Introduction to Quantum Proof Systems and Applications

Liam Salt - APMA 990

April 4, 2022

Abstract

In this paper, we introduce quantum interactive proof systems, demonstrate their advantages over classical systems, and showcase their utility by using them to prove certain properties of finite groups. A quantum proof is a state which, along with some quantum computations, can be used to solve problems in Quantum MA (QMA), a class of decision problems analogous to NP. Using the notion of a black box group, a type of oracle, we exemplify the power of quantum proof systems by considering such problems as group non-membership. We show that this problem is solvable (up to some error) in polynomial time, which is impossible classically. Efficient solving and verification of group non-membership also allows us to approach other group theoretical problems such as: finding the maximal normal subgroup, and whether an integer N divides the order of a group.

Contents

1 Introduction

1.1 Computer Science Background

Before we can begin to define a quantum proof system, we first must introduce some classical computing formalism and terminology. In classical computing, an *interactive proof system*

is a Turing machine that encapsulates the mathematical idea of a proof. It models a proof as a sequence of communications between two actors: a potentially dishonest *prover*, and a skeptical, honest *verifier*. Through various *rounds* of communication, the prover attempts to convince the verifier to accept their proof. The default convention is to assume the verifier is constrained to be a deterministic polynomial-time TM, whereas we make no constraints to the prover's computational power. Additionally, an interactive proof system is assumed to satisfy both *completeness* and *soundess*, defined below:

Definition. *An interactive proof system is said to be complete or to satisfy completeness if for any true statement, the prover can always convince the verifier of its validity.*

Definition. *An interactive proof system is said to be sound if for any false statement, the prover can only with negligible probability convince the verifier of its validity.*

Remark. *Often, many of these conditions are loosened in various ways, e.g. the prover only needing to convince the verifier up to some probability.*

Now we can more formally define a (deterministic) interactive proof system.

We say that a formal language L has a deterministic k-round interactive proof system with error probability ε with prover verifier pair (P, V) if all of the following:

1. (P, V) either accepts or rejects any input after k rounds
2. $\forall x \in L, (P, V)$ accepts x with probability 1
3. $\forall x \notin L, (P, V)$ accepts x with probability ε

Given this definition, a familiar, degenerate example is to show that any language in **NP** has a $1-$round interactive proof system with error probability 0. We recall that one definition of **NP** is as the set of decision problems whose positive answers can be verified in polynomial time by a deterministic Turing machine. Accordingly, if a problem is in **NP**, then by definition on input our prover can produce a polynomial-sized certificate, which can be verified in polynomial-time exactly when the input is in the language. In fact, the class of languages which have a deterministic interactive proof system, termed **dIP**, is exactly equal to **NP**.

We can generalize the above definition by allowing the verifier to also have access to a random number generator, with which it may generate and make use of random bits in each round. The addition of the verifier's access to randomness brings us to the full definition of an interactive proof system. The complexity class of all languages that have interactive proof systems is termed **IP**, and it can be shown [potentially in this paper] that **IP**=**PSPACE**. In other words, problems which can be accepted by an interactive proof system are exactly those problems which require only polynomial-sized space (memory) to answer.

1.2 Arthur-Merlin Protocol

One particular subclass of **IP** which will become important in the quantum case is **MA**, which stands for Merlin-Arthur (of King Arthur fame), and is comprised of languages which have a particular type of interactive proof system known as an Arthur-Merlin protocol. This class was originally described in [Bab] and [Gol].

Definition. *A language L is said to have an Arthur-Merlin protocol with prover-verifier pair (Merlin, Arthur) if:*

1. *Arthur has access to random bits $y \in B_m$, which are public (known to Merlin)*
2. *If $x \in L$, then $\exists z \in B_n$ such that $P(\text{Arthur accepts } x|y,z) \geq \frac{2}{3}$.*
3. *If $x \notin L$, then $\forall z \in B_n$ such that $P(\text{Arthur accepts } x|y,z) \leq \frac{1}{3}$.*
4. *n, m are both polynomial-length*

Equivalently, we say $L \in \mathbf{MA}$. For clarity, in the above definition z is the polynomial-sized proof certificate provided by Merlin, and whether Arthur accepts Merlin's proof depends on both the string sent by Merlin and the random string produced by Arthur.

The probability conditions given above are quite flexible. Given a language meeting the definition above, it can be shown that we can induce a new Arthur-Merlin pair for that language that achieves perfect completeness, and soundness with error probability at most $\frac{1}{2}$.

2 Quantum Interactive Proof Systems

dd

2.1 Relationship Between QMA and PP

d

3 Group Theoretic Applications

d

3.1 Group Non-Membership

d

3.2 Other Applications/Open Problems

j

References

[Bab] L. Babai, Trading Group Theory for Randomness, Proceedings of the Seventeenth Annual ACM Symposium on the Theory of Computing, 1985, 421-429.

[Wat1] J. Watrous, PSPACE has 2-round quantum interactive proof systems Annual ACM Symposium on the Theory of Computing, 1985, 421-429.