Frontiers of Quantum complexity *Instructors: Anand Natarajan, Anurag Anshu*

Lecture 12: Quantum PCP and interactive proofs

Oct 14, 2025 *Scribe: Anastasiia Struss*

1 Gap amplification

Quantum PCP conjecture is one of the central open problems in Hamiltonian complexity. We recall the statement of the conjecture.

Conjecture 1.1 (Quantum PCP by gap amplification)**.** Let $\mathcal{H} = \frac{1}{m}\sum_{i=1}^{m} H_i$ be a k-local Hamiltonian with normalized local terms H_i. Then there exists a constant $\varepsilon = b - a = \Omega(1)$ such that this is QMA-hard to decide whether $\lambda_{\min} \leq a$ or $\lambda_{\min} \geq b$.

The hardness of approximation version of the conjecture says that computing an approximation to the minimum energy of Hamiltonian $\mathcal{H}$ up to a precision that is better than ε is QMA-hard.

Recall that one formulation of the classical PCP theorem is that there exists a constant ε such that deciding whether a given Boolean formula is satisfied or at most $(1-\varepsilon)$ fraction of its clauses are satisfied is NP-hard. Since 3-SAT can be viewed as a 3-local Hamiltonian problem [AAV13], we know that the local Hamiltonian problem with a constant promise gap is at least NP-hard.

The Feynman-Kitaev theorem says that the local Hamiltonian problem is QMA-hard where $\varepsilon = 1/\operatorname{poly}(n)$. One approach to prove Conjecture 1.1 is to take some transformation that blows up the promise gap ε until one gets up to $\Omega(1)$. There has been some progress, but so far, no one has made this work. However, gap amplification technique was used by Dinur [Din07] to prove the classical PCP theorem. We will return to this approach in the following lectures.

2 Proof verification

The next approach also does not work yet, but has had some interesting consequences which are more aligned with how classical PCP theorem was originally proved in [AS98], [ALM$^+$98]. The original proof of the classical PCP theorem was inspired by the result in complexity theory stating that MIP = NEXP [BFL91]. Researchers were looking for a similar characterization of NP. In the MIP model, we have $k = \operatorname{poly}(n)$ computationally unbounded provers, that are not allowed to communicate, and a probabilistic polynomial time verifier. The verifier exchanges polynomial length messages with the provers using polynomially many rounds. It turns out that there is a way to restrict this model to characterize NP.

Definition 2.1. A 2-Prover-1-Round-Game protocol (V, P_1, P_2) is a protocol between a probabilistic polynomial time verifier V and two provers. In the beginning of the protocol all parties hold the same input x. The verifier samples $\mathcal{O}(\log n)$ random bits, sends a message of length $\mathcal{O}(\log n)$ to each prover, receives a response of length $\mathcal{O}(1)$ from each of them, and then decides to accept or reject. We say that (V, P_1, P_2) computes a language L if there exists a constant ε such that

1. For every $x \in L$ we have $\Pr\left[(V, P_1, P_2)(x) = 1\right] = 1$.

2. For every $x \notin L$ and for all P_1^*, P_2^* we have $\Pr[(V, P_1^*, P_2^*)(x) = 1] < 1 - \varepsilon$.

Having this definition we can state the classical PCP theorem.

Theorem 2.2. *The class of problems that have a 2-Prover-1-Round-Game protocol denoted by* $\mathsf{MIP}[\mathcal{O}(\log(n)), \mathcal{O}(1)]$ *is precisely* NP.

Remember that PCP stands for probabilistically checkable proofs. The name comes from the fact that NP is in $\mathsf{MIP}[\mathcal{O}(\log(n)), \mathcal{O}(1), c - s = \Omega(1)]$ if and only if for all $L \in \mathsf{NP}$ there exists a PCP verifier.

Definition 2.3. A PCP verifier V_{PCP} is a probabilistic polynomial time TM that gets an input x as well as oracle access to a proof $\pi \in \Gamma^*$, where Γ is a finite alphabet. The verifier uses $\mathcal{O}(\log n)$ random bits, and based on them and on x chooses two locations in the proof π. Then, verifier decides to accept or reject. We say that V_{PCP} decides language L if

1. For every $x \in L$ there exists a proof π such that $\Pr[V_{\mathsf{PCP}}(x, \pi) = 1] = 1$.

2. For every $x \notin L$ and for all π we have $\Pr[V_{\mathsf{PCP}}(x, \pi) = 1] < 1 - \varepsilon$.

Theorem 2.4. *There exists a PCP verifier for SAT if and only if SAT* $\in \mathsf{MIP}[\mathcal{O}(\log(n)), \mathcal{O}(1), c - s = \Omega(1)]$.

Proof. Let (V, P_1, P_2) be a 2-Prover-1-Round-Game protocol for SAT. Think of the questions of V to P_1 and P_2 as strings $q_1, q_2 \in \{0,1\}^{C \log n}$, respectively. Without loss of generality, P_1 and P_2 are deterministic algorithms and hence can be represented by functions $f_{P_1}, f_{P_2} : \{0,1\}^{C \log n} \to \Gamma$, where Γ is a finite alphabet. Let π be the concatenation of truth tables of f_{P_1} and f_{P_2} of length $2n^C$. The PCP verifier V_{PCP}

1. Uses V to generate questions q_1 and q_2.

2. Looks at the two locations $a, b \in \Gamma$ of π corresponding to q_1 and q_2.

3. Accepts if and only if V accepts a and b when asked q_1 and q_2.

Now let V_{PCP} be a PCP verifier for SAT. Let Φ be a 3CNF formula. The verifier V on input Φ

1. Tosses the same random bits V_{PCP} would toss to obtain indices i, j, k corresponding to a clause of Φ.

2. Picks a uniform $y \in \{i, j, k\}$.

3. Sends (i, j, k) to P_1 and y to P_2.

4. Receives $(a_i, a_j, a_k) \in \Gamma^3$ from P_1 and $b \in \Gamma$ from P_2.

5. Accepts if and only if (a_i, a_j, a_k) satisfies the clause, $b = a_y$ and V_{PCP} accepts (a_i, a_j, a_k) as content of the part of the proof π corresponding to the random bits tossed in the first step.

Analysis of the above algorithms should be straightforward. □

The above definition enables a new characterization of NP. A language is in NP if and only if there is a PCP verifier that decides the language.

It turns out that Conjecture 1.1 also has an equivalent statement in terms of proof verification [AAV13].

Conjecture 2.5 (Quantum PCP by proof verification)**.** For any language in QMA there exists a polynomial time verifier, which acts on the classical input string x and a witness $|\xi\rangle$, a quantum state of poly(n) qubits, such that the verifier accesses only a constant number of qubits from the witness and decides to accept or reject with constant error probability.

3 Exponential-size PCP system

Returning to the classical PCP theorem, notice that it basically says that an NP-witness, that a verifier needs to check in its entirety in order to accept or reject, can be encoded in such a way that this encoding is very robust. Any error in the proof gets magnified to errors in many locations in the encoding.

Now assume that the verifier is allowed to use $\mathcal{O}(n)$ random bits. This amount of randomness now allows exponentially long proofs π. The corresponding relaxed version of the protocol is often referred to as "exponential-size PCP system", which is crucial in the final construction for the classical PCP theorem. Note that an analog relaxed version of quantum PCP theorem is open as well.

Consider an NP-hard problem QuadEq.

Definition 3.1. Given a matrix $A \in \mathbb{F}_2^{m\times n^2}$ and a vector $b \in \mathbb{F}_2^m$, problem QuadEq is to find an assignment $x \in \mathbb{F}_2^n$ such that

$$A(x \otimes x) = b.$$

In the exponential-size PCP system, every assignment $x \in \mathbb{F}_2^n$ corresponds to some proof $\pi = \pi(x)$ of exponential size. We would like that satisfying assignments get mapped to proofs that are far in Hamming distance from proofs that non-satisfying assignments get mapped to. Hadamard code yields such a mapping.

Definition 3.2. Hadamard code $H \subseteq \mathbb{F}_2^{2^n}$ is the image of the following linear map

$$\begin{aligned} \mathrm{Enc} : \mathbb{F}_2^n &\to \mathbb{F}_2^{2^n} \\ x &\mapsto y = (y_a)_{a\in\mathbb{F}_2^n}, \end{aligned}$$

where $y_a = \langle a, x\rangle$.

Notice that for all $y \in H$, $y_0 = 0$ and $y_{e_i} = x_i$, where e_i is the ith unit vector. One important property of the code is that it has an enormous relative Hamming distance $1/2$. Moreover, it is locally testable [Ste16]. In fact, querying three positions of a string $r \in \mathbb{F}_2^{2^n}$ it is possible to check whether it is close to H.

Theorem 3.3. *Let $r \in \mathbb{F}_2^{2^n}$ and $\varepsilon < 1/2$. If*

$$\Pr_{a,b\in\mathbb{F}_2^n}[r_{a+b} = r_a + r_b] \geq 1 - \varepsilon,$$

then the relative Hamming distance is smaller than or equal to ε: $\mathrm{dist}(r, H) \leq \varepsilon$.

Furthermore, H turns out to be locally decodable. More precisely, there exists a probabilistic algorithm $\mathcal{A}$ that given $a \in \mathbb{F}_2^n$ and query access to $r \in \mathbb{F}_2^{2^n}$ outputs $\mathrm{Enc}(x)_a$ with probability $1-2\varepsilon$ by making only 2 queries to r.

Theorem 3.4. *Let $r \in \mathbb{F}_2^{2^n}$ and $x \in \mathbb{F}_2^n$ such that $\mathrm{dist}(r, \mathrm{Enc}(x)) \leq \varepsilon$, then for every $a \in \mathbb{F}_2^n$,*

$$\Pr_{b \in \mathbb{F}_2^n} [\mathrm{Enc}(x)_a = r_{a+b} + r_b] \geq 1 - 2\varepsilon.$$

Now we will explain the construction of the PCP from the Hadamard code. Given an instance $A \in \mathbb{F}_2^{m \times n^2}, b \in \mathbb{F}_2^m$ of the QuadEq problem, the honest prover computes a satisfying assignment $x \in \mathbb{F}_2^n$, computes $y = \mathrm{Enc}(x) \in \mathbb{F}_2^{2^n}$ and $z = \mathrm{Enc}(x \otimes x) \in \mathbb{F}_2^{2^{n^2}}$. The corresponding proof $\pi = \pi(x)$ is the concatenation of y and z. The verifier will

1. Check that y and z are encodings using a local tester for H.
2. Pick $\beta \in \mathbb{F}_2^m$ corresponding to a linear combination of equations (if there is one equation that is not satisfied, then any linear combination that happens to touch that equation should not be satisfied as well).
3. Compute $\gamma = A^T\beta$ and $\eta = \beta^T b$. Check that $z_\gamma = \eta$ using a local decoder.
4. Pick random $a, b \in \mathbb{F}_2^n$ and check that $z_{a \otimes b} = y_a y_b$ using a local decoder.

This polynomial time verifier uses a constant number of queries, $\mathcal{O}(n)$ random bits and decides to accept or reject with constant error probability.

The actual proof of the original PCP theorem uses the same paradigm, but with a different code that is based on low degree polynomials rather than linear functions.

The natural question to ask is whether we can use a similar paradigm for the quantum PCP theorem. However, there are many kinds of problems that occur when we turn to the quantum setting.

1. Local indistinguishability. Consider states $|\psi\rangle$ and $|\phi\rangle$ and their encodings $\mathrm{Enc}(|\psi\rangle)$ and $\mathrm{Enc}(|\phi\rangle)$ corresponding to a quantum code with non-trivial distance. Then looking at these encodings on regions that are smaller than the code distance, one cannot distinguish them. This means that the encoding of the assignment tells absolutely nothing unless one looks at regions bigger than the distance.
2. Notice that step 4 in the verifier's algorithm exploits the property of the code that it is multiplicatively homomorphic. However, finding quantum codes with a similar property seems to be hard.

4 Interactive proofs and quantum PCP

Remember that the classical PCP verifier for NP corresponds to a 2-Prover-1-Round-Game protocols. It turns out that PCPs and MIPs are not so clearly related quantumly. In the quantum world, there are multiple notions of MIP. In any of these notions, we do not know how to convert them back into a quantum PCP in any reasonable way. This is because one can not write down a quantum strategy for quantum provers as a table that one can check locally. It might involve

measuring quantum states in some interesting non-local way and that will not translate back into local checks. On the other hand, one believes that quantum PCPs imply quantum MIPs.

The complexity class MIP^* is the class of languages that can be decided by a protocol in which a polynomial-time classical verifier interacts with quantum provers who may share a finite-dimensional entangled state. One can show that allowing the verifier to be quantum does not make the class bigger, $\mathsf{QMIP}^* = \mathsf{MIP}^*$ [RUV13]. One natural question to ask is whether $\mathsf{MIP} \subseteq \mathsf{MIP}^*$. This question is quite non-trivial because if a protocol is sound in the MIP model, that protocol may not be sound anymore in the MIP^* model, since there might be a way to cheat using entanglements. In particular, the clause variable transformation does not work in the presence of shared entanglement between the provers, as is demonstrated by the so-called Magic Square game example [VW$^+$16]. Nevertheless, the containment was shown to be true [IV12].

The next interesting result is that $\mathsf{QMA} \subseteq \mathsf{QMIP}^*[\mathcal{O}(\log(n)), \mathcal{O}(1), c-s = \Omega(1/\operatorname{poly}(n))]$ [FV15]. This statement is meant to be an analog of the statement that NP is contained in $\mathsf{MIP}[\mathcal{O}(\log(n)), \mathcal{O}(1), c-s = \Omega(1)]$. The construction uses five provers and a quantum version of the clause variable construction. Recall that one essential feature of the classical clause variable construction was that provers both compute the same assignment to the given formula. In the quantum setting, in order for provers to answer consistently according to an assignment to a given QMA problem, it seems to be necessary to clone states, which is impossible. So the question remained how to make sure that provers are consistent with each other. The problem was solved using an error-correcting code with rate $1/5$ and splitting the codeword among five provers. It was shown that this behaves quantumly like copying behaves classically.

Next, it was claimed that $\mathsf{QMA} \subseteq \mathsf{MIP}^*[\operatorname{polylog}(n), \operatorname{polylog}(n), c-s = \Omega(1)]$ [NV18], which was called quantum entangled games PCP. The idea was to use an error-correcting code with rate $1/7$ and hence seven provers and techniques from the classical PCP theorem. However, there is a bug in the construction [NN24].

Interestingly, MIP^* turned out to be extremely powerful in the sense that $\mathsf{MIP}^* = \mathsf{RE}$ [JNV$^+$21]. The refinement of this statement is $\mathsf{RE} \subseteq \mathsf{MIP}^*[\mathcal{O}(1), \operatorname{polylog}(n), c-s = \Omega(1)]$ [NZ23], which can be thought of as a kind of quantum PCP for RE.

We understand the class MIP^* now. It looks very different from quantum PCP and all the straightforward equivalences fail. And yet there is still an open problem to recover $\mathsf{QMA} \subseteq \mathsf{MIP}^*[\operatorname{polylog}(n), \operatorname{polylog}(n), c-s = \Omega(1)]$ using an efficient prover that does not do much apart from having copies of the QMA ground state. Recovering it seems to involve designing a good gap amplification for a certain kind for Hamiltonians which was already the type of problem we want to solve for regular Hamiltonian PCP. It is possible that trying to fix this construction might yield some insight into regular Hamiltonian PCP.

References

[AAV13] Dorit Aharonov, Itai Arad, and Thomas Vidick. Guest column: the quantum pcp conjecture. *Acm sigact news*, 44(2):47–79, 2013.

[ALM$^+$98] Sanjeev Arora, Carsten Lund, Rajeev Motwani, Madhu Sudan, and Mario Szegedy. Proof verification and the hardness of approximation problems. *Journal of the ACM (JACM)*, 45(3):501–555, 1998.

[AS98] Sanjeev Arora and Shmuel Safra. Probabilistic checking of proofs: A new characterization of np. *Journal of the ACM (JACM)*, 45(1):70–122, 1998.

[BFL91] László Babai, Lance Fortnow, and Carsten Lund. Non-deterministic exponential time has two-prover interactive protocols. *Computational complexity*, 1(1):3–40, 1991.

[Din07] Irit Dinur. The pcp theorem by gap amplification. *Journal of the ACM (JACM)*, 54(3):12–es, 2007.

[FV15] Joseph Fitzsimons and Thomas Vidick. A multiprover interactive proof system for the local hamiltonian problem. In *Proceedings of the 2015 Conference on Innovations in Theoretical Computer Science*, pages 103–112, 2015.

[IV12] Tsuyoshi Ito and Thomas Vidick. A multi-prover interactive proof for nexp sound against entangled provers. In *2012 IEEE 53rd Annual Symposium on Foundations of Computer Science*, pages 243–252. IEEE, 2012.

[JNV+21] Zhengfeng Ji, Anand Natarajan, Thomas Vidick, John Wright, and Henry Yuen. Mip*= re. *Communications of the ACM*, 64(11):131–138, 2021.

[NN24] Anand Natarajan and Chinmay Nirkhe. The status of the quantum pcp conjecture (games version). *arXiv preprint arXiv:2403.13084*, 2024.

[NV18] Anand Natarajan and Thomas Vidick. Low-degree testing for quantum states, and a quantum entangled games pcp for qma. In *2018 IEEE 59th Annual Symposium on Foundations of Computer Science (FOCS)*, pages 731–742. IEEE, 2018.

[NZ23] Anand Natarajan and Tina Zhang. Quantum free games. In *Proceedings of the 55th Annual ACM Symposium on Theory of Computing*, pages 1603–1616, 2023.

[RUV13] Ben W Reichardt, Falk Unger, and Umesh Vazirani. Classical command of quantum systems. *Nature*, 496(7446):456–460, 2013.

[Ste16] David Steurer. Pcp theorem: exponential-size proofs. `https://complexity16.dsteurer.org/lectures/exponentialpcp.pdf`, 2016. Lecture notes, CS 6810 (Spring 2016).

[VW+16] Thomas Vidick, John Watrous, et al. Quantum proofs. *Foundations and Trends® in Theoretical Computer Science*, 11(1-2):1–215, 2016.