Quantum Computation **(CMU 15-859BB, Fall 2015)**

Lecture 26: QIP and QMIP

December 9, 2015

Lecturer: John Wright *Scribe: Kumail Jaffer*

1 Introduction

Recall QMA and QMA(2), the quantum analogues of MA and AM. Today we will look at QIP, the quantum analogue of the classical IP, the class of languages with polynomial size "interactive proof systems."

An interactive proof system consists of two algorithms, P, the prover, and V, the verifier. P has access to infinite computational power, while V is limited to computing with the assumptions of BPP. They are allowed to exchange polynomially messages, and at the end the verifier outputs whether or not the input is in the language. More precisely,

Definition 1.1. a language L is in IP if there exists a BPP verifier V such that the following two conditions hold:

1. (completeness) If $x \in L$, there exists a prover P such that the system consisting of P and V accepts x with probability $\geq 2/3$.
2. (soundness) If $x \notin L$, for all provers Q, the system consisting of Q and V accepts x with probability $\leq 1/3$.

This is like MA and AM, except the machines can exchange polynomially many instead of 1 or 2 messages.

The discovery of the fact that IP = PSPACE is one of the great stories of classical complexity [Sha92][LFKN92]. Let's dive into the quantum version of this story, which is being written today.

2 Definition of QIP

Definition 2.1. a language L is in QIP if there exists a BQP verifier V such that the completeness and soundness conditions hold. The prover and the verifier are additionally allowed to exchange quantum messages.

Let's also define another useful notion:

Definition 2.2. QIP(k) is the set of languages with quantum interactive proof protocols of length k.

Using this definition, the following facts are clear

Fact 2.3. $\mathsf{QIP} = \bigcup_{f \text{ is polynomial}} \mathsf{QIP}(f(n))$

Fact 2.4. $\mathsf{QIP}(0) = \mathsf{BQP} \subseteq \mathsf{QIP}(1) = \mathsf{QMA} \subseteq \mathsf{QAM} \subseteq \mathsf{QIP}(2) \subseteq \mathsf{QIP}(3) \subseteq \ldots \mathsf{QIP}$.

Here's another definition:

Definition 2.5. $\mathsf{QIP}(k, p_1, p_2)$ is the set of languages with quantum interactive proof protocols of length k, correctness probability of accepting at least p_1 and soundness probability of accepting at most p_2.

It is not immediately clear that QIP is robust to tweaking the probabilities of the soundness and correctness conditions, but it turns out that it is indeed the case that

Fact 2.6. $\mathsf{QIP}(k, p_1, p_2) = \mathsf{QIP}(k, 1, 2^{-poly})$, *as long as* $p_1 - p_2 \geq poly^{-1}$ *[KW00].*

Finally, Here are two grab bag facts:

Fact 2.7. $\mathsf{IP} \subseteq \mathsf{QIP}$

The proof is very similar to the proof that $\mathsf{MA} \subseteq \mathsf{QMA}$, with the key being to measure the quantum state before doing anything else.

Fact 2.8. *Without loss of generality, we can assume in any interactive proof protocol that the last message sent is from the prover to the verifier.*

Clearly, the verifier accomplishes nothing by sending a message the other way and outputting before receiving a response.

With definitions out of the way, let's look at a problem in QIP to better understand the nature of the class.

3 Quantum State Distinguishability

You are given two mixed states, $\rho_0, \rho_1 \in \mathbb{C}^{d \times d}$ in the form of classical descriptions of quantum circuits that output these states. The task is to output YES if $d_{tr}(\rho_0, \rho_1) \geq .9$, and NO if $d_{tr}(\rho_0, \rho_1) \leq .1$.

Where d_{tr} is the trace distance. Equivalently, we saw in a previous lecture that

Fact 3.1.

$$d_{tr}(\rho_0, \rho_1) = 2 \left\{ \begin{array}{c} \textit{optimal probability of guessing } b, \\ \textit{given a uniformly random } \rho_b \textit{ from } b = \{0,1\} \end{array} \right\} - 1$$

One natural thing we might want to try is to send the optimal POVM for distinguishing the two states as a certificate. Then the verifier can just measure random states with the POVM many times, and if it gets back the same one it put in sufficiently many times, we can say YES. The trouble with this is that it's unclear how we can encode a POVM efficiently into a message. Even if we just use the Pretty Good Measurement, we don't know an efficient way to encode it. If we could find a nice way to encode it, we'd solve this problem in QMA.

But if we allow ourselves a polynomial length protocol, we have more freedom. In fact, it turns out we don't need to send any information about how the prover solves the problem at all (!). We can use the standard coke-pepsi trick from classical zero knowledge proofs.

Claim 3.2. *The following protocol solves the problem in* QIP*:*

1. *Verifier picks $b \in \{0,1\}$ uniformly at random, and sends the prover ρ_b without saying which one it is.*
2. *Prover responds with a guess, b', for b*
3. *Verifier accepts if and only if $b = b'$.*

Proof. If $d_{tr}(\rho_0, \rho_1) \geq 0.9$, then by fact 3.1, there's a measurement which gives the prover a probability $\geq (1 + 0.9)/2 = 0.95$ of guessing b correctly. So the correctness condition is satisfied.

On the other hand, if $d_{tr}(\rho_0, \rho_1) \leq 0.1$, then again by fact 3.1, there's no measurement which gives the prover a probability better than $(1 + 0.1)/2 = 0.55$ of guessing correctly. So the soundness condition is satisfied. □

This protocol has the nice property that is "zero-knowledge", that is, the protocol gives the verifier no information about how to solve the problem itself. It only allows it to conclude that the prover knows what it's doing.

Further, it turns out out that the Quantum State Distinguishability problem is complete for QSZK, the class of languages with zero-knowledge quantum interactive proof protocols [Wat02].

Now let's turn to the related problem of quantum circuit distinguishability.

3.1 Quantum Circuit Distinguishability

You are given two classical descriptions of quantum circuits, c_0 and c_1. The task is to determine if they compute "noticeably" different functions.

To make the notion of "noticeably" different more concrete, we use the following definition

Definition 3.3 (Diamond distance)**.**

$$d_\diamond(c_0, c_1) = \max_{\text{pure states } |\chi\rangle} d_{tr}(c_0 |\chi\rangle, c_1 |\chi\rangle)$$

The task, then, is to output YES if $d_\diamond(c_0, c_1) \geq 0.9$, and NO if $d_\diamond(c_0, c_1) \leq 0.1$.

Claim 3.4. *The following protocol solves the problem in* QIP*:*

1. *Prover sends $|\chi\rangle$, a state claimed to cause noticeably different outputs on c_0 and c_1.*
2. *Verifier picks $b \in \{0,1\}$ uniformly at random, calculates $c_b|\chi\rangle$ and sends it, without saying which one it is.*
3. *Prover responds with a guess, b' for b.*
4. *Verifier accepts if and only if $b = b'$.*

Proof. This clearly works for the same reason that our protocol for Quantum State Distinguishability worked. □

4 QIP collapses to QIP(3) (!!)

Kitaev and Watrous showed in 2000 that every protocol in QIP can be turned into a protocol using only 3 messages [KW00]. In particular, they showed

Theorem 4.1. $\mathsf{QIP}(k, 1, 1-\epsilon) = \mathsf{QIP}(3, 1, 1-\frac{\epsilon^2}{4k^2})$

Proof (sketch). Note first that any quantum interactive proof looks something like the following diagram.

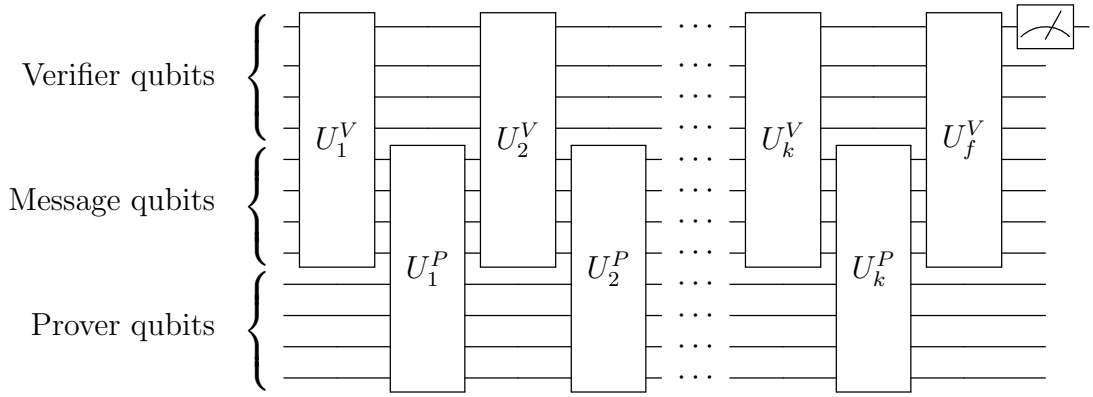

Figure 1: Quantum Interactive Proof System

That is, it consists of multiple rounds of the verifier and the prover applying unitary transformations (U_i^V for the verifier, and U_i^P for the prover) to a set of qubits, where they share some "message qubits", and both the prover and the verifier have their own private qubits that the other cannot interfere with. The protocol ends with the Verifier applying a final unitary transformation, U_f^V, and measuring the first qubit. If it is a 0 it rejects, if it is a 1 it accepts.

Let's call the state before the i'th verifier unitary $|\boldsymbol{V}_i, \boldsymbol{M}_i, \boldsymbol{P}_i\rangle$, where the entries of the tuple are the verifier's qubits, the message qubits, and the prover's qubits, respectively. A transcript is a list of all the states. The following protocol proves the theorem.

1. Prover generates a transcript for the full run of the original length $2k+1$ protocol on the given input, and sends it to the verifier. Verifier preforms U_f^V on $|\boldsymbol{V}_k, \boldsymbol{M}_k\rangle$ and rejects if the first bit of the result is not 1.

2. Verifier prepares entangled state $|b, b'\rangle = \frac{1}{\sqrt{2}}(|0\rangle|0\rangle + |1\rangle|1\rangle)$, and picks $i \in \{1 \dots k-1\}$ uniformly at random. Verifier applies V_i to $|\boldsymbol{V}_i, \boldsymbol{M}_i\rangle$, controlled swaps $\boldsymbol{V}_i$ and $\boldsymbol{V}_{i+1}$ with control bit b, and sends $\boldsymbol{V}_i, \boldsymbol{V}_{i+1}, b', i$ to the prover.

3. Prover applies U_i^P to $|\boldsymbol{M}_i, \boldsymbol{P}_i\rangle$ and controlled swaps it with $|\boldsymbol{M}_{i+1}, \boldsymbol{P}_{i+1}\rangle$ with control bit b', and sends it back. Verifier receives b' from prover, controlled nots it with b, applies hadamard to b and accepts if and only if result is 0

The high level idea of this protocol is that the verifier picks a random step of the proof to check, and checks it by doing the computation for that step itself, preforming a swap test to confirm that it is what the prover claimed it was. It is easy to show that if the original protocol accepts with certainty, this protocol does as well.

Proving soundness is a bit more difficult. For more details, see [KW00].

□

In fact, it turns out that the above protocol can be somehow formulated as a semidefinite program solvable in exponential time, showing $\mathsf{QIP} \subseteq \mathsf{NEXP}$ [KW00]. It even turns out to be true that $\mathsf{QIP} = \mathsf{PSPACE} = \mathsf{IP}$, so quantum gives us no additional power in the domain of interactive proofs [JJUW10].

5 QMIP

One interesting variation on the IP setting that's been widely studied is to allow multiple independent non-communicating provers, resulting in the classical class MIP. Intuitively, this would seem to give us more power because it is the mathematical formalism for the classic policework trick of separating two co-conspirators before interrogating them in an effort to discover the truth. A famous result of Babai, Fortnow, and Lund says that $\mathsf{MIP} \subseteq \mathsf{NEXP}$ [BFL90].

We can generalize this to the quantum setting where we are again allowed multiple non-communicating provers, except that the verifier is in BQP, and the provers are allowed to prepare an entangled state before beginning the protocol. This is, of course, a highly realistic assumption, since you couldn't prevent two co-conspirators from walking into their interrogation rooms with halves of EPR pairs in their pockets. The resulting class is called QMIP.

Currently, we don't know much about QMIP. It is not immediately clear even that $\mathsf{MIP} \subseteq \mathsf{QMIP}$, since the entanglement of the provers prevents us from simply measuring

to avoid sneaky quantum behavior. In 2012, however, it was shown that this is the case, as a corollary of the fact that NEXP $\subseteq$ QMIP [IV12]. It's also true that QMIP = MIP*, where MIP* is MIP except the provers are allowed to share an entangled state before the protocol (note that the verifier is of BPP and not BQP), so the power of QMIP lies in prover entanglement and not in quantum messaging [RUV13].

The most glaring gap in our knowledge is the lack of an analogue of MIP $\subseteq$ NEXP. In fact, it's quite a bit worse than that: We don't know any upper bound at all! Not even the class of decidable languages is known to be an upper bound. For all we know, QMIP could be powerful enough to solve the halting problem, or indeed all possible problems.

References

[BFL90] L. Babai, L. Fortnow, and C. Lund. Nondeterministic exponential time has two-prover interactive protocols. In *Foundations of Computer Science, 1990. Proceedings., 31st Annual Symposium on*, volume 1, pages 16–25, October 1990.

[IV12] Tsuyoshi Ito and Thomas Vidick. A multi-prover interactive proof for NEXP sound against entangled provers. In *Proceedings of the 2012 IEEE 53rd Annual Symposium on Foundations of Computer Science*, FOCS '12, pages 243–252, Washington, DC, USA, 2012. IEEE Computer Society.

[JJUW10] Rahul Jain, Zhengfeng Ji, Sarvagya Upadhyay, and John Watrous. QIP = PSPACE. *Commun. ACM*, 53(12):102–109, December 2010.

[KW00] Alexei Kitaev and John Watrous. Parallelization, amplification, and exponential time simulation of quantum interactive proof systems. In *Proceedings of the Thirty-second Annual ACM Symposium on Theory of Computing*, STOC '00, pages 608–617, New York, NY, USA, 2000. ACM.

[LFKN92] Carsten Lund, Lance Fortnow, Howard Karloff, and Noam Nisan. Algebraic methods for interactive proof systems. *J. ACM*, 39(4):859–868, October 1992.

[RUV13] Ben W. Reichardt, Falk Unger, and Umesh Vazirani. A classical leash for a quantum system: Command of quantum systems via rigidity of CHSH games. In *Proceedings of the 4th Conference on Innovations in Theoretical Computer Science*, ITCS '13, pages 321–322, New York, NY, USA, 2013. ACM.

[Sha92] Adi Shamir. IP = PSPACE. *J. ACM*, 39(4):869–877, October 1992.

[Wat02] J. Watrous. Limits on the power of quantum statistical zero-knowledge. In *Foundations of Computer Science, 2002. Proceedings. The 43rd Annual IEEE Symposium on*, pages 459–468, 2002.