, that you can check line-by-line whether π is a valid deduction, using the standard mathematical axioms that we've learned in math class, of the statement X.

However, with QMA, we've changed this notion of a proof: someone can try to convince you of the truth of some statement X by handing you a *quantum proof* $|\pi\rangle$. And to verify this proof, you can perform some quantum measurement on the state to determine whether you're convinced or not. However the quantum measurement do not need to resemble the traditional notion of line-by-line proof checking. It can be something more exotic, like first performing a Quantum Fourier Transform on $|\pi\rangle$.

Furthermore, the fact that the proof is a quantum state makes it very distinct from a traditional proof, because for one you can't *copy* the proof and share it with your friends, due to the No-Cloning Theorem. Also, it may be difficult to extract any information from the quantum proof other than the fact that X is true. Let's say you make a measurement on part of the quantum proof to try to "read" it. This will generally collapse the quantum proof, and change the state. So it's a really unusual notion of proof.

Despite their strangeness, we believe that quantum proofs can be much more efficient for verifying the truth of certain types of statements X than if you were forced to check an equivalent classical proof. Examples of such statements X include:

1. (**Local Hamiltonians**) "Local Hamiltonian H on n qubits has a ground state with energy less than $a = 0.1$."
2. (**Consistency of local density matrices**) "Here is a collection of two-qubit density matrices on every pair (i, j) of qubits: $\{\rho_{ij}\}$. There exists a global n-qubit state $|\psi\rangle$ where the reduced density matrix of $|\psi\rangle$ on qubits (i, j) is exactly ρ_{ij}."
3. (**Group Non-Membership**) "An element h of a finite group G is not in the subgroup generated by elements $g_1, \ldots, g_k \in G$". Here, think of G as having exponentially many elements.

As an example, we now detail this last problem.

3.2.4 Group Non-Membership Problem

Recall that a group G is a set with an identity element e, and it's closed under an invertible binary operation (that we'll call *multiplication*). If you have a subset of elements $\{g_1, \ldots, g_k\} \subseteq G$, the subgroup H generated by this subset is simply all possible products of the g_i's and their inverses.

Here is the problem, suppose you have some finite group G with $\exp(n)$ many elements, so you can represent each element $g \in G$ using $\text{poly}(n)$-many bits. Furthermore, given two elements $g, h \in G$, you can efficiently multiply them together to obtain a representation of $gh \in G$ in $\text{poly}(n)$ time, and also you can compute inverses g^{-1} in $\text{poly}(n)$ time.

Now suppose you're given elements $\{g_1, \ldots, g_k\} \subseteq G$, and an element $h \in G$. One question you might be interested in is the **Group Membership Problem**: determine whether h is in the subgroup H generated by $\{g_1, \ldots, g_k\}$. How might someone convince you that $h \in H$? Well, they could give you a sequence of products of the g_i's and their inverses that multiply to h. That would be fine, except this sequence of products might be exponentially long – so this wouldn't constitute a polynomial-sized proof.

Fortunately, there's a nontrivial result in group theory that ensures there is always a polynomial-length sequence of products of generators and their inverses that multiply to any given subgroup element h. So this implies that the Group Membership Problem is in NP.

What about the opposite problem, the **Group *Non*membership Problem**? Here, you're given generators for a subgroup H and an element h from the parent group G. How can someone convince you that h is *not* in H? This seems to be a much harder problem, because it's basically asking for the *non-existence* of a way to multiply elements of H together to get h. Proving non-existence seems to be much harder than proving existence.

We don't know what the classical complexity of the Group Nonmembership (GNM) Problem is (it might even be in NP), but it turns out using *quantum proofs* it is possible to efficiently solve the GNM problem.

Here's what Arthur (the quantum polynomial-time verifier) does when he gets a purported proof $|\pi\rangle$, which he is going to treat as a superposition $\sum_g \alpha_g |g\rangle$ over group elements of G.

1. He initializes an additional qubit Q in the state $|+\rangle = \frac{1}{\sqrt{2}}(|0\rangle + |1\rangle)$.
2. He flips a coin. If it is heads, then he performs the **Uniformity test**. Otherwise, if the coin is tails, he performs the **Membership test**.
3. **Uniformity test**
   (a) Arthur samples a random element b of the subgroup H (one can do this in polynomial time due to a result of Babai).
   (b) Controlled on this qubit Q, he's going to coherently left-multiply the proof $|\pi\rangle$ by this element b. In other words, if the qubit Q is in the state $|0\rangle$, then Arthur does nothing. If it's in the state $|1\rangle$, then he performs the following unitary on the proof register: $|a\rangle \mapsto |ba\rangle$.
   (c) He applies a Hadamard gate to the qubit Q, and measures Q in the standard basis. If the outcome is $|1\rangle$, he REJECTs the result. Otherwise, he ACCEPTs.
4. **Membership test**, for which the quantum circuit can be seen in figure 3.2.
   (a) Controlled on this qubit Q, he's going to coherently left-multiply the proof $|\pi\rangle$ by this element h.
   (b) He applies a Hadamard gate to the qubit Q, and measures Q in the standard basis. If the outcome is $|1\rangle$, then ACCEPT. Otherwise, REJECT.

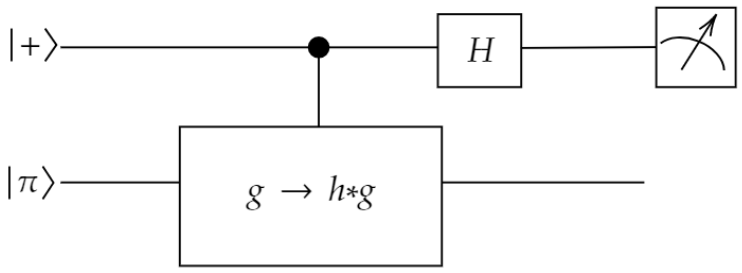


Figure 3.2: Quantum circuit that implements the membership test.

To see why this procedure works, let's work out the YES and NO instances of the problem.

YES case Here, the YES case is that h is *not* an element of H. We therefore need to argue that there is a proof $|\pi\rangle$ that Merlin could send to convince Arthur of this fact. This proof is easy to describe: it's the uniform superposition over H:

$$|\pi\rangle = \frac{1}{\sqrt{|H|}} \sum_{g \in H} |g\rangle \,.$$

Note: this state is easy to describe, but as far as we can tell, it cannot be created in polynomial time by a quantum computer – it requires quantum sorcery by Merlin.

Arthur is going to perform one of two tests at random. The first test is to check whether $|\pi\rangle$ is indeed a uniform superposition over the subgroup H. Indeed, while Arthur isn't able to create this uniform superposition on his own, he can verify it by checking whether the state $|\pi\rangle$ is invariant under left-multiplication by a random element of H. Certainly, multiplying H by elements of itself leaves it invariant, so we can analyze what happens. The state $|\pi\rangle$ will therefore always pass the uniformity test.

What about the Membership test? Since $h \notin H$, left-multiplying the state $|\pi\rangle$ by h will yield a uniform superposition of elements that are disjoint from H. Thus the resulting state $|\sigma\rangle$ will be *orthogonal* to $|\pi\rangle$. Therefore, after the controlled-multiplication operation the state of the algorithm is

$$\frac{1}{\sqrt{2}} \left(|0\rangle \otimes |\pi\rangle + |1\rangle \otimes |\sigma\rangle \right).$$

Performing a Hadamard gate on the first qubit we get

$$\frac{1}{2} \left(|0\rangle \otimes (|\pi\rangle + |\sigma\rangle) + |1\rangle \otimes (|\pi\rangle - |\sigma\rangle) \right).$$

Since $|\pi\rangle$ and $|\sigma\rangle$ are orthogonal, the outcome of measuring qubit Q will be $|0\rangle$ with probability $1/2$ and $|1\rangle$ with probability $1/2$. Therefore, Arthur will ACCEPT with probability $1/2$ if he performs the Membership test. Overall, Arthur will ACCEPT with probability at least $3/4$ in the YES case.

NO case We won't go into the proof here, but in the NO case (when $h \in H$), if Merlin sends Arthur the uniform superposition over H, then Arthur will reject 100% of the time in the membership test. What if Merlin tries to be more devious and sends some other quantum state? One

can show that no proof $|\pi\rangle$ can Merlin send that will get Arthur to accept the membership test with probability greater than 1/3. The uniformity test, on the other hand, doesn't depend on the case, so Arthur will still always accept as long as $|\pi\rangle$ is the superposition that we expect. Overall, Arthur will have a 50% chance of accepting in the NO case. Finally, this shows that GNM is a problem in QMA.

Note: The probabilities of accepting are 75% and 50% for $x \in L_{yes}$ and $x \in L_{no}$, respectively. 50% is higher than our previous 1/3 criteria, but as long as there is a gap between the two probabilities, the proof still holds.

We do not know how to convince someone of the truth of these statements using classical proofs only, unless the classical proofs are exponentially long. However, it is possible to use polynomial-sized quantum proofs to certify the truth of these statements.

This fact is very interesting because we're somehow taking advantage of the exponentiality of quantum states in a concrete and useful way. Naively, one might think that since an n-qubit quantum state $|\psi\rangle$ requires 2^n parameters to describe, one can compress a 2^n-sized classical string (such as a proof) into an n-qubit state $|\psi\rangle$ and transmit exponential amounts of information that way. However, that's not possible because of Holevo's theorem: when the recipient tries to measure this state $|\psi\rangle$, it collapses and the recipient can only recover at most n qubits of classical information. In fact, an exponential amount of measurements is required to fully determine the 2^n parameters. This process is known as *state tomography*.

On the other hand, if we're not concerned with transmitting messages but doing verification of proofs, then we can use this exponentiality of quantum states to our advantage. It's much more sophisticated than stuffing an exponentially-long classical string into the amplitudes of a quantum state.

3.2.5 Local Hamiltonians are in QMA

Let's go back to the k-LOCAL-HAM$_{a,b}$ problem. Some questions arise: is there an easy classical way to convince someone of a YES instance of the Local Hamiltonians problem? Perhaps by providing a ground state?

The answer is no. Indeed, a quantum ground state consists of n qubits and, classically, this may take 2^n parameters to describe. Measuring the energy of that quantum state classically would require exponentially large matrix multiplications. The classical proof is simply too large to check. However, this task can be achieved more efficiently on a quantum computer. More precisely, we need a quantum verifier that measures or estimates the energy of a state with respect to H. There are a couple ways of doing this:

Local measurements. In the k-LOCAL-HAM$_{a,b}$ problem, we want to distinguish between

$$\langle\psi| H |\psi\rangle \leq a \qquad \text{or} \qquad \langle\psi| H |\psi\rangle \geq b.$$

Expanding, we have

$$\langle\psi| H |\psi\rangle = \sum_{i=1}^{m} \langle\psi| H_i |\psi\rangle \ .$$

Notice that H_i is a Hermitian matrix (since it's a Hamiltonian term), and therefore we can interpret H_i as an observable that corresponds to a measurement. Since H_i is k-local, we can write

$$H_i = h_i \otimes I$$

for some $2^k \times 2^k$ Hermitian matrix h_i that acts on a subset $S_i \subseteq [n]$ of k qubits. We can diagonalize h_i as

$$h_i = \sum_j \lambda_{ij} P_{ij}$$

for some k-qubit projective measurement $\{P_{ij}\}$. Therefore, we can interpret

$$\langle\psi| H_i |\psi\rangle = \sum_j \lambda_{ij} \langle\psi| P_{ij} \otimes I |\psi\rangle$$

as denoting the average of the "weights" $\{\lambda_{ij}\}_j$ if λ_{ij} is sampled with probability $\langle\psi| P_{ij} \otimes I |\psi\rangle$, which is simply the probability of obtaining outcome j if we measure the S_i qubits of $|\psi\rangle$ using the projective measurement $\{P_{ij}\}_j$. Since $k = O(1)$, this projective measurement can be performed efficiently on a quantum computer.

This suggests the following algorithm for Arthur, where we will set T to some large quantity of the order of poly(n).

1. We assume that the input is T copies of $|\psi\rangle$.
2. Set the energy $E = 0$.
3. For $t = 1, \ldots, T$:
   (a) Pick a uniformly random $i \in [m]$.
   (b) Load a fresh copy of $|\psi\rangle$, and measure the S_i qubits using the projective measurement $\{P_{ij}\}$. If outcome j is obtained, then set $X_t = m\lambda_{ij}$.
4. Let $E = \frac{1}{T}\sum_{t=1}^{T} X_t$.
5. If $E \leq a$, then ACCEPT. Otherwise, REJECT.

YES case Suppose there is a ground state $|\psi\rangle$ of energy less than a (YES case). Then Merlin can provide T copies of this ground state to Arthur. Arthur tries to estimate $\langle\psi| H |\psi\rangle$ very precisely by repeating the following many times : picking a random term $i \in [m]$, and measuring the observable H_i, and estimating the average energy. The random variable X_t has average

$$\frac{1}{m}\sum_{i=1}^{m}\sum_{j=1}^{2^k} \langle\psi| P_{ij} \otimes I |\psi\rangle \cdot m\lambda_{ij} = \langle\psi| H |\psi\rangle .$$

Each X_t is independently realized, so the average $\frac{1}{T}\sum_t X_t$ converges to $\langle\psi| H |\psi\rangle$ exponentially fast in T (by standard law-of-large-numbers/concentration bounds type arguments).

Therefore, with high probability, we find

$$\langle\psi| H |\psi\rangle - \varepsilon \leq \frac{1}{T}\sum_t X_t \leq \langle\psi| H |\psi\rangle + \varepsilon.$$

By taking $T = \text{poly}(n)$ sufficiently large, we can guarantee with high probability that $\epsilon \ll b - a$, the gap we're trying to distinguish (we have to assume that $b - a \geq 1/\text{poly}(n)$). Thus Arthur can tell whether $\langle\psi| H |\psi\rangle$ is larger than b or at most a, with high probability.

Question: We've used a few assumptions here: the Hamiltonian is reasonably local, *i.e.* $k = O(1)$, $b - a \geq 1/\text{poly}(n)$ (the gap can't be too small). What about the assumption that the norm of the terms H_i is at most 1? Where does that come in? This ensures that the λ_{ij} eigenvalues don't get too large – they're at most 1, in fact. This means that the *variance* of the random variables X_t is controlled, and therefore we can obtain good bounds on the number of times we need to estimate the energy before the law of large numbers applies.

NO case Suppose for all states $|\psi\rangle$, the energy $\langle\psi| H |\psi\rangle \geq b$. Then if Merlin really provides a T tensor product copies of a state $|\psi\rangle$, then this energy estimation algorithm will realize that the energy is at least b, and Arthur will reject.

What if Merlin were more devious, however, and tried to give some other kind of quantum state that would screw up Arthur's verification? For example, what if instead of giving mT copies of the same state, what if Merlin switched it up and gave a tensor product of different states (i.e. $|\psi_1\rangle \otimes |\psi_2\rangle \otimes \cdots$)? That's fine – Arthur's estimate of the energy will still be high. That's because no matter what $|\psi_t\rangle$ Merlin cooks up, the average value of the random variable X_t is going to be at least b.

The more nefarious kind of thing Merlin could do is to send giant entangled state $|\Lambda\rangle$ on nT qubits that doesn't factor into a tensor product of states, and use this entanglement to confuse Arthur into thinking that the ground energy of H is at most a. Could this kind of entanglement mess things up for Arthur?

Fortunately, Merlin can't cheat this way. This requires a short proof to argue that entanglement can't help Merlin cheat in this way, but we won't cover it here (it might show up on the problem set). For now we can just accept this is true.

Phase estimation Just to mention in passing, there's another way to do the energy estimation, and that's to use the phase estimation algorithm. This requires a more sophisticated algorithm that actually requires using Hamiltonian simulation as a black box, but the benefit of doing so is that Merlin only needs to send one copy of the purported ground state $|\psi\rangle$ (rather than many copies). However we won't dwell on this here – if you're interested I encourage you to ponder how this algorithm might work.

3.3 Quantum Cook-Levin Theorem

So far, so good. We've followed the classical-quantum dictionary pretty well. Local Hamiltonians are the quantum analogue of CSPs. QMA is the quantum analogue of NP. Just as CSPs are NP problems, local Hamiltonians are in QMA. Clearly the next thing we want to explore is an analogue of NP-completeness: just as 3SAT is complete for NP, is there a version of the local Hamiltonians problem that are QMA-complete?

There is, and this brings us to the quantum analogue of the Cook-Levin theorem. In particular,

we're going to show for every decision problem L in QMA, there exists a polynomial-time computable transformation from instances x of L to instances $H = H_1 + \cdots + H_m$ of k-LOCAL-HAM$_{a,b}$ for certain k, a, b such that if $x \in L_{yes}$, then the ground energy of H, λ_{min}, is at most a, and if $x \in L_{no}$, then the ground energy of H is at least b.

$$\lambda_{min} = \begin{cases} \leq a, & \text{if } x \in L_{yes} \\ \geq b, & \text{if } x \in L_{no} \end{cases}$$

Since $L \in$ QMA there exists a family of verifier circuits $\{V_n\}$, one for each input length $n \in \mathbb{N}$, such that for $n = |x|$, if $x \in L_{yes}$ then there exists a quantum proof $|\psi\rangle$ such that V_n accepts $|x\rangle \otimes |\psi\rangle$ with probability at least $2/3$, and otherwise for all quantum states $|\psi\rangle$ the verifier V_n accepts $|x\rangle \otimes |\psi\rangle$ with probability at most $1/3$.

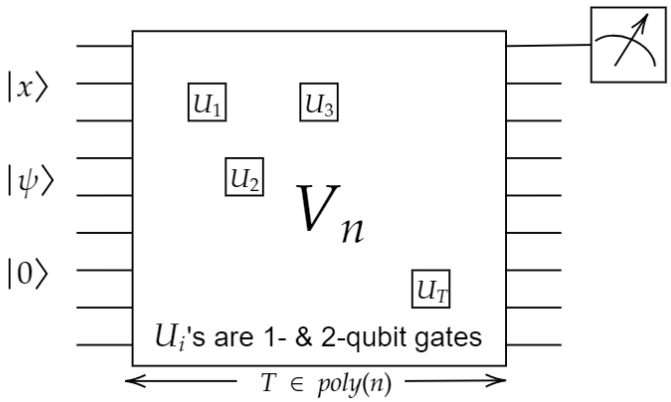


Figure 3.3: Verifier Circuit V_n composed of many one- and two-qubit unitaries U_i.

We're going to just fix n and omit it for notational clarity.

Let $U_1, U_2, \ldots, U_T$ denote the single- and two-qubit gates of the circuit V_n.

We're going to create a quantum version of the Cook-Levin 3SAT instance and the Cook-Levin tableau. From x we're going to design a local Hamiltonian H – called the *Feynman-Kitaev Hamiltonian*, named after Richard Feynman and Alexei Kitaev who developed this idea – such that any ground state of H has the following form:

$$|\Omega\rangle = \frac{1}{\sqrt{T+1}} \sum_{t=0}^{T} |t\rangle \otimes |\Omega_t\rangle$$

where

$$|\Omega_t\rangle = U_t U_{t-1} \cdots U_1 (|x\rangle \otimes |\psi\rangle \otimes |0\rangle)$$

for some quantum proof $|\psi\rangle$ that maximizes the probability that the verifier circuit V accepts $|x\rangle \otimes |\psi\rangle$.

These states are called *history states*, because they're superpositions over the *history* of the verifier circuit V as each gate is being applied, starting with the initial state $|x\rangle \otimes |\psi\rangle \otimes |0\rangle$. This is like

the Cook-Levin tableau, except the snapshots are in superposition. Furthermore, the $|t\rangle$ state is a *clock register* – it keeps track of which timestep the snapshot $|\psi_t\rangle$ is in.

Just like the Cook-Levin SAT formula consists of a large collection of local SAT constraints which altogether enforce that the underlying variables must satisfy Starts OK, Evolves OK, Ends OK, the Feynman-Kitaev Hamiltonian will also have a large collection of Hamiltonian terms that altogether enforce the underlying qubits of the state satisfy the following quantum analogue:

1. (**Starts OK**) The initial snapshot state $|\Omega_0\rangle = |x\rangle \otimes |\psi\rangle \otimes |0\rangle$ for some quantum state $|\psi\rangle$.
2. (**Evolves OK**) Each pair of consecutive snapshot states are related by the following $|\Omega_t\rangle = U_t |\Omega_{t-1}\rangle$.
3. (**Ends OK**) Measuring the output qubit of the final snapshot state $|\Omega_T\rangle$ yields $|1\rangle$ with high probability.

Suppose we had a quantum state $|\Omega\rangle = \frac{1}{\sqrt{T}} \sum_{t=0}^{T} |t\rangle \otimes |\Omega_t\rangle$ satisfying all of these quantum constraints. Then we can conclude that, just like in the classical case, there exists a quantum proof $|\psi\rangle$ such that if you executed the verifier V on input $|x\rangle \otimes |\psi\rangle \otimes |0\rangle$, the verifier would accept with high probability, thus certifying $x \in L_{yes}$.

What do these Hamiltonian terms look like? Let's divide up our qubit space into different registers: C register, which consists of $O(\log T)$-qubits, for the clock register. X register, to hold the initial input $|x\rangle$. P register, to hold the initial proof $|\psi\rangle$. And the A register, to hold the ancillas $|0\rangle$.

For the "Starts OK", we need to ensure that the X register of $|\Omega_0\rangle$ is in the $|x\rangle$ state, and that the A register qubits are in the $|0\rangle$ state. We can enforce the $|x\rangle$ part by using terms of the form

$$H_i^{(X)} = |0\rangle\langle 0|_C \otimes |\overline{x_i}\rangle\langle \overline{x_i}|_{X,i}$$

for $i = 1, 2, \ldots, n$, where $|0\rangle\langle 0|_C$ is the projector onto the clock being in the $|0\rangle$ state, and $|1\rangle\langle 1|_{X,i}$ is the projector onto the i-th qubit of the X register being in $|\overline{x_i}\rangle$ state, where $\overline{x_i}$ is the *complement* of x_i.

What this is saying is, the ground states of $H_i^{(in)}$ are those where either the clock register is not in the $|0\rangle$ time, in which case we don't care what's going on. Otherwise, if the clock register is at time $t = 0$, then the i-th qubit of the X register better not in $|\overline{x_i}\rangle$ state, so in other words it should be in the $|x_i\rangle$ state.

Similarly, to enforce the ancillas, we can have

$$H_i^{(A)} = |0\rangle\langle 0|_C \otimes |1\rangle\langle 1|_{A,i}.$$

The "Ends OK" term is very simple:

$$H_{End} = |T\rangle\langle T|_C \otimes |0\rangle\langle 0|_O$$

where O is the output qubit.

The "Evolves OK" terms are more interesting: for every $t = 0, 1, 2, \ldots, T$,

$$H^{(t \to t+1)} = \frac{1}{2}\Big(|t\rangle\langle t|_C \otimes I + |t+1\rangle\langle t+1|_C \otimes I - |t+1\rangle\langle t|_C \otimes U_{t+1} - |t\rangle\langle t+1|_C \otimes U_{t+1}^\dagger \Big).$$

Thus our Feynman-Kitaev Hamiltonian is the sum

$$H = \sum_{i=1}^{n} H_i^{(X)} + \sum_{j=1}^{\#\text{ ancilla}} H_j^{(A)} + \sum_{t=0}^{T-1} H^{(t \to t+1)} + H_{End}.$$

Chapter 4

QMA-completeness continued

Scribes: Logan Murphy, Yuval Efron

4.1 QMA and its properties

In the last lecture, we showed that the local hamiltonians problem (LH) is in the complexity class QMA, and began talking about a quantum analogue of the Cook-Levin Theorem. Before we continue that discussion, let's go over some basic properties of QMA.

Let's recall what it means for a decision problem L to be in QMA. In the last lecture, we said that $L \in \mathsf{QMA}$ if there exists a family of verifier circuits $\{V_n\}$, one for each input length $n \in \mathbb{N}$, given and instance x with $|x| = n$,

- if $x \in L_{yes}$ then there exists a quantum proof $|\psi\rangle$ such that V_n accepts $|x\rangle \otimes |\psi\rangle \otimes |0...0\rangle$ with probability at least $2/3$ (this number is called the *completeness*, and the $|0..0\rangle$ refers to some amount of ancilla bits that might be required).
- otherwise for all quantum states $|\psi\rangle$ the verifier V_n accepts $|x\rangle \otimes |\psi\rangle \otimes |0...0\rangle$ with probability at most $1/3$ (this number is called the *soundness*).

As mentioned before, the $2/3$ vs $1/3$ probability bounds to distinguish between the YES and NO instances is not so crucial for the definition of QMA. We will now state this fact formally.

Let $\mathsf{QMA}_{a,b}$ denote the set of all problems where all the YES instances have quantum proofs that are accepted with probability at least a and all NO instances are accepted with probability at most b no matter what the proof is. Our claim is that

$$\mathsf{QMA}_{a,b} = \mathsf{QMA}_{2/3,1/3}$$

as long as $a - b \geq 1/\text{poly}(n)$. This is because we can *amplify* the completeness-soundness gap from something that's inverse-polynomial to any constant we like, simply by *repeating* the verification procedure. In fact, we can make the completeness exponentially close to 1 and soundness exponentially close to 0. Let's now prove that this is true.

Suppose we have a verifier V whose completeness is a and whose soundness is b, where $a - b \geq 1/\text{poly}(n)$. Let's say that it expects a proof state consisting of $m = \text{poly}(n)$ qubits.

We can construct a second verifier V' for the same language L whose completeness and soundness are much, much better. Construct V' as follows:

1. V' should expect a proof state of the form $|\psi_1\rangle \otimes |\psi_2\rangle \otimes \cdots \otimes |\psi_T\rangle$, where each $|\psi_i\rangle$ is m qubits.(Of course, a dishonest Merlin could give an arbitrary proof state on mT qubits). We will specify exactly what T should be shortly.
2. Set a counter $X = 0$.
3. For $t = 1, \ldots, T$, execute the verifier circuit V on input $|x\rangle \otimes |\psi_t\rangle \otimes |0\rangle$ (where $|\psi_t\rangle$ denotes the t-th block of m qubits of the proof state), and measure the output qubit of V. If the output is 1, then increment X.
4. If $X \geq (a - \epsilon)T$, then accept. Otherwise, reject.

This verifier should look quite similar to the way we showed that k-LOCAL-HAM is in QMA. Having specified the verifier, let's now analyze its accept/reject behaviour for the decision problem L.

YES case: suppose that $x \in L_{yes}$. What is the maximum acceptance probability of this amplified verifier V'?

Well, let's take an m-qubit proof $|\psi\rangle$ that V would've accepted with probability at least a. Then let's feed the proof $|\psi\rangle^{\otimes T}$ to V'. Then we're verifying T independent copies of the proof and estimating how often the verifier circuit V accepts.

Each verification is denoted by an independent random variable X_t that is 1 if the t-th verification accepts and 0 if it doesn't. The average of X_t is the probability of acceptance of V, which is at least a. So by a Chernoff/Hoeffding bound, we have that

$$\Pr\Big[\sum_{t=1}^{T} X_t < (a - \varepsilon)T\Big] \leq \Pr\Big[\sum_{t=1}^{T} X_t < \sum_{t=1}^{T} \mathbb{E}[X_t] - \epsilon T\Big] \leq \exp(-c\epsilon^2 T)$$

for some universal constant $c > 0$.

Thus, the probability that $\sum_t X_t \geq (a - \epsilon)T$ is at least $1 - \exp(-c\varepsilon^2 T)$.

NO case: again for simplicity let's assume that the proof given by Merlin is of the form $|\psi_1\rangle \otimes |\psi_2\rangle \otimes \cdots \otimes |\psi_T\rangle$. The average of $\sum_{t=1}^{T} \mathbb{E}[X_t]$ is going to be at most bT, because each $|\psi_t\rangle$ is accepted with probability at most b. Using the same Chernoff bound, we have

$$\Pr\Big[\sum_{t=1}^{T} X_t > (b + \varepsilon)T\Big] \leq \Pr\Big[\sum_{t=1}^{T} X_t > \sum_{t=1}^{T} \mathbb{E}[X_t] + \epsilon T\Big] \leq \exp(-c\epsilon^2 T)\ .$$

So the probability that V' accepts is going to be at most $\exp(-c\epsilon^2 T)$.

If we set $\epsilon = \frac{a-b}{2}$, and $T = \frac{n}{c\epsilon^2}$, then the completeness of V' is $1 - \exp(-n)$ and the soundness is $\exp(-n)$, which is quite good.

What if Merlin doesn't send a tensor product proof state? That's a bit harder to analyze, but it turns out that sending a giant entangled state doesn't help Merlin, and the soundness parameter will still be exponentially small, i.e., $\exp(-n)$.

So the soundness and completeness parameters we use to define QMA are pretty flexible. We'll now use this result to consider the Quantum Cook-Levin Theorem in more detail. Just like how the original Cook-Levin Theorem showed that any problem in NP can be reduced to SAT, we will show that any problem in QMA can be reduced to LH.

4.2 Quantum Cook-Levin Theorem

Let's turn to the Quantum Cook-Levin Theorem, which we started talking about last time. We have a decision problem $L \in \mathsf{QMA}$ with a corresponding verifier V. As we just saw, we can assume without loss of generality that the completeness of V is $1 - \exp(-n)$ and the soundness is $\exp(-n)$.

We're going to just fix an input length n and omit it for notational clarity. Let $U_1, U_2, \ldots, U_T$ denote the single- and two-qubit gates of the circuit $V = V_n$.

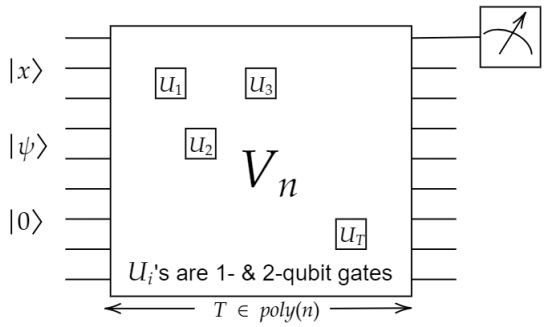


Figure 4.1: Alexandre and Hao's diagram for V_n (previous lecture notes)

We're going to map instances x of L to a local Hamiltonian H (called the *Feynman-Kitaev Hamiltonian*) such that

1. The locality k of the Hamiltonian is $O(\log T)$.
2. Ground states of H are *history states* of the circuit V

$$|\Omega\rangle = \frac{1}{\sqrt{T+1}} \sum_{t=0}^{T} |t\rangle \otimes |\Omega_t\rangle$$

   where $|\Omega_t\rangle$, which we call the *snapshot state*, is

$$|\Omega_t\rangle = U_t U_{t-1} \cdots U_1(|x\rangle \otimes |\psi\rangle \otimes |0 \cdots 0\rangle)$$

   for some quantum proof $|\psi\rangle$ that maximizes the probability that the verifier circuit V accepts $|x\rangle \otimes |\psi\rangle$, and t is a $\log T$ qubit register which stores the snapshot we are recording.
3. If $x \in L_{yes}$, then the ground energy of H will be *at most* $\exp(-n)$.

4. If $x \in L_{no}$, then the ground energy of H will be *at least* $\frac{c}{T^3}$ for some constant $c > 0$.

Since $T = \text{poly}(n)$, the gap between the ground energies in the YES and NO cases is inverse polynomial. This implies that k-LOCAL-HAM$_{a,b}$ is QMA-complete for $a - b \geq 1/\text{poly}(n)$.

The ground states of the Feynman-Kitaev Hamiltonian are called history states because they're superpositions over the *history* of the verifier circuit V as each gate is being applied, starting with the initial state $|x\rangle \otimes |\psi\rangle \otimes |0 \cdots 0\rangle$. This is like the Cook-Levin tableau, except the snapshots are in superposition. Recall, the $|t\rangle$ state is a *clock register* – it keeps track of which timestep the snapshot $|\Omega_t\rangle$ is in.

Recall that the Cook-Levin SAT formula consists of a large collection of local SAT constraints which, together, enforce that the underlying variables must satisfy **Starts OK**, **Evolves OK**, and **Ends OK**.

Similarly, our Feynman-Kitaev Hamiltonian will also have a large collection of Hamiltonian terms that altogether enforce the underlying qubits of the state satisfy the following quantum analogue:

1. (**Starts OK**) The initial snapshot state $|\Omega_0\rangle = |x\rangle \otimes |\psi\rangle \otimes |0 \cdots 0\rangle$ for some quantum state $|\psi\rangle$.
2. (**Evolves OK**) Each pair of consecutive snapshot states are related by the following $|\Omega_t\rangle = U_t |\Omega_{t-1}\rangle$.
3. (**Ends OK**) Measuring the output qubit of the final snapshot state $|\Omega_T\rangle$ yields $|1\rangle$ with high probability.

Suppose we had a quantum state $|\Omega\rangle = \frac{1}{\sqrt{T}} \sum_{t=0}^{T} |t\rangle \otimes |\Omega_t\rangle$ satisfying all of these quantum constraints. Then we can conclude that, just like in the classical case, there exists a quantum proof $|\psi\rangle$ such that if you executed the verifier V on input $|x\rangle \otimes |\psi\rangle \otimes |0 \cdots 0\rangle$, the verifier would accept with high probability, thus certifying $x \in L_{yes}$.

What do these Hamiltonian terms look like? Let's divide up our qubit space into different registers:

- the C register, which consists of $O(\log T)$-qubits, for the clock register,
- the X register, to hold the initial input $|x\rangle$,
- the P register, to hold the initial proof $|\psi\rangle$, and
- the A register, to hold the ancillas $|0 \cdots 0\rangle$.

For the "Starts OK", we need to ensure that the X register of $|\Omega_0\rangle$ is in the $|x\rangle$ state, and that the A register qubits are in the $|0 \cdots 0\rangle$ state. We can enforce the $|x\rangle$ part by using terms of the form

$$H_i^{(X)} = |0\rangle\langle 0|_C \otimes |\overline{x_i}\rangle\langle \overline{x_i}|_{X,i}$$

for $i = 1, 2, \ldots, n$, where $|0\rangle\langle 0|_C$ is the projector onto the clock being in the $|0\rangle$ state, and $|\overline{x_i}\rangle\langle \overline{x_i}|_{X,i}$ is the projector onto the i-th qubit of the X register being in $|\overline{x_i}\rangle$ state, where $\overline{x_i}$ is the *complement* of x_i.

What this is saying is, the ground states of $H_i^{(in)}$ are those where either the clock register is not in the $|0\rangle$ time, in which case we don't care what's going on. Otherwise, if the clock register is at time $t = 0$, then the i-th qubit of the X register had better not be in the $|\overline{x_i}\rangle$ state, so in other words it should be in the $|x_i\rangle$ state.

Similarly, to enforce the ancillas, we can have

$$H_i^{(A)} = |0\rangle\langle 0|_C \otimes |1\rangle\langle 1|_{A,i}.$$

The "Ends OK" term is very simple:

$$H_{End} = |T\rangle\langle T|_C \otimes |0\rangle\langle 0|_O$$

where O is the output qubit.

The "Evolves OK" terms are more interesting: for every $t = 0, 1, 2, \ldots, T$,

$$H^{(t\to t+1)} = \frac{1}{2}\Big(|t\rangle\langle t|_C \otimes I + |t+1\rangle\langle t+1|_C \otimes I - |t+1\rangle\langle t|_C \otimes U_{t+1} - |t\rangle\langle t+1|_C \otimes U_{t+1}^\dagger\Big).$$

Thus our Feynman-Kitaev Hamiltonian is the sum

$$H = \sum_{i=1}^{n} H_i^{(X)} + \sum_{j=1}^{\#\text{ ancilla}} H_j^{(A)} + \sum_{t=0}^{T-1} H^{(t\to t+1)} + H_{End}.$$

Locality. The locality of the Feynman-Kitaev Hamiltonian is $O(\log T)$, because each term has a part that examines the "clock" register, which is $O(\log T)$ qubits wide, and then has a part that acts a $O(1)$-qubits of the "snapshot register".

The locality of this construction can be improved to constant (e.g., 3) using a few tricks (see, e.g., [KR03]).

YES case. Let's verify that the ground energy is indeed $\exp(-n)$ in the YES case.

Let $x \in L_{yes}$ and let $|\psi\rangle$ be a quantum proof for x that is accepted by V with probability at least $1 - \exp(-n)$.

Consider the history state $|\Omega\rangle$ for V on input $|x\rangle \otimes |\psi\rangle \otimes |0\cdots 0\rangle$. Its energy with respect to H is

$$\langle\Omega| H |\Omega\rangle = \sum_{i=1}^{n} \langle\Omega| H_i^{(X)} |\Omega\rangle + \sum_{j=1}^{\#\text{ ancilla}} \langle\Omega| H_j^{(A)} |\Omega\rangle + \sum_{t=0}^{T-1} \langle\Omega| H^{(t\to t+1)} |\Omega\rangle + \langle\Omega| H_{End} |\Omega\rangle.$$

We just need to check that the sum of all these expectation values is at most $\exp(-n)$.

1. $H^{(X)}$ terms. Fix an i. Then we have

$$\begin{aligned}
\langle\Omega| H_i^{(X)} |\Omega\rangle &= \frac{1}{T+1}\sum_{s,t}(\langle s|\otimes\langle\Omega_s|)H_i^{(X)}(|t\rangle\otimes|\Omega_t\rangle) \\
&= \frac{1}{T+1}\sum_{s,t}(\langle s|\otimes\langle\Omega_s|)\cdot(|0\rangle\langle 0|\otimes|\overline{x_i}\rangle\langle\overline{x_i}|_{X,i})\cdot(|t\rangle\otimes|\Omega_t\rangle) \\
&= \frac{1}{T+1}\sum_{s,t}\langle s|\ |0\rangle\langle 0|\ |t\rangle\cdot\langle\Omega_s|\ |\overline{x_i}\rangle\langle\overline{x_i}|_{X,i}\ |\Omega_t\rangle \\
&= \frac{1}{T+1}\langle\Omega_0|\ |\overline{x_i}\rangle\langle\overline{x_i}|_{X,i}\ |\Omega_0\rangle
\end{aligned}$$

   The third line comes from reorganizing the bra's and ket's according to which tensor factor they live in, and the fourth comes from the fact that $\langle s|\,|0\rangle\langle 0|\,|t\rangle$ is nonzero only when $s = t = 0$. Since $|\Omega_0\rangle = |x\rangle_X \otimes |\psi\rangle_W \otimes |0\cdots 0\rangle_A$, and the projector $|\overline{x}_i\rangle\langle\overline{x}_i|$ only acts on the i-th qubit of the X register, which is in the state $|x_i\rangle$, so this inner product is 0.

2. Same thing for ancilla terms.

3. Let's check evolution terms. Fix a time $r \in \{0, 1, \ldots, T-1\}$.

$$\begin{aligned}
&\langle\Omega| H^{(r\to r+1)} |\Omega\rangle \\
&= \frac{1}{T+1}\sum_{s,t}(\langle s|\otimes\langle\Omega_s|)H^{(r\to r+1)}(|t\rangle\otimes|\Omega_t\rangle) \\
&= \frac{1}{2(T+1)}\sum_{s,t}(\langle s|\otimes\langle\Omega_s|)\cdot\Big(|r\rangle\langle r|_C\otimes I + |r+1\rangle\langle r+1|_C\otimes I \\
&\qquad\qquad - |r+1\rangle\langle r|_C\otimes U_{r+1} - |r\rangle\langle r+1|_C\otimes U_{r+1}^\dagger\Big)\cdot(|t\rangle\otimes|\Omega_t\rangle) \\
&= \frac{1}{2(T+1)}\sum_{s,t}\langle s|\,|r\rangle\langle r|\,|t\rangle\cdot\langle\Omega_s|\Omega_t\rangle + \langle s|\,|r+1\rangle\langle r+1|\,|t\rangle\cdot\langle\Omega_s|\Omega_t\rangle \\
&\qquad\qquad - \langle s|\,|r+1\rangle\langle r|\,|t\rangle\cdot|\Omega_s\rangle\,U_{r+1}\,|\Omega_t\rangle - \langle s|\,|r\rangle\langle r+1|\,|t\rangle\cdot\langle\Omega_s|\,U_{r+1}^\dagger\,|\Omega_t\rangle
\end{aligned}$$

   We can compute the four terms of each sum:

$$\sum_{s,t}\langle s|\,|r\rangle\langle r|\,|t\rangle\cdot\langle\Omega_s|\Omega_t\rangle = \langle\Omega_r|\Omega_r\rangle = 1$$

   For the second term:

$$\sum_{s,t}\langle s|\,|r+1\rangle\langle r+1|\,|t\rangle\cdot\langle\Omega_s|\Omega_t\rangle = \langle\Omega_{r+1}|\Omega_{r+1}\rangle = 1$$

   For the third term:

$$-\sum_{s,t}\langle s|\,|r+1\rangle\langle r|\,|t\rangle\cdot|\Omega_s\rangle\,U_{r+1}\,|\Omega_t\rangle = -\,\langle\Omega_{r+1}|\,U_{r+1}\,|\Omega_r\rangle = -1$$

   For the fourth term we get -1 as well. Thus in total the expectation is zero.

4. Let's check the final term:

$$\langle\Omega| H_{End} |\Omega\rangle = \frac{1}{T+1} \langle\Omega_T| \cdot |0\rangle\langle 0|_O \cdot |\Omega_T\rangle .$$

Here, $|0\rangle\langle 0|_O$ is the projector onto the output qubit of $|\Omega_T\rangle$, which is the very final snapshot of the verifier V, onto the state $|0\rangle$. Thus $\langle\Omega_T| \cdot |0\rangle\langle 0|_O \cdot |\Omega_T\rangle$ denotes the probability that if we ran V on input $|x\rangle \otimes |\psi\rangle \otimes |0 \cdots 0\rangle$, the output is 0. This is precisely at most $\exp(-n)$, by the soundness property of the verifier.

So the energy is going to be $\exp(-n)/(T+1)$.

Thus in total the energy is at most $\exp(-n)$.

NO case. As usual, the NO cases is more complicated to analyze – we basically want to show that no matter what state we look at, the energy of that state with respect to H is going to be at least $\Omega(T^{-3})$, which is appreciably larger than $\exp(-n)$.

To get some intuition, let's say we use a history state of V. Then the calculations we performed show that the energy comes from the last term H_{End}. The probability of acceptance (V outputting 1) is at most $\exp(-n)$, or turning it around, the probability of V outputting 0 is at least $1-\exp(-n)$. Thus the energy is at least

$$\frac{1-\exp(-n)}{T+1},$$

which is $\Omega(T^{-1})$.

Of course, we can't just consider history states for the NO case. We won't go through all the details of the analysis here, but instead you will work through some of the steps in the next problem set.

To conclude, we've just shown how to efficiently transform instances x of a QMA decision problem L into instances H of k-LOCAL-HAM$_{a,b}$ where $a = \exp(-n)$ and $b = \Omega(1/T^3)$, with T being the running time of the verifier for L.

Thus estimating ground energies to $1/\text{poly}(n)$ precision is as hard a problem as any other QMA problem.

4.3 Probabilistically checkable proofs and the hardness of approximating CSPs

Going back to our favourite classical-quantum dictionary, we've pretty much covered all of the correspondences thus far. Where do we go from here?

Well, where did things go from the Cook-Levin theorem and the discovery of NP-completeness? Let's review some history of classical theoretical computer science.

Beyond NP-completeness. 1970s: Cook-Levin Theorem [Coo71] and NP-completeness. Identified a bonanza of NP-complete problems everywhere. NP-completeness is a generic phenomenon

in computing in the real world (scheduling, route planning, chip design, logistics, AI, predicting how proteins fold, ...). Indicates many problems we'd like to solve are intractable in the worst case.

If we can't solve these problems efficiently in the worst case, perhaps we can get around that by trying to get approximate solutions, rather than exact solutions. For example, given a graph $G = (V, E)$, instead of trying to find the optimal max cut of the graph, what if we just settled for a "pretty big" cut that was, say, 99% of the optimal cut in size? Could that be solvable in polynomial time? Or what if, when given a 3SAT instance φ, instead of trying to find an assignment to satisfies *all* of the clauses, why not find one that satisfies up to 90% of them?

Starting in the 1980s, people came up with very clever polynomial time approximation algorithms for NP-complete problems. Some include: an approximation algorithm for Max-Cut that gives you a 87% approximation of the optimum cut [GW95], approximation algorithms for Travelling Salesman Problem [KKG20], Set Cover [Chv79], Knapsack [Vaz01], and so on. People started thinking: this is great! Even though NP-completeness, seems like a scary barrier for computing, it could be a very fragile barrier: maybe we can get really good fast approximation algorithms for the NP-hard problems we care about! Or are there limits to how well some problems can be approximately solved?

Radical notions of proof. The 1980s also saw intense developments in the area of generalizing the notion of *proof*. Recall that NP is fundamentally a notion of proof-checking: It's the set of all problems where proofs for YES instances can be checked in polynomial time by a classical algorithm. It coincides with our traditional notion of proof checking.

We already saw one way that this notion of proof checking can be generalized: That's QMA! There, Merlin can give a quantum proof for a quantum polynomial-time Arthur to verify.

In the 1980s, researchers came up with the notion of an *interactive proof* [GMR85; Bab85]. Here the idea is pretty natural: Instead of Arthur just checking a static proof that's received by Merlin, Arthur can also have an interactive dialogue with Merlin. It's a two-way conversation where Merlin tries to convince Arthur that some instance x is a YES instance, and Arthur gets to ask "follow-up questions" (See Figure 4.2). Thus a decision problem L has an interactive proof if there is a way for Arthur to interact with Merlin, such that if $x \in L_{yes}$, then Arthur accepts with probability at least $2/3$, and if $x \in L_{no}$, no matter what Merlin says to Arthur, Arthur will accept with probability at most $1/3$.

Does adding the element of interaction change the class of decision problems that can be verified? It turns out that interaction is incredibly powerful. The class of decision problems admitting interactive proofs is called IP. One of the most celebrated results in computational complexity theory is the characterization IP = PSPACE [LFK+92; Sha92], which was proved by Shamir and Lund, Fortnow, Karloff, Nisan in 1991. The class PSPACE is incredibly large: it not only contains NP but also BQP and QMA and many other classes of hard problems. In problem set 1 you saw how the k-Local Hamiltonians problem can be solved in PSPACE. The result IP = PSPACE shows that, a classical polynomial time Arthur can also solve the local Hamiltonians problem, provided that he gets to have a conversation with Merlin. You may find this surprising, especially since Merlin can no longer hand Arthur a quantum state to check (Arthur is completely classical)!

Shortly after IP = PSPACE, there was another breakthrough in complexity theory that demonstrated the amazing power of interaction in proofs. This considers the model of *multiprover inter-*

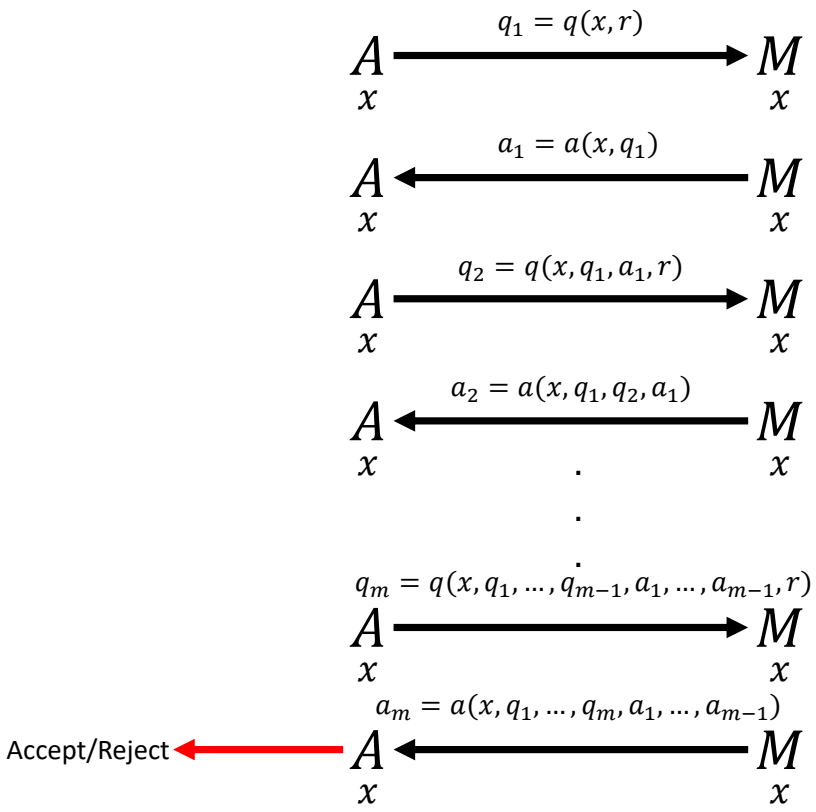


Figure 4.2: An illustration of some interactive protocol between Arthur and Merlin on some input x, consisting of $2m$ rounds. q and a are the functions with which Arthur and Merlin compute questions and answers, respectively. At the end of the interaction, Arthur either accepts or rejects x.

active proofs, where Arthur gets to talk to not one, but *two* (or more) Merlins. What's the point of this? Arthur doesn't trust Merlin, remember, and is worried that maybe even with the ability to ask "follow-up questions" to Merlin, the wizard might be able to trick Arthur into believing that x is a YES instance (even though x may be a NO instance to begin with).

However, imagine that Arthur has at his disposal two Merlins, who he places into separate rooms. The Merlins can't talk to each other, but instead can only be interrogated by Arthur. So now Arthur not only has the ability to generate "follow-up questions", he can *cross-interrogate* the Merlins. In other words, cross-check their answers to see if they're consistent. It's like a policeman cross-interrogating two suspects to see if they committed a crime. The policeman uses to his advantage the fact that each suspect doesn't know what was asked in the other room.

The set of decision problems that can be verified in this model of Arthur interacting with multiple Merlins is denoted by MIP. Babai, Fortnow and Lund showed that this cross-interrogation power is even *more* powerful: it's more than PSPACE. It's more than EXP (which stands for problems solvable in exponential time). It reaches all the way to NEXP [BFL91], which is the exponential-time version of NP.

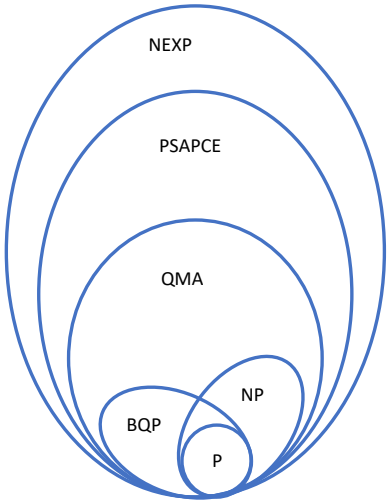


Figure 4.3: Diagram representing inclusion relationships between several complexity classes. The only known strict separation between the discussed classes is $NEXP \neq NP$, by the time hierarchy theorem.

Probabilistically checkable proofs. It didn't take long after IP = PSPACE and MIP = NEXP were proved, that people were led to discover what is now thought of as one of the crown jewels of complexity theory: the Probabilistically Checkable Proofs (PCP) Theorem.

It considers this twist on NP proof checking. We're back to the non-interactive setting: there's an instance x that Arthur wants to determine is a YES instance or not. Merlin generates a static, polynomial-length proof string y for Arthur to verify. Arthur is a classical polynomial-time Turing machine.

However, instead of Arthur reading into the entire proof y, Arthur will pick a small number of bits of y to read, and based on those bits, determine whether x is a YES instance or not. And by small, I mean *really small* – constantly many bits. Think 3.

If x is a NO instance, then the guarantee is that no matter what proof string y is sent by Merlin, Arthur will accept with low probability.

This kind of proof is called a PCP [AS98], for obvious reasons. What decision problems can have their YES instances verified by a PCP?

It turns out, all of NP.

Theorem 1 (PCP Theorem, proof checking version [ALM+98])**.** *For all $\varepsilon > 0$, for all decision problems $L \in$ NP, there exists a randomized polynomial-time verifier V that, when given instance x and query access to a proof string y, makes 3 random queries and has the following behavior:*

1. *If $x \in L_{yes}$, then there exists a proof string y such that $V(x, y)$ accepts with probability at least $1 - \varepsilon$.*
2. *If $x \in L_{no}$, then for all proofs y, $V(x, y)$ accepts with probability at most $1/2 + \varepsilon$.*

This (with worse parameters) was first proved by Arora-Safra, and Arora-Lund-Motwani-Sudan-Szegedy. The version of the PCP Theorem with 3-bit queries is due to Hastad.

To appreciate how this is a mind-bending result, let's consider the following implications:

1. For any satisfiable 3SAT formula φ, there's a proof y that I can generate where, just by looking at constant number of random locations of y (say 1000), you will be convinced with high probability that φ is satisfiable. The number 1000 is independent of the number of variables and clauses of φ.
2. Suppose an alien mathematician comes to Earth and claims that it has a marvelous, wonderful proof of the Riemann Hypothesis. However, the alien's proof is stored on a hard disk that is twenty times the size of the Andromeda galaxy. Good luck checking that!

   Fortunately, the PCP Theorem says that us puny humans can verify the correctness of this alien's proof, with high statistical confidence, just by examining 1000 randomly chosen bits of the proof.

Some intuition for the PCP Theorem. Unfortunately, we won't be able to cover the beautiful proofs of the $\mathsf{IP} = \mathsf{PSPACE}$, $\mathsf{MIP} = \mathsf{NEXP}$, or the PCP theorem in this course. The proof of the PCP theorem is quite involved, although people have found simplifications over the years. Instead, let me try to convey some pieces of intuition for some aspects of it.

So one super basic question, almost bordering on super silly, that one might ask is the following: why doesn't the Cook-Levin theorem already prove the PCP Theorem?

Let L be an NP decision problem with standard NP verifier V. We're going to try to convert this into a probabilistic, 3-query verifier W. For every instance x of L, let φ_x denote the Cook-Levin SAT formula corresponding to $V(x, y)$. Remember that if $x \in L_{yes}$, then φ_x is satisfiable, otherwise it isn't. Let's say φ_x is a SAT formula on t variables and has m clauses. Each clause, remember, involves 3 variables.

Suppose Merlin sends a purported proof in the form of an assignment y to the variables that is supposed to satisfy all clauses of φ_x. Of course, Arthur could just check all of the clauses. But this would require Arthur to examine every bit of the proof. Instead, what if Arthur picks a random clause C of φ_x, which involves 3 variables, and examine the corresponding variables in y to see if C is satisfied. If they are, Arthur accepts. Otherwise, Arthur rejects.

As always let's examine the YES and NO cases. When $x \in L_{yes}$, the Cook-Levin theorem guarantees that φ_x is satisfiable, so therefore Merlin could send a satisfying assignment y for φ_x. When the verifier W chooses a random clause, it's going to be satisfied with probability 1. So Arthur always accepts.

What about the NO case? All clauses except one could be satisfied. Remember that φ_x consists of constraints of the form "Starts OK", "Evolves OK", "Ends OK". The "Ends OK" is just one constraint: it checks whether in the very final snapshot of V, the output bit is set to 1. If we sacrifice this constraint, then we can easily find an assignment y that satisfies every single constraint except for the last one. How many constraints are there? There are $\text{poly}(n)$ constraints, and the chance that the PCP verifier W picks "Ends OK" is at most $1/\text{poly}(n)$, so otherwise the verifier is going to accept. Thus in this NO case, the verifier will still accept with probability $1 - 1/\text{poly}(n)$. This is a far cry from $1/2$.

In this sense, the Cook-Levin theorem is considered "brittle". The Cook-Levin SAT formula that's produced, it's very easy to satisfy it with something that does not remotely resemble the tableau of an accepting proof verification.

What the PCP Theorem really is, is a *robust* version of the Cook-Levin theorem: we can transform every polynomial-time verification algorithm V and instance x into a 3SAT formula φ_x where, if there is no proof that makes V accept, then all possible assignments for φ_x will violate a sizable fraction of clauses (by sizable, I mean a constant fraction like $\frac{1}{10}$ that's independent of the size of x). In other words, there's no way to violate one clause without violating many other clauses.

Connection to hardness of approximation. Now we loop back to our discussion about approximation algorithms for NP-hard problems. Can all NP-hard problems be can be solved with good approximations in polynomial time?

It turns out that the answer is *no*, and it turns out that it's the PCP Theorem that gives us a general theory for the limits on polynomial-time approximations of NP-hard problems. This is because the PCP Theorem, even though it's ostensibly about a funky kind of proof-checking, is *equivalent* to a statement about the hardness of finding approximate solutions to NP-hard problems:

Theorem 2 (PCP Theorem, hardness of approximation version [ALM+98])**.** *The following decision problem L is NP-complete: given a 3SAT formula φ, determine whether*

- *(YES instance): there exists an assignment that satisfies 99% of clauses of φ*
- *(NO instance): all assignments satisfy at most 88% of clauses of φ*

promised that one is the case.

First, this implies that, unless P = NP, there is no polynomial-time algorithm to approximate the maximum number of satisfiable clauses of a 3SAT formula to within 10%. Why is this?

Second, I now claim that the hardness of approximation and proof checking versions of the PCP Theorem are equivalent:

- (**Hardness of approximation ⇒ Proof checking**) To get a PCP for an arbitrary NP decision problem L, we first use the Hardness of Approximation statement to transform every instance x of L to a 3SAT formula φ_x such that

  1. If $x \in L_{yes}$, then there exists an assignment that satisfies 99% of clauses of φ_x
  2. If $x \in L_{no}$, then all assignments satisfy at most 88% of clauses of φ_x.

  Arthur doesn't know which of these is the case, so he asks Merlin to send a supposedly good assignment y of variables to φ_x. Arthur will pick 1000 random clauses and estimate whether at least $99\% - \varepsilon$ fraction of them are satisfied by the assignment y. This only involves examining at most 3000 variables (which is a constant independent of size of φ_x). If so, then Arthur accepts. Otherwise, Arthur rejects.

  If we're in the YES case, Arthur will accept with very high probability. If we're in the NO case, Arthur will accept with very low probability.

- (**Proof checking ⇒ Hardness of approximation**) Suppose now we had a PCP for every problem in NP. Take an NP complete problem L, say 3SAT, and consider the PCP verifier V for it, which uses at most $c \log n$ random bits, for some $c > 0$. Our goal is to reduce 3SAT to the hardness of approximation formulation of the PCP theorem. Given some input x, denote by $V_{x,r}$ the function that receives some alleged proof π as input and outputs 1 iff V accepts x with the proof π and random bits r. Note that since r is fixed, the computation after querying the proof is completely deterministic. Thus, with a fixed r, the output value of $V_{x,r}$ can be characterized by a 3CNF formula of the 3 bits that V reads from π, where each clause represents one possible assignment to the queried bits that would make V reject x. Now consider the 3CNF formula produced by the conjunction of all formulas $\phi_{x,r}, r \in \{0,1\}^{c \log n}$ for some constant c. This formula can be computed in polynomial time since each $\phi_{x,r}$ has constant size and the number of such formulas is polynomial. Denote the resulting formula by ϕ. Now note that by the completeness property of V for L, if $x \in$ 3SAT, there is an assignment (a.k.a a proof π) that satisfies at least 99% of the clauses of ϕ (V accepts x with the proof π with probability arbitrarily close to 1). Now, respectively, due to the soundness property of V, if $x \notin$ 3SAT, then any assignment (a.k.a any proof π) satisfies at most $\frac{2}{3}$ of ϕ's clauses (V accepts x with probability arbitrarily close to 1/2).

  This completes the reduction, and proves this direction of the equivalence.

This is an amazing connection between proof checking and limits of finding approximate solutions to NP-hard problems. It's become a beautiful area of theoretical computer science, and for many fundamental optimization problems, we've been able to pinpoint the *precise* transition between when a problem is poly-time approximable and when it's NP-hard: for example, we know that to find a 7/8-ths approximation of 3SAT is doable in polynomial time (it's trivial, actually: just pick a uniformly random assignment), whereas to find $7/8 + \varepsilon$ approximation is NP-hard [Hås01].

4.4 A quantum PCP Theorem?

Now that we've gotten a whirlwind tour of how NP-completeness and complexity theory has evolved from the 1970's, let's get back to quantum information theory and quantum complexity theory. We have the quantum analogue of the Cook-Levin theorem, but is there a quantum analogue of the PCP Theorem?

Just like the classical case, we can try to formulate a proof checking version and a hardness of approximation version:

Conjecture 3 (Quantum PCP Conjecture, proof checking version)**.** *For all $\varepsilon > 0$, for all decision problems $L \in$ QMA, there exists a quantum polynomial-time verifier V that, when given instance x and and query access to a quantum proof $|\psi\rangle$, makes measurements on $O(1)$ randomly chosen qubits of $|\psi\rangle$ and has the following behavior:*

1. *If $x \in L_{yes}$, then there exists a proof $|\psi\rangle$ such that $V(x, |\psi\rangle)$ accepts with probability at least $1 - \varepsilon$.*
2. *If $x \in L_{no}$, then for all proofs $|\psi\rangle$, $V(x, |\psi\rangle)$ accepts with probability at most $1/2 + \varepsilon$.*

and

Conjecture 4 (Quantum PCP Conjecture, hardness of approximation version)**.** *There exists* $0 \leq \alpha < \beta \leq 1$ *such that the following decision problem* L *is* QMA*-complete: given a local Hamiltonian* $H = H_1 + \cdots + H_m$ *acting on* n *qubits where each* H_i *is positive semidefinite and* $\|H_i\| \leq 1$*, determine whether*

- *(YES instance): the ground energy of* H *is at most* αm*.*
- *(NO instance): the ground energy of* H *is at least* βm*.*

promised that one is the case.

The Proof Checking Version is fairly natural. But why is the hardness of approximation version an appropriate generalization? This is because of we think of each of the H_i's as "clauses", then this is akin to saying that there it is QMA-hard to determine, up to precision $\pm(\beta - \alpha)$, the maximum *fraction* of quantum clauses that can be satisfied.

Just like in the classical case, these two versions are equivalent (this may be on the next problem set). So we interchangeably refer to either them as The Quantum PCP Conjecture, although most people tend to talk about the local Hamiltonian version.

This is one of the holy grails of quantum complexity theory now, to determine whether the Quantum PCP Conjecture is true. Why are people so interested in this? Well, it's not just that it would be really cool. And it's not even to argue that approximating ground state energies of local Hamiltonians is hard: we already know this from the *classical* PCP Theorem (because classical CSPs are just a special family of local Hamiltonians).

The importance stems from the fact that, if the Quantum PCP Conjecture were true, then this would have (theoretical, but also possibly practical) implications for condensed matter physics and many-body physics: namely, that in principle there are physical systems that, even at relatively high energies, can retain complicated patterns of entanglement that admit no efficient description.

We'll talk about this next time.

Chapter 5

The Quantum PCP Conjecture

Scribes: Kyle Oppenheimer and Hugh Goatcher

5.1 The Classical PCP Theorem

Previously, we saw that the Quantum Cook-Levin Theorem tells us that estimating the ground energy of a local Hamiltonian to within $\pm 1/poly(n)$ error is a QMA-hard problem, where n is the number of qubits in the instance. So as n gets larger, we know that this problem is QMA-hard to estimate within this requirement of vanishing error. A natural next question to ask is what happens to the complexity when we relax this requirement? For example, what if we wanted to make a courser approximation (ie. within 1%) that does not depend on the size of n? We do not yet have an answer to this problem. We do, however, have an amazing result for the classical analogue of this question. This result is the Classical Probabilistically Checkable Proofs (PCP) Theorem.

Theorem 5 (PCP Theorem, proof checking version)**.** *For all $\varepsilon > 0$, for all decision problems $L \in \mathsf{NP}$, there exists a randomized polynomial-time verifier V (whose running time scales with $\frac{1}{\varepsilon}$) that, when given instance x and query access to a proof string y, makes 3 random queries and has the following behavior:*

1. *If $x \in L_{yes}$, then there exists a proof string y such that $V(x, y)$ accepts with probability at least $1 - \varepsilon$.*
2. *If $x \in L_{no}$, then for all proofs y, $V(x, y)$ accepts with probability at most $1/2 + \varepsilon$.*

Some intuition for the PCP Theorem. Unfortunately, we won't be able to cover the beautiful proofs of the $\mathsf{IP} = \mathsf{PSPACE}$, $\mathsf{MIP} = \mathsf{NEXP}$, or the PCP theorem in this course. The proof of the PCP theorem is quite involved, although people have found simplifications over the years. Instead, let me try to convey some pieces of intuition for some aspects of it.

So one super basic question, almost bordering on super silly, that one might ask is the following: why doesn't the Cook-Levin theorem already prove the PCP Theorem? Here is an attempt to do so that will motivate why the Cook-Levin Theorem is not strong enough to directly imply the PCP theorem.

Let L be an NP decision problem with standard NP verifier V. We're going to try to convert this into a probabilistic, 3-query verifier W. For every instance x of L, let φ_x denote the Cook-Levin SAT formula corresponding to $V(x, y)$. Remember that if $x \in L_{yes}$, then φ_x is satisfiable, otherwise it isn't. Let's say φ_x is a SAT formula on t variables and has m clauses. Each clause, remember, involves 3 variables.

Suppose Merlin sends a purported proof in the form of an assignment y to the variables that is supposed to satisfy all clauses of φ_x. Of course, Arthur could just check all of the clauses. But this would require Arthur to examine every bit of the proof. Instead, what if Arthur picks a random clause C of φ_x, which involves 3 variables, and examines the corresponding variables in y to see if C is satisfied. If they are, Arthur accepts. Otherwise, Arthur rejects.

As always let's examine the YES and NO cases. When $x \in L_{yes}$, the Cook-Levin theorem guarantees that φ_x is satisfiable, so therefore Merlin could send a satisfying assignment y for φ_x. When the verifier W chooses a random clause, it's going to be satisfied with probability 1. So Arthur always accepts.

What about the NO case? All clauses except one could be satisfied. Remember that φ_x consists of constraints of the form "Starts OK", "Evolves OK", and "Ends OK". The "Ends OK" is just one constraint: it checks whether in the very final snapshot of V, the output bit is set to 1. If we sacrifice this constraint, then we can easily find an assignment y that satisfies every single constraint except for the last one. How many constraints are there? There are $\text{poly}(n)$ constraints, and the chance that the PCP verifier W picks "Ends OK" is at most $1/\text{poly}(n)$, so otherwise the verifier is going to accept. Thus in this NO case, the verifier will still accept with probability $1 - 1/\text{poly}(n)$. This is a far cry from $1/2$.

In this sense, the Cook-Levin theorem is considered "brittle". It is very easy to satisfy the Cook-Levin SAT formula that is produced with something that does not remotely resemble the tableau of an accepting proof verification.

The PCP Theorem is really about making a *robust* version of the Cook-Levin theorem. It says that we can transform every polynomial-time verification algorithm V and instance x into a 3SAT formula φ_x where, if there is no proof that makes V accept, then all possible assignments for φ_x will violate a sizable fraction of clauses (by sizable, I mean a constant fraction like $\frac{1}{10}$ that's independent of the size of x). In other words, there's no way to violate one clause without violating many other clauses.

Connection to hardness of approximation. Now we loop back to our discussion about approximation algorithms for NP-hard problems. Can all NP-hard problems be solved with good approximations in polynomial time?

It turns out that the answer is *no*, and it turns out that it's the PCP Theorem that gives us a general theory for the limits on polynomial-time approximations of NP-hard problems. This is because the PCP Theorem, even though it's ostensibly about a funky kind of proof-checking, is *equivalent* to a statement about the hardness of finding approximate solutions to NP-hard problems:

Theorem 6 (PCP Theorem, hardness of approximation version)**.** *The following decision problem L is NP-complete: given a 3SAT formula φ, determine whether*

- *(YES instance): there exists an assignment that satisfies 99% of clauses of φ*

- *(NO instance): all assignments satisfy at most* 88% *of clauses of* φ

promised that one is the case.

This implies that, unless $\mathsf{P} = \mathsf{NP}$, there is no polynomial-time algorithm to approximate the maximum number of satisfiable clauses of a 3SAT formula to within 5% error. If we were able to approximate within 5% then we could simply reject when we estimate 93% or lower, and accept for 94% or higher, as Figure 5.1 displays.

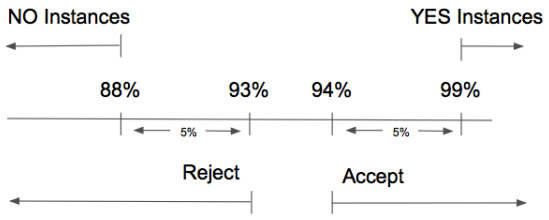


Figure 5.1: If a polynomial time algorithm exists that could approximate the maximum number of satisfiable clauses to within 5%, then we could reject if within 5% of 88%, and accept if within 5% of 99%, implying $\mathsf{P} = \mathsf{NP}$ by the Classical PCP Theorem.

Let us now show that the hardness of approximation and proof checking versions of the PCP Theorem are indeed equivalent:

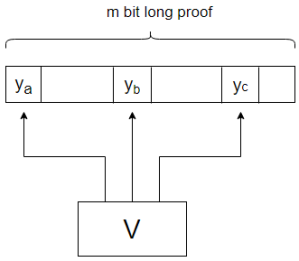
- (**Hardness of approximation** $\Rightarrow$ **Proof checking**) To get a PCP for an arbitrary NP decision problem L, we first use the hardness of approximation statement to transform every instance x of L to a 3SAT formula φ_x such that

  1. If $x \in L_{yes}$, then there exists an assignment that satisfies 99% of clauses of φ_x
  2. If $x \in L_{no}$, then all assignments satisfy at most 88% of clauses of φ_x.

  Arthur doesn't know which of these is the case, so he asks Merlin to send a supposedly good assignment y of variables to φ_x. Arthur will pick 1000 random clauses and estimate whether at least $99\% - \varepsilon$ fraction of them are satisfied by the assignment y. This only involves examining at most 3000 variables (which is a constant independent of size of φ_x). If so, then Arthur accepts. Otherwise, Arthur rejects.

  If we're in the YES case, Arthur will accept with very high probability. If we're in the NO case, Arthur will accept with very low probability.

- (**Proof checking** $\Rightarrow$ **Hardness of approximation**) Suppose now we had a PCP for every problem in NP. Take an NP complete problem L, and consider the PCP verifier V for it. V will query 3 bits at random and in the YES case, will accept with probability at least 2/3, and in the NO case will accept with probability no higher than 1/3. Suppose that the proof is $m = poly(n)$ bits long. There are at most $\binom{m}{3}$ possible checks that V could perform. If V queries at locations a, b, and c and receives bits with values y_a, y_b, and y_c, there is a constraint (or clause) $C_{abc}(y_a, y_b, y_c)$ that determines whether to accept or reject.



We can aggregate all of these clauses together and view this as a three-local constraint satisfaction problem. In other words, the maximum probability of V accepting is equivalent to determining the maximum proportion of clauses that can be satisfied by an assignment y. By similar logic used in 5.1, if we can approximate the maximum number of satisfiable clauses to within some error $\epsilon < 1/6$, we will be able to tell whether V accepts with probability greater than $2/3$ or less than $1/3$, telling us if x is a YES instance or a NO instance. So we have shown that the proof checking formulation reduces to the hardness of approximating the maximum fraction of satisfiable clauses in a 3-local CSP to within ϵ.

This is an amazing connection between proof checking and limits of finding approximate solutions to NP-hard problems. It's become a beautiful area of theoretical computer science, and for many fundamental optimization problems, we've been able to pinpoint the *precise* transition between when a problem is poly-time approximable and when it's NP-hard: for example, we know that to find a 7/8-ths approximation of 3SAT is doable in polynomial time (it's trivial, actually: just pick a uniformly random assignment), whereas to find $7/8+\varepsilon$ approximation is NP-hard. So the Classical PCP Theorem gives us a lot of insight into the question of complexity for course approximations of classical algorithms. Perhaps there is a direct analogue we could draw to help answer this question for quantum algorithms. In other words, is there a Quantum PCP Theorem?

5.2 A Quantum PCP Theorem?

Now that we've gotten a whirlwind tour of how NP-completeness and complexity theory has evolved from the 1970's, let's get back to quantum information theory and quantum complexity theory. We have the quantum analogue of the Cook-Levin theorem, but we have not yet been able to prove a quantum analogue of the PCP Theorem. So it remains today a conjecture, and just like the classical case, we have formulated a proof checking version and a hardness of approximation version:

Conjecture 7 (Quantum PCP Conjecture, proof checking version)**.** *For all $\varepsilon > 0$, for all decision problems $L \in$ QMA, there exists a quantum polynomial-time verifier V that, when given instance x and query access to a quantum proof $|\psi\rangle$, makes measurements on $O(1)$ randomly chosen qubits of $|\psi\rangle$ and has the following behavior:*

1. *If $x \in L_{yes}$, then there exists a proof $|\psi\rangle$ such that $V(x, |\psi\rangle)$ accepts with probability at least $1 - \varepsilon$.*
2. *If $x \in L_{no}$, then for all proofs $|\psi\rangle$, $V(x, |\psi\rangle)$ accepts with probability at most $1/2 + \varepsilon$.*

and

Conjecture 8 (Quantum PCP Conjecture, hardness of approximation version)**.** *There exists a k and $0 \leq \alpha < \beta \leq 1$ such that the following decision problem L is* QMA*-complete: given a k-local Hamiltonian $H = H_1 + \cdots + H_m$ acting on n qubits where each H_i is positive semidefinite and $\|H_i\| \leq 1$, determine whether*

- *(YES instance): the ground energy of H is at most αm.*
- *(NO instance): the ground energy of H is at least βm.*

promised that one is the case.

In other words, the Quantum PCP Conjecture posits that the k-LOCAL-HAM$_{\alpha m, \beta m}$ problem is QMA-complete, where m is the number of terms in the local Hamiltonian instance.

The Proof Checking Version is fairly natural. But why is the hardness of approximation version an appropriate generalization? This is because if we think of each of the H_i's as "clauses", then this is akin to saying that it is QMA-hard to determine, up to precision $\pm(\beta - \alpha)$, the maximum *fraction* of quantum clauses that can be satisfied.

Just like in the classical case, these two versions are equivalent (this may be on the next problem set). So we interchangeably refer to either of them as The Quantum PCP Conjecture, although most people tend to talk about the local Hamiltonian version.

This is one of the holy grails of quantum complexity theory now, to determine whether the Quantum PCP Conjecture is true. Why are people so interested in this? Well, it's not just that it would be a really cool quantum analogue of a classical result. And it's not even to argue that approximating ground state energies of local Hamiltonians is hard: we already know this from the *classical* PCP Theorem (because classical CSPs are just a special family of local Hamiltonians).

The importance stems from the fact that, if the Quantum PCP Conjecture were true, then this would have (theoretical, but also possibly practical) implications for condensed matter physics and many-body physics: namely, that in principle there are physical systems that, even at relatively high energies, can retain complicated patterns of entanglement that admit no efficient description.

5.3 The complexity of ground state entanglement

What can complexity theory tell us about many-body physics? Recall that the physics of a system of interacting particles are modelled by local Hamiltonians, and ground states describe when the system is in its lowest energy state. The QMA-completeness of the local Hamiltonians problem already tells us something interesting. Suppose we make the *complexity-theoretic assumption* that NP $\neq$ QMA, which implies that there are problems in QMA whose YES instances can be efficiently verified via a *quantum proof*, but cannot be efficiently verified if you only gave a *classical proof*. In other words, quantum proofs in general cannot be *convincingly* replaced by a polynomial-sized piece of text.

Since the k-LOCAL-HAM problem is QMA-complete, this implies that ground states of local Hamiltonians in general do not have *useful* classical descriptions that are polynomial-sized! What do I mean by "useful"? Well, here are a couple examples of non-useful descriptions of a ground state:

- Writing out the ground state as a vector in $(\mathbb{C}^2)^{\otimes n}$. This is not polynomial-sized.
- Pointing to the local Hamiltonian $H = H_1 + \cdots + H_m$ (which has polynomial-sized description) and saying, "The eigenvector corresponding to the minimum eigenvalue of H". If H has a unique ground state, then this is unambiguous and indeed counts as a "classical description" of the ground state. However, it's not apparent how one can use this description to verify anything about the ground state. The usefulness of the quantum ground state, for example, is that one can efficiently estimate the energy of the state with respect to H. Or estimate statistics of local measurements on the state.

The assumption $\mathsf{NP} \neq \mathsf{QMA}$ implies that there are local Hamiltonians for which all polynomial-sized classical descriptions of their ground states are not going to be generally useful. You really need to present ground states in quantum form if you want to do anything interesting with them.

Let's say that's true then. Then one of the implications is that there are local Hamiltonians whose ground states are going to be entangled. To see this, suppose that ground states $|\psi\rangle$ were in general unentangled product states:

$$|\psi\rangle = |\psi_1\rangle \otimes |\psi_2\rangle \otimes \cdots \otimes |\psi_n\rangle$$

where the $|\psi_i\rangle$ are single-qubit states. Then this would admit an efficient and useful description of the ground state: a classical description would be simply to describe the n qubits, so overall would take $O(n)$ bits to describe. Furthermore, the energy of $|\psi\rangle$ can be estimated in polynomial time:

$$\langle\psi| H |\psi\rangle = \sum_{i=1}^{m} \langle\psi| H_i |\psi\rangle .$$

Suppose that term $H_i = h_i \otimes I$ acts only nontrivially on qubits $(i_1, \ldots, i_k)$. Then

$$\langle\psi| H_i |\psi\rangle = (\langle\psi_{i_1}| \otimes \cdots \otimes \langle\psi_{i_k}|) h_i (|\psi_{i_1}\rangle \otimes \cdots \otimes |\psi_{i_k}\rangle)$$

which can be computed in $\mathrm{poly}(2^k)$ time, because it involves vector-matrix-vector multiplication of dimension 2^k. So the energy can be estimated in $\mathrm{poly}(n, 2^k)$ time, which is polynomial time for $k = O(1)$.

So, unentangled ground states of this form are definitely out of the question. This is maybe not so surprising, because we have known about the existence of local Hamiltonians whose ground states are highly entangled for a long time, as well as physical systems that upon being cooled down to near absolute zero exhibit quantum effects like superfluidity, or superconductivity, or Bose-Einstein condensates (which all exhibit complex entanglement).

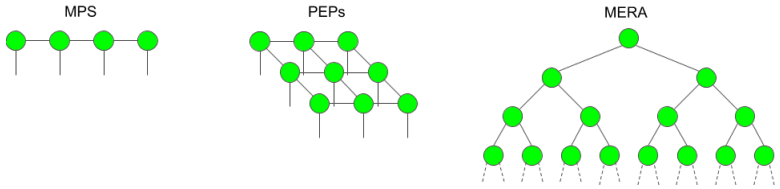
We can say more though. It's not just that ground states are generally entangled. They in fact must possess *complex* entanglement, that can't be easily described via a classical description. The reason this is significant is that over the years actually people have developed really sophisticated ways of succinctly and effectively describing entangled states. One of the most powerful ways of doing this is via something called *tensor networks*, which is a very expressive framework to describe entanglement in many-particle systems. If you come from the world of AI or machine learning, then tensor networks is the physics analogue of graphical models, which is a succinct and efficient way to describe complicated probability distributions.

A particularly well-known success story in tensor networks is that of *matrix product states* (MPS), which have been very successful in describing ground states of one-dimensional local Hamiltonians,

where the particles are arranged on a line and they interact via nearest-neighbour interactions. There is a famous classical algorithm known as DMRG which, when given a description of a one dimensional local Hamiltonian H satisfying a spectral gap condition, can very quickly produce the description of a ground state of H in the form of a matrix product state. Given an MPS representation of a ground state, you can do things like estimate the energy of the state or estimate local observables on the state. So this is an example of a special case of the k-LOCAL-HAM problem that can be solved in polynomial-time:

Theorem 9 (Arad, Landau, Vazirani, Vidick)**.** *There exists an algorithm that, given as input a one-dimensional Hamiltonian H on n particles with nearest-neighbour interactions, outputs a matrix product state description of the ground state of H in time* $\mathrm{poly}(n, 2^{1/\gamma})$*, where γ is the* spectral gap *of the Hamiltonian H (i.e. the difference between the smallest and second-smallest eigenvalue of H).*

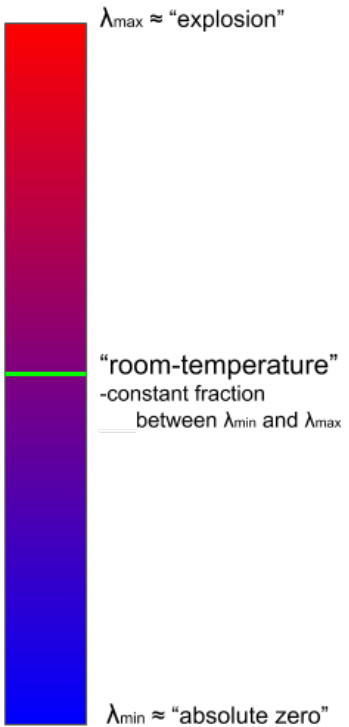
There are other types of tensor networks, that get more expressive: there's something called (projected entangled pairs) PEPs, which are used to describe two-dimensional systems, and multiscale entanglement renormalization ansatz (MERA), which describe fractal-like, scale-invariant systems. Each of these are good for describing different types of entanglement.



So given the effectiveness of certain kinds of tensor networks, one might be inclined to think, maybe the entanglement in ground states of local Hamiltonians can always be described using a sufficiently clever class of tensor networks. But $\mathsf{NP} \neq \mathsf{QMA}$ implies otherwise.

5.4 The complexity of entanglement near room temperature?

As I mentioned though, this implication for physics shouldn't be too surprising. If we have a local Hamiltonian H, we can think of λ_{min} as representing "absolute zero" – the minimum possible energy for the system – and λ_{max} as representing an upper cut-off on the energy that we can meaningfully talk about in the system (physicists generally believe that if you pack too much energy into a confined volume of space and time, it collapses into a black hole, at which point our Hamiltonian description will no longer be valid). So λ_{min} to λ_{max} represents the range of energies that are physically relevant for the system of interest.



We've shown that states of the Hamiltonian with energy really close to λ_{min} will generally be complex to describe, which matches our experimental evidence that really weird quantum stuff can happen at temperatures near absolute zero. On the other hand, our everyday experience is that we don't really see funky quantum effects

happening at "room temperature". The standard physics explanation for that is that quantum effects such as superposition and entanglement are very fragile, and when you're in an environment that's not close to absolute zero, there's too much noise from the environment and whatever quantum effects are present get wiped out, and the system effectively becomes classical.

This is one of the biggest questions in physics today: can large-scale quantum effects be witnessed at energy scales that correspond to something closer to "room temperature"? One of the holy grails of materials engineering, which is to discover a superconductor that can operate at room temperature, is an instantiation of this question. Maybe this is not possible, because there's a fundamental reason that always prevents complex entanglement from persisting once you go to room temperature.

We can rigorously formulate this question in the language of quantum complexity theory. First, we can equate "room temperature" of a physical system with n particles as denoting energy levels that are *macroscopically large*; in other words, the amount of energy, considered as a *fraction* of the total energy range $\lambda_{max} - \lambda_{min}$, is a *constant* that does not go to zero as the number of particles grows. (This is in contrast to the notion of "absolute zero", where the amount of energy, considered as a fraction, is very very close to 0.) Then the Quantum PCP Conjecture (the local Hamiltonian version), *plus* the assumption that $\mathsf{NP} \neq \mathsf{QMA}$, essentially implies the existence of local Hamiltonians (i.e., physical systems) where, all states at "room temperature" retain complex entanglement, complex in the sense that the states cannot be efficiently described in a classical way.

Why is this?

Suppose that, on the contrary, for all local Hamiltonians H (that satisfy the standard assumptions of being positive semidefinite, all terms have operator norm at most 1, etc.), if you go up to high enough energy such as βm where β is the constant from the Quantum PCP Conjecture, then there always exists a state $|\psi\rangle$ with energy at most βm that admits a succinct and useful classical description. Then this would imply that the k-LOCAL-HAM$_{\alpha m,\beta m}$ problem is actually solvable in NP, because to convince someone that a local Hamiltonian H has ground energy less than βm, you can just provide them with one of these classically describable low-energy states, which they can then estimate the energy from. However, this contradicts the assumption that k-LOCAL-HAM$_{\alpha m,\beta m}$ problem is QMA-complete.

So the Quantum PCP Conjecture has interesting consequences for physics, if true. It would indicate the existence – at least, in principle – of these really exotic quantum systems whose quantum effects are extremely robust. Researchers are quite divided on whether this is possible. As I mentioned, standard physics intuition suggests that complex entanglement cannot persist at high temperatures. There are also a number of "no-go" results that I will discuss shortly. On the other hand, a computer scientist might counter that the *classical* PCP Theorem itself was very surprising and unexpected, so perhaps we shouldn't trust our standard intuition that much. Regardless of the truth of the Quantum PCP Conjecture, it will be an extremely interesting scientific journey to resolve it one way or another.

Chapter 6

Complexity of quantum states, and no-go results for Quantum PCP

Scribes: Juan Castaneda, Stephen Zhang

6.1 A closer look at the implications of Quantum PCP

2.1 Review of the Quantum PCP conjecture

Conjecture 1: Quantum PCP conjecture (Hamiltonian formulation)

$\exists k$ and $0 \leq \alpha < \beta \leq 1$ such that the following decision problem L is QMA-complete:

Given a k-local Hamiltonian $H = H_1 + \cdots + H_m$ acting on n qubits, where each H_i is positive semidefinite and $\|H_i\| \leq 1$, determine whether the Hamiltonian is one of the following instances:

- (YES instance): the ground energy of H is at most αm (ie. $\lambda_{min}(H) \leq \alpha m$).
- (NO instance): the ground energy of H is at least βm (ie. $\lambda_{min}(H) \geq \beta m$).

Note: it is promised that it falls into one of these cases.

Recall that the Quantum Cook-Levin Theorem tells us that this problem is QMA complete if we let the distance between the YES and NO cases be some vanishingly small number (ie. The YES case is: $\lambda_{min}(H) \leq \alpha$, and the NO case is: $\lambda_{min}(H) \geq \beta$, for $\alpha - \beta \geq \frac{1}{\text{poly}(n)}$).

What the Quantum PCP conjecture is saying, is that the problem is still QMA complete when the difference between the YES and NO cases is much larger.

2.2 a) Implications for many-body physics: Complexity of quantum states

Last time we had an informal discussion of the implications that the Quantum PCP Conjecture would have for many-body physics. Intuitively, if the Quantum PCP Conjecture were true, and we

made some natural complexity theory assumptions, it would imply the existence of local Hamiltonians whose "room temperature" states would exhibit high complexity.

Let's make this connection a bit more precise. First, I should define what I mean by complex. We can introduce a quantitative measure of complexity of quantum states as follows:

Given an n-qubit state $|\psi\rangle$, define

$$\text{Complexity}(|\psi\rangle) = \text{minimum depth of circuit } C \text{ such that } C\,|0\rangle^{\otimes n} = |\psi\rangle\ .$$

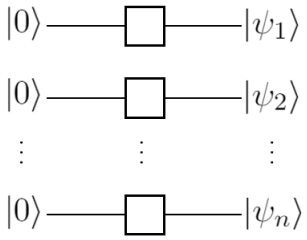
That is to say, the minimum depth of a circuit of 1-qubit and/or 2-qubit unitary gates required to create the desired state $|\psi\rangle$, if starting with the state of all zeros.

Some examples:

1. Any product state $|\psi\rangle = |\psi_1\rangle \otimes \cdots \otimes |\psi_n\rangle$ has circuit complexity $O(1)$.

   This is because each $|\psi_i\rangle$ only requires up to one gate to prepare, and we can apply all the gates in parallel.

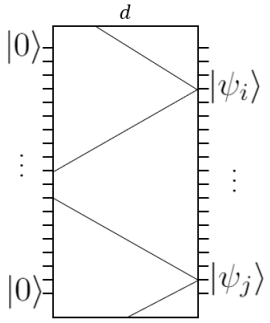
Figure 6.1: Circuit Depth of Preparing a Product State



2. The CAT state (a.k.a. GHZ state), denoted by $\frac{1}{\sqrt{2}}(|0\rangle^{\otimes n} + |1\rangle^{\otimes n})$, has circuit complexity $O(\log n)$. You will see this on the problem set.

   You can also show that you need at least a depth of $\log n$ for the CAT state, using the "light cone" argument. The following is a sketch of the argument:

Figure 6.2: Light Cone Argument for CAT State



Since the circuit is built out of 2-qubit gates, each output qubit can only be affected by 2^d input qubits, where d is the depth of the circuit. If the circuit depth is too small, as in the figure above, the light cones of some pair of output qubits will be disjoint, meaning the qubits act independently. This isn't possible in the CAT state, since once one qubit is measured to be, for example, zero, all the other qubits must also be in the zero state. Therefore, we need the depth to be on the order of $\log n$ so that no two light cones are disjoint.

3. A random quantum state, with overwhelmingly high probability, has circuit complexity $\exp(\Omega(n))$. Recall that Big-Ω notation means the complexity can't be any better than exponential. Note that you can approximate any quantum state arbitrarily well using exponential size circuits, so this is actually the maximum circuit complexity.

The larger Complexity($|\psi\rangle$) is, the more we think of it as expressing more and more complex entanglement. If Complexity($|\psi\rangle$) $\leq$ poly(n), then the state is creatable in (quantum) polynomial time. If a state has more than polynomial complexity (such as exponential), then we think of it as being an extremely complex piece of matter, with entanglement that's irreducibly complicated.

This motivates the following question:

Which local Hamiltonians have ground states or low-energy states with polynomial complexity?

A fascinating potential answer to this question would be "All physically-relevant local Hamiltonians.", because for states involving hundreds or thousands of particles, if their complexity was exponential, the universe wouldn't be old enough for these states to be created by physical processes (that could be simulated by a quantum computer that operates using 2-qubit gates). That would mean that we can always give polynomial-sized *classical descriptions* of low-energy states of physical systems! This would be a breakthrough in many-body physics – we don't know that this is true.

2.2 b) Implications for many-body physics: QCMA complexity class

This motivates the following variant of QMA, called QCMA, which stands for "Quantum-Classical Merlin-Arthur". This is actually a terrible name because it completely misrepresents what it means, so that's unfortunate.[1] In this model, Arthur can do quantum polynomial-time computations, but Merlin can only send classical proofs (ie. a quantum state with polynomial complexity), which Arthur verifies. Just like the k-LOCAL-HAM problem is complete for QMA, the version of the local Hamiltonian problem where you're further guaranteed that in the YES case there is a ground state $|\psi\rangle$ with Complexity($|\psi\rangle$) = poly(n), this problem is complete for QCMA.

So we can think of QCMA as capturing the problem of trying to find a local Hamiltonian's ground state with polynomial description complexity. It is not known whether QMA = QCMA or not, although there are complexity-theory arguments that this is true. If we assume that QMA $\neq$ QCMA, then this implies the existence of local Hamiltonians whose ground states exhibit super-polynomial description complexity.

Let's come back to the Quantum PCP Conjecture. Assume that QMA $\neq$ QCMA. Then this implies the existence of local Hamiltonians where, not only are their ground states super-polynomially complex, *all* states of energy at most βm must have super-polynomial complexity!

[1] At the very least, it should be called "Classical-Quantum Merlin-Arthur".

Why is this? Well, the Quantum PCP Conjecture implies that it is QMA-hard to decide whether a local Hamiltonian has ground energy below αm or above βm. If there was a state with energy below βm that has a polynomial-sized classical description, then Merlin could provide that as proof that the local Hamiltonian was in the YES case, which would mean that QMA = QCMA, a contradiction.

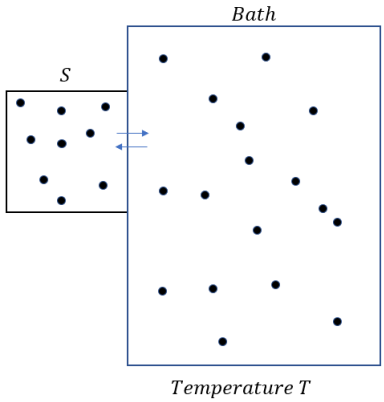
2.2 c) Implications for many-body physics: Connection to "room temperature"

Now what does this have to do with temperature? In the previous lecture, I was loosely equating temperature with energy of a state, but we can make this a little more precise. A local Hamiltonian H describes the constraints and interactions of a physical system S. Statistical mechanics tells us that, if the physical system S in an arbitrary initial state (say, its ground state, or the all zeroes state, doesn't matter), but the system S is placed in contact with an infinitely large heat bath B at temperature T, and left to equilibrate, the state of system S would eventually converge to the *Gibbs state*:

$$\rho(H,T) = \frac{1}{Z}e^{-H/T}$$

where $Z = \mathrm{Tr}(e^{-H/T})$.

Figure 6.3: System Described by Local Hamiltonian H in Contact with Heat Bath



There sometimes is a constant in the denominator known as Boltzmann's constant but let's ignore that for now. The Gibbs state is a density matrix, meaning that it's a probabilistic mixture of pure states, where an eigenstate $|\psi\rangle$ that has energy E with respect to H has probability $e^{-E/T}/Z$ in the mixture $\rho(H,T)$.

The density matrix $\rho(H,T)$ defines a typical state of a system S, governed by a Hamiltonian H, at temperature T.

So given this, we can connect the notion of ground state with the notion of "absolute zero": you will show on the next Problem Set that as $T \to 0$, the Gibbs state $\rho(H,T)$ approaches a uniform mixture over the ground state of H. And if we believe that QMA $\neq$ QCMA, we believe that in general, $\rho(H,T)$ for very small T will be mixtures of super-polynomially complex states.

On the other hand, if you crank up the temperature $T \to \infty$, then you can see that $\rho(H,T)$ will approach the maximally *mixed* state $I/2^{-n}$, which is a completely featureless density matrix that looks like the uniform distribution over all possible classical states. Importantly, there is no entanglement whatsoever in the maximally mixed state – thus, $\rho(H,T)$ for large T will be

supported on states with trivial complexity. So this makes formal the idea that, as you increase the temperature, entanglement effects eventually get wiped out.

So for each Hamiltonian H we can ask, what is the cross-over point in temperature between which the Gibbs state exhibits high complexity entanglement, versus the Gibbs state being a mixture of mostly low complexity entanglement?

The Quantum PCP Conjecture posits that there are local Hamiltonians H where you have to crank up T to some quantity that scales with n before you start seeing complex entanglement disappear?

You will explore this more in depth in the Problem Set.

6.2 No-go results

6.2.1 Quantum PCP cannot hold for local Hamiltonians defined on a grid

Consider an $n \times n$ grid with n^2 qubits on each vertex. Between neighbouring qubits (i, j) there is a Hamiltonian term H_{ij} that describes the interactions between the qubits (see 6.4). Consider the local Hamiltonian $H = \sum_{ij} H_{ij}$. Such a class of local Hamiltonians is QMA-complete in the sense that there are Quantum Cook-Levin Theorems out there where arbitrary QMA verifiers can be transformed into a local Hamiltonian defined on a grid. However, the *promise gap* (the difference between the ground state energies in the YES and NO cases) is $1/\text{poly}(n)$, as is standard with Quantum Cook-Levin Theorem.

Assuming NP $\neq$ QMA, we claim that such 2D Hamiltonians cannot be candidates for Quantum PCP, in the sense that we cannot reduce arbitrary QMA problems to 2D local Hamiltonians with a large promise gap. This is because for these geometrically local Hamiltonians, we can always find a state $|\psi\rangle$ with relatively high energy (up to some constant fraction of the entire system's energy) that admits a succinct classical description.

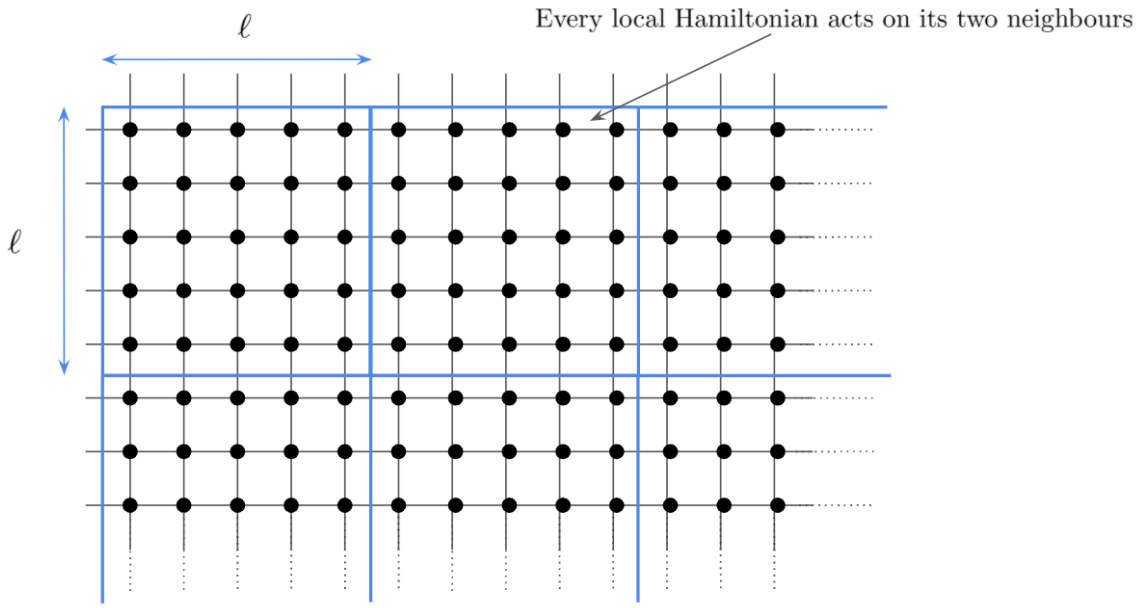
To show this, we are going to divide the $n \times n$ grid into into a bunch of $\ell \times \ell$ square patches (see 6.4). For a given patch, we can group the terms of $H = \sum_{i,j} H_{ij}$ that only act on the qubits in that patch. We will call such a group a "super-term". Thus we can write $H = H'_1 + H'_2 + \cdots + H'_T + H'_{boundary}$ where H'_i is the super-term acting on patch i (so there are at most n^2/ℓ^2 super-terms), and $H'_{boundary}$ is the sum of all the Hamiltonian terms H_{ij} where qubits i and j belong to two different patches. We call such terms "boundary terms". We can bound the number of boundary terms in the following manner: each patch has at most 4ℓ boundary terms and thus, there are at most $4n^2/\ell$ boundary terms.

Each super-term H'_i has a ground state on ℓ^2 qubits. If $\ell = O(1)$, then a ground state of each super-term can be found in $2^{O(\ell^2)}$ time by just brute forcing to solve for the ground state. Let $|\psi_i\rangle$ denote a ground state of super-term H'_i. We can put these together to form a state

$$|\psi\rangle = \bigotimes_i^{n^2/\ell^2} |\psi_i\rangle \,.$$

This is a state on $n \times n$ qubits but since it is a product state, it has a very succinct description. In fact, it can be described using $O(n^2/\ell^2) \cdot 2^{O(\ell^2)}$ bits which is $poly(n)$ bits. Let us compute its

Figure 6.4: 2D Hamiltonians defined on a Grid with Square Patches



energy:

$$\begin{aligned}
\langle\psi|\, H\, |\psi\rangle &= \Big(\sum_i^{n^2/\ell^2} \langle\psi|\, H_i' \otimes I\, |\psi\rangle \Big) + \langle\psi|\, H'_{boundary}\, |\psi\rangle \\
&= \Big(\sum_i^{n^2/\ell^2} \langle\psi_i|\, H_i'\, |\psi_i\rangle \Big) + \langle\psi|\, H'_{boundary}\, |\psi\rangle \\
&\leq \Big(\sum_i^{n^2/\ell^2} \langle\psi_i|\, H_i'\, |\psi_i\rangle \Big) + \|H'_{boundary}\| \\
&\leq \Big(\sum_i^{n^2/\ell^2} \langle\psi_i|\, H_i'\, |\psi_i\rangle \Big) + \frac{4n^2}{\ell}
\end{aligned}$$

where the last inequality is because each H_{ij} has norm of at most 1 so we can bound $||H'_{boundary}||$ by the number of boundary terms. On the other hand, let $|\theta\rangle$ be an eigenvector of H with minimum energy $\lambda_{min}(H)$. Using the fact that the energy of θ with respect to the boundary terms will always be non-negative, we have that

$$\begin{aligned}
\lambda_{min}(H) &= \langle\theta|\, H\, |\theta\rangle \\
&\geq \sum_i^{n^2/\ell^2} \langle\theta|\, H_i' \otimes I\, |\theta\rangle \qquad \text{(dropping the } H'_{boundary} \text{ term)} \\
&\geq \sum_i^{n^2/\ell^2} \langle\psi_i|\, H_i'\, |\psi_i\rangle\,.
\end{aligned}$$

The last inequality follows because regardless of what the state $|\theta\rangle$ is on the qubits of patch i, it can't have a lower energy than $|\psi_i\rangle$ with respect to the super-term H_i' as $|\psi_i\rangle$ is a ground state of H_i'. Putting everything together we have that

$$\langle\psi| H |\psi\rangle \leq \lambda_{min}(H) + \frac{4n^2}{\ell} = \lambda_{min}(H) + O(m/\ell)$$

where we let $m = O(n^2)$ be the number of Hamiltonian terms in H.

Thus we have just constructed a state $|\psi\rangle$ that only has $O(m/\ell)$ energy above the ground energy of H, and furthermore $|\psi\rangle$ has a polynomial sized classical description. Thus if we let ℓ be a large enough constant so that $\frac{m}{\ell} \ll (\beta - \alpha)m$, then we could use this classical description as a way to prove to someone that the ground energy of H is below αm or above βm. (Given the classical description of $|\psi\rangle$, the energy can also be estimated in classical polynomial time).

There's nothing special about 2D grids in this argument; this can be extended to any finite-dimensional grid. Thus, physical systems with nearest-neighbour interactions in a lattice will not be able to exhibit this really exotic quantum behaviour that the Quantum PCP Conjecture predicts.

6.2.2 Quantum PCP cannot hold for local Hamiltonians defined on a high-degree and expanding graphs

The argument above suggests that local Hamiltonians with nearest-neighbour interactions on some kind of lattice or graph that can be chopped up into clusters cannot be candidates for the Quantum PCP Conjecture. Meaning, the "medium-energy" states of these local Hamiltonians can be approximated by unentangled product states. So if we're looking to prove the Quantum PCP Conjecture, then we will need to construct local Hamiltonians with more interesting types of interactions.

This shouldn't be too surprising from the classical complexity theory point of view. The argument presented above actually applies to the classical PCP Theorem as well: CSPs defined on grids can not be candidates for the classical PCP Theorem because one can always find "pretty good" solutions to the CSPs in polynomial-time.

If we look at the CSPs that are constructed by the classical PCP Theorem, the constraints are defined on what are called *expander graphs*. These are graphs where it is impossible to find a subset of nodes that have many connections within the subset, but have few edges going outside the subset. In other words, every subset of nodes has many edges escaping the set – hence the term "expander" and one would not be able to decompose the graph into small pieces without cutting lots of edges. It is known that any candidate CSP for the classical PCP Theorem must be more or less an expander.

Knowing this, it's natural to start thinking about local Hamiltonians defined on expander graphs. So here, since we're talking about graphs, we're focused on 2-local Hamiltonians (because each edge represents a term acting on two qubits). If you want to talk about 3-local or k-local Hamiltonians in general, then you have to talk about *hypergraphs*, but for now let's stick with 2-local Hamiltonians.

In the classical case, generally speaking, the better the expansion of your CSP constraint graph, the better the parameters of the PCP theorem that you will be able to prove. Fascinatingly, this is *not* the case for quantum Hamiltonians. Brandao and Harrow proved the following result:

Theorem 10. *Let H be a 2-local Hamiltonian defined on a graph $G = (V, E)$ with n nodes. Then*

there exists a product state $|\psi\rangle = |\psi_1\rangle \otimes \cdots \otimes |\psi_t\rangle$ *where each* $|\psi_j\rangle$ *is a pure state on a collection of* n/t *qubits, such that*

$$\langle\psi| H |\psi\rangle \leq \lambda_{min}(H) + O\Big(\Big(\frac{1}{2} - \Phi_G\Big)^{1/3}\Big) \cdot m$$

where m *denotes the number of terms in* H*, and* Φ_G *denotes the average expansion of the sets of qubits that the* $|\psi_i\rangle$ *are defined on.*

For a graph G, Φ_G ranges from 0 to 1/2. If it's close to zero, it's a terrible expander, meaning that you can easily disconnect pieces of the graph without cutting too many edges. If it's close to 1/2, then it's a great expander. Thus, as Φ_G approaches 1/2, the theorem above shows that there exist better and better product state approximations of the ground states of H. If $n/t = O(1)$ and $\frac{1}{2} - \Phi_G \leq \varepsilon$, then that means that Merlin can send a classical description of a product state that has energy at most $\lambda_{min}(H) + \varepsilon^{1/3} \cdot m$.

There's another version of their theorem where they show that something similar happens when you have a graph with very *high degree* (how many neighbours each vertex is connected to). If the constraint graph G has degree D, then there exists a product state $|\psi\rangle = |\psi_1\rangle \otimes \cdots \otimes |\psi_n\rangle$ that has energy at most

$$\langle\psi| H |\psi\rangle \leq \lambda_{min}(H) + O\Big(D^{-1/6}\Big) \cdot m$$

So as D increases, you can find better and better product state approximations to the ground space. This sort of behaviour doesn't happen in the classical setting: using a constraint graph with more edges can only make your life *harder*, because you have more constraints to satisfy. However, in the quantum setting, adding more constraints on a quantum system can force it to become more *classical*; this can be seen as another example of the monogamy of entanglement phenomenon.

What this tells us is that if we want to prove the Quantum PCP Conjecture, then we have to look for a "Goldilocks" family of local Hamiltonians: the Hamiltonian constraints can't be too sparse nor too dense, they can't be arranged nicely in grid-like geometries, but they can't be too wild and have an average expansion, Φ_G, that is too close to $\frac{1}{2}$.

The best candidates so far for Hamiltonians that may be useful for any Quantum PCP construction are those that come from *quantum error-correcting codes*. That should make some intuitive sense, because quantum error correcting codes is by definition a way of making entanglement robust, and furthermore quantum error correcting codes give rise to Hamiltonians. However, the big challenge now is to make these code Hamiltonians *local*.

Chapter 7

Classical verification of quantum systems

7.1 The motivation

The motivating question is this:

Is it possible to classical verify that a quantum system (such as a quantum computer) is behaving as intended?

There's a number of reasons why we would be interested in this question.

Suppose QApple (pronounced "quapple") sells you a qMac for the cool price of a gazillion dollars. You take it out of the box, run it through its paces. Can it factor numbers? You choose random primes p, q and ask the computer to factor $N = pq$. It returns p, q. Impressive.

How about simulating the evolution of a Hamiltonian? You plug in your favorite Hamiltonian H, run the latest algorithm to approximate e^{-iHt} applied to some state $|\psi_{init}\rangle$, and measure the resulting state $|\psi\rangle = e^{-iHt}|\psi_{init}\rangle$ using some observable M. You obtain the answer 42. How can check if the computer was really behaving honestly, instead of outputting junk? Is there a way of classically verifying whether

$$\langle\psi| M |\psi\rangle \approx 42?$$

This seems challenging, because ostensibly the whole point of building a quantum computer is that it's supposed to be doing something that's exponentially more complex than what a classical computer can do.

How do we know that QApple sold us a bonafide, working quantum computer?

It's not just about being paranoid. Companies and laboratories are building quantum computers suffering from noise and imperfection. It's important to be able to certify when quantum computers are giving us the right answer.

There's a more philosophical motivation for this question. Quantum computers are not just bigger and faster versions of classical computers. Quantum physics predicts that they'll be *qualitatively*

different from classical computers, and our best attempts at trying to classically describe their behavior necessarily requires exponential resources. It's so unlike anything we've built or designed before, that it justifies additional scrutiny on whether such fantastic machinery could actually be made to work. There is a lot of skepticism about quantum computing out there – some more scientific than others. At the forefront of this skepticism is Gil Kalai, a well-known and serious mathematician, who proposes that there is some noise mechanism that prevents scalable and fault-tolerant quantum computation from being possible. That is, as the numbers of qubits of your system grows, and as you try to increase the complexity of the computations you are performing, there will be correlated errors that occur that will ultimately force the system to be classically simulable. Now, it's not clear how much water his proposal holds. But Carl Sagan had a famous saying: "Extraordinary claims require extraordinary evidence," and so far, it still is an extraordinary claim that large-scale quantum computation can be made to work. So it's scientifically important – and interesting – to figure out ways of providing strong evidence that a quantum computer is working as intended.

Over the last decade, there's been significant progress in designing methods to verify quantum systems, ranging from the very theoretical to experimental protocols for benchmarking the quality of individual gates. In this second half of the class, we'll discuss the more theoretical angle of verification. We'll see how this motivating question has not only led to protocols for classically verifying quantum computations, but has also led to applications to problems that have nothing to do with quantum computing. We will cover:

- Alex Grilo's two-prover protocol for verifying quantum computations.
- A complexity-theoretic result known as $\mathsf{MIP}^* = \mathsf{RE}$ that has implications for pure mathematics and mathematical physics.
- (Time permitting): Urmila Mahadev's single-prover protocol for verifying quantum computations, using cryptography as leverage.

7.2 Nonlocal games and entanglement testing

At the core of the two main results we'll talk about (Grilo's protocol and $\mathsf{MIP}^* = \mathsf{RE}$) is using *nonlocal games* to command quantum systems.

Recall the CHSH game from the very first lecture in class. It's a game played between a classical referee and two separated, non-communicating players we call Alice and Bob. We saw how the CHSH game can be used to prove Bell's theorem, that there is no local classical theory that reproduces all of the predictions of quantum mechanics. In other words, if Alice and Bob use a classical strategy, then their maximum success probability in the game (called the *classical value of CHSH*) is $\omega(CHSH) = 3/4$. On the other hand, if Alice and Bob use quantum entanglement and perform local measurements on the entangled state, they can achieve a higher winning probability: the *quantum value of CHSH*, denoted by $\omega^*(CHSH)$, is $\cos^2(\pi/8) \approx .854\ldots$. Thus the CHSH game can be seen as a *test* for quantum entanglement. If you as the referee play this game between Alice and Bob, and see that they're winning with higher than $3/4$ probability, then you know that they must be using some kind of quantum entanglement in their strategy.

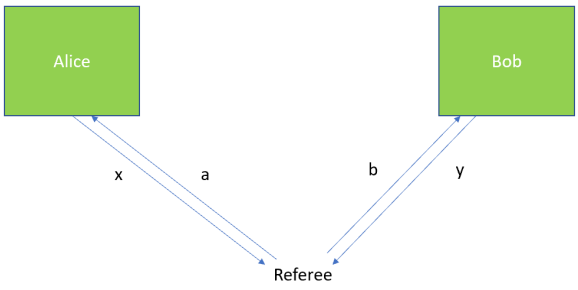


Figure 7.1: Image credit: Michael Dangana

Recall that there is a simple strategy to obtain this quantum value: Alice and Bob share an EPR pair $|EPR\rangle = \frac{1}{\sqrt{2}}\Big(|00\rangle + |11\rangle \Big)$, and Alice uses the following observables depending on her question:

$$A_0 = Z \qquad \text{and} \qquad A_1 = X$$

and Bob measures uses the following observables:

$$B_0 = \frac{1}{\sqrt{2}}(X+Z) \qquad \text{and} \qquad B_1 = \frac{1}{\sqrt{2}}(Z-X)\ .$$

The CHSH game has a special property called *rigidity*, which says that this simple canonical strategy is essentially the *unique* optimal strategy for the CHSH game. So, if you're playing the CHSH game and observe a winning probability that is close to the quantum value, then you actually know behind the scenes Alice and Bob must have been using a quantum strategy that is (close to) "isomorphic" to this EPR strategy.

So we can view the CHSH game as not just a test for the presence of quantum entanglement and quantum measurements, but it's also a test for a *specific* entanglement and *specific* measurements. This is a very powerful tool, because using it you can start to put many CHSH games together to test whether the players are performing a sequence of complicated quantum operations. Put enough of the games together, interleaved in clever ways, and you can actually test whether Alice and Bob are performing entire quantum computations. This is what we'll see in the next lecture.

In this lecture, we'll prove that the CHSH game is rigid, and discuss extensions of it.

Modeling general strategies for the CHSH game. We can model an arbitrary quantum strategy for the CHSH game in the following way: it is a triple $\mathcal{S} = (|\psi\rangle, A, B)$ where

1. **Choice of bipartite state**: $|\psi\rangle \in \mathbb{C}^d \otimes \mathbb{C}^d$ for some choice of d.
2. **Choice of measurements**: Alice chooses a d-dimensional projective measurement for each question $x \in \{0,1\}$. The projective measurement for each question x has two outcomes,

represented by two orthogonal projectors $\{A_{x,0}, A_{x,1}\}$ that sum to the identity. We can further package the measurements into an observable:

$$A_x = A_{x,0} - A_{x,1} \ .$$

Similarly Bob chooses a projective measurement for each $y \in \{0,1\}$, $\{B_b^y\}$ and these form corresponding observables $B_y = B_{y,0} - B_{y,1}$.

The success probability of the strategy of a strategy $\mathcal{S}$ in the CHSH game can be calculated as follows:

$$\omega^*(CHSH, \mathcal{S}) = \sum_{x,y} \Pr(x,y) \sum_{a,b:a\oplus b = x\wedge y} \Pr(\text{output } (a,b)) = \frac{1}{4} \sum_{\substack{x,y,a,b \\ a\oplus b = x\wedge y}} \langle\psi| A_{x,a} \otimes B_{y,b} |\psi\rangle \, .$$

For the CHSH game, it will be more convenient to analyze an equivalent quantity called the *bias* of the strategy, which is how much better the strategy does better than success probability $1/2$, or equivalently the difference between the winning and losing probabilities:

$$\Pr(win) - \Pr(lose) = 2\omega^*(CHSH, \mathcal{S}) - 1 = \frac{1}{4} \sum_{x,y} (-1)^{x\wedge y} \cdot \langle\psi| A_x \otimes B_y |\psi\rangle$$

Note that this bias quantity is written in terms of *observables* A_x, B_y rather than the projectors – this will be convenient for us. You can verify that this equation is true on your own, just by plugging in the definition of what the observables are.

7.2.1 Tsirelson's bound

The first thing we'll prove is that this $\cos^2(\pi/8) = \frac{1}{2} + \frac{1}{2\sqrt{2}} \approx .854\ldots$ value is actually the optimal quantum value.

This is equivalent to saying that the maximum bias for the CHSH game is at most $\frac{1}{2\sqrt{2}}$, or that

$$\langle\psi| \left(A_0 \otimes B_0 + A_1 \otimes B_0 + A_0 \otimes B_1 - A_1 \otimes B_1\right) |\psi\rangle \leq 2\sqrt{2}.$$

This is known as the famous *Tsirelson Bound*, named after Boris Tsirelson, a mathematical physicist who only just passed away this January.

Since A_x, B_y are binary observables, we know that they square to the identity (i.e. $A_x^2 = B_y^2 = \mathbb{I}$). We can rewrite the expression we want to prove as

$$\|A_0 \otimes C_0 + A_1 \otimes C_1\| \leq 2.$$

where $C_0 = \frac{B_0 + B_1}{\sqrt{2}}$ and $C_1 = \frac{B_0 - B_1}{\sqrt{2}}$ and $\|\cdot\|$ denotes the operator norm. (We're using the fact that for any Hermitian operator M, $\|M\| = \sup_{|\psi\rangle} \langle\psi| M |\psi\rangle$).

To prove this, let's first prove that

$$\|(A_0 \otimes C_0 + A_1 \otimes C_1)^2\| \leq 4.$$

If we can prove this, then we obtain the Tsirelson bound. This is because $\|M^2\| = \|M\|^2$ for any operator M.

The operator norm of this squared quantity is easy to bound, however:

$$\|(A_0 \otimes C_0 + A_1 \otimes C_1)^2\| = \|A_0^2 \otimes C_0^2 + A_1^2 \otimes C_1^2 + A_0A_1 \otimes C_0C_1 + A_1A_0 \otimes C_1C_0\| \tag{7.1}$$

and we can evaluate

$$A_0^2 \otimes C_0^2 = \frac{1}{2}\mathbb{I} \otimes (B_0^2 + B_1^2 + B_0B_1 + B_1B_0) = \mathbb{I} \otimes (\mathbb{I} + \frac{1}{2}(B_0B_1 + B_1B_0))$$
$$A_1^2 \otimes C_1^2 = \mathbb{I} \otimes (\mathbb{I} - \frac{1}{2}(B_0B_1 + B_1B_0))$$

Putting everything together, the right hand side of Equation (7.1) looks like

$$\begin{aligned}\|2\mathbb{I} \otimes \mathbb{I} + A_0A_1 \otimes C_0C_1 + A_1A_0 \otimes C_1C_0\| &\leq 2 + \|A_0A_1 \otimes C_0C_1\| + \|A_1A_0 \otimes C_1C_2\| \\ &\leq 2 + 2\|A_0A_1\| \cdot \|C_0C_1\|\end{aligned}$$

where we used the facts that the operator norm of the identity is 1, $\|A \otimes B\| = \|A\| \cdot \|B\|$, and $\|AB\| = \|BA\|$.

First, $\|A_0A_1\| \leq \|A_0\| \cdot \|A_1\|$, and since A_0, A_1 are binary observables this is at most 1. On the other hand,

$$\|C_0C_1\| = \|\frac{1}{2}(B_0^2 - B_1^2 - B_1B_0 + B_1B_0)\| = \frac{1}{2}\|B_1B_0 - B_0B_1\| \leq 1.$$

Thus the right hand side of Equation (7.1) is at most 4, as desired.

7.2.2 Rigidity of the CHSH game

Now we've established that the maximum bias of CHSH is $\frac{1}{2\sqrt{2}}$, or equivalently that the maximum winning probability is $\cos^2(\pi/8)$, we now turn to showing that any strategy that comes *close* to this optimal quantum value must in fact be close to the canonical textbook strategy for CHSH we all know and love.

For simplicity, we're going to prove this in the *exact* case. The theorem we will spend the rest of this lecture proving is the following. To clarify things we will call Alice's Hilbert space $\mathcal{A}$ and Bob's space $\mathcal{B}$ (so that state $|\psi\rangle \in \mathcal{A} \otimes \mathcal{B}$, for example).

Theorem 11 (Exact CHSH rigidity)**.** *Let $\mathcal{S} = (|\psi\rangle, A, B)$ be a d-dimensional strategy for CHSH that succeeds with the optimal quantum value. Then there exist isometries $V : \mathcal{A} \to \mathcal{A}_1 \otimes \mathcal{A}_2, W : \mathcal{B} \to \mathcal{B}_1 \otimes \mathcal{B}_2$ where $\mathcal{A}_1, \mathcal{A}_2$ are isomorphic to $\mathbb{C}^2$, such that, letting $|\theta\rangle = (V \otimes W)|\psi\rangle$, we have*

$$|\theta\rangle = |EPR\rangle_{\mathcal{A}_1\mathcal{B}_1} \otimes |\phi\rangle_{\mathcal{A}_2\mathcal{B}_2}$$

for some pure state $|\phi\rangle$, and

$$(V \otimes W)A_0|\psi\rangle = Z_{\mathcal{A}_1}|\theta\rangle \qquad (V \otimes W)B_0|\psi\rangle = Z_{\mathcal{B}_1}|\theta\rangle$$
$$(V \otimes W)A_1|\psi\rangle = X_{\mathcal{A}_1}|\theta\rangle \qquad (V \otimes W)B_1|\psi\rangle = X_{\mathcal{B}_1}|\theta\rangle.$$

First, what is an isometry? It's like a unitary, except it can map into a bigger space. Formally, a linear map $V : \mathcal{A} \mapsto \mathcal{A}'$ is an isometry if for all $a, b \in \mathcal{A}$, $\|Va\| = \|Vb\|$. Another way of thinking about what an isometry is that it's a unitary with some of its columns deleted.

Let's interpret what this theorem is saying: it says that given an arbitrary optimal strategy for the CHSH game, there exists a local "change of basis" V, W for Alice and Bob, respectively, where if Alice performs basis change V, and Bob performs the basis change W, then their shared state $|\psi\rangle$ (which *a priori* could look like a crazily complicated entangled state in d dimensions), actually factors into a product of an EPR pair, and some auxiliary state $|\phi\rangle$ (typically called a "junk" state because we don't care what it is). Furthermore, if Alice's observable A_0 (which is a unitary, remember) is applied to Alice's share of the state $|\psi\rangle$, and then the basis change $V \otimes W$ is applied, then this is exactly the same as applying the Pauli Z observable to Alice's part of the EPR pair. Similarly, observable A_1 gets mapped to the Pauli X observable.

A symmetrical statement holds for Bob's observables B_0, B_1; they get mapped to Pauli Z and X on Bob's share of the EPR pair. But wait a minute! you might say. In the canonical textbook strategy, Bob's observables don't look like Z or X – they look like $(Z+X)/\sqrt{2}$ and $(Z-X)/\sqrt{2}$. True, but it's not too hard to find a 2×2 unitary U such that

$$UZU^\dagger = \frac{Z+X}{\sqrt{2}} \qquad \text{and} \qquad UXU^\dagger = \frac{Z-X}{\sqrt{2}}.$$

So, really, up to a local unitary rotation on Bob's space, the canonical observables for Bob look like the Z and X Pauli operators.

Theorem 14 proved in three steps:

1. *Deducing anticommutativity*: we will show that

$$A_0 A_1 \otimes \mathbb{I} \, |\psi\rangle = -A_1 A_0 \otimes \mathbb{I} \, |\psi\rangle \qquad \text{and} \qquad \mathbb{I} \otimes B_0 B_1 \, |\psi\rangle = -\mathbb{I} \otimes B_1 B_0 \, |\psi\rangle$$

2. *Extracting a qubit strategy*: we will show that the anticommutation relations above imply the existence an isometry V that maps A_0, A_1 to Z, X on a qubit on Alice's side, and an isometry W that maps B_0, B_1 to Z, X on a qubit on Bob's side.

3. *Extracting an EPR pair*: we will show that $V \otimes W \, |\psi\rangle$ must be the tensor product of an EPR pair and a "junk' auxiliary state.

7.2.3 Deducing anticommutativity

Suppose there's a strategy $\mathcal{S} = (|\psi\rangle, A, B)$ that wins with probability $\cos^2(\pi/8)$. Then this saturates Tsirelson's bound, so we have that

$$\langle\psi| \, A_0 \otimes C_0 + A_1 \otimes C_1 \, |\psi\rangle = 2 \tag{7.2}$$

This algebraic relation will in fact force stronger, more surprising constraints on the operators: optimal play in the CHSH game implies that

$$(A_0 A_1 \otimes \mathbb{I}) \, |\psi\rangle = -(A_1 A_0 \otimes \mathbb{I}) \, |\psi\rangle$$

In other words, on the state $|\psi\rangle$, Alice's two observables must *anti-commute*. By symmetry one can deduce the same thing for Bob's observables:

$$(\mathbb{I} \otimes B_0 B_1) \, |\psi\rangle = -(\mathbb{I} \otimes B_1 B_0) \, |\psi\rangle$$

This is a really fascinating deduction. During the actual process of playing the CHSH game, Alice only ever performs one measurement – either A_0 or A_1. There is no situation in which Alice performs both measurements on the state. Yet just from observing statistics of how often Alice and Bob win, one can deduce that Alice's operators must satisfy a strict algebraic relationship between each other.

We're going to use the Cauchy-Schwarz inequality, which says that for any two vectors $v, w \in \mathbb{C}^d$,

$$|\langle v, w\rangle|^2 \leq \langle v, v\rangle \cdot \langle w, w\rangle.$$

If we let $v = (A_0 \otimes \mathbb{I})\,|\psi\rangle$ and $w = (\mathbb{I} \otimes C_0)\,|\psi\rangle$, we get that

$$\langle\psi|\, A_0 \otimes C_0\,|\psi\rangle \leq \sqrt{\langle\psi|\, A_0^2\,|\psi\rangle \cdot \langle\psi|\, C_0^2\,|\psi\rangle} = \sqrt{\langle\psi|\,(\mathbb{I} + \frac{1}{2}(B_0B_1 + B_1B_0))\,|\psi\rangle} = \sqrt{1 + \frac{1}{2}\,\langle\psi|\,(B_0B_1 + B_1B_0)\,|\psi\rangle}$$

where we used the fact that $\langle\psi|\, A_0^2\,|\psi\rangle = 1$ and that $\langle\psi|\, A_0 \otimes C_0\,|\psi\rangle$ is a real number (because $A_0 \otimes C_0$ is Hermitian). Similarly,

$$\langle\psi|\, A_1 \otimes C_1\,|\psi\rangle \leq \sqrt{1 - \frac{1}{2}\,\langle\psi|\,(B_0B_1 + B_1B_0)\,|\psi\rangle}$$

If we let $\alpha = \langle\psi|\,(B_0B_1 + B_1B_0)\,|\psi\rangle$ then we get

$$\sqrt{1 + \frac{\alpha}{2}} + \sqrt{1 - \frac{\alpha}{2}} \geq \langle\psi|\, A_0 \otimes C_0 + A_1 \otimes C_1\,|\psi\rangle = 2.$$

The only way to satisfy this is if $\alpha = 0$. Which implies that $\langle\psi|\, C_x^2\,|\psi\rangle = 1$, or in other words the vector $C_x\,|\psi\rangle$ has unit norm.

This, combined with Equation (7.2), implies that

$$\langle\psi|\, A_0 \otimes C_0\,|\psi\rangle = 1$$
$$\langle\psi|\, A_1 \otimes C_1\,|\psi\rangle = 1$$

Since A_x is unitary, the vector $(A_x \otimes \mathbb{I})\,|\psi\rangle$ has unit norm. But now we have that the inner product between $(A_0 \otimes \mathbb{I})\,|\psi\rangle$ and $(\mathbb{I} \otimes C_0)\,|\psi\rangle$ is 1, so in fact these two vectors must be equal:

$$(A_0 \otimes \mathbb{I})\,|\psi\rangle = (\mathbb{I} \otimes C_0)\,|\psi\rangle\ .$$

Similarly

$$(A_1 \otimes \mathbb{I})\,|\psi\rangle = (\mathbb{I} \otimes C_1)\,|\psi\rangle\ .$$

This already tells you something really cool: Alice applying the observable A_0 (applying as a unitary – which isn't the same as performing a measurement) to the state $|\psi\rangle$ yields the same effect as Bob applying the observable C_0. In other words, we can perform "operator switching" between the A_x's and C_x's.

We can now deduce the following relationship:

$$\begin{aligned}(A_0A_1 + A_1A_0)\,|\psi\rangle &= (A_0 \otimes C_1 + A_1 \otimes C_0)\,|\psi\rangle \\ &= (\mathbb{I} \otimes C_1)(A_0 \otimes \mathbb{I})\,|\psi\rangle + (\mathbb{I} \otimes C_0)(A_1 \otimes \mathbb{I})\,|\psi\rangle \\ &= (C_1C_0 + C_0C_1)\,|\psi\rangle \\ &= 0.\end{aligned}$$

where we used operator switching twice, and expanded out what $C_1C_0 + C_0C_1$ is. This is precisely the anticommutativity statement we were looking for.

7.2.4 Extracting a qubit strategy

Natively, Alice's strategy takes place in the space $\mathbb{C}^d$. *A priori*, it may seem that the operators A_0, A_1 can be really wild and look nothing like the canonical 1-qubit strategy. However, we'll see that the anticommutation relations we just derived will actually allow us to *extract* a qubit structure from Alice's space, and we'll see that all the action just takes place in that one qubit, and the rest of the space is left untouched.

This is captured by the following Proposition.

Proposition 12. *If $|\psi\rangle \in \mathcal{A} \otimes \mathcal{B}$ and binary observables A_0, A_1 acting on $\mathcal{A}$ are such that*

$$A_0 A_1 |\psi\rangle = -A_1 A_0 |\psi\rangle$$

then there exists an isometry $V : \mathcal{A} \to \mathcal{A}_1 \otimes \mathcal{A}_2$ where $\mathcal{A}_1$ is isomorphic to $\mathbb{C}^2$ and $\mathcal{A}_2$ is isomorphic to $\mathbb{C}^{d'}$ for some d', such that

$$V(A_0 \otimes \mathbb{I}) |\psi\rangle = (Z_{\mathcal{A}_1} \otimes \mathbb{I}_{\mathcal{A}_2\mathcal{B}}) V |\psi\rangle$$
$$V(A_1 \otimes \mathbb{I}) |\psi\rangle = (X_{\mathcal{A}_1} \otimes \mathbb{I}_{\mathcal{A}_2\mathcal{B}}) V |\psi\rangle$$

Here, $Z_{\mathcal{A}_1}$ (resp. $X_{\mathcal{A}_1}$) denotes the Pauli Z operator (resp. the Pauli X operator) acting on the space $\mathcal{A}_1$.

Let's interpret what this Proposition is saying. For simplicity let's imagine that V were a unitary instead of an isometry. It says that there's a local change of basis V on Alice's side, where if you first apply A_0 to the state, and then apply the change of basis, this is the same thing as, locally changing the basis of Alice's side, and then applying the standard Pauli Z operator on the qubit in register $\mathcal{A}_1$. If instead you applied A_1, then this looks like the Pauli X operator in the rotated space.

Thus, anticommuting binary observables are always equivalent to Pauli Z and X operators, and they in fact specify a qubit in the ambient space!

We won't prove this Proposition in lecture, but it may appear on your next Problem Set.

We can also apply Theorem 12 to Bob's side (because his operators also anticommute on the state), so we also get an isometry $W : \mathcal{B} \to \mathcal{B}_1 \otimes \mathcal{B}_2$ where $\mathcal{B}_1$ is isomorphic to $\mathbb{C}^2$ under which B_0 looks like Z and B_1 looks like X.

7.2.5 Extracting an EPR pair

We're almost done – we've determined that Alice's and Bob's observables are correct. What about the state they're using? Let's go back to some relations we discovered, such as

$$A_0 \otimes C_0 |\psi\rangle = A_1 \otimes C_1 |\psi\rangle = |\psi\rangle \ .$$

Using our isometries, this is equivalent to saying

$$(V \otimes W)(A_0 \otimes C_0) |\psi\rangle = (Z \otimes Z)_{\mathcal{A}_1\mathcal{B}_1} (V \otimes W) |\psi\rangle = (V \otimes W) |\psi\rangle$$

and similarly

$$(X \otimes X)_{\mathcal{A}_1\mathcal{B}_1}(V \otimes W)\,|\psi\rangle = (V \otimes W)\,|\psi\rangle$$

Let $|\theta\rangle = (V \otimes W)\,|\psi\rangle$. Since V, W are isometries, $|\theta\rangle$ is a unit vector (and thus a valid quantum state) in the space $(\mathcal{A}_1 \otimes \mathcal{A}_2) \otimes (\mathcal{B}_1 \otimes \mathcal{B}_2)$. The above conditions can be expressed as saying

$$(Z \otimes Z)_{\mathcal{A}_1\mathcal{B}_1}\,|\theta\rangle = (X \otimes X)_{\mathcal{A}_1\mathcal{B}_1}\,|\theta\rangle = |\theta\rangle\,.$$

As you will show on the next Problem Set, these equations imply the following: if you measure the $\mathcal{A}_1$ and $\mathcal{B}_1$ qubits of $|\theta\rangle$ in the standard basis, you're always going to get the same outcome. Similarly, if you measure the $\mathcal{A}_1, \mathcal{B}_1$ qubits of $|\theta\rangle$ in the X-basis (which is $\{|+\rangle\,, |-\rangle\}$ basis), you're always going to get the same outcome. From the problem you solved in Problem Set 1, this implies that the state of the registers $\mathcal{A}_1 \otimes \mathcal{B}_1$ is actually an EPR pair:

$$|\theta\rangle = |EPR\rangle_{\mathcal{A}_1\mathcal{B}_1} \otimes |\phi\rangle_{\mathcal{A}_2\mathcal{B}_2}$$

for some pure state $|\phi\rangle$.

Thus we've located the canonical qubit strategy for the CHSH game inside an arbitrary optimal strategy $\mathcal{S} = (|\psi\rangle\,, A, B)$, and we have proved Theorem 14.

What about strategies that are *near*-optimal? After all, in practice one can never test whether a given strategy for Alice and Bob win with *exactly* $\cos^2(\pi/8)$ probability – we can only take statistics and estimate that their winning probability is at least $\cos^2(\pi/8) - \varepsilon$.

In that case, the above analysis still holds, except one has to use approximations instead of exact equalities, and keep track of all the errors. In the end, we get a similar statement:

Theorem 13 (Approximate CHSH rigidity)**.** *Let $\mathcal{S} = (|\psi\rangle\,, A, B)$ be a d-dimensional strategy for CHSH that succeeds with probability $\omega^*(CHSH) - \varepsilon$. Then there exist isometries $V : \mathcal{A} \to \mathcal{A}_1 \otimes \mathcal{A}_2, W : \mathcal{B} \to \mathcal{B}_1 \otimes \mathcal{B}_2$ where $\mathcal{A}_1, \mathcal{A}_2$ are isomorphic to $\mathbb{C}^2$, such that, letting $|\theta\rangle = (V \otimes W)\,|\psi\rangle$, we have*

$$\|\,|\theta\rangle\,|EPR\rangle_{\mathcal{A}_1\mathcal{B}_1} \otimes |\phi\rangle_{\mathcal{A}_2\mathcal{B}_2}\,\| \leq O(\sqrt{\varepsilon})$$

for some pure state $|\phi\rangle$, and

$$\|(V \otimes W)A_0\,|\psi\rangle - Z_{\mathcal{A}_1}\,|\theta\rangle\,\| \leq O(\sqrt{\varepsilon}) \qquad \|(V \otimes W)B_0\,|\psi\rangle - Z_{\mathcal{B}_1}\,|\theta\rangle\,\| \leq O(\sqrt{\varepsilon})$$
$$\|(V \otimes W)A_1\,|\psi\rangle - X_{\mathcal{A}_1}\,|\theta\rangle\,\| \leq O(\sqrt{\varepsilon}) \qquad \|(V \otimes W)B_1\,|\psi\rangle - X_{\mathcal{B}_1}\,|\theta\rangle\,\| \leq O(\sqrt{\varepsilon})$$

In other words, *approximately* optimal strategies for CHSH are *approximately* close to the canonical textbook strategy.

Chapter 8

Verifying quantum computations via nonlocal game rigidity

Scribes: Adrian She, Sumner Alperin-Lea

8.1 Recap of CHSH rigidity

Last time, we proved Tsirelson's bound, which says that the optimal quantum winning probability for the CHSH game is $\cos^2(\pi/8) \approx .854\ldots$. Then, building on top of that proof, we saw that the CHSH game is *rigid*:

Theorem 14 (Exact CHSH rigidity)**.** *Let $\mathcal{S} = (|\psi\rangle, A, B)$ be a d-dimensional strategy for CHSH that succeeds with the optimal quantum value. Then there exist isometries $V : \mathcal{A} \to \mathcal{A}_1 \otimes \mathcal{A}_2, W : \mathcal{B} \to \mathcal{B}_1 \otimes \mathcal{B}_2$ where $\mathcal{A}_1, \mathcal{A}_2$ are isomorphic to $\mathbb{C}^2$, such that, letting $|\theta\rangle = (V \otimes W)|\psi\rangle$, we have*

$$|\theta\rangle = |EPR\rangle_{\mathcal{A}_1\mathcal{B}_1} \otimes |\phi\rangle_{\mathcal{A}_2\mathcal{B}_2}$$

for some pure state $|\phi\rangle$, and

$$(V \otimes W)A_0|\psi\rangle = Z_{\mathcal{A}_1}|\theta\rangle \qquad (V \otimes W)B_0|\psi\rangle = Z_{\mathcal{B}_1}|\theta\rangle$$
$$(V \otimes W)A_1|\psi\rangle = X_{\mathcal{A}_1}|\theta\rangle \qquad (V \otimes W)B_1|\psi\rangle = X_{\mathcal{B}_1}|\theta\rangle.$$

There's also a robust version of this statement, which says that if you have a strategy $\mathcal{S}$ that succeeds with probability $\omega^*(CHSH) - \varepsilon$, then there exist local isometries under which the strategy is $O(\sqrt{\varepsilon})$-close to the canonical textbook strategy. A proof of approximate rigidity can be found at [MYS12] that uses techniques similar to the rigidity statement proved last lecture. Another proof of approximate rigidity can be found at [Vid18], which uses tools in representation theory.

Let's recall what this rigidity statement is saying. It's saying that no matter what strategy Alice and Bob use for the CHSH game, if the strategy attains the optimal winning probability, then it is possible for Alice and Bob to have perform local basis changes so that their strategy really looks like the canonical textbook strategy which involves Pauli Z/X measurements on an EPR pair, with an auxiliary junk state that is not used at all.

An important thing to note is that, any the rigidity statement *needs* to have this choice of local basis changes V, W in order for it to be correct. This is because Alice and Bob can always start with the canonical textbook strategy $\mathcal{S}$, and apply local basis changes to obtain an equivalent strategy $\mathcal{S}'$ with exactly the same success probability. So any rigidity statement needs to be able to account for this.

Rigidity of CHSH is really powerful, because this is a way for a purely classical verifier to start getting a precise handle on what quantum operations the two players are performing. Today, we're going to see how this rigidity phenomenon can be leveraged to check entire quantum computations. The basic idea is we start with a simple protocol that allows a classical verifier, augmented with a device with very limited quantum capabilities, to verify arbitrary polynomial-time quantum computations. This extra device, by itself, cannot perform universal quantum computations, but it can help verify them.

Then, to rid the verifier of the need to use the extra quantum device, the verifier will instead *delegate* the device's functionality to two untrusted players by exploiting the rigidity of games like CHSH.

Before we describe this protocol, we'll need to describe a couple other nonlocal games whose rigidity properties are more convenient to use.

8.2 Rigidity of other nonlocal games

8.2.1 Magic Square Game

One downside of the CHSH game is that its optimal quantum success probability is not 100%, it's this $\approx 85.4\%$ value. If a classical verifier is playing the CHSH game with two players, how can it tell if the players have a strategy for winning exactly $\cos^2(\pi/8)$ probability? It's not possible – the verifier would have to play many CHSH games with the players to get an estimate of the winning probability.

It's often much nicer to deal with games whose optimal quantum success probability is 100%. There's a game known as the *Magic Square game*. It's based on the following setup:

Consider a 3×3 grid. The goal is to fill in the grid with 0's and 1's satisfying the following constraints: each row must sum to an even number, and each column must sum to an odd number. Of course, it's impossible to satisfy all of these constraints: there's always going to be at least one row/column that is not satisfied.

We can turn this setup into a nonlocal game MS that demonstrates quantum advantage. In this game, the referee selects a random row or column $x \in \{r_1, r_2, r_3, c_1, c_2, c_3\}$, and then selects a random cell $y \in x$. For example, the referee could select $x = r_2$, and chooses $y = 4$ to denote the 4th cell. The referee sends x to Alice, and y to Bob. Importantly, Alice has no idea what cell Bob received, and Bob has no idea whether it was a row/column that Alice received.

Alice is supposed to answer with three bits (a_1, a_2, a_3) corresponding to an assignment to the cells in her row/column, and Bob responds with a bit b corresponding to his assignment to his received cell.

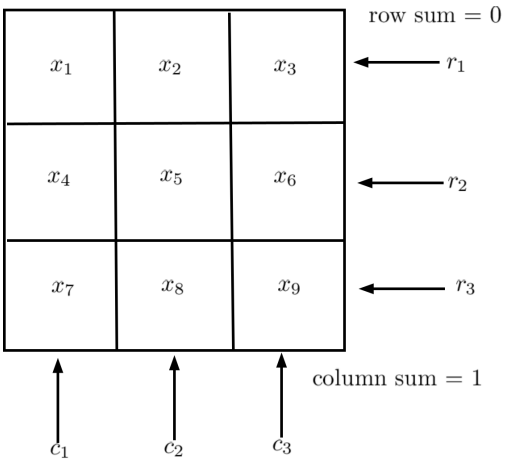


Figure 8.1: Diagram of Magic Square

The players win if Alice's assignment satisfies the row/column constraint, and Bob's bit matches Alice's bit for Bob's cell.

It is not too hard to see that classical value of the Magic Square game (denoted by $\omega(MS)$), is less than 1 (in fact it is 17/18). That is because if Alice and Bob succeeded with probability 1 using a classical, deterministic strategy, then that would imply a consistent way to fill out all the squares with 0's and 1's that satisfies all the constraints, which is impossible.

On the other hand, there exists a *quantum* strategy to win with probability 1. For this reason, this game is often called a *pseudotelepathic*, because an undiscerning external observe might think that Alice and Bob are performing some kind of telepathy in order to win this game 100% of the time.

In fact, no telepathy is going on – just quantum entanglement. The canonical textbook quantum strategy for the Magic Square game is as follows: Alice and Bob share two EPR pairs:

$$|\psi\rangle = \left(\frac{1}{\sqrt{2}}|00\rangle + \frac{1}{\sqrt{2}}|11\rangle\right)^{\otimes 2}.$$

Then, their measurements are specified by the following *operator solution* to the Magic Square:

$I \otimes Z$	$Z \otimes I$	$Z \otimes Z$
$X \otimes I$	$I \otimes X$	$X \otimes X$
$X \otimes Z$	$-Z \otimes X$	$Y \otimes Y$

Figure 8.2: Operator Solution to Magic Square

Here, I, X, Y, Z denote the standard Pauli observables that we talked about in the first lecture, and two put together mean the tensor product of them.

How are they supposed to use this? Well, suppose Alice receives $x = c_2$, i.e., the second column. Then she measures her two qubits using the $Z \otimes I$ observable, followed by the $I \otimes X$ observable, and then the $-Z \otimes X$ observable. Each observable is a Hermitian matrix with ± 1 eigenvalues, so the outcomes for each measurement will be binary. As a simple example, the observable $Z \otimes Z = \Pi_0 - \Pi_1$ where Π_0 is the projector $\Pi_0 = |00\rangle\langle 00| + |11\rangle\langle 11|$ and $\Pi_1 = |01\rangle\langle 01| + |10\rangle\langle 10|$. Thus Alice will get a sequence of bits (a_1, a_2, a_3).

Does it matter which order she measures these observables? It doesn't, because each of the observables within a given row/column *commute* with each other, which means that in fact she can simultaneously measure all of them.

Furthermore, the sequence of bits she gets will always satisfy the parity constraint of her received row/column. Why is that? This comes from the fact that, if you multiply all of the observables in a given row, you get I, and if you multiply all of the observables in a given column, you get $-I$. If you work through the calculations, you will see that this implies that Alice will always give a sequence of bits whose parity is even for rows, and odd for columns.

And as you might expect, when Bob receives a cell, he will just measure his two qubits using the observable corresponding to his received cell. This will match whatever observable Alice measured for that cell, and by the properties of the EPR pair you worked out in Problem Set 1, this implies that their assignment to that cell will always match.

Thus, this is a value-1 quantum strategy for the Magic Square game.

Rigidity of the Magic Square game Similarly to the CHSH game, the Magic Square game is also rigid: *any* strategy that succeeds with optimal probability (or close to it) must be locally isometric to this canonical, 2-EPR pair strategy (or close to it). Thus it is very similar to the CHSH game in the sense that winning with high probability certifies the existence of EPR pairs and Pauli observable measurements (although slightly more involved than that of the CHSH game).

8.2.2 Certifying larger numbers of qubits

So the CHSH game and the Magic Square game certify one and two EPR pairs respectively. What if we wanted to certify a much larger amounts of entanglement?

Here's a simple game to certify $2N$ EPR pairs: this will be the *N-fold parallel repeated Magic Square* game, denoted by MS^N. Here, the referee plays N independent instances of the Magic Square game with Alice and Bob all in parallel: the referee samples row/column choices $x_1, \ldots, x_N$ independently, and then samples cell choices $y_1 \in x_1, y_2 \in x_2, \ldots$ independently. Alice then receives $(x_1, \ldots, x_N)$ and Bob receives $(y_1, \ldots, y_N)$.

Alice is supposed to answer with $3N$ bits and Bob with N bits, and they win if all N instances of the Magic Square game are won.

Clearly, the quantum value of MS^N is 1, because they can use N copies of the optimal quantum strategy for the Magic Square game, and they will win all instances with probability 1. This uses $2N$ EPR pairs.

The classical value of this game is going to be exponentially small in N. It's not $(17/18)^N$ as you might expect, it's actually much more complicated than that. But it is on the order of e^{-cN} for some constant c.

Furthermore, this repeated Magic Square game is rigid: any strategy that succeeds with probability at least 1 must use N parallel copies of the optimal Magic Square strategy. We also have a robust version as follows:

Theorem 15 (Coudron-Natarajan [CN16])**.** *Any strategy for MS^N that succeeds with probability $1-\varepsilon$ must be $O(N\varepsilon^{1/4})$-close, under local isometries, to N parallel copies of the optimal Magic Square strategy.*

An important point to observe about this Theorem is that it is only meaningful when ε is very small, at most N^{-4}. That means that if you want to certify a large number N of EPR pairs, then you can only guarantee this when the players' winning probability is extremely close to 1.

This is undesirable from an experimental point of view because it means that in order to test for larger amounts of entanglement, you need to have increasingly precise estimates of the players' winning probability.

What if you wanted something with better robustness? There is a more sophisticated nonlocal game, called the Pauli Braiding Test, which can certify N EPR pairs without requiring winning probability that gets closer to 1:

Theorem 16 (Natarajan-Vidick [NV17])**.** *Any strategy for the Pauli Braiding Test that succeeds with probability $1-\varepsilon$ must be $O(\varepsilon^{1/2})$-close, under local isometries, to a canonical strategy (not specified here) that uses N EPR pairs and Pauli measurements on each of these EPR pairs.*

Observe that the closeness does *not* depend on N. Thus, we still get meaningful guarantees if the players win with probability, say, 99%. No matter what N is, we can still guarantee that they'll be, say, .01-close to a strategy that uses N EPR pairs. (Don't take the choice of numbers here too seriously, they're just meant to illustrate the point.)

An open question about rigidity is what happens for games when the quantum strategy succeeds with a low constant probability (eg. 25%) and when any classical strategy will succeed with much smaller probability than the quantum strategy. The parallel repeated magic square is one example where such a rigidity statement about the quantum strategy might hold.

8.3 Using rigidity to verify quantum computations

Having seen more examples of various nonlocal games with increasingly elaborate rigidity properties, one can put them to use to verify quantum computations.

To begin: a classical verifier V seeks to validate the execution of some n-qubit quantum circuit C. V seeks to determine if running the circuit C on $|0\rangle^{\otimes n}$ followed by measuring some output qubit would yield $|1\rangle$ with probability at least 2/3 (YES case) or at most 1/3 (NO case) – a standard BQP problem. Without a quantum computer the classical verifier cannot determine this independently. V has access to quantum computers, but doesn't trust them (what if the quantum computers are defective? what if they're just a hoax?). Instead of simple trust, V will execute an *interactive*

protocol with some quantum computers to check, with high confidence, whether the circuit C falls in the YES case or falls in the NO case. For simplicity assume that in the YES case the circuit is accepted with probability 1, and in the NO case the circuit is accepted with probability 0.

In such a protocol the verifier interacts with two untrusted quantum computers that cannot communicate with each other, designated Alice and Bob. One desires a protocol that satisfies the following properties:

- **Verifier resources**: The verifier is a classical computer that runs in polynomial time, and can send/receive classical messages.
- **Completeness**: if C is a YES instance, then there exists a *strategy* for Alice and Bob to convince the verifier to accept with high probability. Furthermore, this strategy must be executable in polynomial time on quantum computers.
- **Soundness**: if C is a NO instance, then no matter what strategy Alice and Bob use – even if this strategy requires Alice and Bob to perform exponential time computations (or more) – the verifier will accept with very low probability.

The following is a protocol developed by Alex Grilo in a paper titled *A simple protocol for verifiable delegation of quantum computation in one round.* [Gri17].

8.3.1 Verifying quantum computations using a trusted measurement device

The following is a simple *measurement-based* verification protocol. This fits in a slightly different model than previously discussed: a classical verifier with access to an extra quantum measurement device that is very simple, but *trusted*, meaning that the verifier knows exactly how the device behaves and what it is meant to do. The measurement device can receive an K-qubit state $|\psi\rangle$, and it can be commanded to measure each qubit in either the Z or X basis, yielding an K-bit string corresponding to the outcomes. After that, the state has been destroyed.

This device is quite simple, and is incapable of performing universal quantum computation. However, it is enough to allow the classical verifier to verify arbitrary polynomial-time quantum computations, in particular determine whether a given circuit C falls in the YES or NO cases.

To do so, the verifier makes use of the Feynman-Kitaev transformation. Using a slightly more sophisticated version of the argument seen previously, there exists a mapping from an n-qubit circuit C to a description of a Hamiltonian $H = H_1 + \cdots + H_m$ with the following properties:

- H acts on $R = \text{poly}(n)$ qubits.
- Each term H_i of H can be written as a tensor product of projectors onto either a standard basis state $|0\rangle, |1\rangle$, or an X-basis state $|+\rangle, |-\rangle$. For example:

  $$H_i = |0\rangle\langle 0|_{i_1} \otimes |+\rangle\langle +|_{i_2} \otimes \cdots \otimes |-\rangle\langle -|_{i_k}$$

  where H_i acts on qubits $(i_1, \ldots, i_k)$.
- There are exponentially many terms, $m = exp(n)$. However, the verifier doesn't have to write out the description of all m terms: given index i, it can compute a description of the term H_i in time $\text{poly}(n)$.

- If C is a YES instance, then there exists a ground state $|\psi\rangle$ such that $\langle\psi| H |\psi\rangle = 0$.
- If C is a NO instance, then for all states $|\psi\rangle$ we have $\langle\psi| H |\psi\rangle \geq (2/3)m$.

The reason we can get such a dramatic difference between the energies in the YES and NO cases here is because H is not a *local* Hamiltonian: each term may act on all R qubits. Furthermore, there are exponentially many terms. However, the verifier can compute the mapping $(C, i) \mapsto H_i$ in polynomial time, and need only work term by term.

Given this, the verifier can determine whether C is a YES circuit or a NO circuit as follows: V asks a quantum computer to prepare a supposed ground state $|\psi\rangle$ for H. The quantum computer will generate some state $|\psi\rangle$ and send it to the verifier's measurement device.

The verifier then picks a random integer $i \in \{1, \ldots, m\}$, and computes the description of the i-th term H_i. As promised above, H_i is a tensor product of measurements in the X or Z basis. The verifier prepares a string $M = (M_1, \ldots, M_R)$ of "measurement commands", where $M_j \in \{I, X, Z\}$, and sends it to the measurement device.

The measurement device measures each qubit of $|\psi\rangle$ according to M and obtains an outcome string $(a_1, \ldots, a_R)$. Suppose the term H_i was the one described above. Then the measurement specification will look like:

$$M_{i_1} = Z, \quad M_{i_2} = X, \quad \cdots \quad M_{i_k} = X$$

and $M_j = I$ everywhere else. The verifier will accept only if the following conditions were met:

$$a_{i_1} = 1 \quad \text{OR} \quad a_{i_2} = 1 \quad \text{OR} \; \cdots \; \text{OR} \qquad a_{i_k} = 0 \; .$$

Why is this? The outcome $a_{i_1} = 1$ corresponds to measuring the i_1-th qubit of $|\psi\rangle$ in the Z basis and obtaining outcome $|1\rangle$, which means that it's annihilated by the term H_i. Similarly, any of these other outcomes implies that the state is annihilated by the term H_i.

Otherwise, the verifier will reject.

I claim that this measurement-based protocol satisfies the requisite Completeness and Soundness properties we need.

Completeness Suppose C is a YES case. Then, there exists a ground state $|\psi\rangle$ such that $\langle\psi| H |\psi\rangle = 0$. The quantum prover can supply this state to the measurement device.

Since H is positive semidefinite, this implies that $H_i |\psi\rangle = 0$ for all i. This means if we were to measure $|\psi\rangle$ in the basis specified by H_i, then the outcomes will all be annihilated by H_i, which means the verifier accepts with probability 1.

Soundness Suppose C is a NO case. Then, *no matter* what state $|\psi\rangle$ is supplied to the measurement device, we have $\langle\psi| H |\psi\rangle \geq (2/3)m$. In other words

$$\frac{1}{m} \sum_i \langle\psi| H_i |\psi\rangle \geq \frac{2}{3}.$$

So if we picked a term H_i uniformly at random, on average the energy of $|\psi\rangle$ with respect to H_i is going to be around $2/3$. Using the example H_i from above again, this would imply that the qubits $(i_1, \ldots, i_k)$ is going to have large overlap with the states

$$|0\rangle_{i_1} \otimes |+\rangle_{i_2} \otimes \cdots \otimes |-\rangle_{i_k}$$

and thus if the verifier picks term H_i to measure, the measurement outcomes are going to be $(0, 0, \ldots, 1)$ with probability $2/3$, and the verifier will reject.

Thus overall the verifier will reject with probability at least $2/3$.

Using this simple, trusted measurement device, classical verifiers can certify the results of polynomial-time quantum computations. This protocol is due to Fitzsimons, Hajdusek and Morimae, in a paper called *Post-hoc verification of quantum computation.* [FHM18]

8.3.2 Delegating the trusted measurement device to untrusted provers

The previous protocol is elegant, but the constraint of a trusted measurement device may prove onerous in practice. Is there a way to certify quantum computations using a *purely* classical verifier, interacting with untrusted quantum computers?

We can do in the *two prover model* (i.e. the same model as in nonlocal games) by taking the measurement-based protocol we just saw, and *offloading* the trusted measurement device to one of the provers. We can use rigidity properties of certain nonlocal games in order to convert untrusted provers into trusted measurement devices.

A classical verifier can play a nonlocal game, such as the parallel repeated Magic Square game MS^R with untrusted device-wielders Alice and Bob to test whether they are sharing at least R EPR pairs, and are performing Pauli measurements on those EPR pairs (provided that they're passing with probability ~ 1). The procedure for doing so turns Bob into a trusted measuring device. We organize Bob's qubits in the following way. He has R pairs of qubits (for $2R$ qubits total). We will always ignore the first qubit in each pair.

If we wanted to command Bob to measure the sequence $IXZZ\cdots$ on R qubits, the classical verifier can send him the question string $(y_1, \ldots, y_R)$ where $y_1 = \perp, y_2 = 5, y_3 = 1, y_4 = 1$, where $\perp$ indicates "we don't care", $y_2 = 5$ indicates we want Bob to give an assignment to the fifth cell in the second instance of the Magic Square game, $y_3 = 1$ indicates we want Bob to give an assignment to the first cell in the third instance of the Magic Square game, and so on. Because of rigidity, we know that Bob's measurement will be $I \otimes X$ for the second instance, $I \otimes Z$ for the third instance, and so on.

That's great, but we're missing an important component: so far Bob is just doing trusted measurements on his share of EPR pairs, not a state $|\psi\rangle$ that is supposed to prove that C is a YES case.

To solve this, we will ultimately need to employ *quantum teleportation*, to be covered next time.

Chapter 9

MIP* Part I

Scribes: Junqiao Lin

9.1 Delegating the trusted measurement device to untrusted provers

Last time, we have seen that using a trusted measurement device, a verifier can (classically) verify quantum computations even with a untrusted prover using a more sophisticated version of the Feymann-Kitaev transformation [BL08]. Today we are going continue this discussion and see how the verifier can force one of the prover to act as a trusted XZ measurement device in a two prover mode, first developed by Grilo [Gri20].

9.1.1 Quantum Teleportation

Recall that quantum teleportation is the following process, if Alice and Bob initially shares an EPR pair and Alice receives a qubit state $|\psi\rangle$ that she wants to send it over to Bob. She can first performs a joint measurement between $|\psi\rangle$ and her share of the EPR pair and obtain 2 bits, (a, b) as the desire outcome. Conditioned on this, the residual state on Bob's side is $X^a Z^b |\psi\rangle$. Without knowing (a, b), from Bob's point of view the state looks completely mixed, so he has no information what the state is. However, if Alice communicates (a, b) (which we call *teleportation keys*) to Bob, then Bob can *correct* his state by un-doing the $X^a Z^b$ operations to obtain $|\psi\rangle$. Thus by presharing entanglement and sending 2 classical bit to Bob, Alice can communicate an arbitrary quantum state to Bob.

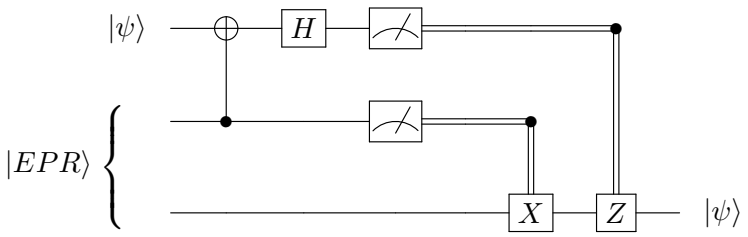


Figure 9.1: Circuit for Quantum teleportation on one qubit.

Going back to our verification protocol, the idea is that if Bob is treated as the trusted measurement device, then Alice is treated as the provider of the purported ground state $|\psi\rangle$ of the Hamiltonian H by teleporting it to Bob. Note since the teleportation protocol preserves entanglement (i.e. if Alice were to perform the teleportation protocol of each of the qubit in $|\psi\rangle$ individually, Bob will still recieve $|\psi\rangle$ as his final state). Base on this idea we can construct the full protocol as follows:

With probability 1/2, the classical verifier does the following:

1. (**Rigidity test**) Play the R-parallel magic square with Alice and Bob.
2. (**Energy test**)
   (a) Tell Alice to teleport a ground state of H to Bob. Alice then responds with a sequence of teleportation keys $(a_1, b_1, \ldots, a_R, b_R)$.
   (b) Pick a random term $i \in [m]$, and let $M \in \{I, X, Z\}^R$ denote the basis corresponding to H_i. Let $(y_1, \ldots, y_R)$ denote the mapping from the basis sequence M to cell locations in the 3×3 Magic Square as described above. Send Bob the string $(y_1, \ldots, y_R)$. Bob responds with outcomes $(c_1, \ldots, c_R)$.
   (c) The classical verifier computes the corrections to $(d_1, \ldots, d_R)$ based on the teleportation keys $(a_1, b_1, \ldots, a_R, b_R)$, as well as the measurement choices M. Where:

$$d_i = \begin{cases} c_i & \text{If } M_i = I \\ c_i \oplus b_i & \text{If } M_i = X \\ c_i \oplus a_i & \text{If } M_i = Z \end{cases}$$

   (d) The classical verifier decides to accept or reject based on i and $(c_1, \ldots, c_R)$ just like in the measurement-based protocol.

We divide the analysis of the above protocol into the following two parts:

Completeness Suppose that C was a YES instance. Then there is a strategy for Alice and Bob to win with probability 1 in this protocol: they simply perform the "honest" strategy: Alice and Bob share $2R$ EPR pairs. When the players receives Magic Square-type questions, they will perform the canonical textbook strategy on their share of the EPR pairs.

When Alice receives the command to "teleport", she will take the R-qubit ground state of H, denoted by $|\psi\rangle$, and then perform the teleportation measurement, qubit-by-qubit, using R EPR pairs. For each qubit j she will obtain teleportation keys (a_j, b_j), and she forwards them to the classical verifier.

In the Energy test, after Alice teleports her state, the state on Bob's side will look like:

$$X^{a_1} Z^{b_1} \otimes \cdots \otimes X^{a_R} Z^{b_R} |\psi\rangle .$$

When Bob is give questions $(y_1, \ldots, y_R)$ which correspond to some measurement string $M \in \{I, X, Z\}^R$, he will measure each qubit correspondingly to obtain outcomes $(c_1, \ldots, c_R)$. The outcomes are corrected by the verifier in the following manner:

1. If $M_j = I$, then we don't do anything to c_j.
2. If $M_j = X$, then $c_j \mapsto c_j \oplus b_j$ (this is because X is measuring in the $|+\rangle, |-\rangle$ basis, and a Z correction flips between the two states).

3. If $M_j = Z$, then $c_j \mapsto c_j \oplus a_j$ (this is because Z is measuring in the $|0\rangle, |1\rangle$ basis, and an X correction flips between the two states).

Thus, after the correction, the processed outcomes $(c_1, \ldots, c_R)$ will be *as if* Bob had measured the plain state $|\psi\rangle$ using the measurement basis sequence M, which is exactly like what happened in the measurement-based verification protocol.

Since $|\psi\rangle$ is a ground state of H, the verifier will accept the outcomes with probability 1.

Soundness Assume that C is a NO case. Suppose for contradiction Alice and Bob are accepted in this protocol with very high probability probability $1 - \varepsilon$ (think of ε as being very small, say inverse polynomial in n). We're going to then argue that this means H must have a ground state with energy 0.

They must be accepted in the Rigidity test with probability at least $1 - 2\varepsilon$. Thus from the rigidity of the parallel repeated Magic Square game, we know that their measurements and state must be $O(R\varepsilon^{1/4})$-close to $2R$ EPR pairs, and in particular Bob must be performing the proper Pauli measurements when he gets input $(y_1, \ldots, y_R)$.

On the other hand, Alice and Bob must also be accepted in the Energy test with probability at least $1 - 2\varepsilon$ as well. The key thing to notice is while Alice can tell that she's not playing the Magic Square game anymore (because she is told to teleport something), Bob is oblivious to this: he *still* gets a question of the form $(y_1, \ldots, y_R)$, and thus he cannot tell whether he's playing the Rigidity test or the Energy test! This is how we keep Bob "honest", in that he cannot adapt his strategy depending on what subgame we're playing.

Thus, in the Energy Test, Bob is measuring $X^{a_1} Z^{b_1} \otimes \cdots \otimes X^{a_R} Z^{b_R} |\psi\rangle$ for *some* state $|\psi\rangle$ – we don't know if $|\psi\rangle$ is a ground state of H or anything. We don't know what Alice did on her qubits when she was told to teleport something, but whatever she did, we're calling $X^{a_1} Z^{b_1} \otimes \cdots \otimes X^{a_R} Z^{b_R} |\psi\rangle$ the post-measurement state on Bob's side.[1]

And since Bob passes the Energy Test with probability $1 - 2\varepsilon$, this means that Bob's measurement of this state yields answers that pass the measurement-based protocol with high probability. This implies that, in fact, $|\psi\rangle$ must have very low energy with respect to H – so small that it must mean that H has a 0 eigenvalue. Which means that C is a YES case.

This is a contradiction. Thus Alice and Bob could not have passed this protocol with probability that is too close to 1.

9.2 Complexity of Nonlocal Games

We learned about how some nonlocal games, such as the CHSH game, the Magic Square game, etc. have very strong rigidity properties. Not only are they tests for entanglement – they are tests for *specific* entanglement and specific quantum operations on them. This rigidity can be leveraged in

[1] To be more precise: Alice will perform some unknown measurement (which may be completely unrelated to the honest teleportation measurement) on her state to obtain bits $(a_1, b_1, \ldots, a_R, b_R)$, and then Bob's share of the post-measurement state will be some density matrix that depends on $(a_1, b_1, \ldots, a_R, b_R)$. We then think of Bob sampling a pure state from the probabilistic mixture represented by the density matrix.

a protocol to classically verify arbitrary quantum computations. Rigidity will also be leveraged in a central way to answer the following fundamental question about nonlocal games:

What is the complexity of computing the quantum value of nonlocal games?

The quantum value of a game is the maximum winning probability of the players when they are allowed to use quantum strategies.

Let's make things more formal. A general nonlocal game can be defined as follows:

Definition 17 (Nonlocal game)**.** *A nonlocal game is a tuple $G = (X, Y, A, B, \mu, D)$ where:*

- *X and Y are question sets for Alice and Bob respectively.*
- *A and B are answer sets for Alice and Bob respectively.*
- *μ is the question distribution and it is a distribution over $X \times Y$.*
- *$D : X \times Y \times A \times B \to \{0, 1\}$ is the payoff function, where $D(x, y, a, b) = 1$ iff Alice and Bob wins with output (a, b) given the question (x, y).*

Quantum strategies for nonlocal games are defined as follows:

Definition 18 (Strategy for a nonlocal game)**.** *A strategy S for a nonlocal game $G = (X, Y, A, B, \mu, D)$ is a tuple $(\{A_x^a\}, \{B_y^b\}, |\psi\rangle)$ where*

- *$|\psi\rangle \in \mathbb{C}^d \otimes \mathbb{C}^d$ is the shared entangled state between Alice and Bob, for some $d > 0$.*
- *For every $x \in X$, $\{A_x^a\}_{a \in A}$ is a d-dimensional POVM with outcomes in the answer set A. This is the measurement that Alice performs on her share of $|\psi\rangle$ when she receives question x.*
- *Similarly, $\{B_y^b\}$ is a POVM with outcomes in the answer set B. This is the measurement that Bob performs on his share of $|\psi\rangle$ when he receives question y.*

We can define the optimal success probability (also known as the *quantum value* of the game) as the following:

Definition 19 (Quantum value of a nonlocal game)**.** *Given a nonlocal game $G = (X, Y, A, B, \mu, D)$, the quantum value of the game $\omega^*(G)$ is defined as*

$$\omega^*(G) = \sup_{(\{A_x^a\}, \{B_y^b\}, |\psi\rangle)} \left\{ \sum_{x \in X, y \in Y} \mu(x, y) \cdot \sum_{a \in A, b \in B} D(x, y, a, b) \cdot \langle \psi | A_x^a \otimes B_y^b | \psi \rangle \right\}$$

Now that we have a more formal definition of the quantum value of a nonlocal game, we can consider the following two computational problems related to games:

- (**Exact version**) Given a nonlocal game G, compute $\omega^*(G)$.

- (**Approximate version**) Given a nonlocal game G, compute v such that $|\omega^*(G) - v| \leq \varepsilon$. In other words, compute an ε-approximation of the quantum value of the game.

Are there algorithms to solve these problems? Clearly, if one can solve the exact version, then one can solve the approximate version. It also may be more reasonable to ask for an algorithm to solve the approximate version; after all, what if the optimal winning probability of G has infinitely many digits (such as $\cos^2(\pi/8) \approx 0.854\ldots$, or worse, is a transcendental number without any closed form).

It turns out that there is *no algorithm whatsoever* to solve either of the problems above – even if the algorithms are allowed to take arbitrary amounts of time! To make it clear, it's not that we just don't know of any algorithm – we can *provably* show there *cannot* be an algorithm!

It also may seem surprising that there is no algorithm for even the approximate case; couldn't there be a "brute force" algorithm that iterates through all possible quantum strategies to find an approximately optimal strategy for a given game G? Note that for every fixed d, it is possible to compute an approximation of the best winning probability using d-dimensional strategies: this is because it is possible to discretize the space of d-dimensional quantum strategies into a finite number of different strategies, and identify which one of those give the highest winning probability. For example, the space of states $|\psi\rangle \in \mathbb{C}^d \otimes \mathbb{C}^d$ can be divided into ε-balls, and similarly the space of d-dimensional measurements can also be divided up into ε-balls.

However, the issue is that for any given game G, it is not clear *a priori* what dimension d to search over. Is dimension $d = 2$ enough? $d = 100$? $d = 10^{100}$? If one found a 100-dimensional strategy for a game G that succeeds with probability .50, how does one know that there isn't a strategy with .51 winning probability if one searches dimension 101? And so on.

More specifically, let α_d be the $\varepsilon-$approximation to the winning probability of the best d-dimensional strategy. Note that $\alpha_d \leq \alpha_{d+1} \leq \omega^*(G)$ for all d since increasing the dimensionality of the strategies considered cannot not worsen the maximum winning probability. Furthermore, α_d *does* converge to $\omega^*(G)$ as $d \to \infty$. However, it is not obvious how fast α_d converges to $\omega^*(G)$ for any given G.

The non-existence of algorithms for approximating the quantum value of nonlocal games has some really unexpected consequences for questions that people have been wondering about for a while. These questions come from a number of different areas of science:

1. Complexity theory: what is computation complexity of MIP^*?
2. Mathematical physics: Is there a difference between finite and infinite-dimensional entanglement?
3. Functional analysis: the Connes' embedding problem from function analysis and operator algebras.

9.3 Complexity of MIP^*

Recall from one of the earlier lectures, we talked about different notions of proof, a few examples includes: static proofs, quantum proofs, interactive proofs, probabilistically checkable proofs

(PCPs). Each of these proof concepts have very interesting developments in complexity theory. In particular, one of the most important results in classical complexity theory is that interactive proofs are extremely powerful, as $\mathsf{IP} = \mathsf{PSPACE}$ and $\mathsf{MIP} = \mathsf{NEXP}$.

In earlier 2000s, some computer scientists started wondering about quantum analogues of interactive proof systems. The first class that they consider is the quantum analogue of IP which is known as QIP [Wat99], which involve a quantum polynomial-time verifier, interacting with a single quantum prover; they can communicate qubits with each other. The additional of quantum computation to the single-prover interactive proof model, interestingly, does not increase the class of decision problems that can be verified in this way; it was shown that $\mathsf{QIP} = \mathsf{IP} = \mathsf{PSPACE}$ by Jain, Ji, Upadhyay, and Watrous [JJU+09].

Later in 2004, the computer science community considered a quantum analogue of MIP, the class of problems verifiable between a classical verifier and multiple (classical) provers, and introduced the complexity class MIP^* [CHT+10]. Similar to MIP, in the MIP^* model the verifier remains classical, and can only run in polynomial-time. The difference is that the provers are allowed to use preshared entanglement with each other in order to coordinate their answers (they however are still not allowed to communicate with each other). Nonlocal games are a special case of MIP^* protocols, where there are only two provers and one round of communication. It turns out that the complexity of nonlocal games is closely linked to understanding the complexity of MIP^*.

Note that it is not immediately clear what the relationship between MIP and MIP^* is, since sharing entanglement could allow the provers to more easily "fool" the verifier into accepting that an instance x is a YES instance (whereas in reality it's a NO instance) – this would suggest that MIP^* is a *weaker* model of verification. On the other hand, the verifier could take advantage of this extra quantum resource available to the provers to ask them to do things they couldn't do before in the classical, unentangled setting. This could potentially give the verifier an adavantage in verifying whether an instance x is a YES instance or not. Thus it is not *a priori* clear who ends up having the upper hand in an interactive proof: the verifier or the entangled provers?

For a long while we only had the following trivial complexity bounds $\mathsf{IP} \subseteq \mathsf{MIP}^* \subseteq \mathsf{RE}$. The lower bound $\mathsf{IP} \subseteq \mathsf{MIP}^*$ follows from the fact that a classical verifier could simply just interact with only one prover, and ignore all other provers. In this case, anything that can be verified with a single-prover interactive proof can be verified here (the fact that the single prover may share entanglement with other provers doesn't matter).

The class RE is the set of all decision problems that are reducible to the Halting problem. In particular, the Halting problem is in RE and thus RE contains undecidable problems (problems that cannot be solved by any algorithm, given any amount of time)! Another way to define the class RE is that it is the set of all decision problems $L = (L_{yes}, L_{no})$ where there exists an algorithm A (depending on L), such that for all instances x, if $x \in L_{yes}$, then $A(x)$ eventually halts, otherwise $A(x)$ does not halt.

The upper bound $\mathsf{MIP}^* \subseteq \mathsf{RE}$ essentially comes from the fact that we can give a "one-sided" method for approximating the quantum value of a nonlocal game. Consider the following decision problem: given a nonlocal game G, determine whether $\omega^*(G) = 1$ (YES instance) or $\omega^*(G) \leq \frac{1}{2}$ (NO instance). For reasons that we won't explain here, the complexity of this problem is *essentially* related to the complexity of MIP^*.

This decision problem is contained in RE, because given a game G, we can compute the sequence

of values $\alpha_1 \leq \alpha_2 \leq \cdots$ until we find a value $\alpha_d > \frac{1}{2}$. If this happens, then we know for sure that G is a YES instance (because this means there exists a finite dimensional strategy to win G with probability strictly greater than $\frac{1}{2}$). Thus the algorithm can halt and output "YES".

On the other hand, if $\alpha_d \leq \frac{1}{2}$ for all d, then this algorithm will never terminate!

This seems like an embarrassing state of affairs for quantum complexity theory; *surely* one could be slightly more clever to come up with an algorithm that always terminates in some finite of time regardless of whether G is a YES or NO instance!

As we'll discuss more in depth, it turns out that in fact there is no clever way; in fact the upper bound is tight, and $\mathsf{MIP}^* = \mathsf{RE}$.

9.4 Mathematical Physics

Now we turn to the connections between the complexity of nonlocal games and mathematical physics. The primary motivating question here is

Should quantum physics be modeled as a finite-dimensional theory, or as an infinite-dimensional one? Does it make a difference?

Typically, quantum information theory is introduced as a finite dimensional theory: all Hilbert spaces are finite-dimensional, operators can be represented as matrices, and so on. However, much of quantum physics and quantum field theory is more naturally formulated as an *infinite-dimensional theory*: the Hamiltonian of the hydrogen atom, for example, is an operator that acts on the space of all square-integrable functions on $\mathbb{R}^3$, which is an infinite-dimensional space.

Does it make a meaningful difference whether we stick to finite-dimensional Hilbert spaces, or allow ourselves to talk about infinite-dimensional spaces? For example, could infinite-dimensional quantum mechanical phenomena be approximated arbitrarily well by finite dimensional descriptions? If one believes that the universe is fundamentally discrete and finite, then a finite-dimensional theory is all you would need to model nature.

This is a pretty deep philosophical question that we won't be able to fully resolve, and especially not in this class. However, we can try to formalize this question in a well-defined, mathematical way, and thus open ourselves to the possibility of giving a precise, mathematical answer to it:

Does allowing Alice and Bob to use infinite-dimensional entanglement help them in a nonlocal game?

In the way we've defined quantum strategies for nonocal games, we've implicitly assumed that Alice and Bob use *finite dimensional* entanglement. But what if Alice and Bob shared an infinite-dimensional state, and performed measurements on that? Could that allow them to win with higher probability than if they only restricted themselves to finite-dimensional strategies?

Perhaps surprisingly, our exploration of the computational complexity of nonlocal games is deeply connected to this question. To explain this, we first have to describe several extensions to the notion of a quantum strategy. We've defined $\omega^*(G)$ as the supremum of Alice and Bob's winning

probability over the choice of finite dimensional strategies. We'll still call this the *quantum value* as before.

Here's an obvious extension of strategies to infinite dimensions:

Definition 20 (Quantum spatial strategies)**.** *A tuple* $(\{A_x^a\}, \{B_y^b\}, |\psi\rangle)$ *is a* quantum spatial strategy *if*

- $|\psi\rangle \in \mathcal{H}_A \otimes \mathcal{H}_B$*, where* $\mathcal{H}_A$ *and* $\mathcal{H}_B$ *are two arbitrary Hilbert spaces (not necessarily finite dimensional).*
- *For all* x*, the set* $\{A_x^a\}$ *is a POVM that acts on* $\mathcal{H}_A$*.*
- *For all* y*, the set* $\{B_y^b\}$ *is a POVM that acts on* $\mathcal{H}_B$*.*

Let $\omega^{qs}(G)$ *denote the supremum over the winning probability for game* G *using a quantum spatial strategy. We call this the* quantum spatial value *of* G*.*

This definition looks nearly identical to the definition of finite dimensional strategy, except the Hilbert spaces are allowed to be infinite-dimensional. Could this be a more powerful model of strategies for Alice and Bob in a nonlocal game? Clearly, $\omega^*(G) \leq \omega^{qs}(G)$ because quantum spatial strategies contain all finite dimensional strategies. Could there be a game G for which the two values are different?

It turns out that it's not more helpful: given any infinite-dimensional quantum spatial strategy, we can approximate it arbitrarily well using a finite-dimensional strategy.[2] Since the quantum value is defined as the *supremum* over finite-dimensional strategies, we have that

$$\omega^*(G) = \omega^{qs}(G).$$

Thus it would seem that infinite dimensional quantum entanglement can be approximated by finite dimensional entanglement, after all!

Not so fast. There's yet another model of quantum strategies that one can consider. We can go one step further and remove the restriction of having Alice and Bob's operators living in different spaces, and instead having the operators *commute* with each other. More precisely:

Definition 21 (Commuting operator strategy)**.** *A tuple* $(\{A_x^a\}, \{B_y^b\}, |\psi\rangle)$ *is a* commuting operator strategy *if*

- $|\psi\rangle \in \mathcal{H}$*, where* $\mathcal{H}$ *is a Hilbert space (not necessarily finite dimensional).*
- *For all* x*, the set* $\{A_x^a\}$ *is a POVM that acts on* $\mathcal{H}$*.*
- *For all* y*, the set* $\{B_y^b\}$ *is a POVM that acts on* $\mathcal{H}$*.*
- *For all* x, y, a, b*, the operators* A_x^a *and* B_y^b commute*, i.e.* $A_x^a B_y^b = B_y^b A_x^a$*.*

[2] More precisely, one can consider the Schmidt decomposition of the state $|\psi\rangle$ and truncate it to its largest d Schmidt coefficients. This yields a d-dimensional state, and furthermore the measurements can be taken to be d-dimensional. By taking d to a large enough value, the measurement statistics can be approximated to within any desired accuracy.

Let $\omega^{co}(G)$ denote the supremum over the winning probability for game G using a commuting operator strategy. We call this the commuting operator value *of the game G.*

Where does this model of entanglement come from? First of all, on its surface it seems like a strange model: there is no separate Hilbert space for Alice and Bob; their measurements all act on the same Hilbert space. However, their measurement operators are required to commute on this space, which means that it doesn't matter what order their measurements are performed – their measurements cannot be used to send signals between each other.

The commuting operator model is a generalization of the tensor product model, because the Hilbert space $\mathcal{H}$ could be the tensor product $\mathcal{H}_A \otimes \mathcal{H}_B$, and Alice's measurement operators can be written as $A_x^a = \hat{A}_x^a \otimes \mathbb{I}$ and $B_y^b = \mathbb{I} \otimes \hat{B}_y^b$. Their measurement operators clearly commute in this case. Thus it is easy to see that

$$\omega^*(G) \leq \omega^{co}(G).$$

Commuting operator strategies are only more general than tensor product strategies.

The commuting operator model is motivated by mathematical formulations of quantum field theory. In finite dimensional quantum theory, it is very natural to model two spatially separated pieces of lab equipment as living in different Hilbert spaces; to describe their joint behaviour, we consider the tensor product of their Hilbert spaces. However, in quantum field theory, the notion of spatiotemporal separation gets very tricky (apparently), because of relativistic effects. Thus it is not always obvious whether the Hilbert space describing two regions of spacetime that are causually separated (meaning they cannot affect each other) has a natural tensor product decomposition. So in QFT, correlations between two causally-separated parties are naturally modeled in the commuting operator model.

Does the commuting operator model give rise to a larger set of correlations? More concretely, is there a game G for which

$$\omega^*(G) \neq \omega^{co}(G)?$$

This is known as *Tsirelson's problem.*[3]

It's not easy to find such a game G. All the games we've talked about so far (CHSH, Magic Square, etc) all have commuting operator values that are the same as the tensor product values.

One important thing to note is that if there is such a separating game G, then the commuting operator strategy that achieves a higher value than any tensor-product strategy must be intrinsically infinite-dimensional. That's because if we restrict ourselves to finite-dimensional strategies, the tensor product and commuting operator models are the same (one can always find a tensor product decomposition between Alice and Bob).

Thus, finding a game G such that $\omega^*(G) < \omega^{co}(G)$ would be tantamount to finding – at least in principle – an experimental test of inherently infinite-dimensional quantum physics. Just like how observing a winning probability greater than 3/4 in the CHSH game is an experimental demonstration of the non-classicality of Nature, observing a winning probability that's larger than $\omega^*(G)$ would give experimental evidence that Nature cannot be adequately described by finite-dimensional quantum mechanical theory.

[3]Technically, his question asks about whether there's a difference between commuting operator *correlations* vs tensor product correlations, not just nonlocal games. But it's close enough to think of his question as being about nonlocal games.

Of course, the catch is, even if we could find a description of such a game G, it's not clear whether we would know how to implement this infinite-dimensional commuting operator strategy in the lab. But at the very least, it would show that it is in principle possible for an experiment to distinguish between these two possibilities.

Chapter 10

MIP* Part II

Scribes: Dong Hao Ou Yang, Junqiao Lin

10.1 Complexity of nonlocal Game

Last time, we discussed the following question:

What is computational complexity of approximating quantum value of nonlocal game?

Let's recall ourselves what a nonlocal game is:

Definition 22. A *nonlocal game* is a 6-tuple $G = (X, Y, A, B, \mu, D)$ where

- X, Y are question sets, consist of all possible questions we can send to Alice and Bob, respectively;
- A, B are answer sets, consist of the answer we receive from Alice and Bob, respectively;
- μ is the distribution over $X \times Y$, which is the probability distribution of sending questions from X and Y to Alice and Bob;
- D is a function $D : X \times Y \times A \times B \to \{0, 1\}$, which is known as the *decision predicate* (or *decision procedure*).

We saw that it's closely connected to the complexity of multiprover interactive proofs with entangled provers, or in other words understanding the complexity of MIP^*. As mentioned for a long time the only bounds known were that $\mathsf{IP} \subseteq \mathsf{MIP}^* \subseteq \mathsf{RE}$. It turns out that it's the upper-bound that is tight, i.e., $\mathsf{MIP}^* = \mathsf{RE}$.

Now, if someone comes up to us and provides a description of a nonlocal game $G = (X, Y, A, B, \mu, D)$, can we compute the maximum winning probability? In other words, is there an algorithm to compute this probability? To measure the optimal win probability, we have to talk about what types of strategies the players are allowed to use. In this course, we've mostly focused our attention on *finite-dimensional* quantum strategies, defined as follows:

Definition 23. A *finite-dimensional (quantum) strategy* for a nonlocal game G is a tuple $S = (|\psi\rangle, \{A_x^a\}, \{B_y^b\})$, where

- $|\psi\rangle \in \mathbb{C}^d \otimes \mathbb{C}^d$ is an entangled state between Alice and Bob for some finite d;
- For every $x \in X$ and $y \in Y$, $\{A_x^a\}_{a \in A}$ and $\{B_y^b\}_{b \in B}$ are measurements, which are some d-dimensional positive operators that add up to identity. When Alice receives question x and Bob receives question y, they simultaneously measure the shared state $|\psi\rangle$ using the joint measurement $\{A_x^a \otimes B_y^b\}_{a,b}$ to obtain outcomes $(a, b) \in A \times B$.

Their success probability for the game G and strategy S is defined as

$$\omega(G, S) = \sum_{x \in X, y \in Y} \mu(x, y) \sum_{\substack{a \in A, b \in B, \\ D(x,y,a,b)=1}} \langle\psi| A_x^a \otimes B_x^b |\psi\rangle .$$

We define the *quantum value* of the game G as

$$\omega^*(G) = \sup_{\substack{\text{finite dimensional} \\ \text{strategy } S}} \omega(G, S)$$

Our task is to estimate $\omega^*(G) \pm \epsilon$ given G, where $\epsilon > 0$ is some small number, for example, $1/10$. As we'll see, it turns out there is no algorithm to accomplish this task!

10.1.1 Connections to Mathematical Physics

Recall from last time we had an interesting discussion on whether Nature could be accurately described by a finite-dimensional quantum theory, or does it require an infinite-dimensional theory?

We can formulate this question in the context of nonlocal games: does it make a difference if Alice and Bob are allowed to use infinite-dimensional quantum strategies? In other words, does it give them some advantage over using finite-dimensional strategies? As we discussed last time, simply allowing Alice and Bob to share an entangled state $|\psi\rangle \in \mathcal{H} \otimes \mathcal{H}$ for infinite-dimensional $\mathcal{H}$ and allowing them to perform measurements on their respective portions of $|\psi\rangle$ does not give them any advantage – any such infinite-dimensional strategy can be approximated arbitrarily well by finite-dimensional strategies. Since the $\omega^*(G)$ is defined as a *supremum* over finite-dimensional strategies, we have that $\omega^*(G)$ is equal to $\omega^{qs}(G)$, which is the maximum success probability when allowing the players to use infinite-dimensional strategies in the tensor-product model.

There is another generalization called the *commuting operator model*.

Definition 24. A strategy $S = (|\psi\rangle, \{A_x^a\}, \{B_y^b\})$ is a *commuting operator strategy* if

- $|\psi\rangle \in \mathcal{H}$ for some Hilbert space $\mathcal{H}$;
- $\{A_x^a\}, \{B_y^b\}$ are measurements acting on $\mathcal{H}$
- $A_x^a B_y^b = B_y^b A_x^a$ for all x, y, a, b.

In a commuting operator strategy, since measurements of Alice and Bob commute, the statistics of their outcomes will always the same regardless of whether Alice measures first or Bob measures first, which means that the correlations generated by their measurements are *non-signaling*: in other words, Alice and Bob cannot use this strategy to send signals to each other[1]. The success probability of a commuting operator strategy in a game G is defined as

$$\omega(G,S) = \sum_{x\in X, y\in Y} \mu(x,y) \sum_{\substack{a\in A, b\in B,\\ D(x,y,a,b)=1}} \langle\psi| A_x^a B_x^b |\psi\rangle .$$

The *commuting operator value* of a game G as

$$\omega^{co}(G) = \sup_{\substack{\text{commuting operator}\\ \text{strategy } S}} \omega(G,S).$$

One easy thing to see is that

$$\omega^*(G) \leq \omega^{co}(G)$$

since the commuting operator strategy is just a strict generalization of finite dimensional one.

Now, an interesting question would be: is there a game G such that $\omega^*(G) \neq \omega^{co}(G)$? If there is a game G such that this held, then it means that the commuting operator strategy achieving $\omega^{co}(G)$ inherently infinite dimensional because finite dimensional tensor product strategies are equivalent to finite dimensional commuting operator strategies. In other words, if there is such game, there is an inherent advantage between finite and infinite dimensional strategies. Such a game can be viewed as an experimental test of infinite dimensionality. Whether $\omega^*(G) = \omega^{co}(G)$ for all G is known as the *Tsirelson's problem.*[2]

From the nonlocal game we have seen so far, none of these exhibit any difference between finite and infinite dimensional values. For example,

$$\omega^*(\mathsf{CHSH}) = \omega^{co}(\mathsf{CHSH}),$$

$$\omega^*(\mathsf{MS}) = \omega^{co}(\mathsf{MS}).$$

So it is not easy to find a separating game.

The connection between this mathematical physics question to complexity theory comes from the following statement: if $\omega^*(G) = \omega^{co}(G)$ for *all* nonlocal games G (meaning that commuting operator strategies can be approximated arbitrarily well by finite-dimensional strategies), then this there exists an algorithm to approximate the quantum value of nonlocal games.

Why is this? Well, this comes from combining two different "semi-algorithms": one semi-algorithm for approximating $\omega^*(G)$ from below, and one semi-algorithm for approximating $\omega^{co}(G)$ from above. We already saw the semi-algorithm for approximating $\omega^*(G)$ from below: fix ε to be a small constant such as $1/100$. The semi-algorithm simply computes the values $\alpha_1 \leq \alpha_2 \leq \cdots \leq \omega^*(G)$ where α_d is the best success probability, up to additive $\pm\varepsilon$, attainable by a d-dimensional quantum strategy.

[1] Formally speaking, we mean that the marginal distribution of Alice's answer a conditioned on x and y, does not depend on y. Similar for Bob's answer.

[2] Technically speaking, Tsirelson's problem is a bit more general; it asks whether the closure of all correlations generated by finite-dimensional strategies is equal to the set of correlations generated by commuting operator strategies.

As $d \to \infty$, the value α_d converges to $\omega^*(G) \pm \varepsilon$. However as we mentioned before we have no *a priori* way of determining how fast the sequence α_d converges.

The semi-algorithm for approximating $\omega^{co}(G)$ computes an infinite sequence of values $\beta_1 \geq \beta_2 \geq \cdots$ where each $\beta_d \geq \omega^{co}(G) \pm \varepsilon$. For each d, the value β_d is computed as the result of a specific convex optimization problem; in technical terms β_d is the smallest value such that the nonnegativity of $\beta_d - \omega^{co}(G)$ can be proved via a degree-d sum-of-squares polynomial in noncommutative variables. What a mouthful! It's not so important for this lecture what that means exactly, but the important thing is that for a fixed d, β_d is computable in finite time. This semi-algorithm for nonlocal games was formulated simultaneously by Navascues, Pironio, Acin [NPA08] and also Doherty, Liang, Toner, Wehner [DLT+08] in two simultaneous papers in 2008. The interested reader should consult John Watrous's wonderful lecture notes on this convex optimization algorithm (often called the NPA hierarchy): https://cs.uwaterloo.ca/~watrous/QIT-notes/QIT-notes.08.pdf.

Suppose that $\omega^*(G) = \omega^{co}(G)$. Then we can put these two semi-algorithms together in the obvious way to obtain a bonafide algorithm for approximating the quantum value of G: alternate between computing the sequences $\alpha_1 \leq \alpha_2 \leq \cdots$ and $\beta_1 \geq \beta_2 \geq \cdots$ until they come within ε of each other, at which point we know we have identified $\omega^*(G) \pm \varepsilon$. Furthermore, we know that this will occur in some finite amount of time, because the two sequences converge to each other, and they are monotonically increasing/decreasing.

If $\omega^*(G) = \omega^{co}(G)$ for all games G, then this algorithm will approximate the value for all games.

10.1.2 Connections to Pure Mathematics

Tsirelson's problem then connects to a deep question from pure mathematics. It turns out that something known as the *Connes' embedding problem* from operator algebras, if it were true, would imply that $\omega^{co}(G) = \omega^*(G)$ always (i.e. a positive resolution to Tsirelson's problem), which would imply the existence of an algorithm!

What is the statement of this problem? It's rather abstract and difficult to understand if you haven't taken a course in von Neumann algebras, but the gist of it is as follows:

A von Neumann algebra M is a set of bounded operators on a Hilbert space $\mathcal{H}$ that satisfy certain closure properties. Not only can elements within M be added, multiplied, etc. M also contain the limits of sequences of operators. The Hilbert space $\mathcal{H}$ can be finite dimensional, infinite dimensional. They are very general, and over the years one of the main goals of the field of operator algebras has been to develop a classification of all von Neumann algebras.

The Connes' embedding problem asks whether a certain class of vNAs, called Type II_1 algebras, can always be embedded in a (ultrapower of a) specific algebra called the hyperfinite factor [Con76; MN36]. Morally speaking, if they were, then this would mean that all Type II_1 algebras, which are in general extremely complicated infinite dimensional objects, can be approximated arbitrarily well with finite dimensional matrix algebras.

The point is, CEP is very similar to Tsirelson's problem, in the sense that it captures whether infinite dimensional objects can be approximated with finite dimensional ones. It was discovered that these questions are actually equivalent.

If CEP has a positive answer, then $\omega^*(G) = \omega^{co}(G)$ for all G. In particular, this implies the existence

of an algorithm for approximating the optimal winning probability of nonlocal games (using either finite-dimensional or commuting operator strategies; they're the same by assumption). Taking the contrapositive, if we prove that there cannot be *any* algorithm for this, then this yields a negative answer to Tsirelson's problem and thus a negative answer to CEP.

10.2 $\mathsf{MIP}^* = \mathsf{RE}$

We will show that there is no such algorithm. How do we know that no such algorithm can exist? The complexity-theoretic result $\mathsf{MIP}^* = \mathsf{RE}$ establishes this.

Theorem 25 ([JNV+20], $\mathsf{MIP}^* = \mathsf{RE}$)**.** *There exist an algorithm R that, given a Turing machine M, outputs a nonlocal game G_M such that*

1. *If M halts on the empty input, then $\omega^*(G_M) = 1$*
2. *Otherwise, we have $\omega^*(G_M) < \frac{1}{2}$*

Turing machines are a model of computation first formulated by Alan Turing in 1936. Intuitively, we can think of Turing machines as a "source code" for algorithms; we think of it as a Python program, for example.

Informally, this theorem tells you that there is an algorithm R to transform any algorithm M into a nonlocal game such that the quantum value of the game depends on whether the algorithm M halts or not.

To connect this theorem with what we are trying to show, we recall a fundamental result from computer science that the *Halting problem*, which is to determine whether any given computer program halts or not, is unsolvable. Hence, if there exist an algorithm which calculates ω^* within $\frac{1}{2}$, then one can pair this "game solver" algorithm with the algorithm R above in order to decide the Halting problem. However, this is impossible; thus there cannot be such a "game solver" algorithm.

In particular, the combination of the two "semi-algorithms" described previously cannot succeed in approximating the value of nonlocal games. In order for this to fail, it must be that there is a game G (in fact, infinitely many of them) for which $\omega^*(G) \neq \omega^{co}(G)$. Thus Tsirelson's problem and CEP have negative resolutions.

Before we go any further into understanding this result, it might be helpful to first review why the Halting problem is undecidable.

10.2.1 The undecibility of the Halting problem

The Halting Problem can be more formally described as the following:

Definition 26 (Halting Problem)**.** *Given a description of a Turing Machine M, determine whether:*

- *M halts on the empty input*
- *M runs forever on the empty input.*

In the same paper in which he proposed his eponymous model of computation, Turing also proved that the Halting problem was undecidable by any Turing machine.

We prove this via contradiction. Suppose there was such an algorithm A to decide the Halting problem. We can design a new algorithm B that behaves as follows:

Algorithm 1: The "self-defeating" algorithm B

1 Run algorithm A on the description of B.
2 If A says B halts, then enter infinite loop. If A says B does not halt, then stop immediately.

Note that in step 1 the description of B is used within the algorithm B itself. This might seem like a circularity, but computer programs are allowed to be self-referential [3].

This is a well-defined algorithm, so we can now ask, does B halt or not? If B halts, then the algorithm A says it halts, but then B would then go ahead and get stuck in an infinite loop – contradiction. If B does not halt, then the algorithm A says it does not halt, but then B will halt – another contradiction. Hence, the algorithm A cannot exist.

10.2.2 Compression of nonlocal games

How do we show that we can construct nonlocal games whose value reflects whether a given Turing machine halts or not? Intuitively, we need to somehow encode computation into a series of nonlocal games.

To facilitate this, let's first model nonlocal games in a way that makes it easier to talk about both computations and games at the same time. First, we want to be able to talk about *infinite families* of nonlocal games in some uniform way. We can more formally define a family of nonlocal games as the following:

Definition 27 (Uniform Game Sequence)**.** *Let $\mathcal{G} = (G_1, G_2, \ldots)$ be an infinite sequence where each $G_n = (X_n, Y_n, A_n, B_n, \mu_n, D_n)$ is a nonlocal game, such that there exist a single Turing machine V, which computes the following efficiently (on input n, represented in binary, the run time should be* $\mathrm{poly}(\log n)$*):*

- *The sizes of the question and answer sets (i.e. $|X_n|, |Y_n|, |A_n|, |B_n|$)*
- *A Turing machine S_n (called the* sampler*) which specifies how to sample a pair $(x, y) \in X_n \times Y_n$ from the distribution μ_n.*
- *A Turing machine D_n (called the* decider*) that computes the function $D_n(x, y, a, b)$ (the decision rule for the game G_n).*

The sequence $\mathcal{G}$ is known as an uniform game sequence *(UGS).*

We can think of V as a way to encode an infinite number of nonlocal games $\{G_n\}$. For example, if a referee is interested in playing G_{17} with the two players Alice and Bob, the referee can generate the

[3] If one is curious how this is possible, one should look up "quines" on Google; these are programs that print out their own source code. This is also a consequence of the *Kleene recursion theorem* from computability theory.

game by running the Turing Machine V with the input 17 and ask V to generate S_{17}, in order to figure out to sample (x, y) from μ_{17}. Then after receiving (a, b) from Alice and Bob, the referee can run V again to obtain another Turing machine D_{17}, which can then used to compute the decision procedure for G_{17} in order to decide whether Alice and Bob have won the game or not.

The insistence that the Turing machine V runs in time $\text{poly}(\log n)$ is to ensure that it runs in polynomial time in its input, which is an integer n represented in binary (requiring $O(\log n)$ bits). The fact that this is polylogarithmic is not super essential, but we just need to have *some* bound on its complexity that is asymptotically less than n^c for a universal constant $c > 0$.

Before we get into the compression theorem, we need just one more bit of notation, which is to introduce the concept of *entanglement lower bound for nonlocal games*. For a nonlocal game G, let $\mathcal{E}(G, p)$ denote the minimum dimension d such that there exists a d-dimensional strategy that succeeds with probability at least p in the game G. If there is no such finite d, then we define $\mathcal{E}(G, p) = +\infty$. So for example:

- $\mathcal{E}(CHSH, \frac{3}{4}) = 0$, because we can win CHSH with probability $\frac{3}{4}$ classically (i.e. with no entanglement)
- $\mathcal{E}(CHSH, \cos^2(\pi/8)) = 2$, as witnessed by the canonical textbook strategy as described within Lecture 6.
- $\mathcal{E}(CHSH, 1) = \infty$, as shown by Tsirelson's bound, it is impossible to win CHSH with probability higher than $\cos^2(\pi/8)$ with any entanglement.

Now we are ready to discuss the *compression theorem* for uniform game sequences.

10.2.3 Compression Theorem for nonlocal games

In the rest of this section, we'll assume all game sequences $\mathcal{G}$ consist of *polynomial-time computable games*. In other words, each of the games G_n in the sequence $\mathcal{G}$ can be executed in time $\text{poly}(n)$: sampling the questions (x, y) from μ_n takes $\text{poly}(n)$ time, and computing the decision procedure D_n takes only $\text{poly}(n)$ time.

The compression theorem gives a procedure for transforming a UGS $\mathcal{G} = (G_n)$ to another UGS $\mathcal{G}' = (G'_n)$ where the game G'_n *simulates* the game G_n. More precisely:

Theorem 28 (Compression Theorem)**.** *There exists an algorithm C that takes as input Turing machines and outputs Turing machines, and has the following property: Let $\mathcal{G} = (G_n)$ denote a UGS such that every game G_n is complexity at most $O(n^2)$, and let V denote the Turing machine that computes (G_n). Then if the output of C on input V is a Turing machine V' which computes a UGS $\mathcal{G}' = (G'_n)$ of polynomial-time computable games where for all $n \in \mathbb{N}$:*

- *If $\omega^*(G_n) = 1$, then $\omega^*(G'_n) = 1$*
- $\mathcal{E}(G'_n, \frac{1}{2}) \geq \max\{\mathcal{E}(G_n, \frac{1}{2}), n\}$
- *The time complexity of G'_n is $\text{poly}(\log n)$.*

We measure the complexity of a game by the amount of time spent by the verifier generating the questions as well as computing the decision procedure.

There's quite a lot of moving pieces in this Compression theorem but let's take a moment to parse what's going on. The compression theorem transforms one sequence of games $\mathcal{G} = (G_1, G_2, \ldots)$ where each game has time complexity at most $O(n^2)$ into another $\mathcal{G}' = (G'_1, G'_2, \ldots)$ where for each n, the game G'_n (called the n-th "compressed game") simulates the game G_n (called the n-th "original game") in a certain sense, except that G'_n has much smaller complexity than G_n – hence the name "compressed".Thus, the compressed games G'_n are *exponentially* more efficient than the original games G_n, but still capture key aspects of their winnability (as indicated by the first two bullet points).

This notion of compressing nonlocal games was first introduced by Zhengfeng Ji [Ji16], albeit in a weaker and more general form. In particular, it could not be recursively invoked. The power of the compression theorem presented here is that it can be composed recursively.

10.3 Recursive self-compression

Fix a Turing machine M. We would like to encode its halt/non-halt behaviour into the quantum value of a nonlocal game.

We'll define a sequence of games $\mathcal{G} = (G_1, G_2, \ldots)$ that depends on the Turing machine M. The n-th game G_n has the following behaviour, described in pseudocode:

Nonlocal game 2: The verifier behaviour in the game G_n

1 Run M on the empty input for n time steps. If M halts in this time, accept.
2 Otherwise, run the compression procedure C on the Turing machine V, which uniformly computes $\mathcal{G}$ to obtain a Turing machine V'.
3 Run V' on input $n+1$ to get the description of a game G'_{n+1}.
4 Play the game G'_{n+1} with Alice and Bob.

First, something to clarify here. What is V? This is the Turing machine that uniformly computes the sequence $\mathcal{G}$. But wait a minute – there seems to be something circular going on here! We're defining the sequence $(G_1, G_2, \ldots)$ *in terms* of a Turing machine that defines the same sequence. Is this allowed? It turns out that this is possible, for the same reason as in our explanation of the undecidability of the Halting problem.

First, let's convince ourselves that this really describes a uniform sequence of games. One could imagine writing out the code of a Turing machine V that on input n, outputs say the Python code that implements the sampling and decision procedure of game G_n (assuming that the compression procedure C has a Python implementation).

Next, what is the complexity of the games in this sequence? Let's do an accounting of the time required to execute game G_n:

- Running M takes n steps.
- Computing the compression procedure C on input V takes $O(1)$ time, independent of n. This is because V does not depend on n at all.

- Computing the description of G'_n is $\text{poly}(\log n)$, by definition of UGS.
- The Compression Theorem guarantees that the game G'_n runs in polynomial time $\text{poly}(\log n)$.

Adding this up, we get that the time compexity is $n + O(1) + \text{poly}(\log n)$ which is asymptotically less than $O(n^2)$, which satisfies the requirements of the Compression Theorem.[4]

Now we want to reason about the quantum value $\omega^*(G_n)$, for all n. Let's consider two cases.

M halts on the empty input. Suppose M eventually halts on the empty input, say in time T. Then clearly for all $n \geq T$, $\omega^*(G_n) = 1$ (because the verifier just automatically accepts). What about $n < T$? Well, the verifier always moves on to Line 2, and computes the compressed sequence $\mathcal{G}' = (G'_n)$, and plays the game G'_{n+1}. Thus by definition

$$\omega^*(G_n) = \omega^*(G'_{n+1}).$$

Let's see what the Compression Theorem promises us. Since we know that $\omega^*(G_T) = 1$, this means $\omega^*(G'_T) = 1$. So therefore $\omega^*(G_{T-1}) = 1$ as well. One can continue this line of reasoning to get that in fact

$$\omega^*(G_n) = 1$$

for all n.

M never halts. Suppose M never halts. Then for all n, the verifier in the game G_n always moves ahead to Line 2 and we have that

$$\omega^*(G_n) = \omega^*(G'_{n+1}).$$

It seems like we're stuck because it seems like the only thing we know about $\omega^*(G'_{n+1})$ is that it's 1 if $\omega^*(G_{n+1})$ is 1...

This is where we need to use the entanglement lower bound. Not only do we know that the values of G_n and G'_{n+1} are equal, they're identical games, meaning that their entanglement lower bounds are the same: for all p,

$$\mathcal{E}(G_n, p) = \mathcal{E}(G'_{n+1}, p)$$

Let's set $p = \frac{1}{2}$. Then the Compression Theorem guarantees us that

$$\begin{aligned}
\mathcal{E}(G'_{n+1}, p) &\geq \mathcal{E}(G_{n+1}, p) \\
&= \mathcal{E}(G'_{n+2}, p) \\
&\geq \mathcal{E}(G_{n+2}, p) \\
&= \cdots
\end{aligned}$$

[4] You might be rightly suspicious that we've used yet another piece of circularity here: in order to do this accounting, we invoked the conclusions of the Compression Theorem, before we even knew that the assumptions on $\mathcal{G}$ were satisfied! This issue can be handled by adding a forced "time-out" to the games G_n; they are essentially forced to run in time $O(n^2)$ (if they don't finish what they're doing within the given time window, then verifier is supposed to automatically accept). Thus by definition they run in $O(n^2)$, automatically satisfying the requirements of the Compression Theorem. We then conclude this artificially-imposed "time-out" never occurs.

So in other words, $\mathcal{E}(G_n, p) = \mathcal{E}(G'_{n+1}, p) \geq \mathcal{E}(G'_m, p)$ for all $m > n$. On the other hand, $\mathcal{E}(G'_m, p) \geq m$ for all m. So $\mathcal{E}(G_n, p) \geq m$ for all integer m, hence, we have $\mathcal{E}(G_n, p) = \infty$, and thus there is no finite-dimensional strategy to win with probability p. Therefore

$$\omega^*(G_n) \leq p = \frac{1}{2}.$$

for all n. This is because $\omega^*(\cdot)$ is defined as the limit of maximum success probabilities over all finite-dimensional strategies.

Finishing the reduction. We're basically there. Consider the transformation that takes an arbitrary Turing machine M, computes the description of the Turing machine V (which depends on M), and executes V on input $n = 1$ to get the description of the first game G_1. We will let $G_M = G_1$. If M halts, then $\omega^*(G_M) = 1$. If M never halts, then $\omega^*(G_M) \leq \frac{1}{2}$.

Next time. Clearly, this Compression Theorem is quite powerful. Using it, one can construct a family of self-referential games where the optimal success probability of quantum players depends on whether a given Turing machine M halts or not.

Next time, we'll get a glimpse of ideas that go into proving the Compression Theorem. As a sneak preview, the two main pillars are: rigidity of nonlocal games, and probabilistically checkable proofs from classical complexity theory.

Chapter 11

MIP* Part III

Scribes: Alex Rodriguez, Andrew Nader

11.1 How to compress uniform game sequences

Recall that last time we discussed the following Compression Theorem:

Theorem 29 (Compression Theorem)**.** *There exists an algorithm C that takes as input Turing machines and outputs Turing machines, and has the following property: Let $\mathcal{G} = (G_n)$ denote a UGS such that every game G_n has time complexity at most $O(n^2)$, and let V denote the Turing machine that computes (G_n). Then if the output of C on input V is a Turing machine V' which computes a UGS $\mathcal{G}' = (G'_n)$ of polynomial-time computable games where for all $n \in \mathbb{N}$:*

- *If $\omega^*(G_n) = 1$, then $\omega^*(G'_n) = 1$*
- *$\mathcal{E}(G'_n, 1/2) \geq \max\{\mathcal{E}(G_n, 1/2), n\}$*
- *The time complexity of G'_n is $\text{poly}(\log n)$.*

(For a refresher of the definitions of UGS, complexity of games, and so on, see Lecture 10).

We used this Compression Theorem to engineer a reduction from the Halting Problem to the problem of deciding whether a nonlocal game has quantum value 1 or at most $\frac{1}{2}$, promised that one is the case. Today, we'll see how to prove a weaker version of the Compression Theorem, but has the central ideas.

Theorem 30 (Simpler Compression Theorem)**.** *There exists an algorithm C that takes as input Turing machines and outputs Turing machines, and has the following property: Let $\mathcal{G} = (G_n)$ denote a UGS such that every game G_n is complexity at most $O(n^2)$, and let V denote the Turing machine that computes (G_n). Then if the output of C on input V is a Turing machine V' which computes a UGS $\mathcal{G}' = (G'_n)$ of polynomial-time computable games where for all $n \in \mathbb{N}$:*

- *(YES case) If $\omega^*(G_n) = 1$, then $\omega^*(G'_n) = 1$*

- *(NO case) If* $\omega^*(G_n) < 1$*, then* $\omega^*(G'_n) < 1$
- *The time complexity of* G'_n *is* $\text{poly}(\log n)$.

11.1.1 High-level, intuitive idea

The inner workings of the (simpler) Compression procedure will draw upon many of the concepts and ideas we've discussed in class:

1. Rigidity of nonlocal games
2. Verification of computations using multiple provers
3. The uncertainty principle
4. Probabilistically checkable proofs

The way we put all of these together as follows: fix an index n. We would like to design a nonlocal game G'_n (which we'll call the *compressed game*) "simulates" G_n (which we'll call the *original game*, or OG) in the sense that:

1. The winning behaviour of G'_n reflects that of G_n, in the sense that the OG has value 1 if and only if the compressed game G'_n has value 1. This by itself is a trivial requirement, because one can simply say that the compressed game is exactly the same as the OG, but...
2. ...we also demand that the *complexity* of the compressed game is *smaller* than the complexity of the OG.

Since the verifier in the compressed game G'_n doesn't have enough time to just play the OG G_n, it will *offload* the work to the provers Alice and Bob: it will have *them* simulate the OG verifier in G_n playing the game, and them reporting whether the simulated OG verifier would've accepted or not.

So this is conceptually related to the Grilo verification protocol we covered earlier: the verifier, instead of wanting to know whether a quantum circuit C accepts with high probability, now wants to know whether an *interactive protocol* between a verifier and two quantum provers accepts with high probability.

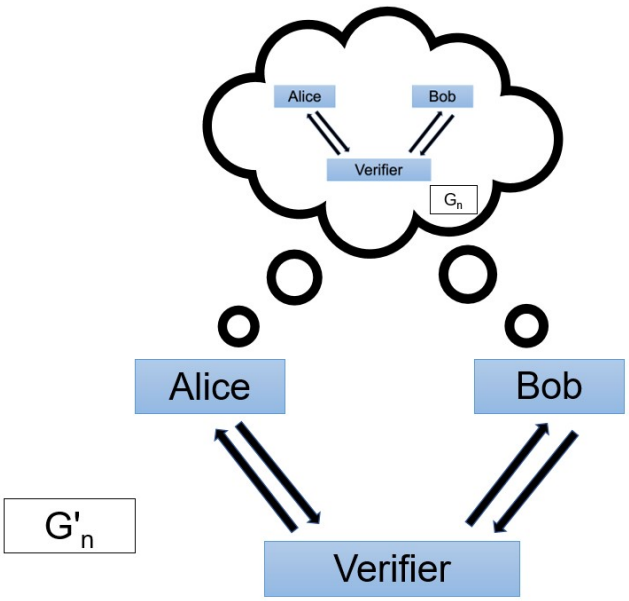


Figure 11.1: G'_n simulating G_n.

Compressing a game involves two transformations: *Question Reduction* and *Answer Reduction*. Here's a high level explanation of the two steps:

- *Question Reduction*: The verifier, instead of sampling the questions (x, y) from the distribution μ_n in the OG G_n itself, will instead offload the question generation process to the provers: through a sophisticated rigid nonlocal game, the verifier will check that Alice and Bob are sampling the questions (x, y) from μ_n themselves, where only Alice knows x, and only Bob knows y. The players then have to generate answers to the questions they sampled. The verifier will check that their sampled questions (x, y) and answers (a, b) satisfy the original decision procedure D_n.
- *Answer Reduction*: The verifier now wants to offload the work of computing D_n. Instead, it asks the players to not only generate (x, y, a, b), but also compute $D_n(x, y, a, b)$, report the result, and give a succinct *proof* that their reported result was computed correctly. The verifier checks all of this in less time than it takes to compute D_n itself.

The following table keeps track of the important properties throughout these transformations.

	Sampler Complexity	*Decider Complexity*	*Oracularizable*
Original game	$\text{poly}(n)$	$\text{poly}(n)$	yes
Question Reduction	$\text{poly}(\log n)$	$\text{poly}(n)$	yes
Answer Reduction	$\text{poly}(\log n)$	$\text{poly}(\log n)$	yes

Here's a brief explanation of the properties:

- *Sampler complexity* is the complexity of running the Turing machine S_n for the n-th game G_n. Recall that the way a nonlocal game is described is via a pair of Turing machines called a sampler and decider; the sampler gives instructions for how to sample from the question distribution μ_n.
- *Decider complexity* is the complexity of running the Turing machine D_n that computes the decision procedure.
- *Oracularizable* refers to a property that I will describe later on. It only matters for the Answer Reduction transformation.

The paper that introduced this idea of performing Question Reduction, followed by Answer Reduction to compress games is due to Anand Natarajan and John Wright, who proved that NEEXP (the complexity class of *nondeterministic doubly-exponential time*) is contained in MIP^*. Once this result was established, it became clear that one could — with a lot of work — refine these two procedures to obtain a recursively composable Compression Theorem (and thus obtain $\mathsf{MIP}^* = \mathsf{RE}$).

11.2 Question reduction

As mentioned, in this transformation we're going to reduce the complexity of the verifier's question sampling procedure by offloading the duty of sampling questions in the game G_n to the provers Alice and Bob. The verifier will instead just ask a smaller number of questions to carefully check that Alice and Bob are indeed sampling the questions in a proper way. This is where the verifier takes advantage of rigidity phenomenon.

How can we do this? Let's consider the simplest possible question distribution: the uniform distribution. In particular, suppose that μ_n is the uniform distribution over $\{0,1\}^n \times \{0,1\}^n$: Alice and Bob each get n bit strings x, y chosen uniformly at random.

We would like to run a game where Alice reports a uniformly random string x and Bob reports a uniformly random string y. We can use nonlocal games to certify that Alice and Bob are indeed generating (approximately) uniform randomness.

Here's a warmup: let's put aside the game G_n for now, imagine that we instead play the Magic Square game and the players win with probability 1 (or close to it). Since the game is rigid, we know that they must be (up to local isometries) sharing two EPR pairs and performing Pauli observables on their EPR pairs, as specified by this grid:

$$\begin{array}{ccc} IZ & ZI & ZZ \\ XI & IX & XX \\ -XZ & -ZX & YY \end{array}$$

The way we described the game before a few lectures ago, Alice was the "constraint" player (because she received a row/column as her question) and Bob was the "variable" player (because he received one of the nine squares as his question). We'll consider a *symmetrized* version of the Magic Square

game where the choice of who is the constraint player and who is the variable player is randomized. Sometimes Alice is the constraint player, but sometimes she's chosen as the variable player. Each player knows which role they're getting. It's not hard to see that this game is essentially equivalent to the original Magic Square game; it has quantum value 1, it's rigid, and we use the same strategy as described by the table above.

Now, we'll do something similar to the Grilo verification protocol where we used the Magic Square game (well, the repeated Magic Square game technically) as a subgame to ensure that the players are doing what we want them to. Imagine that we do the following: with probability 1/2, we play one of these games at random:

- **Rigidity test**. Play (Symmetrized) Magic Square.
- **Sampling test**. Tell both players to be a "Variable" player, and tell them to report the value of square 1 (top left corner). They both respond with a bit, and accept if their bits are equal.

Suppose they win this game with probability (close to) 1. What can we deduce about their answers in the Sampling Test?

They must be winning the rigidity test with probability close to 1. So by the rigidity of Magic Square, when each player is told to be a variable player and report the value of the first square, they will measure their qubits with the two-qubit observable $I \otimes Z$, which is equivalent to measuring their second qubit in the standard basis. Since their second qubits form an EPR pair, they will report identical outcomes that are uniformly random.

So we've just designed a way to test whether Alice and Bob are generating uniform randomness. However we're only certifying one bit of randomness from Alice and Bob, and the verifier has to put in more than one bit of randomness in to play the game to begin with! This is not so useful, so instead we use a more sophisticated family of rigidity tests, called the *Quantum Low-Degree Test* (first developed by Natarajan and Vidick), which I'll abbreviate by QLD. This actually is an infinite family of nonlocal games $(\mathsf{QLD}_1, \mathsf{QLD}_2, \ldots)$ where for each integer n,

1. QLD_n has complexity $\mathrm{poly}(\log n)$,
2. QLD_n has quantum value 1, and is (robustly) rigid with an optimal strategy that involves n EPR pairs, and furthermore there if each player gets a question labelled "TOTAL-Z", they measure their n EPR pairs in the standard basis and report their n-bit outcomes. If a player receives a question labelled "TOTAL-X", they measure their n EPR pairs in the X-basis and report their n-bit outcomes.

Using the Quantum Low-Degree Test as our rigidity test, and asking the "TOTAL-Z" question to Alice and Bob in the sampling test, we (as the verifier) can certify that Alice and Bob are generating uniformly random n-bit strings, all the while only expending $\mathrm{poly}(\log n)$ amounts of computation to do so.

We are moving closer to our goal: the idea is that the verifier can force Alice and Bob to generate this uniform randomness, which then Alice and Bob could go ahead and treat as questions in the original game G_n. There still remains a big problem that needs to be addressed: in the scheme we

just described, when Alice and Bob both perform the "TOTAL-Z" measurement, they get *identical* n-bit strings!

However, in the game G_n, we're assuming that Alice and Bob receive *independent* n-bit strings. And it is very important that Alice doesn't know what question Bob received and vice versa. How can the verifier certify that Alice and Bob sample independent strings, and have no idea what string the other player got?

Let's tackle the first problem of sampling independent strings: in the Rigidity test we'll have Alice and Bob play *two* parallel instances of QLD_n. This will certify that they share two blocks of n EPR pairs, and on each block they can measure "TOTAL-Z/X". Let's call the four possible "TOTAL" questions TX_1, TX_2 (which represent measuring the first and second block in the X basis, respectively), and TZ_1, TZ_2 (which represent measuring the first and second block in the Z basis, respectively).

Let's start putting this together into what we call the *Introspection protocol*: instead of directly asking the players the question x and y, we'll get them to introspectively ask themselves the questions, and provide their answers.

- **Rigidity test**. Play two parallel instances of QLD_n.
- **Introspection test**. Ask Alice and Bob to "INTROSPECT". Alice will send a pair (x, a), and Bob sends a pair (y, b). The verifier will accept if $D_n(x, y, a, b) = 1$.
- **Alice sampling check**. Ask Alice to "INTROSPECT". Alice will send a pair (x, a). Ask Bob to answer TZ_1. Bob will send a string x'. The verifier will accept if $x = x'$.
- **Bob samping check**. (Analogous to above).

Let us describe pictorially what is going on. The original game G_n is shown below in figure 2. Here, we ask Alice x, Bob y, and they respond with a, b respectively, and then we compute if $D_n(x, y, a, b) = 1$ or not.

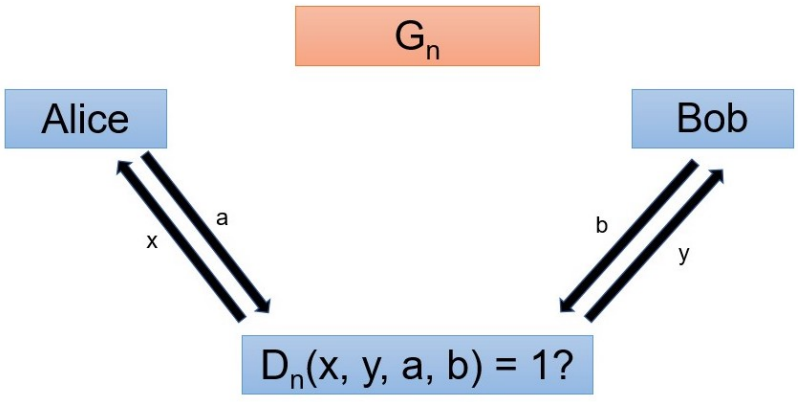


Figure 11.2: Original game G_n.

In the introspection protocol (depicted in Figure 11.3), however, the protocol is slightly different. Instead of sending x, y, we send Alice and Bob a single command to introspect, and they send back x, a and y, b respectively, and then we still compute if $D_n(x, y, a, b) = 1$ or not.

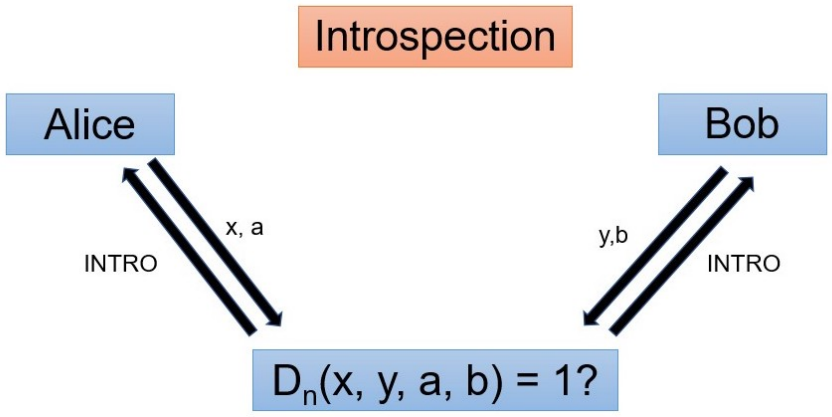


Figure 11.3: Introspection.

What are we hoping is accomplished by the introspection protocol? This is depicted below in Figure 11.4. First, note that we know that by playing this rigidity test, we know they're sharing the two blocks of EPR pairs depicted below. We also have the auxiliary entanglement which we cannot characterize yet. What we are hoping is that when we ask them to introspect Alice will measure her first block using the measurement TZ_1. She then gets a uniformly generated string x. When Bob is asked to introspect he'll measure his second block using TZ_2 and he'll get a independently random string y. Alice is then going to go to her auxiliary quantum states, and measure using whatever strategy she would have used in the original game. Bob does the same with his auxiliary quantum states.

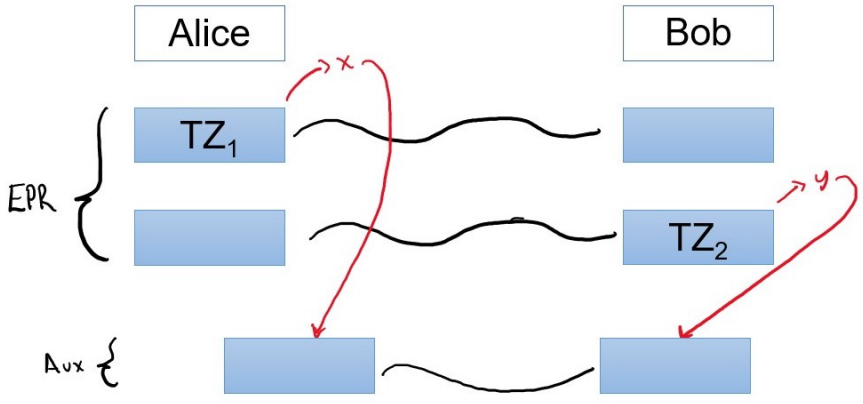


Figure 11.4: Intended Introspection Process

There's still a problem: we need to ensure that when Alice and Bob introspects, her answer a *only* depends on Alice's introspected question x and not Bob's introspected question y. Same thing with Bob. The thing we're worried about is the following sneaky strategy: when Alice is asked to "INTROSPECT", she will measure *both* blocks in the Z basis and obtain both (x, y). We're no longer able to ensure that we're simulating G_n properly if Alice or Bob know each other's questions.

Thus, what we need is for Alice to *prove* to us that she's not using any Z-basis information from the second block in her generation of her answer a. She can do this by measuring the second block of qubits in the X basis! Here we invoke the uncertainty principle: if she's measured the second block of qubits in the X-basis (which we can ensure she does by tying it to the Quantum Low-Degree test), then she cannot have *any* information about what Bob's outcomes would be if he measured the same second block in the Z-basis.

So our final Introspection protocol looks like the following (see Figure 11.5):

- **Rigidity test**. Play two parallel instances of QLD_n.
- **Introspection test**. Ask Alice and Bob to "INTROSPECT". Alice will send a pair (x, u, a), and Bob sends a pair (y, v, b). Accept if $D_n(x, y, a, b) = 1$.
- **Alice sampling check**. Ask Alice to "INTROSPECT". Alice will send a pair (x, u, a). With probability $1/2$, perform one of the following:
  1. Ask Bob TZ_1. Bob will send a string x'. Accept if $x = x'$, or
  2. Ask Bob TX_2. Bob will send a string u'. Accept if $u = u'$.
- **Bob samping check**. (Analogous to above).

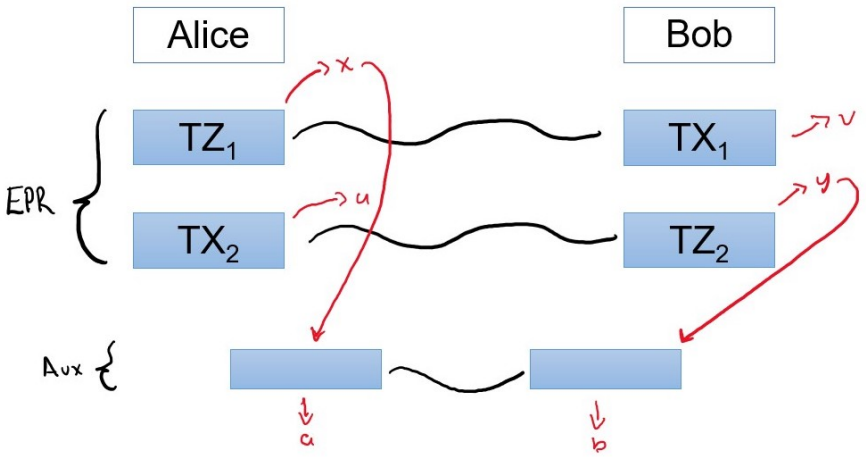


Figure 11.5: Final Introspection Process

I now claim that this Introspection protocol simulates the game G_n, but with the added feature that we've reduced the question sampling complexity to $\text{poly}(\log n)$. This is because the sampling complexity is dominated by this rigidity test, and we're promised its complexity is $\text{poly}(\log n)$.

11.3 Answer reduction

So we've succeeded in offloading the process of generating questions to the provers, and we now have a question sampling complexity of $\text{poly}(\log n)$. We now want to deal with the problem of *answer reduction*, i.e the problem of reducing the complexity of computing the decision predicate D_n. We will follow the same strategy as before, and try to offload the work to the provers as much as we can. To do this, we will make use of classical probabilistically checkable proofs, which were discussed in earlier lectures. First of all, we're going to temporarily forget that we transformed a game G_n using question reduction; for now let's just imagine that we're given a game G_n with $\text{poly}(\log n)$ question sampling complexity and $O(n^2)$ decision complexity, and we want to reduce the latter to $\text{poly}(\log n)$. We will assume that G_n has the property of being *oracularizable*, which I'll explain in a moment.

In the game G_n, we have a verifier who sends questions x and y to Alice and Bob, who then respond with answers a and b. We want the provers Alice and Bob to prove to the verifier that $D_n(x, y, a, b) = 1$. But there's a conceptual difficulty here: neither Alice nor Bob has all of the information needed! Alice only knows x and a, and Bob only knows b and y. So each player alone has no idea whether the statement $D_n(x, y, a, b) = 1$ is true, and they are not allowed to communicate with each other. How do we fix this issue? Well, we have to use something called the oracularization transformation. This oracularization might seem like a nonsensical idea at first, but it turns out that it works. First, let's define a new game G_n^{orac}.

As we can see from Figure 6, this game also has two players, but this time, we call them P1 and P2. We do this renaming to avoid confusion, since P1 will play the role of both Alice and Bob here. In this new game, the first player P1 will get *both* questions (x, y) and will return a pair of answers (a, b). The verifier will then check that $D_n(x, y, a, b) = 1$. By itself, this is problematic because P1 now knows both questions (x, y), and can easily generate answers (a, b) that satisfy the decision procedure – this may not be a faithful simulation of G_n where Alice only knows x to generate a and Bob only knows y to generate b.

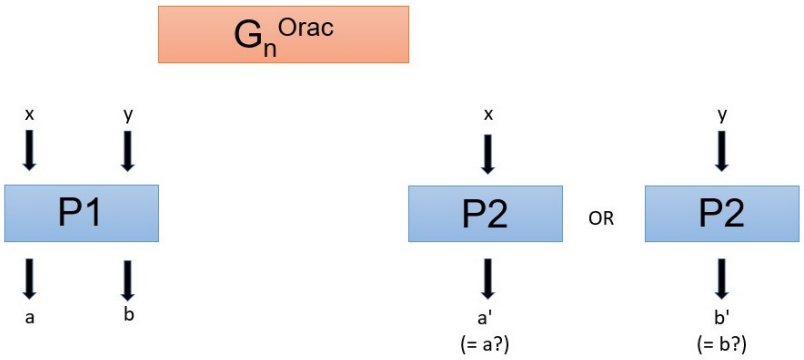


Figure 11.6: Oracularization transformation

This is where we're going to use the second player P2 to make sure that a only depends on x and b only depends on y. We're just going to send P2 a single question chosen with probability $\frac{1}{2}$. Let's say that we send the question x, then we get a single answer a' back. Then, we're going to check for consistency between a and a'. Or we can do the same thing with question y: we send y to P2, get b' back, and check for consistency between b and b'. The point of this is that we know that a' cannot depend on y, and b' cannot depend on x since P2 does not get both questions. So if a' happens to agree with a in this setting, by transitivity, we know that a also cannot depend on y, and the same holds for b', b, and x. So in this new game, we're saying to P1: "I'm going to give you both questions x and y, and you're going to give me both answers a and b. I'm then going to check to see if x, y, a, and b satisfy the original game G_n. But to make sure that you're doing what you're supposed to, I'm going to use P2 to keep you honest." The reason this transformation is called *oracularization* is because one of the players (P1) is treated as an "oracle" who knows the answers to both (x, y).

We can always take any game G_n and oracularize it to a game G_n^{orac}. How does the value of G_n^{orac} relate to that of G_n, though? A property called *soundness* is preserved, meaning that if $\omega^*(G_n) < 1$, then $\omega^*(G_n^{\text{orac}}) < 1$. The way this is proved is by arguing the contrapositive: suppose there was a quantum strategy for G_n^{orac} that succeeds with probability 1 (or that there is a sequence of strategies with success probability approaching 1). Then from this oracularized strategy, one can "extract" a perfect quantum strategy for the original game G_n.

What's less clear is whether the *completeness* property holds: we would like that if $\omega^*(G_n) = 1$, then $\omega^*(G_n^{\text{orac}}) = 1$ as well. In other words, if it is possible to win the original game perfectly, then it is also possible to do the same for the oracularized game. A game with this property is called *oracularizable*.

Not all games are oracularizable. For example, consider the following variant of the Magic Square game: instead of Alice getting a row/column and Bob getting a cell in that row/column, Alice gets a random row, and Bob gets a random column. Their assignments to the cells have to satisfy the Magic Square constraints, and their assignments to the cell at which the row and column overlap (there must always be such a cell) must match.

Imagine that we now oracularize this variant of the Magic Square game. Player P1 gets a row and a column, and Player P2 gets either the row or column. If P2's answers always consistent with P1's answers, then we can deduce the following: P1's measurement to generate its answers can be factored into two simultaneous measurements: one measurement that only depends on the row to produce the row cell assignments, and another measurement that only depends on the column to produce the column cell assignments. These measurements must commute with each other if they can be simultaneously measured. Furthermore, these measurements can be used to obtain a perfect quantum strategy for the variant of the Magic Square game. However, this is not possible: due to the rigidity of the Magic Square game, the row and column measurements *cannot* commute with each other. Thus this variant of Magic Square is not oracularizable.

The formulation of the Magic Square game that we originally presented, however, *does* have this oracularizability property; this is essentially due to the fact that Bob's cell measurement always commutes with Alice's row/column measurement (when considered as operators acting on the same space $\mathbb{C}^2 \otimes \mathbb{C}^2$).

Since we are assuming the "input" game to Answer Reduction is oracularizable, we have the following: $\omega^*(G_n) = 1$ if and only if $\omega^*(G_n^{\text{orac}}) = 1$.

The complexity of this game has not been reduced yet (the sampling complexity is still $\text{poly}(\log n)$ and the decision complexity is still $O(n^2)$), but we have a game G_n^{orac}, equivalent to the original game G_n, such that one of the players has all the information needed to generate a succinct *proof* that given question pair (x, y), the player can generate answers (a, b) such that

1. a only depends on x, and b only depends on y, and
2. $D_n(x, y, a, b) = 1$.

The complexity of checking this proof will only be $\text{poly}(\log n)$, instead of $O(n^2)$.

We're going to transform the oracularized game G_n^{orac} to the final *answer reduced* game G_n^{ar} which has the desired $\text{poly}(\log n)$ complexity for both the sampler and decider. In the answer reduced game, the high level idea is the following: the oracle player who gets both questions (x, y) from the input game G_n, instead of reporting the answers (a, b) back to the verifier, is instead supposed to compute a *probabilistically checkable proof* (PCP) of the statement that "$D_n(x, y, a, b) = 1$". Note that, given the questions (x, y) and answers (a, b), the player P1 can deterministically compute a proof string Π of this statement (if the statement is true). Recall that such a PCP has the following wondrous checkability property: by examining $O(1)$ random bits from the proof Π, a PCP verifier can determine with high confidence whether it is a valid proof of the statement. More precisely, if the proof is valid, then the PCP verifier accepts with probability 1; otherwise the PCP verifier accepts with probability that is some constant strictly less than 1 (for example 1/2). Furthermore the efficiency of the PCP implies that the PCP verifier only requires $\text{poly}(\log n)$ time – very convenient! For simplicity let's assume the number of queries to the proof is 3.

Thus, roughly speaking, the verifier in the answer reduced game G_n^{ar} wants to run this PCP checking procedure to select $O(1)$ random bits from this purported proof Π to verify that the statement is true.

There are a number of immediate challenges to this idea:

1. The PCP string Π is going to have length $\text{poly}(n) \geq n^2$. The oracle player can generate

this, but it cannot send the string in its entirety to the verifier, because just reading that string alone would take at least n^2 time, which is much larger than the poly($\log n$) decision complexity we were aiming for.

2. On the other hand, the verifier cannot generate random three proof locations (i_1, i_2, i_3), send these to P1, and expect P1 to honestly return $(\Pi_{i_1}, \Pi_{i_2}, \Pi_{i_3})$. This is because if the statement happens to be *false*, but P1 knows what proof locations the verifier is interested in, P1 can concoct "fake proof bits" (r, s, t) that convinces the verifier that the proof was valid.

3. We still have to check consistency between P1 and P2 to ensure that a only depends on x and b only depends on y. However the answers a, b will also generally have length poly(n). So somehow this consistency checking between strings of length poly(n) has to be done in time poly($\log n$).

On the surface, these challenges seem pretty serious. But surprisingly, it is possible to deal with them. I won't be able to get into all the details here, but here are the high level ideas: to deal with the issue that player P1 might generate the proof bits adaptively based on the proof locations sent by the verifier, the verifier will use P2 sometimes to check that P1's responses are as if P1 was just reading from an underlying proof string Π. This is similar to how oracularization forces a to depend on x only and b to depend on y only.

To perform the consistency check between P1 and P2 to ensure that a only depends on x and b only depends on y, the verifier will instead ask P1 and P2 to provide "proofs" of consistency using error-correcting codes. Again there are more consistency checks to be performed to ensure that these proofs are not generated in an adaptive fashion.

The details get quite gnarly, but the upshot is that all of this can be done with total complexity poly($\log n$). Thus we have transformed an oracularizable game G_n with small sampling but large decision complexity into a game G_n^{ar} with small sampling and small decision complexity. Furthermore, the answer reduced game is equivalent to the input game in the sense that $\omega^*(G_n) = 1$ if and only if $\omega^*(G_n^{\text{ar}}) = 1$. Finally, the answer reduced game is also oracularizable.

Putting everything together, we can combine Question Reduction, plus Answer Reduction to get a sequence of transformations on oracularizable nonlocal games that shrink the complexity from $O(n^2)$ to poly($\log n$). Furthermore, the output game has value 1 if and only if the original game had value 1. This establishes the Simpler Compression Theorem described at the beginning of lecture.

Stepping back, we now have the ingredients that go into proving the general compression theorem. To do that, we use many of the tools that we talked about in class like PCPs, rigidity, non-local games and the uncertainty principle to squeeze down the complexity of the games. Using the compression theorem, we can get all of the consequences we talked about last time. This wraps up the course's discussion of $\mathsf{MIP}^* = \mathsf{RE}$, where we saw how all of these different ideas from computer science, quantum physics, and pure mathematics interact. We expect all of these different ideas to be pointing to some central or common phenomenon, though we're not sure what that phenomenon is, exactly. This is an open research question, and a very interesting one, indeed.

Chapter 12

QMA(2) and the power of unentanglement

Scribes: Shiquan Zhang

12.1 Motivation

Throughout this class we've seen examples of the power and complexity of entanglement:

1. **Quantum proofs**: Using quantum entangled states, Merlin can prove to Arthur that a local Hamiltonian has low ground energy.
2. **Quantum PCP Conjecture**: Assuming the Quantum PCP Conjecture, there exist local Hamiltonians where at relatively high energies, the states of the system possess irreducibly complex entanglement.
3. **Verification of quantum computations**: By playing nonlocal games with quantum provers, a classical verifier can certify the precise structure of the entanglement and measurements used by the provers. This can be bootstrapped to verify full-fledged quantum computations.
4. **Complexity of nonlocal games**: How hard is it to estimate the optimal winning probability of quantum players in a nonlocal game? Extremely hard – beyond the ability of any algorithm to solve. Put another way, it is impossible to algorithmically optimize (even approximately) over the set of all quantum correlations.

In this last lecture of the course, we're going to end on something completely different: we're going to explore the power of *unentanglement*. We're going to talk about the complexity class imaginatively called QMA(2): this is a model of quantum proofs where Merlin sends you not one quantum state, but *two* quantum states, with the promise that they are *not entangled*.

12.1.1 QMA(2)

To be more precise, QMA(2) is the set of decision problems (L_{yes}, L_{no}) where there exists a quantum polynomial-time verifier V which has not one but two register for a quantum proof as shown in Figure 12.1, such that for all instances x,

- If $x \in L_{yes}$, then there exists a quantum proof state of the form $|\psi\rangle_A \otimes |\theta\rangle_B$ such that $V(x, |\psi\rangle_A \otimes |\theta\rangle_B)$ accepts with probability at least $2/3$.
- If $x \in L_{no}$, then for *all* quantum proof states of the form $|\psi\rangle_A \otimes |\theta\rangle_B$, the verifier V accepts $(x, |\psi\rangle_A \otimes |\theta\rangle_B)$ with probability at most $1/3$.

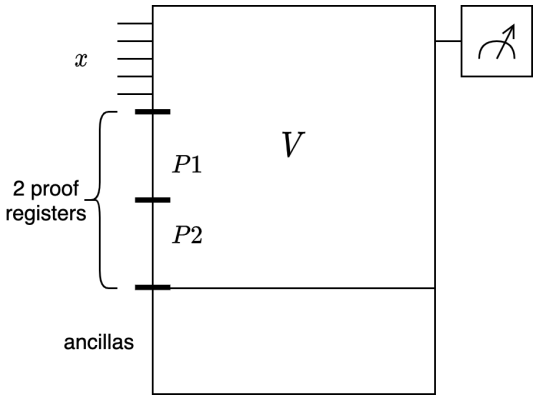


Figure 12.1: Circuit of the Verifier for QMA(2).

So this complexity class is very similar to the one we've talked about a lot, except there are a couple key differences. The first is that we now expect Merlin to provide not just an arbitrary state but one that is the *tensor product* of two quantum states. The second key difference is that in the NO case, we are only guaranteed that the verifier will reject with high probability *if Merlin provides a tensor product of two quantum states*. There is no guarantee what will happen if Merlin provides a general state that's entangled across both proof registers A and B – the verifier could very well be confused and accept even in the NO case.

12.1.2 QMA(2) VS QMA

What is the effect of forcing Merlin to provide two quantum states that are unentangled with each other to the verifier, rather than just one quantum state? This unentanglement guarantee doesn't decrease the ability of the verifier to check quantum proofs: any decision problem in QMA can always be verified in the QMA(2) model: the verifier can always ignore the second proof register. So QMA $\subseteq$ QMA(2). The more interesting question is, can this unentanglement guarantee give additional verification power to Arthur? This is where things start to get very interesting.

First, one might wonder: why can't the QMA verification model simulate the QMA(2) verification model? Suppose we have a QMA(2) verification procedure V for a decision problem L, and suppose

Merlin hands Arthur a monolithic quantum state $|\psi\rangle_{AB}$ on two registers that is possibly entangled. Can Arthur verify whether the state $|\psi\rangle_{AB} = |\varphi\rangle_A \otimes |\theta\rangle_B$? If so, then $\mathsf{QMA}(2) = \mathsf{QMA}$, because we can always ensure the unentanglement guarantee.

It's not so clear how to verify whether an arbitrary quantum state, when given in quantum form, is unentangled! Of course, if one had exponentially many copies of the state, then one could perform tomography to obtain a classical description of the state, and then classically compute whether the state is (approximately) unentangled or not. However there is only one copy of the state, so you can't do this. In fact, in the most naive formulation, it's impossible to tell whether a given state is unentangled or not: there is no measurement M that accepts only unentangled states $|\varphi\rangle_A \otimes |\theta\rangle_B$. Does anyone see why? This is because, since the set of unentangled states span all of the AB space, M must in fact accept *all* states, including entangled ones.

This gives some indication that the unentanglement guarantee is pretty special. In fact, we'll see an example of something you can do with an unentanglement guarantee that is unlikely doable without.

12.2 Verifying NP using short quantum proofs

We'll see how we can verify any NP decision problem using very short quantum proofs in the $\mathsf{QMA}(2)$ model. Let's take a canonical NP-complete problem called 3-COLORING. Here you're given a graph $G = (V, E)$, and the goal is to colour every vertex with one of three colours (say red, green, blue) such that neighbouring vertices have distinct colours. It is NP-hard to decide whether a given graph is 3-colourable or not. Clearly, given an n-vertex graph G, one can give a proof of its 3-colorability that's verifiable in (classical) polynomial time: provide the colouring $c : V \to \{r, g, b\}$. This colouring can be represented with $O(n)$ bits.

Would it be possible to provide a shorter proof of 3-colourability? Say, can Merlin convince Arthur that a graph is 3-colourable by sending a proof of length $n^{.99}$. It seems unlikely; first of all, this would imply a $3^{n^{.99}} = 2^{O(n^{.99})}$ time algorithm to solve 3-COLORING: given a graph G, we can simply enumerate over all possibly length-$n^{.99}$ proofs to see if Arthur (the verifier). If he does, then that means the graph is 3-colourable. If he doesn't, then the graph isn't.

To date, we do not know of any such algorithm: the best algorithms we have take 2^{cn} time for some constant $c > 0$, which is asymptotically much greater than $2^{O(n^{.99})}$. Due to the web of NP-completeness reductions, this is the case for most NP-complete problems: 3-SAT, SET COVER, MAX-CUT, and so on. The fact that for all of these problems we haven't been able to come up with algorithms that beat strictly exponential time (we mean that 2^n time is strictly exponential, whereas $2^{n^{1-\varepsilon}}$ is considered "subexponential"). This motivates the following conjecture:

Exponential Time Hypothesis *(ETH): Any deterministic algorithm that solves 3-SAT (or 3-COLORING) must take at least 2^{cn} time for some constant $c > 0$ in the worst case.*

This is a strengthening of the $\mathsf{P} \neq \mathsf{NP}$ conjecture; not only are there no polynomial time algorithms for 3-SAT but there are no subexponential time ones. Given the empirical evidence, this seems true to the best of our knowledge.

So, assuming the ETH, we do *not* expect short(er) proofs for 3-COLORING (or 3-SAT, or many

other NP-complete problems).

12.2.1 Quantum proofs for 3-COLORING

What about *quantum* proofs? Suppose Arthur wants to verify whether a graph is 3-colourable, and is willing to accept quantum proofs in order to determine this. Would it be possible for Merlin to send a compressed quantum proof on $n^{.99}$ qubits to convince Arthur of a graph's 3-colorability? In the QMA model, this is not possible if we assume the ETH. This is because given a QMA verifier that takes in a quantum witness of size m qubits, to classical compute its maximum acceptance probability, we really just need to compute the operator norm of a matrix that acts on m qubits, or in other words has dimension 2^m. [1] This takes time at most $\text{poly}(2^m) = 2^{O(m)}$ in order to compute. Thus if there was a QMA verifier for 3-COLORING that only required $m = n^{.99}$ qubits, again we would have a subexponential time *classical* algorithm for solving 3COLORING, which would violate the ETH.

What if Arthur had an unentanglement guarantee on the quantum proof? Suddenly the situation looks very different. It turns out that there is a QMA(2)-proof system for 3-COLORING that requires only two $O(\log n)$-qubit proofs, provided that they are unentangled with each other. This is an *exponential improvement* in proof size!

This proof system is due to Blier and Tapp [BT09]. The idea is as follows. Fix an n-vertex graph G. What Arthur hopes Merlin will provide is $|\psi\rangle \otimes |\psi\rangle$ where $|\psi\rangle$ is of the following form:

$$|\psi\rangle = \frac{1}{\sqrt{n}} \sum_{v \in V} |v\rangle \otimes |c_v\rangle$$

where the first register (called the *vertex register*) stores a vertex of the graph G, and the second register (called the *colour register*) stores a coloring of that vertex. So this proof state $|\psi\rangle$ is supposed to encode a purported 3-colouring of the graph G. It is succinct: it is only $O(\log n)$ qubits because the vertex register only requires $O(\log n)$ qubits and the colour register only requires $O(1)$ qubits.

Let's imagine for now that Merlin really did provide two copies of such a state $|\psi\rangle$. Then Arthur wants to check that this assignment of colours $(c_v)_{v \in V}$ really corresponds to a valid 3-colouring of the graph. Let's give names to the four registers: vertex registers $V1, V2$, and colour registers $C1, C2$. Arthur can measure all four registers in the standard basis to get two pairs (v_1, c_{v_1}) and (v_2, c_{v_2}) where v_1, v_2 are uniformly random and independently sampled vertices from V. First, if v_1 is not a neighbour of v_2, then Arthur just automatically accepts (there's nothing to check). If v_1 is a neighbour of v_2 (which happens with probability at least $1/|E|$), then Arthur checks if $c_{v_1} \neq c_{v_2}$ (i.e. their assigned colours are different). If they're the same colour, then Arthur rejects. Otherwise, Arthur accepts.

What is the completeness and soundness of this proof system?

[1] One can see this as follows: let V denote the QMA-verification circuit that takes as input $|G\rangle \otimes |\psi\rangle \otimes |0 \cdots 0\rangle$ where G is the input graph, $|\psi\rangle$ is the m-qubit quantum state, and the zeroes represent ancillas. The acceptance probability of V, for a fixed graph G, is $\| \Pi \cdot V |G\rangle \otimes |\psi\rangle \otimes |0 \cdots 0\rangle \|^2$ where Π is the projector onto the output qubit being in the state $|1\rangle$ (for "accept"). We can "fold" the input $|G\rangle$ and ancillas $|0 \cdots 0\rangle$ into the operator ΠV to get a matrix M that acts on 2^m-dimensional space. The acceptance probability is thus equal to $\langle \psi | M^\dagger M | \psi \rangle$; when maximized over $|\psi\rangle$ this is simply the operator norm of $M^\dagger M$, which is a $2^m \times 2^m$ matrix.

- Suppose G was 3-colourable. Then Merlin can indeed generate proof states $|\psi\rangle \otimes |\psi\rangle$ for some valid 3-colouring $c : V \to \{r, g, b\}$. When Arthur performs his check, he will accept with probability 1.
- Suppose G was not 3-colourable. Then for any colouring $c : V \to \{r, g, b\}$, there exists at least one edge $e = (x, y) \in E$ that violates the colouring constraint. Suppose Merlin sends $|\psi\rangle \otimes |\psi\rangle$ where $|\psi\rangle$ corresponds to a colouring c. Then the probability Arthur samples a violated edge $e = (x, y)$ is going to be at least $2/n^2$. Thus Arthur accepts with probability at most $1 - \frac{2}{n^2}$.

Thus there is an inverse-polynomial gap between the YES and NO cases. We'll talk more about how to boost this gap to a constant later, but first we have an important issue to deal with: while Arthur is guaranteed that Merlin is providing a state of the form $|\varphi\rangle \otimes |\theta\rangle$, how does Arthur know that (a) $|\varphi\rangle = |\theta\rangle$, and (b) they're of the form $\frac{1}{\sqrt{n}} \sum_{v \in V} |v\rangle \otimes |c_v\rangle$? In general, an arbitrary state on the vertex and colour registers could look like $\sum_{v,c} \alpha_{v,c} |v\rangle \otimes |c\rangle$ for arbitrary amplitudes $\{\alpha_{v,c}\}$; which need not be uniform and a vertex could have multiple colours assigned to it. A cheating Merlin could try to only have a superposition over a subset of vertices for which it can colour properly.

Fortunately there is a way for Arthur to check the well-formed-ness of the proof state. Given a state of the form $|\varphi\rangle \otimes |\theta\rangle$, Arthur can pick one of three tests at random to perform:

- **Colouring test**: what we described before; measure all registers in the standard basis and check if the colouring constraints are satisfied.
- **Swap test**: Arthur checks that $|\varphi\rangle \otimes |\theta\rangle$ by performing the *swap test*. This is given by the following circuit:

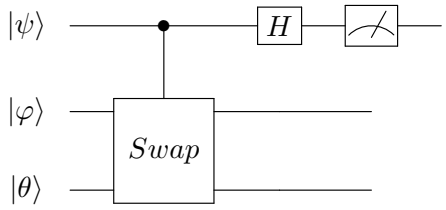


Figure 12.2: Circuit for Swap Test.

The probability that the output qubit measures 1 is $\frac{1}{2} + \frac{1}{2} \cdot |\langle\varphi|\theta\rangle|^2$. Thus the swap test succeeds with probability 1 if the states are equal; if they are orthogonal, the swap test succeeds with probability 1/2. For states that are somewhere in between, the swap test succeeds with probability in between 1/2 and 1.

- **Uniformity test**: Conditioned on the swap test succeeding with high probability, we can assume that Merlin's proof state is (approximately) of the form $|\psi\rangle \otimes |\psi\rangle$. Arthur now wants to check whether the state $|\psi\rangle$ is an equal superposition over all vertices. It will do so using

the *Quantum Fourier Transform* over $\mathbb{Z}_r$; this is the unitary map F_r that has the following behaviour:

$$|x\rangle \mapsto \frac{1}{\sqrt{r}} \sum_{y=0}^{r-1} \omega_r^{x \cdot y} |y\rangle$$

where x, y are interpreted as integers from 0 to $r-1$ and $\omega_r = e^{2\pi i/r}$ is the r-th root of unity. After Arthur applies the Quantum Fourier Transform F_n to the vertex register and F_3 to the colour register, it measures both registers in the standard basis and rejects if the colour register is in the $|0\rangle$ state but the vertex register is not in the $|0\rangle$ state.

If the state $|\psi\rangle$ is properly formatted, then the uniformity test will succeed with probability 1. Conversely, if the uniformity test succeeds with probability very close to 1 (according to Blier and Tapp's calculations, at least $1 - O(n^{-6})$), then the state $|\psi\rangle$ is very close to being properly formatted (i.e. it is a uniform superposition over vertices and a colour assignment for each vertex).

Putting these three tests together, we get a $\mathsf{QMA}(2)$ proof system for 3-COLORING where in the YES case, there is a $O(\log n)$-qubit proof Merlin can provide that is accepted with probability 1, and in the NO case, Arthur accepts with probability at most $1 - O(n^{-6})$.

12.2.2 Improving the completeness-soundness gap

This is pretty impressive, and suggests that the unentanglement guarantee is powerful for verification. However there is one major downside to the Blier-Tapp protocol: the completeness-soundness gap is atrocious. It goes to 0 with the instance size n at an inverse-polynomial rate. Normally for randomized complexity classes (such as BPP, BQP, QMA, etc) this is not a big deal because we can just repeat a polynomial times to amplify our completeness-soundness gap to a constant (say 2/3 vs 1/3).

Here there's a couple issues: first, it is *not* known that repeating the protocol in parallel (by having Merlin send two giant states $|\Phi\rangle \otimes |\Theta\rangle$ that is supposed to represent $|\varphi\rangle^{\otimes k} \otimes |\theta\rangle^{\otimes k}$, say) is still sound. (In contrast, we know that parallel repetition is fine for QMA proofs). Second, even if it *were* fine, we would have to repeat the protocol roughly n^6 times to get the completeness-soundness gap to a constant, which would mean n^6-qubit states – we no longer have short proofs.

A couple follow-up papers showed that the completeness-soundness gap could be improved while still maintaining shorter proofs: Aaronson, Beigi, Drucker, Fefferman and Shor [ABD+08] proved that, for 3-SAT instances of size m, there is a $\mathsf{QMA}(m)$ proof system for 3-SAT where Merlin provides $\sqrt{m} \cdot \operatorname{poly}\log(m)$ unentangled proofs of size $O(\log m)$ each (for a total of $\sqrt{m} \cdot \operatorname{poly}(\log m)$ qubits), *and* the completeness-soundness gap is a constant. This result ostensibly relies on a stronger unentanglement guarantee: there's unentanglement not just between two registers, but $\widetilde{O}(\sqrt{m})$ registers!

Their protocol is quite sophisticated, and uses *probabilistically checkable proofs* (PCPs)! Funny how these things keep on making an appearance. The idea to use PCPs in this setting is quite natural: one of the major reasons why the Blier-Tapp protocol has such poor completeness-soundness is because for 3-COLORING, in the NO Case, we are only guaranteed that there exists just *one* edge whose colouring constraint is violated. The probability that Arthur has in finding this one violated edge is going to be vanishingly small; it's roughly $1/n^2$. What if, on the other hand, we had the

guarantee that the graph is either 3-colourable or very far from being 3-colourable, meaning that for any 3-colouring, there is a significant fraction of edges that violate the colouring constraint? Arthur will then have an easier time of finding these violations via random sampling. PCPs are precisely a tool for securing this guarantee: given a graph G, by applying a PCP reduction to it, as shown in Figure 12.3, we obtain a graph G' where it's either 3-colourable (if G is a YES instance) or very far from being 3-colourable (if G is a NO instance).

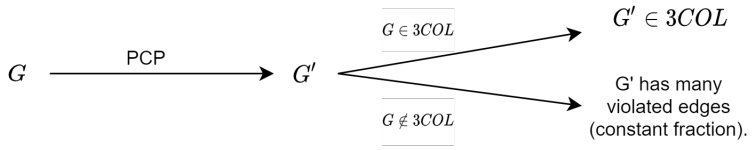


Figure 12.3: 3-colourable problem using PCP.

What about the increased number of unentangled proofs, though? This is not a big issue. In another paper, Harrow and Montanaro [HM13] proved that for all $k \geq 2$, $\mathsf{QMA}(k) = \mathsf{QMA}(2)$. That is, as long as you have an unentanglement guarantee between *two* registers, you can efficiently bootstrap that into an unentanglement guarantee between k registers. In other words, from the proof verification point of view, it's not more helpful to have many unentangled proofs as compared to having two unentangled proof states. Thus, the protocol of Aaronson, et al. [ABD+08] can be converted into an equivalent $\mathsf{QMA}(2)$ protocol.

12.3 The complexity of detecting mixed-state entanglement

The investigations into $\mathsf{QMA}(2)$ is not interesting just because it gives us a clever way to give shorter proofs of 3SAT and other NP-complete problems. The complexity of $\mathsf{QMA}(2)$ connects directly to a very basic and very fundamental questions about quantum information theory: if I give you a classical description of a quantum state, can you tell me if it's entangled or not?

There are two relevant formulations here. One is about *detecting pure state entanglement*: given a classical description of a state $|\psi\rangle \in \mathbb{C}^d \otimes \mathbb{C}^d$, determine whether $|\psi\rangle = |\varphi\rangle \otimes |\theta\rangle$. This is an easy linear algebra problem; one can for example take a partial trace of the state $|\psi\rangle$ on the first subsystem to see if it's a pure state. This takes time $\mathrm{poly}(d)$ on a classical computer. Just to be clear – this is different from what we discussed earlier about it being impossible to tell whether a given quantum state is entangled or not; that was when the state was given in *quantum form*. Here, the state is given to you as a list of amplitudes (i.e. a classical description).

The second formulation is about *detecting mixed state entanglement*: given a classical description of a density matrix ρ on two registers A and B (each of dimension d), determine whether ρ is *separable*. A density matrix is separable if it can be written as a convex combination of product states:

$$\rho = \sum_i p_i \, \sigma_i \otimes \tau_i$$

for density matrices $\sigma_i, \tau_i \in \mathbb{C}^{d \times d}$. Separable states are not considered to have any entanglement, but they might have *classical* correlations between the two registers. For example, consider two perfectly synchronized coin flips: both coins are heads or both coins are tails. The coins are correlated, but only in a classical way. On the other hand, if we take two qubits that are in the EPR state, this is not separable; the two qubits are entangled in a genuinely quantum fashion.

Suddenly this problem doesn't look so easy anymore. It turns out that distinguishing between classical correlations and quantum correlations is quite a difficult task! Gurvits [Gur04] proved that this problem, called the SEPARABLE STATES problem, as shown in Figure 12.4, is NP-complete in the worst case. But what about approximations? That is, what if we're asked to determine whether a given density matrix ρ is separable or is ε-far from any separable state, promised that one is the case?

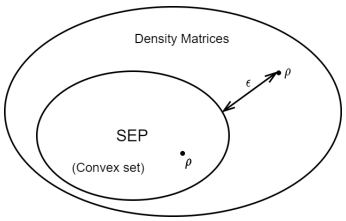


Figure 12.4: SEPARABLE STATES problem.

One of the biggest open problems in quantum information theory is whether an efficient algorithm exists for solving the approximate SEPARABLE STATES problem. Some people speculate that there is a *quasipolynomial*-time algorithm for this, meaning an algorithm that runs in $d^{\log d}$ time. Others conjecture that you really need some kind of exponential time in order to solve it. Nobody really knows.

But generally the belief seems to be, that *detecting pure state entanglement is easy, but detecting mixed state entanglement is hard.*

This question about the complexity of the SEPARABLE STATES problem is related to the complexity of QMA(2) in the following way. Right now, QMA(2) could be anywhere between QMA and NEXP. This is a ridiculously large expanse of complexity space; it could be anywhere in between such as EXP or PSPACE. If there were a quasipolynomial-time algorithm for the SEPARABLE STATES problem, then this would imply QMA(2) $\subseteq$ EXP. This is because we can reduce the problem of deciding a QMA(2) decision problem to whether a certain $2^m \times 2^m$ matrix M has a large value when applied to vectors that come from the set of separable states. If we had a quasipolynomial time (here quasipolynomial is in the dimension of the states, which is 2^m) algorithm for the set of separable states, then we could optimize over the set of separable states in $2^{\text{poly}(m)}$ time.

One of the fascinating things about this question is that people keep on discovering surprising connections to other areas of computer science and quantum information theory. For example, a paper by Barak, Brandao, Harrow, Kelner, Steurer, and Zhou [BBH+12] shows that there's an interesting web of reductions that relate the complexity of the SEPARABLE STATES problem to questions in *classical complexity theory*, namely the *Unique Games Conjecture* and its variants. These questions about the hardness of approximation of some fundamental optimization problems. It's remarkable that the SEPARABLE STATES problem is so closely connected to these questions, and it suggests that making progress on the quantum information theory side could give some

insight into some central classical computer science questions, and vice versa.

Bibliography

[ABD+08] S. Aaronson, S. Beigi, A. Drucker, B. Fefferman, and P. Shor. "The Power of Unentanglement". In: *2008 23rd Annual IEEE Conference on Computational Complexity*. 2008, pp. 223–236. DOI: 10.1109/CCC.2008.5.

[ALM+98] Sanjeev Arora, Carsten Lund, Rajeev Motwani, Madhu Sudan, and Mario Szegedy. "Proof Verification and the Hardness of Approximation Problems". In: *J. ACM* 45.3 (1998), pp. 501–555. DOI: 10.1145/278298.278306. URL: https://doi.org/10.1145/278298.278306.

[AS98] Sanjeev Arora and Shmuel Safra. "Probabilistic Checking of Proofs: A New Characterization of NP". In: *J. ACM* 45.1 (1998), pp. 70–122. DOI: 10.1145/273865.273901. URL: https://doi.org/10.1145/273865.273901.

[Bab85] László Babai. "Trading Group Theory for Randomness". In: *Proceedings of the 17th Annual ACM Symposium on Theory of Computing, May 6-8, 1985, Providence, Rhode Island, USA*. Ed. by Robert Sedgewick. ACM, 1985, pp. 421–429. DOI: 10.1145/22145.22192. URL: https://doi.org/10.1145/22145.22192.

[BBH+12] Boaz Barak, Fernando G.S.L. Brandao, Aram W. Harrow, Jonathan Kelner, David Steurer, and Yuan Zhou. "Hypercontractivity, Sum-of-Squares Proofs, and Their Applications". In: *Proceedings of the Forty-Fourth Annual ACM Symposium on Theory of Computing*. STOC '12. New York, New York, USA: Association for Computing Machinery, 2012, pp. 307–326. ISBN: 9781450312455. DOI: 10.1145/2213977.2214006. URL: https://doi.org/10.1145/2213977.2214006.

[BFL91] László Babai, Lance Fortnow, and Carsten Lund. "Non-Deterministic Exponential Time has Two-Prover Interactive Protocols". In: *Comput. Complex.* 1 (1991), pp. 3–40. DOI: 10.1007/BF01200056. URL: https://doi.org/10.1007/BF01200056.

[BL08] Jacob D. Biamonte and Peter J. Love. "Realizable Hamiltonians for Universal Adiabatic Quantum Computers". In: *Phys. Rev. A* 78.1 (July 28, 2008), p. 012352. ISSN: 1050-2947, 1094-1622. DOI: 10.1103/PhysRevA.78.012352. arXiv: 0704.1287. URL: http://arxiv.org/abs/0704.1287 (visited on 11/05/2020).

[BT09] Hugue Blier and Alain Tapp. "All languages in NP have very short quantum proofs". In: *2009 Third International Conference on Quantum, Nano and Micro Technologies*. IEEE. 2009, pp. 34–37.

[CHT+10] Richard Cleve, Peter Hoyer, Ben Toner, and John Watrous. "Consequences and Limits of Nonlocal Strategies". In: *arXiv:quant-ph/0404076* (Jan. 11, 2010). arXiv: quant-ph/0404076. URL: http://arxiv.org/abs/quant-ph/0404076 (visited on 11/15/2020).

[Chv79] Vasek Chvátal. "A Greedy Heuristic for the Set-Covering Problem". In: *Math. Oper. Res.* 4.3 (1979), pp. 233–235. DOI: 10.1287/moor.4.3.233. URL: https://doi.org/10.1287/moor.4.3.233.

[CN16] Matthew Coudron and Anand Natarajan. *The Parallel-Repeated Magic Square Game is Rigid*. arXiv:1609.06306. 2016. arXiv: 1609.06306 [quant-ph].

[Con76] A. Connes. "Classification of Injective Factors Cases II_1, II_∞, III_λ, $\lambda \neq 1$". In: *Annals of Mathematics* 104.1 (1976), pp. 73–115. ISSN: 0003486X. URL: http://www.jstor.org/stable/1971057.

[Coo71] Stephen A. Cook. "The Complexity of Theorem-Proving Procedures". In: *Proceedings of the 3rd Annual ACM Symposium on Theory of Computing, May 3-5, 1971, Shaker Heights, Ohio, USA*. Ed. by Michael A. Harrison, Ranan B. Banerji, and Jeffrey D. Ullman. ACM, 1971, pp. 151–158. DOI: 10.1145/800157.805047. URL: https://doi.org/10.1145/800157.805047.

[DLT+08] Andrew C Doherty, Yeong-Cherng Liang, Ben Toner, and Stephanie Wehner. "The quantum moment problem and bounds on entangled multi-prover games". In: *2008 23rd Annual IEEE Conference on Computational Complexity*. IEEE. 2008, pp. 199–210.

[FHM18] Joseph F. Fitzsimons, Michal Hajdu šek, and Tomoyuki Morimae. "Post hoc Verification of Quantum Computation". In: *Phys. Rev. Lett.* 120 (4 Jan. 2018), p. 040501. DOI: 10.1103/PhysRevLett.120.040501. URL: https://link.aps.org/doi/10.1103/PhysRevLett.120.040501.

[GMR85] Shafi Goldwasser, Silvio Micali, and Charles Rackoff. "The Knowledge Complexity of Interactive Proof-Systems (Extended Abstract)". In: *Proceedings of the 17th Annual ACM Symposium on Theory of Computing, May 6-8, 1985, Providence, Rhode Island, USA*. Ed. by Robert Sedgewick. ACM, 1985, pp. 291–304. DOI: 10.1145/22145.22178. URL: https://doi.org/10.1145/22145.22178.

[Gri17] Alex B. Grilo. *A simple protocol for verifiable delegation of quantum computation in one round*. arXiv.171109585. 2017. arXiv: 1711.09585 [quant-ph].

[Gri20] Alex B. Grilo. "A simple protocol for verifiable delegation of quantum computation in one round". In: *arXiv:1711.09585 [quant-ph]* (June 5, 2020). arXiv: 1711.09585. URL: http://arxiv.org/abs/1711.09585 (visited on 11/05/2020).

[Gur04] Leonid Gurvits. "Classical complexity and quantum entanglement". In: *Journal of Computer and System Sciences* 69.3 (2004). Special Issue on STOC 2003, pp. 448–484. ISSN: 0022-0000. DOI: https://doi.org/10.1016/j.jcss.2004.06.003. URL: http://www.sciencedirect.com/science/article/pii/S0022000004000893.

[GW95] Michel X. Goemans and David P. Williamson. "Improved Approximation Algorithms for Maximum Cut and Satisfiability Problems Using Semidefinite Programming". In: *J. ACM* 42.6 (1995), pp. 1115–1145. DOI: 10.1145/227683.227684. URL: https://doi.org/10.1145/227683.227684.

[Hås01] Johan Håstad. "Some optimal inapproximability results". In: *J. ACM* 48.4 (2001), pp. 798–859. DOI: 10.1145/502090.502098. URL: https://doi.org/10.1145/502090.502098.

[HM13] Aram W. Harrow and Ashley Montanaro. "Testing Product States, Quantum Merlin-Arthur Games and Tensor Optimization". In: *J. ACM* 60.1 (Feb. 2013). ISSN: 0004-5411. DOI: 10.1145/2432622.2432625. URL: https://doi.org/10.1145/2432622.2432625.

[Ji16] Zhengfeng Ji. "Compression of Quantum Multi-Prover Interactive Proofs". In: *arXiv:1610.03133 [quant-ph]* (Oct. 10, 2016). arXiv: 1610.03133. URL: http://arxiv.org/abs/1610.03133 (visited on 11/20/2020).

[JJU+09] Rahul Jain, Zhengfeng Ji, Sarvagya Upadhyay, and John Watrous. "QIP = PSPACE". In: *arXiv:0907.4737 [quant-ph]* (Aug. 2, 2009). arXiv: 0907.4737. URL: http://arxiv.org/abs/0907.4737 (visited on 11/14/2020).

[JNV+20] Zhengfeng Ji, Anand Natarajan, Thomas Vidick, John Wright, and Henry Yuen. "MIP*=RE". In: *arXiv:2001.04383 [quant-ph]* (Sept. 29, 2020). arXiv: 2001.04383. URL: http://arxiv.org/abs/2001.04383 (visited on 10/14/2020).

[KKG20] Anna R. Karlin, Nathan Klein, and Shayan Oveis Gharan. "A (Slightly) Improved Approximation Algorithm for Metric TSP". In: *CoRR* abs/2007.01409 (2020). arXiv: 2007.01409. URL: https://arxiv.org/abs/2007.01409.

[KR03] Julia Kempe and Oded Regev. "3-local Hamiltonian is QMA-complete". English (US). In: *Quantum Information and Computation* 3.3 (May 2003), pp. 258–264. ISSN: 1533-7146.

[LFK+92] Carsten Lund, Lance Fortnow, Howard J. Karloff, and Noam Nisan. "Algebraic Methods for Interactive Proof Systems". In: *J. ACM* 39.4 (1992), pp. 859–868. DOI: 10.1145/146585.146605. URL: https://doi.org/10.1145/146585.146605.

[MN36] F. J. Murray and J. v. Neumann. "On Rings of Operators". In: *Annals of Mathematics* 37.1 (1936), pp. 116–229. ISSN: 0003486X. URL: http://www.jstor.org/stable/1968693.

[MYS12] M McKague, T H Yang, and V Scarani. "Robust self-testing of the singlet". In: *Journal of Physics A: Mathematical and Theoretical* 45.45 (Oct. 2012), p. 455304. ISSN: 1751-8121. DOI: 10.1088/1751-8113/45/45/455304. URL: http://dx.doi.org/10.1088/1751-8113/45/45/455304.

[NPA08] Miguel Navascues, Stefano Pironio, and Antonio Acin. "A convergent hierarchy of semidefinite programs characterizing the set of quantum correlations". In: *New J. Phys.* 10.7 (July 8, 2008), p. 073013. ISSN: 1367-2630. DOI: 10.1088/1367-2630/10/7/073013. arXiv: 0803.4290. URL: http://arxiv.org/abs/0803.4290 (visited on 10/14/2020).

[NV17] Anand Natarajan and Thomas Vidick. "A quantum linearity test for robustly verifying entanglement". In: *Proceedings of the 49th Annual ACM SIGACT Symposium on Theory of Computing - STOC 2017* (2017). DOI: 10.1145/3055399.3055468. URL: http://dx.doi.org/10.1145/3055399.3055468.

[Sha92] Adi Shamir. "IP = PSPACE". In: *J. ACM* 39.4 (1992), pp. 869–877. DOI: 10.1145/146585.146609. URL: https://doi.org/10.1145/146585.146609.

[Vaz01] Vijay V. Vazirani. *Approximation algorithms*. Springer, 2001. ISBN: 978-3-540-65367-7. URL: http://www.springer.com/computer/theoretical+computer+science/book/978-3-540-65367-7.

[Vid18] Thomas Vidick. *UCSD Summer School Notes: Quantum multiplayer games, testing and rigidity*. http://users.cms.caltech.edu/ vidick/notes/ucsd/ucsd_games.pdf. 2018.

[Wat99] John Watrous. "PSPACE has 2-round quantum interactive proof systems". In: *arXiv:cs/9901015* (Jan. 27, 1999). arXiv: cs/9901015. URL: http://arxiv.org/abs/cs/9901015 (visited on 11/15/2020).