QALCO: Quantum Algorithms and Complexity **2025 - 2026**

Lecture 6: QIP, MIP*

Lecturer: Alex Bredariol Grilo (alex.bredariol-grilo@lip6.fr)

1 Interactive proof systems

We can see NP as the interaction between an unbounded Prover, who provides an untrusted proof/witness to a polynomially bounded Verifier. The requirements that we have seen are that for positive instances, there is a witness that the Prover can provide that makes the Verifier accept, while for negative instances, all possible witnesses make the verifier reject.

Introduced independently by Goldwasser, Micali, and Rackoff, and Babai in the 1980s, Interactive Proof Systems (IP) generalize the concept of NP by allowing interaction between the Prover and the Verifier, as depicted in Figure 1.

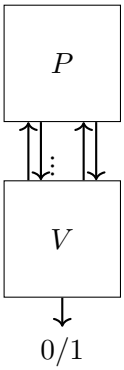

Figure 1: Interaction between the Prover and the Verifier

Definition 1 (Interactive Proof System (IP))**.** *An Interactive Proof System (IP) for a language L is a protocol between two parties: a verifier V and a prover P. The verifier V is a probabilistic polynomial-time machine, while the prover P is computationally unbounded. The protocol satisfies the following properties:*

- ***Completeness**: For any input $x \in L$, there exists a prover P such that the interaction between V and P on input x results in V accepting with probability at least $\frac{2}{3}$:*

$$\Pr[(V \leftrightarrow P)(x) = 1] \geq \frac{2}{3}$$

- ***Soundness**: For any input $x \notin L$, and for any prover strategy P^* (possibly dishonest), the interaction between V and P^* on input x results in V accepting with probability at most $\frac{1}{3}$:*

$$\Pr[(V \leftrightarrow P^*)(x) = 1] \leq \frac{1}{3}$$

Definition 2. *The class IP is the set of all languages L that have an interactive proof system as defined above.*

Let us see an example of a problem that is not known to be in NP, but it is in IP.

Theorem 1. *The problem of Graph Non-Isomorphism is in* IP*. Specifically, there exists an interactive proof system for the language:*

$$GNI = \{(G_0, G_1) \mid G_0 \text{ and } G_1 \text{ are not isomorphic graphs}\}.$$

Proof. We describe the interactive proof system between a polynomial-time verifier V and an unbounded prover P.

1. **Input**: Two graphs G_0 and G_1 with n vertices each.
2. **Verifier's Challenge**: The Verifier picks a random permutation π and a random bit b and sends $\pi(G_b)$ to the Prover.
3. **Prover's Response**: The prover P responds with the guess b'.
4. **Verifier's Check**: The verifier V accepts iff $b = b'$

If G_0 and G_1 are not isomorphic, the prover can always respond correctly to the verifier's challenge.

If G_0 and G_1 are isomorphic, the distributions $\{\pi(G_0)\}_\pi$ and $\{\pi(G_1)\}_\pi$ are identical, so the prover will always fail with probability at least $\frac{1}{2}$. □

A landmark result in this field is the proof that IP equals PSPACE.

Theorem 2 ([8, 6])**.** IP = PSPACE*.*

Proof. IP ⊆ PSPACE is straightforward: in PSPACE we can enumerate all strategies from the provers and compute the maximal acceptance probability.

PSPACE ⊆ IP is much harder and it goes via algebraization of PSPACE-complete problems and the sum-check protocol. □

2 Introduction

Quantum Interactive Proofs (QIP) generalize classical interactive proof systems by allowing quantum messages between a prover and a verifier.

Here, it is simpler to design the interaction between the Prover and the Verifier as a quantum circuit. We have three quantum registers, the Prover's private register, which can be of unbounded size, and the message and Verifier's private registers, which consist of polynomially many qubits. The Prover and the Verifier then alternate applying their operations: the Prover applies their unitaries on their private register and the message register, and the Verifier applies their unitary on their private register and the message register. After the interaction, the Verifier measures the output qubit and decides to accept or reject. We depict a protocol with 3 messages in Figure 2.

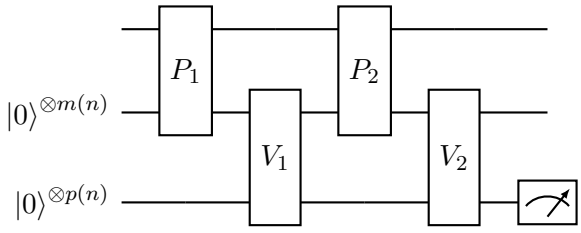

Figure 2: Example of a QIP protocol with 3 rounds of communication

Notice that the Verifier is the party that applies their operation at the end of the protocol. So if the protocol has an even number of messages, the Verifier starts, and if the protocol has an odd number of messages, the Prover starts.

Definition 3. *A problem L is in* QIP *if there exists a polynomial-time quantum verifier V that interacts with the quantum prover such that:*

- *For $x \in L$, there exists a quantum prover strategy that makes V accept with probability $\geq \frac{2}{3}$.*
- *For $x \notin L$, every prover strategy makes V accept with probability $\leq \frac{1}{3}$.*

One of the most striking properties of QIP is its ability to be parallelized into a 3-message protocol. This is in contrast to classical IP, where parallelization is not generally possible without increasing the number of messages.

Theorem 3 ([5])**.** QIP = QIP(3).

This characterization allows us to show that actually quantum interactive proof systems have the same computational power as classical ones (without considering the number of rounds). In particular, it can be shown that

Theorem 4 ([3])**.** QIP= PSPACE.

Proof. The proof goes via showing that QIP has actually a highly structured QIP(3) protocol:

1. The prover sends a quantum state.
2. The verifier answers back with a classical random bit.
3. The prover answers back with a quantum state.

This is called a QMAM protocol. The idea of this reduction is rather simple. We start from a QIP(3) protocol (such as Figure 2) that is complete and sound. We can design a QMAM as follows:

1. The prover sends the Verifier's private register in the QIP(3) protocol after the operation V_1 (notice that this state depends on the Prover's first message)
2. The verifier sends a random bit b to the prover:
   (a) If $b = 0$, the prover sends the message register of the QIP(3) protocol *before* the operation P_2. The verifier then applies $V_1^\dagger$ and accepts iff their private register is $|0\rangle^{\otimes p(n)}$.
   (b) If $b = 1$, the prover is supposed to send the message register of the QIP(3) protocol *after* the operation P_2. The verifier then applies V_2 and accepts iff the output qubit is $|1\rangle$.

If the original QIP(3) protocol was complete, it is not hard to see that the new QMAM protocol is too. One can also show that the same holds for soundness.

Then, it was shown in [3] that the maximal acceptance probability of a QMAM protocol can be solved by a *semi-definite program*, which is a generalization of linear programs, that can be solved in polynomial space. □

2.1 Multiple provers

2.2 Classical

In classical complexity theory, the notion of interactive proof systems has been considered in the model where we have multiple provers. In this setting, the provers share an *strategy*, but after the protocol begins, the provers are not allowed to communicate. We depict an MIP protocol in Figure 3.

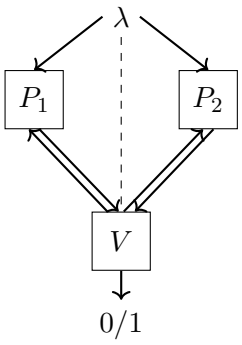

Figure 3: Example of a one round MIP protocol with two provers, who share a classical string λ.

Definition 4 (Multi-Prover Interactive Proof System (MIP))**.** *A language L belongs to the class* MIP *if there exists such a multi-prover interactive proof systems satisfying:*

- ***Completeness:*** *if $x \in L$, then there exist prover strategies such that*

$$\Pr[V \text{ accepts } x] = 1,$$

- ***Soundness:*** *if $x \notin L$, then for any prover strategies,*

$$\Pr[V \text{ accepts } x] \leq \frac{1}{3}.$$

One interesting property of having multiple provers is that you can use one of the provers to cross-check the answers of the second prover. This enables, for example, to reduce the communication for proof verification.

Theorem 5. NP *has an* MIP *protocol with $O(\log(n))$ bits of communication with completeness* 1 *and soundness* $1 - \frac{1}{poly(n)}$.

Proof. Let φ be a 3CNF formula with variables $x_1, \ldots, x_n$ and clauses $C_1, \ldots, C_m$, where each clause contains exactly three literals.

The verifier interacts with two non-communicating provers P_1 and P_2.

1. The verifier samples uniformly at random a clause C_j and then selects uniformly at random one variable x_i appearing in C_j.
2. The verifier sends j to prover P_1 and i to prover P_2.
3. Prover P_1 responds with an assignment to the three variables appearing in clause C_j.
4. Prover P_2 responds with an assignment to the variable x_i.

5. The verifier accepts if and only if:

   - the assignment provided by P_1 satisfies clause C_j, and
   - the value assigned to x_i by P_1 agrees with the value provided by P_2.

As usual, proving the completeness of the protocol is straightforward. One can show that the soundness of the protocol is $1 - \frac{1}{poly(n)}$. □

A surprising characterization of MIP is that it is actually equal to NEXP, the exponential version of NP.

Theorem 6 ([1])**.** MIP= NEXP*.*

The proof of this theorem created many techniques that have several applications in complexity theory (and even in general computer science) such as the low-degree testing, locally testable/decodable codes, oracularization.

Moreover, since the proposed MIP protocol for NEXP has only two provers and one round of communication, it shows that the simple structure defined in Figure 3 has the same computational power as any protocol in MIP.

2.3 Quantum protocols

In the quantum setting, we consider MIP protocols where the provers share a quantum state, but the verifier is classical (thus the communication between the verifier and the provers is also classical). There is an alternative extension of MIP to the quantum setting called QMIP where the verifier and the communication could be quantum. We won't get into details here, but we know that QMIP = MIP* so we will stick to the simpler notion. We depict an MIP* protocol in Figure 4.

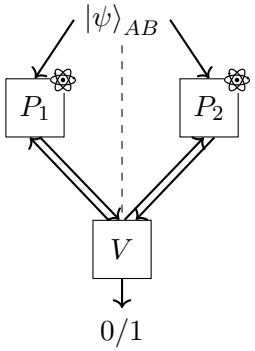

Figure 4: Example of a one round MIP* protocol with two provers, who share the quantum state $|\psi\rangle$.

Definition 5 (Multi-Prover Interactive Proof System with Entanglement (MIP*))**.** *A* Multi-Prover Interactive Proof system with entanglement *(MIP*) is a multi-prover interactive protocol between a probabilistic polynomial-time verifier V and a set of 2 provers P_1 and P_2 who may share an arbitrary prior quantum entangled state but are not allowed to communicate during the protocol.*

The verifier exchanges classical messages with the provers for a polynomial number of rounds and decides whether to accept or reject an input x.

A language L belongs to the class MIP* *if there exists such a protocol satisfying:*

- ***Completeness:*** *if $x \in L$, then there exist entangled prover strategies such that*

$$\Pr[V \text{ accepts } x] = 1,$$

- ***Soundness:*** *if $x \notin L$, then for any entangled prover strategies,*

$$\Pr[V \text{ accepts } x] \leq \frac{1}{3}.$$

First of all, it is not even trivial that $\mathsf{MIP} \overset{?}{\subseteq} \mathsf{MIP}^*$: since the provers have more power, it could be the case the malicious provers have too much power and proving soundness would be impossible. It was first shown by [2] that this is not actually the case, and indeed $\mathsf{MIP} \subseteq \mathsf{MIP}^*$. The idea of this proof is to show that the MIP protocol for NEXP is sound even if the provers share quantum strategies. This proof was first shown with three provers, where the third prover was introduced to break entanglement. Later results showed that this proof actually works with only two provers.

For a long time, it was an open question on what is the exact computational power of MIP^*. We now know it, but before seeing some details, we will make a detour into self-testing.

2.4 Detour on non-local games and self-testing

Definition 6 (Non-Local Game)**.** *A game G is a one-round interaction between a verifier and 2 non-communicating players Alice and Bob, who may share prior resources.*

The verifier samples a tuple of questions (x, y) from a known probability distribution μ and sends x to Alice and y to Bob, who answer with a and b, respectively.

The players win the game if the predicate

$$V(x, y, a, b) = 1.$$

evaluates to 1.

We denote the classical value of the game $\omega(G)$ as the maximum acceptance probability among all strategies where Alice and Bob share classical resources. The quantum value of the game $\omega^*(G)$ is the maximum acceptance probability among all strategies where Alice and Bob share a quantum state. A game G is said to be non-local if $\omega^*(G) > \omega(G)$.

Definition 7 (CHSH Game)**.** *The CHSH game is a two-player non-local game defined as follows:*

- *The verifier samples questions $x, y \in \{0, 1\}$ uniformly at random and sends x to Alice and y to Bob.*
- *Alice outputs an answer $a \in \{0, 1\}$ and Bob outputs an answer $b \in \{0, 1\}$.*
- *The players win if*

$$a \oplus b = x \wedge y,$$

where $\oplus$ denotes addition modulo 2.

Lemma 1. *The maximum winning probability is $3/4$ for classical strategies and*

$$\cos^2\left(\frac{\pi}{8}\right) \approx 0.854$$

for quantum (entangled) strategies.

The non-locality of a game allows us to certify that the players' correlations cannot be explained classically. This was used to "prove" that we need to extend classical Physics to explain experiments. But actually, we can go one step further: some games not only allow us to show that parties are not classical, but they also allow us to characterize their strategies. This is called self-testing. For example, we have the following:

Lemma 2. *Let* $(|EPR\rangle, \{A_x\}_x, \{B_y\})$ *be the textbook strategy that wins the CHSH game with probability* $\cos^2\left(\frac{\pi}{8}\right)$*. Suppose that Alice and Bob share a strategy* $(|\psi\rangle_{AB}, \{M_x\}_x, \{N_y\})$*, where* $|\psi\rangle \in \mathcal{H}_A \otimes \mathcal{H}_B$ *for the CHSH that has success probability at least* $\cos^2\left(\frac{\pi}{8}\right) - \varepsilon$*. Then there exist a state* $|garb\rangle_{A'B'} \in \mathcal{H}_{A'} \otimes \mathcal{H}_{B'}$ *and isometries* $V_D : \mathcal{H}_D \to \mathbb{C}^2 \otimes \mathcal{H}_{D'}$*, for* $D \in \{A, B\}$*, such that*

$$\left\| (V_A \otimes V_B) |\psi\rangle_{AB} - |EPR\rangle_{A'B'} |garb\rangle_{A'B'} \right\|^2 = O(\sqrt{\varepsilon}). \tag{1}$$

Moreover

$$\mathbb{E}_{x\in\{0,1\}} \left\| \left(N_x - V_A^\dagger (A_x \otimes I) V_A\right) \otimes I_B |\psi\rangle \right\|^2 = O(\sqrt{\varepsilon}), \tag{2}$$

and similarly to B*.*

This notion of self-testing has been extended to multiple games with more desired properties:

- **Magic-square game:** A non-local game that achieves quantum value 1 and classical value $\frac{7}{8}$, and allows us to self-test two EPR pairs.
- **Parallel repetition of CHSH/Magic square:** Allows us to self-test n EPR pairs. However, the error in the analog of equations Equations (1) and (2) in this case scales as $O(poly(n, \varepsilon))$. Thus, ε must be $\frac{1}{poly(n)}$ for the bound to be non-trivial.
- **Pauli Braiding test:** A new non-local game that allows the self-testing of n EPR pairs but with robust self-testing, i.e., the error in the analog of equations Equations (1) and (2) in this case scales as $O(poly(\varepsilon))$. However, this game has polynomial communication between the parties.
- **Quantum low-degree testing:** A non-local game that allows robust self-testing of n EPR pairs with $polylog(n)$ communication between the parties.

3 Complexity of MIP*

Using the quantum low-degree testing, [7] showed that actually NEEXP $\subseteq$ MIP*, where NEEXP is the doubly exponential version of NP. Notice that this already shows that MIP $\subsetneq$ MIP*, since we know that NEXP $\subsetneq$ NEEXP.

This was proven using a beautiful idea called introspection. The rough sketch of the idea is the following:

1. With probability $\frac{1}{2}$, run the quantum low-degree testing with $poly(n)$ to self-test $2^{O(n)}$ EPR pairs with $poly(n)$ communication.
2. With probability $\frac{1}{2}$, ask questions that are locally indistinguishable from the quantum low-degree testing, but actually samples from the correct distribution of the scaled up game from NEXP $\subseteq$ MIP.
3. If the provers could answer with exponentially many bits, we would be done. But then we need to use some fancy PCP machinery to reduce the output to constant.

Unfortunately, the game presented in [7] can not be repeated more times, and their final result is:

Theorem 7 ([7])**.** NEEXP $\subseteq$ MIP*.

However, a similar game from [4] with a very careful analysis, allowed them to recurse the introspection idea and the final result is that:

Theorem 8 ([4])**.** MIP* = RE*, i.e., the Halting problem is* MIP**-complete.*

References

[1] László Babai, Lance Fortnow, and Carsten Lund. Non-deterministic exponential time has two-prover interactive protocols. In *31st Annual Symposium on Foundations of Computer Science, St. Louis, Missouri, USA, October 22-24, 1990, Volume I*, pages 16–25. IEEE Computer Society, 1990.

[2] Tsuyoshi Ito and Thomas Vidick. A multi-prover interactive proof for NEXP sound against entangled provers. In *53rd Annual IEEE Symposium on Foundations of Computer Science, FOCS 2012, New Brunswick, NJ, USA, October 20-23, 2012*, pages 243–252. IEEE Computer Society, 2012.

[3] Rahul Jain, Zhengfeng Ji, Sarvagya Upadhyay, and John Watrous. QIP = PSPACE. *J. ACM*, 58(6):30:1–30:27, 2011.

[4] Zhengfeng Ji, Anand Natarajan, Thomas Vidick, John Wright, and Henry Yuen. Mip*=re. *CoRR*, abs/2001.04383, 2020.

[5] Alexei Y. Kitaev and John Watrous. Parallelization, amplification, and exponential time simulation of quantum interactive proof systems. In F. Frances Yao and Eugene M. Luks, editors, *Proceedings of the Thirty-Second Annual ACM Symposium on Theory of Computing, May 21-23, 2000, Portland, OR, USA*, pages 608–617. ACM, 2000.

[6] Carsten Lund, Lance Fortnow, Howard J. Karloff, and Noam Nisan. Algebraic methods for interactive proof systems. *J. ACM*, 39(4):859–868, 1992.

[7] Anand Natarajan and John Wright. NEEXP is contained in MIP. In David Zuckerman, editor, *60th IEEE Annual Symposium on Foundations of Computer Science, FOCS 2019, Baltimore, Maryland, USA, November 9-12, 2019*, pages 510–518. IEEE Computer Society, 2019.

[8] Adi Shamir. Ip=pspace. In *31st Annual Symposium on Foundations of Computer Science, St. Louis, Missouri, USA, October 22-24, 1990, Volume I*, pages 11–15. IEEE Computer Society, 1990.