Parallel approximation of min-max problems with applications to classical and quantum zero-sum games

Gus Gutoski∗ Xiaodi Wu†

∗Institute for Quantum Computing and School of Computer Science
University of Waterloo, Waterloo, Ontario, Canada
†Department of Electrical Engineering and Computer Science
University of Michigan, Ann Arbor, Michigan, USA

April 11th, 2011

Abstract

This paper presents an efficient parallel algorithm for a new class of min-max problems based on the matrix multiplicative weight (MMW) update method. Our algorithm can be used to find near-optimal strategies for competitive two-player classical or quantum games in which a referee exchanges any number of messages with one player followed by any number of additional messages with the other. This algorithm considerably extends the class of games which admit parallel solutions and demonstrates for the first time the existence of a parallel algorithm for any game (classical or quantum) in which one player reacts adaptively to the other.

A special case of our result is a parallel approximation scheme for a new class of semidefinite programs whose feasible region consists of $n$-tuples of semidefinite matrices that satisfy a certain consistency condition. Applied to this special case, our algorithm yields a direct polynomial-space simulation of multi-message quantum interactive proofs resulting in a first-principles proof of $\text{QIP} = \text{PSPACE}$. It is noteworthy that our algorithm establishes a new way, called the min-max approach, to solve SDPs in contrast to the primal-dual approach to SDPs used in the original proof of $\text{QIP} = \text{PSPACE}$. It also follows from our work that several competing-provers complexity classes collapse to PSPACE such as $\text{QRG}(2)$, $\text{SQG}$ and two new classes called $\text{DIP}$ and $\text{DQIP}$. 

It is noteworthy that our algorithm establishes a new way, called the min-max approach, to solve SDPs in contrast to the primal-dual approach to SDPs used in the original proof of $\text{QIP} = \text{PSPACE}$. It also follows from our work that several competing-provers complexity classes collapse to PSPACE such as $\text{QRG}(2)$, $\text{SQG}$ and two new classes called $\text{DIP}$ and $\text{DQIP}$. 

1 Introduction

1.1 Results

Parallel approximation of semidefinite programs and min-max problems

This paper presents an efficient parallel algorithm for a new class of min-max problems with applications to classical and quantum zero-sum games and interactive proofs. A special case of our result is a parallel approximation scheme for semidefinite programs (SDPs) of the form

\[
\begin{align*}
\text{minimize} & \quad \text{Tr}(X_n P) \\
\text{subject to} & \quad \text{Tr}_{C_n}(X_n) = \Phi_{n-1}(X_{n-1}) \\
& \quad \vdots \\
& \quad \text{Tr}_{C_2}(X_2) = \Phi_1(X_1) \\
& \quad \text{Tr}_{C_1}(X_1) = Q \\
& \quad X_1, \ldots, X_n \succeq 0.
\end{align*}
\]

where \(\text{Tr}_{C_1}, \ldots, \text{Tr}_{C_n}\) are partial trace maps and \(\Phi_1, \ldots, \Phi_{n-1}\) are arbitrary completely positive and trace-preserving maps.\(^1\) It has long since been known that the problem of approximating the optimal value of an arbitrary SDP is logspace-hard for \(\mathsf{P}\),\(^2\) so there cannot be a parallel approximation scheme for all SDPs unless \(\mathsf{NC} = \mathsf{P}\). However, the precise extent to which SDPs admit parallel solutions is not known. Our result adds considerably to the set of such SDPs. The result is stated in full generality as follows.

**Theorem 1** (Informal, see Section 6 for details). Let \(A\) denote the feasible region of the SDP (1). There exists an efficient parallel oracle-algorithm for finding approximate solutions to the min-max problem

\[
\begin{align*}
\min_{(X_1, \ldots, X_n) \in A} \max_{P \in \mathcal{P}} \text{Tr}(X_n P)
\end{align*}
\]

with an oracle for optimization over the set \(\mathcal{P}\). The SDP (1) is recovered from the above min-max problem (2) in the special case where \(\mathcal{P} = \{P\}\) is a singleton set.

We also describe parallel implementations of this oracle for certain sets \(\mathcal{P}\), yielding an unconditionally efficient parallel approximation algorithm for the min-max problem (2) for those choices of \(\mathcal{P}\).

Applications to zero-sum games

This algorithm can be used to find near-optimal strategies for a new class of competitive two-player games that are moderated by a referee and obey the following protocol.

(i) The referee exchanges several messages only with Alice.

(ii) After processing this interaction with Alice, the referee exchanges several additional messages only with Bob. After further processing, the referee declares a winner.

---

\(^1\)The partial trace and complete positivity are standard notions from quantum information. A linear map from square matrices to square matrices denotes a quantum channel if and only if it is completely positive and trace preserving.

\(^2\)Hardness of approximation for SDPs follows from hardness of approximation for linear programming [Ser91, Meg92].
Indeed, our algorithm applies even to quantum games, in which the referee and players are free to exchange and process quantum information. Due to the similarity with the oft-studied interactive proof model of computation, games of this form shall be called double interactive proofs: the referee in such a game executes a standard interactive proof with Alice followed by a second interactive proof with Bob. (This protocol is depicted in Figure 2 on page 10. See Sections 4 and 9.1 for further detail.)

If the referee is specified succinctly by circuits rather than in explicit matrix form then our parallel algorithm can be used to find near-optimal strategies in polynomial space (via the relation NC(poly) = PSPACE [Bor77]). This algorithm is optimal in that it is PSPACE-hard even to distinguish games that Alice can win with near certainty from games that Bob can win with near certainty. This strong form of PSPACE-hardness holds even in the special case of two-turn games [FK97] where the referee exchanges only two messages synchronously with each player.

Ordinary interactive proofs could also be cast as a special type of game in which the referee completely ignores Bob. Taking this view, the celebrated proof of IP = PSPACE [LFKN92, Sha92] implies a similar hardness result: it is PSPACE-hard to distinguish interactive proofs that Alice can win with certainty from those which she can win with only exponentially small probability.

Prior to the present work polynomial-space algorithms were known only for two-turn classical games and for quantum interactive proofs. The algorithm for two-turn games is due to Feige and Kilian [FK97]. Algorithms for quantum interactive proofs are presented in proofs of QIP = PSPACE [JJUW10, Wu10a].

Our result unifies and subsumes both of these algorithms. It also demonstrates for the first time the existence of a parallel algorithm for two-turn quantum games and for any game (classical or quantum) in which one player reacts adaptively to the other.

Applications to complexity theory

In complexity theory, our result implies the collapse to PSPACE of several classical and quantum interactive proof classes. Letting DIP and DQIP denote the competing-provers complexity classes associated with classical and quantum double interactive proofs, respectively, we have

**Corollary 1.1.** DQIP = DIP = PSPACE.

In contrast to the classical case, the competing-provers complexity class QRG(2) associated with two-turn quantum games was not known to be a subset of PSPACE prior to the present work. A special case of our result yields the equality

QRG(2) = PSPACE,

thus solving an open problem of Ref. [JJUW10]. Of course, every other complexity class whose protocol can be cast as a double interactive proof also collapses to PSPACE, such as SQG [GW05].

In the special case of the SDP (1) our algorithm yields a direct polynomial-space simulation of multimessage quantum interactive proofs, resulting in a first-principles proof of QIP = PSPACE. By contrast, all other known proofs [JJUW10, Wu10a] rely on the highly nontrivial fact that the verifier and prover in a quantum interactive proof can be assumed to exchange only three messages [KW00]. The original proof of Jain et al. also relies on the additional assumption that verifier’s only message to the prover is a single classical coin flip [MW05].

1.2 Techniques

Our algorithm is an example of the matrix multiplicative weights update method (MMW) as discussed in the survey paper [AHK05] and in the PhD thesis of Kale [Kal07]. We also draw upon the valuable experience
of recent applications of this method to parallel algorithms for quantum complexity classes [JW09, JUW09, JJUW10, Wu10a]. However, our application of the MMW method is somewhat different from all previous ones in the sense that our algorithm is applied twice in a two-level recursive fashion. At the top level, our algorithm makes use of the MMW method to solve a min-max problem. At the bottom level, a special case of our algorithm is used to solve a SDP problem as the implementation of the oracle for any min-max problem required by the MMW method. Previously the MMW was used only in primal-dual approaches to SDPs [AK07, Kal07, JUW09, JJUW10]. By contrast, we do not take a primal-dual approach—our SDP solution arises as a special case of a more general min-max problem. A more detailed comparison can be found below.

A naive approach to find optimal strategies for competitive two-player games is to choose a natural representation for these strategies and optimize over it. For two-turn classical games the natural representation is a table of probabilities—a stochastic matrix. Indeed, Feige and Kilian successfully optimize over this representation in their complicated and highly specialized precursor to the MMW that solves two-turn classical games in polynomial space [FK97].

For two-turn quantum games a strategy is naturally represented by a quantum channel. For more complicated games such as double quantum interactive proofs the most natural representation is a quantum strategy [GW07], which may be viewed as a special type of channel. A quantum channel is typically specified by its Choi-Jamiolkowski matrix [Wat08, Lecture 5]. But optimizing over Choi-Jamiolkowski matrices is a task fraught with difficulty [JUW09]; optimizing over Choi-Jamiolkowski matrices that also represent quantum strategies can only be harder.

Fortunately, double quantum interactive proofs admit another representation for strategies that is more suitable for our purpose. In Kitaev’s transcript representation [Kit02] the actions of a player are represented by a list $\rho_1, \ldots, \rho_n$ of density matrices that satisfy a special consistency condition. Intuitively, these density matrices correspond to “snapshots” of the state of the referee’s qubits at various times during the interaction. (See Figure 3 on page 12.)

The key property of double quantum interactive proofs that we exploit is the ability to draw a “temporal line” in the interaction just after Alice’s last action. Given a transcript $\rho_1, \ldots, \rho_n$ for Alice, the actions of Bob can then be represented by another transcript $\xi_1, \ldots, \xi_m$. By optimizing over all such transcripts one obtains an oracle for “best responses” for Bob to a given strategy of Alice as required by the MMW.

Whereas the MMW in its unaltered form can be used to solve min-max problems over the domain of density operators, we introduce a new extension to this method for min-max problems over the domain of transcripts—a domain consisting of lists of multiple operators, each drawn from a strict subset of the density operators. The high-level approach of our method is as follows:

1. **Extend the domain from a single density matrix to a list of $n$ density matrices.**
   This step is relatively straightforward: the MMW can be applied without complication to all $n$ density matrices at the same time.

2. **Restrict the domain to a strict subset of density matrices.**
   This step is more difficult. It is accomplished by relaxing the game so as to allow all density matrices, with an additional penalty term to remove incentive for the players to use inconsistent transcripts.

3. **Round strategies in the relaxed game to strategies in the original game.**
   For this step one must prove a “rounding” theorem (Theorem 5), which establishes that near-optimal, fully admissible strategies can be obtained from near-optimal strategies in the unrestricted domain with penalty term.
Primal-dual MMW versus min-max MMW

It is interesting to compare the method used in the proof of $\text{QIP} = \text{PSPACE}$ \cite{JJUW10} and the one used here, especially for their applications to SDP problems. Of course, both methods are based on the MMW method and share lots of similarities at the first look. However, significant differences exist for those two methods. The method used by Jain et al. is the so-called primal-dual approach for solving SDPs originally from \cite{AK07}. This method makes use of the duality between the primal and dual problem of any SDP instance. Our method, on the contrary, makes no use of such duality. Instead our method, which works for min-max problems and thus is called the min-max approach, solves the SDPs as a special case when the max part is trivial.

Both methods requires some efficient oracles for different subproblems and rounding theorems which convert approximately feasible solution to exact feasible solution without sacrificing the objective function too much. The sets of possible SDPs solvable by each method respectively are not known to coincide essentially because the existence of such efficient oracle and rounding theorem in one method doesn’t imply their existence in the other method. Since the existence of such oracle and rounding theorem relies heavily on the specific form of SDPs in consideration, it is hard to argue which method is better than the other in general. Nevertheless, some advantages of the min-max approach are known \cite{Wu10b}. For example, there exists a generic design of efficient oracle in the min-max approach while the existence of corresponding rounding theorem is not guaranteed and the approximately feasible solutions obtained in the min-max approach are close to feasible solutions in terms of $\mathcal{L}_1$ norm rather than $\mathcal{L}_\infty$ norm in the primal-dual approach. We now consider the specific forms of SDPs in our comparison. If one rewrites the algorithm solving SDPs in \cite{JJUW10} according to the standard way in Kale’s thesis \cite{Kal07}, one can find out the existence of efficient oracle for the primal-dual approach depends on some additional assumptions from the complexity model which are no longer valid in our case. Moreover, the rounding theorem for the constraints in our problem, namely general partial trace constraints, requires the approximately feasible solution is close to the exact feasible solution in $\mathcal{L}_1$ norm.\footnote{The SDP in \cite{JJUW10} also has partial trace constraints. However, it is solved by the additional assumption that the measurement matrix is invertible and has bounded condition number. This assumption makes it possible that only scaled identity matrix appears in the analysis and the $\mathcal{L}_\infty$ norm bound is sufficient. Such assumption is invalid in our case essentially because our algorithm recursively calls itself as the oracle. No assumption could be made about those inputs to the oracle since they are arbitrary instances generated during the MMW update.} Those difficulties make it hard to apply the primal-dual approach in our case. Instead, we design the min-max approach and add the penalty term to facilitate the use of MMW method. Furthermore, our results establishes a much larger class of SDPs that admits efficient parallel algorithms.

The Bures metric

Finally, it is noteworthy that the proof of our rounding theorem (Theorem \ref{thm:rounding}) contains an interesting and nontrivial application of the Bures metric, which is a distance measure for quantum states that is defined in terms of the more familiar fidelity function.

Properties of the trace norm, which captures the physical distinguishability of quantum states, are often sufficient for most needs in quantum information. When some property of the fidelity is also required one uses the Fuchs-van de Graaf inequalities to convert between the trace norm and fidelity \cite{FvdG99}. However, every such conversion incurs a quadratic slackening of relevant accuracy parameters. Our study calls for repeated conversions, which would incur an unacceptable exponential slackening if done naively via Fuchs-van de Graaf. Instead, we make only a single conversion between the trace norm and...
the Bures metric and then repeatedly exploit the simultaneous properties of (i) the triangle inequality, (ii) contractivity under quantum channels, and (iii) preservation of subsystem fidelity.

Although conversion inequalities between the trace norm and Bures metric are implied by Fuchs-van de Graaf, to our knowledge explicit conversion inequalities have not yet appeared in published literature. The required inequalities are derived in the present paper (Proposition 3).

Organization of the paper

The rest of the paper is organized as follows. We refer curious readers to Section 2 for further comments on related work. A brief preliminaries is provided in Section 3, followed by the formalization of the double quantum interactive proofs in Section 4. The rounding theorem, MMW based oracle-algorithm and the implementation of the oracle for certain choices of the set P are described in Section 5, 6, 7, respectively. The containment of DQIP inside PSPACE is proved in Section 8. We conclude with some extensions of the main results in Section 9.

2 Further comments on related work

2.1 Parallel approximation of semidefinite programs

We noted earlier that there is no parallel approximation scheme for arbitrary SDPs unless NC = P. But that fact does not rule out the existence of parallel algorithms for interesting subclasses of SDP.

Some of what is known about SDPs in this respect is inherited knowledge from linear programs (LPs). For example, Luby and Nisan describe their own precursor to the MMW that yields a parallel approximation scheme for so-called positive LPs where all input numbers are positive [LN93]. By contrast, Trevisan and Xhafa show that it is P-hard to find exact solutions for positive LPs [TX98].

The notion of a positive instance of an LP can be generalized to SDPs as follows. An SDP of the form

\[
\text{minimize } \quad \text{Tr} (X P)
\]

\[
\text{subject to } \quad \Psi(X) \succeq Q
\]

\[
X \succeq 0
\]

is said to be positive if P, Q \succeq 0 and \Psi is a positive map (meaning that \Psi(X) \succeq 0 whenever X \succeq 0). Of course, P-hardness of exact solutions for positive LPs implies P-hardness of exact solutions for positive SDPs. By analogy with the Luby-Nisan algorithm for positive LPs, Jain and Watrous give a parallel approximation algorithm based on MMW for positive SDPs [JW09]. The algorithm is derived from a correspondence between positive SDPs and one-turn quantum refereed games and can therefore be recovered as a special case of the work of the present paper.

Unlike the present paper, the original proof of QIP = PSPACE due to Jain et al. [JJUW10] does not take advantage of the transcript representation for multi-turn strategies. Instead, those authors derive a special SDP based on additional assumptions of the complexity class. It is not difficult to see that their SDP can be written in the form (1) considered in the present paper. Thus, the work of Jain et al. is also subsumed by our algorithm. It is noteworthy that neither the SDP instance from Ref. [JJUW10] nor its generalization (1) from the present paper are positive SDP instances.
2.2 Algorithms for competitive two-player games

Competitive two-player games are often modeled as either a table of payouts (normal form) or a game tree (extensive form). The extensive form model is equivalent to the refereed games model wherein the game is specified by a referee who exchanges messages with the players and declares a winner at the end of the interaction. In this paper we prefer the refereed games model for its simplicity and the ease with which it extends to quantum games.

The normal form is historically the most popular model, though it is not fully general like the extensive form or refereed game models. Indeed, normal form games correspond to the very restricted class of one-turn refereed games in which there is no communication from the referee to the players. Despite this restriction, the problem of computing the exact value of a normal form game is logspace-hard for \( P \) \cite{FIKU08,FKS95}. This hardness result is striking when juxtaposed with the existence of deterministic polynomial-time algorithms for arbitrary, multi-turn games \cite{KM92,KMvS94}.

For succinct games in which the table of payouts, game tree, or referee is specified implicitly by circuits the aforementioned results immediately imply \( \text{EXP} \)-completeness for the problem of computing the exact value of a game. By contrast, the relaxed problem of approximating the value of such a game is much more diverse. For arbitrary multi-turn games \( \text{EXP} \)-hardness extends to the relaxed problem of distinguishing games that Alice can win with near certainty from games that Bob can win with near certainty \cite{FK97}.

But the situation is much different for shorter games. Earlier we mentioned that Feige and Kilian gave both (i) a polynomial-space approximation scheme for succinct two-turn games, and (ii) a matching \( \text{PSPACE} \)-hardness result valid even for weak approximations. Fortnow \textit{et al.} prove that the problem of approximating the value of a succinct game in normal form (\textit{i.e.} a one-turn classical game) is complete for \( \text{S}_2^P \) and they give a \( \text{ZPP}^\text{NP} \) approximation scheme based on the multiplicative weights update method for the related search problem of finding near-optimal strategies for these games \cite{FIKU08}.

All that was known of quantum games prior to the present work is that arbitrary, multi-turn quantum games admit a polynomial-time exact solution \cite{GW07} and that one-turn quantum games admit an efficient parallel approximation scheme \cite{JW09}. For both classical and quantum games it is an interesting open question as to whether there is a parallel algorithm for approximating \( k \)-turn games for some \( k > 2 \).

2.3 Interactive proofs with competing provers

An interactive proof with competing provers consists of a conversation between a randomized polynomial-time verifier and two computationally unbounded provers on some input \( x \). One of the provers—the yes-prover—tries to convince the verifier to accept \( x \), while the other—the no-prover—tries to convince the verifier to reject \( x \). The analogy to competitive games is obvious, and for this reason such interactive proofs are also called refereed games.

A decision problem \( L \) is said to admit a classical refereed game if there exists a randomized polynomial-time referee such that: (i) if \( x \) is a yes-instance of \( L \) then the yes-prover can convince the verifier to accept with probability at least \( 2/3 \) regardless of the no-prover’s strategy, and (ii) if \( x \) is a no-instance of \( L \) then the no-prover can convince the verifier to reject with probability at least \( 2/3 \) regardless of the yes-prover’s strategy. The complexity class of problems that admit classical refereed games is denoted \( \text{RG} \). Polynomial-time algorithms for game trees imply \( \text{RG} \subseteq \text{EXP} \) \cite{KM92,KMvS94}. The reverse containment follows from hardness of approximation for refereed games \cite{FK97}, yielding the characterization \( \text{RG} = \text{EXP} \).

Quantum refereed games are defined similarly except that the referee is a polynomial-time quantum computer who exchanges quantum information with the provers. The class of problems that admit quantum refereed games is denoted \( \text{QRG} \). The polynomial-time algorithm for quantum games implies \( \text{QRG} \subseteq \text{EXP} \).
Prior work on classical refereed games then implies
\[ \text{QRG} = \text{RG} = \text{EXP}, \]
which is the competing-prover analogy of the well-known collapse \( \text{QIP} = \text{IP} = \text{PSPACE} \) for single-prover interactive proofs [LFKN92, Sha92, JJUW10, Wu10a].

For each positive integer \( k \) the complexity classes of problems that admit \( k \)-turn classical and quantum refereed games are denoted \( \text{RG}(k) \) and \( \text{QRG}(k) \), respectively. The results of Fortnow et al. tell us that \( \text{RG}(1) \) is essentially a randomized version of \( \text{SZK} \). The parallel algorithm for one-turn quantum games immediately implies \( \text{QRG}(1) \subseteq \text{PSPACE} \) [JW09]. For two-turn games, Feige and Kilian proved \( \text{RG}(2) = \text{PSPACE} \) [FK97] and the complexity of \( \text{QRG}(2) \) is an open question of Ref. [JJUW10] that is solved in the present paper. The exact complexity of \( \text{RG}(k) \) and \( \text{QRG}(k) \) for all other \( k \) is not known.

Double quantum interactive proofs with exactly two messages per player have been called short quantum games. The associated complexity class is denoted \( \text{SQG} \); it trivially contains \( \text{QRG}(2) \) and is known to contain \( \text{QIP} \) [GW05]. The importance of this class was diminished by the proof of \( \text{QIP} = \text{PSPACE} \). The present paper establishes that \( \text{SQG} \) is also equal to \( \text{PSPACE} \).

Earlier we defined \( \text{DIP} \) and \( \text{DQIP} \) to be the complexity classes of decision problems that admit classical and quantum double interactive proofs, respectively. These classes appear quite large at first glance. For example, it follows immediately from first principles that \( \text{DQIP} \) contains \( \text{SQG} \) (and hence \( \text{QRG}(2) \)) as well as both \( \text{QIP} \) and its complement \( \text{co-QIP} \). That this class should collapse to \( \text{PSPACE} \) could be construed as surprising.

Our results also illustrate a difference in the role of public randomness between single-prover interactive proofs and competing-prover interactive proofs. Any classical interactive proof with single prover can be simulated by another public coin interactive proof where the verifier’s messages to the prover consist entirely of uniformly random bits and the verifier uses no other randomness [GS89]. (Public coin single-prover interactive proofs are also known as Arthur-Merlin games.) Extending the notion of public coin interaction to refereed games, it is easy to see that an arbitrary public-coin refereed game with any number of turns can be simulated by a double interactive proof.\(^5\) We therefore have that the public-coin version of \( \text{RG} \) is a subset of \( \text{DIP} \), which we now know is equal to \( \text{PSPACE} \). Thus, by contrast to the single-prover case where we have \( \text{public-coin-IP} = \text{IP} \), in the competing-prover case we have \( \text{public-coin-RG} \neq \text{RG} \) unless \( \text{PSPACE} = \text{EXP} \).

### 3 Preliminaries

We assume familiarity with standard concepts from quantum information [NC00, Wat08]. This section provides a table describing our notation in Figure 1 followed by a brief survey of parallel computation in Section 3.1 (also known as NC computation). Two rarer but nonetheless simple and fundamental concepts from quantum information are also discussed: the preservation of subsystem fidelity in Section 3.2 and the Bures angle in Section 3.3.

---

\(^3\) The class we call \( \text{RG}(2) \) is called \( \text{RG}(1) \) by Feige and Kilian. This conflict in notation stems from the fact that we measure the length of a game in turns, whereas those authors measure a game in rounds of messages. This switch of notation was instigated by Jain and Watrous [JW09], who required a convenient symbol for one-turn refereed games.

\(^4\) Proof sketch: As the referee’s questions to a player are uniformly random, they cannot depend on prior responses from the other player and can therefore be reordered so that all messages with one player are exchanged before any messages are exchanged with the other.
Finite-dimensional complex vector spaces. \( \mathcal{X} \mathcal{Y} \) is shorthand for \( \mathcal{X} \otimes \mathcal{Y} \).

\[ \mathbb{L}(\mathcal{X}) \] The (complex) space of all linear operators \( A : \mathcal{X} \to \mathcal{X} \).

\[ I_{\mathcal{X}} \] The identity operator acting on \( \mathcal{X} \).

\[ \text{Dens}(\mathcal{X}) \] The compact convex set of all density operators within \( \mathbb{L}(\mathcal{X}) \).

\[ \text{Meas}(\mathcal{X}) \] The compact convex set of all measurement operators within \( \mathbb{L}(\mathcal{X}) \). A measurement operator is a positive semidefinite operator \( M \) with \( M \preceq I_{\mathcal{X}} \).

\[ \mathbb{U}(\mathcal{X}) \] The set of all unitary operators within \( \mathbb{L}(\mathcal{X}) \).

\( A^* \) The adjoint of an operator \( A : \mathcal{X} \to \mathcal{Y} \), which has the form \( A^* : \mathcal{Y} \to \mathcal{X} \).

\[ \langle A, B \rangle \] The standard inner product between \( A, B : \mathcal{X} \to \mathcal{Y} \). Defined by \( \langle A, B \rangle = \text{Tr}(A^* B) \).

Figure 1: Notation and terminology

### 3.1 NC and parallel matrix computations

We denote by \( \text{NC} \) the class of promise-problems that admit efficient parallel algorithms. Since every matrix is of exponential size in term of the input size in quantum computation, we also need the scaled up version of \( \text{NC} \), namely \( \text{NC}(\text{poly}) \). It is through the relation \( \text{NC}(\text{poly}) = \text{PSPACE} \) that we can prove polynomial-space upper bound.

There are many nice facts about these classes that we will make use of in our discussions. The first is that the functions in these classes compose nicely. Thus one can design efficient parallel algorithms for each part of the whole problem and then compose them to get an efficient parallel algorithm for the whole problem as long as the number of parts is bounded. Another useful fact is that many computations involving matrices, such as singular value decompositions and matrix exponentials, can be performed by \( \text{NC} \) algorithms (see the survey [vzG93]). Moreover, the adapted \( \text{NC} \) algorithms especially for the implementation of matrix multiplicative weight update method are also known before. (See Refs. [JUW09, JJUW10] or [GW10, Facts 2–5].) It remains to show some special operations (e.g., computing a purification of a mixed state and computing a unitary that maps one purification to another) in our algorithm can also be implemented efficiently in parallel. All \( \text{NC} \) algorithms for these extra operations can be found in the previous version of this paper [GW10, Lemmas 2–4].

The last concern about the implementation of those parallel algorithms is the precision issue. This issue raises when precision of some of the computations must be truncated because of the irrational number involved. Similar issue might also happen when one compose approximate computations. Fortunately, all the computations involved in our algorithm can be made either exact or approximate to high precision in \( \text{NC} \). Furthermore, we will assume all these computations can be made exact in our proof and refer curious readers to the details of handling these issues in the previous version of this paper [GW10, Appendix B].

### 3.2 Preservation of subsystem fidelity

Consider the following property of the fidelity function, which we call the *preservation of subsystem fidelity*: if \( \sigma, \sigma' \) are states of a quantum system with fidelity \( F(\sigma, \sigma') \) and \( \rho \) is any state of a larger system consistent with \( \sigma \) then it is always possible to find \( \rho' \) consistent with \( \sigma' \) such that \( F(\rho, \rho') = F(\sigma, \sigma') \).

A formal construction of such a \( \rho' \) appears in Jain *et al.* [JUW09]. Since their construction consists entirely of elementary matrix operations mentioned above, such construction hence admits a parallel algorithm that takes as input \( \sigma, \sigma', \rho \) and produces \( \rho' \) in time bounded by a polylogarithm in the dimensions of the input matrices \( \sigma, \sigma', \rho \).
Proposition 2 (Preservation of subsystem fidelity—see Ref. [JUW09, Lemma 7.2], [GW10, Lemma 2]).
Let \( \sigma, \sigma' \in \text{Dens}(\mathcal{V}) \) and \( \rho \in \text{Dens}(A\mathcal{V}) \) be density operators with \( \text{Tr}_A(\rho) = \sigma \). There exists a density operator \( \rho' \in \text{Dens}(A\mathcal{V}) \) with \( \text{Tr}_A(\rho') = \sigma' \) and \( F(\rho, \rho') = F(\sigma, \sigma') \). Moreover \( \rho' \) can be computed efficiently in parallel given \( \sigma, \sigma', \rho \).

3.3 The Bures angle

The Bures angle or simply the angle \( A(\rho, \xi) \) between quantum states \( \rho, \xi \) is defined by
\[
A(\rho, \xi) \triangleq \arccos F(\rho, \xi).
\]
The angle is a metric on quantum states, meaning that it is nonnegative, equals zero only when \( \rho = \xi \), and obeys the triangle inequality [NC00]. Moreover, the angle is contractive, so that
\[
A(\Phi(\rho), \Phi(\xi)) \leq A(\rho, \xi)
\]
for any quantum channel \( \Phi \). The Fuchs-van de Graaf Inequalities establish a relationship between the fidelity and trace norm [FvdG99]. The inequalities are
\[
1 - F(\rho, \xi) \leq \frac{1}{2} \| \rho - \xi \|_{\text{Tr}} \leq \sqrt{1 - F(\rho, \xi)^2}.
\]
These inequalities can be used to derive a relationship between \( A(\rho, \xi) \) and \( \| \rho - \xi \|_{\text{Tr}} \). For example,

Proposition 3 (Relationship between trace norm and Bures angle). For all density matrices \( \rho, \xi \) it holds that
\[
\frac{1}{2} \| \rho - \xi \|_{\text{Tr}} \leq A(\rho, \xi) \leq \sqrt{\frac{\pi}{2}} \| \rho - \xi \|_{\text{Tr}}.
\]

Proof. The upper bound follows immediately from Fuchs-van de Graaf:
\[
\frac{1}{2} \| \rho - \xi \|_{\text{Tr}} \leq \sqrt{1 - \cos A(\rho, \xi)^2} = \sin A(\rho, \xi) \leq A(\rho, \xi)
\]
where we used the identity \( \sin x \leq x \) for all \( x \geq 0 \).

To obtain the lower bound we employ the identity \( \cos x \leq 1 - x^2/\pi \) for \( x \in [0, \pi/2] \), which can be verified using basic calculus. Then we have
\[
\frac{1}{2} \| \rho - \xi \|_{\text{Tr}} \geq 1 - \cos A(\rho, \xi) \geq \frac{A(\rho, \xi)^2}{\pi}
\]
from which the proposition follows.

4 Double quantum interactive proofs

A double quantum interactive proof is completely specified by a referee, which consists of a tuple \( R = (|\psi\rangle, V_1, \ldots, V_{a+b}, \Pi) \) where
\[
|\psi\rangle \in \mathcal{CV}
\]
is a pure state
\[
V_1, \ldots, V_{a+b} \in U(\mathcal{CV})
\]
are unitary operators
\[
\Pi \in \text{Meas}(\mathcal{CV})
\]
is a projective measurement operator
Figure 2: An illustration of a double quantum interactive proof in which the referee \( R = (|\psi\rangle, V_1, \ldots, V_6, \Pi) \) exchanges \( a = 3 \) rounds of messages with Alice followed by \( b = 3 \) rounds of messages with Bob before performing the measurement \( \{\Pi, I - \Pi\} \) and announcing a winner. The register \( C \) is a message register to be exchanged among the referee and players. The registers \( V, A, \) and \( B \) are private memory registers for the referee, Alice and Bob, respectively. Any choice of \( A_1, A_2, A_3 \) and \( B_1, B_2, B_3 \) induces a state \( \rho \) and a measurement operator \( P \) as indicated. Bob’s winning probability is given by \( \langle \rho, P \rangle = \text{Tr}(\rho P) \).

The spaces \( C, V \) correspond to registers \( C, V \). The register \( C \) is a message register to be exchanged with the players and the register \( V \) is a private memory register for the referee.

The actions of the players during each round of interaction are specified by unitary operators acting upon the message register \( C \) and a private memory register for that player. In particular, Alice’s actions are specified by unitary operators \( A_1, \ldots, A_a \in \mathcal{U}(CA) \) where the space \( A \) corresponds to the private memory register \( A \) for Alice. Similarly, Bob’s actions are specified by unitary operators \( B_1, \ldots, B_b \in \mathcal{U}(CB) \) where the space \( B \) corresponds to the private memory register \( B \) for Bob.

The game proceeds as suggested by Figure 2 and is described as follows:

1. The referee prepares the registers \( (C, V) \) in the pure state \( |\psi\rangle \). The players’ private registers \( A, B \) are both initialized to the pure state \( |0\rangle \).

2. For \( i = 1, \ldots, a \): The register \( C \) is sent to Alice, who applies \( A_i \) to the registers \( (C, A) \). The register \( C \) is then returned to the referee, who applies \( V_i \) to the registers \( (C, V) \).

3. For \( i = 1, \ldots, b \): The register \( C \) is sent to Bob, who applies \( B_i \) to the registers \( (C, B) \). The register \( C \) is then returned to the referee, who applies \( V_{a+i} \) to the registers \( (C, V) \).

4. The referee applies the binary-valued measurement \( \{\Pi, I - \Pi\} \) on the registers \( (C, V) \) with the outcome associated with \( \Pi \) indicating victory for Bob.

Basic quantum formalism tells us that if Alice and Bob act according to \( (A_1, \ldots, A_a) \) and \( (B_1, \ldots, B_b) \), respectively, then the probability with which Bob is declared the winner is given by

\[
\text{Pr}[\text{Bob wins} \mid (A_1, \ldots, A_a), (B_1, \ldots, B_b)] = \| \Pi V_{a+b} B_b V_{a+b-1} B_{b-1} \cdots B_1 V_a A_a V_{a-1} A_{a-1} \cdots A_2 V_1 A_1 |\psi\rangle \|^2 .
\]

(For clarity we have suppressed numerous tensors with identity and the initial states \( |0\rangle \) of the players’ private memory registers.)
Of course, Bob wishes to maximize this quantity while Alice wishes to minimize this quantity. It follows immediately from the min-max theorem for zero-sum quantum games [GW07] that every double quantum interactive proof with referee $R$ has a value $\lambda(R)$ given by

$$
\lambda(R) = \min_{(A_1, \ldots, A_a)} \max_{(B_1, \ldots, B_b)} \Pr[\text{Bob wins} \mid (A_1, \ldots, A_a), (B_1, \ldots, B_b)]
$$

$$
= \max_{(B_1, \ldots, B_b)} \min_{(A_1, \ldots, A_a)} \Pr[\text{Bob wins} \mid (A_1, \ldots, A_a), (B_1, \ldots, B_b)]
$$

where the minima are over all private spaces $\mathcal{A}$ for Alice and all unitaries $A_1, \ldots, A_a \in U(CA)$ and the maxima are over all private spaces $\mathcal{B}$ for Bob and all unitaries $B_1, \ldots, B_b \in U(CB)$. In particular, for every double quantum interactive proof with referee $R$ there exist optimal actions $(A_1^*, \ldots, A_a^*)$ for Alice and $(B_1^*, \ldots, B_b^*)$ for Bob such that

$$
\Pr[\text{Bob wins} \mid (A_1^*, \ldots, A_a^*), (B_1, \ldots, B_b)] \leq \lambda(R) \quad \text{for all } (B_1, \ldots, B_b),
$$

$$
\Pr[\text{Bob wins} \mid (A_1, \ldots, A_a), (B_1^*, \ldots, B_b^*)] \geq \lambda(R) \quad \text{for all } (A_1, \ldots, A_a).
$$

From an operational perspective, the min-max expression (4) for $\lambda(R)$ in terms of unitaries $(A_1, \ldots, A_a)$ and $(B_1, \ldots, B_b)$ is natural and intuitive. However, this expression does not lend itself well to the MMW, which is designed to solve min-max problems over domains of density operators—not tuples of unitaries. To address this problem we derive an alternate expression for $\lambda(R)$ that is more amenable to the MMW.

To this end, for any $(A_1, \ldots, A_a)$ and $(B_1, \ldots, B_b)$ let $\rho$ be the reduced state of the registers $(C, V)$ immediately after Alice’s final unitary is applied and let $P$ be the measurement operator on $(C, V)$ obtained by bundling the referee-Bob interaction into a single measurement operator as suggested by Figure 2. The expression (3) for Bob’s probability of victory can be rewritten in terms of $\rho, P$ as

$$
\Pr[\text{Bob wins} \mid (A_1, \ldots, A_a), (B_1, \ldots, B_b)] = \langle \rho, P \rangle.
$$

Similarly, the expression (4) for $\lambda(R)$ can be rewritten as

$$
\lambda(R) = \min_{\rho \in A(R)} \max_{P \in P(R)} \langle \rho, P \rangle = \max_{P \in P(R)} \min_{\rho \in A(R)} \langle \rho, P \rangle
$$

where the sets $A(R) \subset \text{Dens}(CV)$ and $P(R) \subset \text{Meas}(CV)$ are given by

$$
A(R) = \{ \text{Tr}_{\mathcal{A}}(|\phi\rangle\langle\phi|) : |\phi\rangle = A_a V_a^{-1} A_a^{-1} \cdots A_2 V_1 A_1 |\psi\rangle \text{ for some } (A_1, \ldots, A_a) \}, \tag{5}
$$

$$
P(R) = \{ U^* P U : U = V_{a+b} B_b V_{a+b-1} B_{b-1} \cdots B_1 V_a \text{ for some } (B_1, \ldots, B_b) \}. \tag{6}
$$

At this point, we have rewritten $\lambda(R)$ so that the set of all possible actions available to Alice has been identified with a subset $A(R)$ of density operators, as desired. (Bob’s actions will be addressed later.) However, the MMW is designed to solve min-max problems whose domain is the entire set of density operators. In the next section we present a new adaptation of the MMW that applies to min-max problems on strict subsets of density operators. We will see that this adaptation yields a parallel algorithm for the above formulation of $\lambda(R)$.

## 5 Rounding theorem for a relaxed min-max problem

In this section we define a new min-max expression $\mu_\varepsilon(R)$ that approximates the desired quantity $\lambda(R)$ in the limit as $\varepsilon$ approaches zero. The new expression is a relaxation of $\lambda(R)$ that is more amenable to the
The states $\rho_1, \rho_2, \rho_3$ are a transcript of the referee’s conversation with Alice. It follows easily from the unitary equivalence of purifications that a triple $(\rho_1, \rho_2, \rho_3)$ is a valid transcript if and only if it obeys the recursive relation $\text{Tr}_C(\rho_{i+1}) = \text{Tr}_C(V_i \rho_i V_i^*)$ for $i = 0, 1, 2$ where $V_0 = I$.

We prove a “rounding theorem” by which near-optimal points for $\lambda(R)$ are efficiently obtained from near-optimal points for $\mu_\varepsilon(R)$.

We begin in Section 5.1 with a review of the consistency conditions for transcripts, which motivate our definition of $\mu_\varepsilon(R)$. A formal definition of $\mu_\varepsilon(R)$ and proof of the rounding theorem appear in Section 5.2. Section 5.3 contains the proof of a technical lemma and its corollary that is used in the rounding theorem. Our use of the Bures metric occurs in the proof of this lemma.

### 5.1 Consistency conditions for Alice

The set $A(R)$ of density operators that represent admissible actions for Alice as defined in (5) is unwieldy. In order to optimize over this set we begin by writing it not in terms of unitaries $A_1, \ldots, A_a$ but in terms of states $\rho_1, \ldots, \rho_a$ that represent a transcript of the referee’s conversation with Alice. Such a transcript is depicted in Figure 3. It is straightforward to use the unitary equivalence of purifications to characterize those density matrices which constitute valid transcripts. This characterization was first noted by Kitaev [Kit02] and a formal proof can be found in Ref. [Gut05].

**Proposition 4** (Kitaev’s consistency conditions—see Ref. [Gut05]). Let $R = (|\psi\rangle, V_1, \ldots, V_{a+b}, \Pi)$ be a referee and let $A(R)$ be the set of admissible states for Alice as defined in Eq. (5). A given state $\rho$ is an element of $A(R)$ if and only if there exist $\rho_1, \ldots, \rho_a \in \text{Dens}(CV)$ with $\rho_a = \rho$ and

$$\text{Tr}_C(\rho_{i+1}) = \text{Tr}_C(V_i \rho_i V_i^*) \quad \text{for } i = 0, \ldots, a - 1$$

where we have written $V_0 = I$ and $\rho_0 = |\psi\rangle \langle \psi|$ for convenience.

Any states $\rho_1, \ldots, \rho_a$ obeying the consistency condition of Proposition 4 are said to be consistent with $R$. It therefore follows from Proposition 4 that the value $\lambda(R)$ of the game may be written

$$\lambda(R) = \min_{(\rho_1, \ldots, \rho_a)} \max_{P \in \mathcal{P}(R)} \langle \rho_a, P \rangle.$$

(7)
5.2 A relaxed min-max problem and a rounding theorem

Define the relaxation $\mu_\varepsilon(R)$ of $\lambda(R)$ by

$$
\mu_\varepsilon(R) \equiv \min_{(\rho_1, \ldots, \rho_a)} \max_{P \in \mathcal{P}(R)} \langle \rho_\alpha, P \rangle + \frac{a-1}{\varepsilon} \sum_{i=0}^{a-1} \langle \text{Tr}_C(\rho_{i+1}) - \text{Tr}_C(V_i \rho_i V_i^*), \Pi_{i+1} \rangle
$$

$$
= \min_{(\rho_1, \ldots, \rho_a)} \max_{P \in \mathcal{P}(R)} \langle \rho_\alpha, P \rangle + \frac{a-1}{\varepsilon} \sum_{i=0}^{a-1} \frac{1}{2} \| \text{Tr}_C(\rho_{i+1}) - \text{Tr}_C(V_i \rho_i V_i^*) \|_{\text{Tr}}
$$

Here the minimum is taken over all density operators $\rho_1, \ldots, \rho_a \in \mathcal{D}(\mathcal{C} \mathcal{V})$ and the maximum over all $P \in \mathcal{P}(R)$ and over all measurement operators $\Pi_1, \ldots, \Pi_a \in \mathcal{M}(\mathcal{V})$. The second equality follows immediately from the identity $\frac{1}{\varepsilon} \| \rho - \xi \|_{\text{Tr}} = \max_{0 \leq \Pi \leq I} \langle \rho - \xi, \Pi \rangle$ which holds for all density operators $\rho, \xi$.

Notice that the minimum in the definition of $\mu_\varepsilon(R)$ is taken over all density operators, not just those consistent with $R$. Each term in the summation serves to penalize any violation of consistency in the choice of $\rho_1, \ldots, \rho_a$ by adding the magnitude of that violation to Bob’s probability of victory. The $a/\varepsilon$ factor amplifies the penalty so as to remove incentive for Alice to select an inconsistent course of action. Indeed, it is clear that

$$
\lim_{\varepsilon \to 0} \mu_\varepsilon(R) = \lambda(R).
$$

The following “rounding” theorem establishes a specific rate of convergence for this limit and a means by which near-optimal points for $\lambda(R)$ are efficiently computed from near-optimal points for $\mu_\varepsilon(R)$.

Before proving Theorem 5 we need some terminology. Consider any equilibrium value $\lambda$ of the form

$$
\lambda = \min_{a \in \mathcal{A}} \max_{b \in \mathcal{B}} f(a, b) = \max_{b \in \mathcal{B}} \min_{a \in \mathcal{A}} f(a, b)
$$

A pair $(\tilde{a}, \tilde{b})$ is called $\delta$-optimal for $\lambda$ if

$$
\max_{b \in \mathcal{B}} f(\tilde{a}, b) \leq \lambda + \delta \quad \text{and} \quad \min_{a \in \mathcal{A}} f(a, \tilde{b}) \geq \lambda - \delta.
$$

Elements that are $0$-optimal are simply called optimal. Any value $\bar{\lambda}$ is called $\delta$-optimal for $\lambda$ if $|\bar{\lambda} - \lambda| \leq \delta$.

**Theorem 5** (Rounding theorem). The following hold for any referee $R$ and any $\varepsilon, \delta > 0$:

1. $\lambda(R) \geq \mu_\varepsilon(R) > \lambda(R) - \varepsilon$.

2. If $(\tilde{P}, \tilde{\Pi}_1, \ldots, \tilde{\Pi}_a)$ is $\delta$-optimal for $\mu_\varepsilon(R)$ then $\tilde{P} \in \mathcal{P}(R)$ is also $(\delta + \varepsilon)$-optimal for $\lambda(R)$.

3. If $(\tilde{\rho}_1, \ldots, \tilde{\rho}_a)$ is $\delta$-optimal for $\mu_\varepsilon(R)$ then there exists density operators $(\tilde{\rho}'_1, \ldots, \tilde{\rho}'_a)$ consistent with $R$ that can be computed in parallel time $O(a \text{ polylog}(\dim(\mathcal{C} \mathcal{V})))$ such that $\tilde{\rho}'_a \in \mathcal{A}(R)$ is $(\delta + \varepsilon)$-optimal for $\lambda(R)$.

**Proof.** We begin with item 1. The first inequality is easy: let $(\rho^\lambda_1, \ldots, \rho^\lambda_a)$ achieve the minimum for $\lambda(R)$ in Eq. (7) among all density operators consistent with $R$. Let $(P^\mu_1, \ldots, \Pi^\mu_a)$ achieve the maximum for $\mu_\varepsilon(R)$. Then we have

$$
\lambda(R) \geq \langle \rho^\lambda_a, P^\mu \rangle = \langle \rho^\lambda_a, P^\mu \rangle + \frac{a-1}{\varepsilon} \sum_{i=0}^{a-1} \langle \text{Tr}_C(\rho^\lambda_{i+1}) - \text{Tr}_C(V_i \rho_i V_i^*), \Pi_{i+1} \rangle \geq \mu_\varepsilon(R).
$$
The second inequality is more difficult. Choose any density operators \( \rho_1, \ldots, \rho_a \). By Lemma \[7\] (the statement of which appears below in Section \[5.3\]) there exist density operators \( \rho_1', \ldots, \rho_a' \) consistent with \( R \) such that

\[
\frac{1}{2}\|\rho_a - \rho_a'\|_{\text{Tr}} < \frac{1}{\varepsilon} \sum_{i=0}^{a-1} \frac{1}{2}\|\text{Tr}_C(\rho_{i+1}) - \text{Tr}_C(V_i \rho_i V_i^*)\|_{\text{Tr}}.
\]

For any measurement operator \( P \) we have

\[
\langle \rho_a, P \rangle = \langle \rho_a', P \rangle + \langle \rho_a - \rho_a', P \rangle \geq \langle \rho_a', P \rangle - \frac{1}{2}\|\rho_a - \rho_a'\|_{\text{Tr}}
\]

\[
> \langle \rho_a', P \rangle - \varepsilon - \frac{1}{\varepsilon} \sum_{i=0}^{a-1} \frac{1}{2}\|\text{Tr}_C(\rho_{i+1}) - \text{Tr}_C(V_i \rho_i V_i^*)\|_{\text{Tr}}.
\]

The inequality (8) will be employed several times throughout the rest of this proof.

To complete the proof of item 1 let \( \rho_1, \ldots, \rho_a \) be consistent with \( \lambda(R) \). Employing (8) for the choices \( \rho_1, \ldots, \rho_a \) and \( P = P^\lambda \) we obtain

\[
\mu_\varepsilon(R) \geq \langle \rho_a', P^\lambda \rangle + \frac{1}{\varepsilon} \sum_{i=0}^{a-1} \frac{1}{2}\|\text{Tr}_C(\rho_{i+1}) - \text{Tr}_C(V_i \rho_i V_i^*)\|_{\text{Tr}} > \langle \rho_a', P \rangle - \varepsilon \geq \lambda(R) - \varepsilon,
\]

which establishes the desired lower bound on \( \mu_\varepsilon(R) \).

Item 2 follows easily from the above construction. For any \( P \in P(R) \) we may substitute \( \rho_1, \ldots, \rho_a \) into (8) and use the fact that \( \lambda(R) \geq \mu_\varepsilon(R) \) to obtain

\[
\lambda(R) + \delta \geq \mu_\varepsilon(R) + \delta \geq \langle \rho_a, P \rangle + \frac{1}{\varepsilon} \sum_{i=0}^{a-1} \frac{1}{2}\|\text{Tr}_C(\rho_{i+1}) - \text{Tr}_C(V_i \rho_i V_i^*)\|_{\text{Tr}} \geq \langle \rho_a', P \rangle - \varepsilon,
\]

from which it follows that \( \rho_a' \) is \((\delta + \varepsilon)\)-optimal for \( \lambda(R) \).

Item 2 can be proven using the fact that \( \lambda(R) - \varepsilon < \mu_\varepsilon(R) \) without making further use of the above construction. For any \( \rho_1, \ldots, \rho_a \) consistent with \( R \) we have

\[
\lambda(R) - \varepsilon - \delta < \mu_\varepsilon(R) - \delta \leq \langle \rho_a, \bar{P} \rangle,
\]

from which it follows that \( \bar{P} \) is also \((\delta + \varepsilon)\)-optimal for \( \lambda(R) \).

5.3 Rounding lemma for obtaining consistent states

In this subsection we prove a technical lemma and its corollary that appeared in the proof of Theorem 5. Given any states \( \rho_1, \ldots, \rho_a \), this lemma asserts that these states can be “rounded” to valid transcript states \( \rho_1', \ldots, \rho_a' \) in such a way that the distance between the final states \( \rho_a \) and \( \rho_a' \) is bounded by a function of the extent to which \( \rho_1, \ldots, \rho_a \) violate the consistency condition of Proposition 4. The proof of this lemma is interesting because it provides a nontrivial application of the Bures angle.

Lemma 6. For any referee \( R = (|\psi\rangle, V_1, \ldots, V_{a+b}, \Pi) \) and any \( \rho_1, \ldots, \rho_a \in \text{Dens}(CV) \) there exist \( \rho_1', \ldots, \rho_a' \in \text{Dens}(CV) \) consistent with \( R \) such that

\[
A(\rho_a, \rho_a') \leq \sum_{i=0}^{a-1} A(\text{Tr}_C(\rho_{i+1}), \text{Tr}_C(V_i \rho_i V_i^*)).
\]

Moreover, \( \rho_1', \ldots, \rho_a' \) can be computed in parallel time \( O(\text{polylog}(\dim(CV))) \).
Proof. Define $\rho'_1, \ldots, \rho'_a$ recursively as follows. Let $\rho'_0 = \rho_0$. For each $i = 0, \ldots, a - 1$ by the preservation of subsystem fidelity (Proposition \ref{prop:subsystem-fidelity}) there exists $\rho'_{i+1}$ (which can be efficiently computed) with $\text{Tr}_C(\rho'_{i+1}) = \text{Tr}_C(V_i \rho'_i V_i^*)$ and

$$A(\rho_{i+1}, \rho'_{i+1})$$



(preservation of fidelity)

$$\leq A(\text{Tr}_C(\rho_{i+1}), \text{Tr}_C(V_i \rho_i V_i^*)) + A(\text{Tr}_C(V_i \rho'_i V_i^*), \text{Tr}_C(V'_i \rho'_i V'_i^*))$$

(triangle inequality)

$$\leq A(\text{Tr}_C(\rho_{i+1}), \text{Tr}_C(V_i \rho_i V_i^*)) + A(\rho_i, \rho'_i)$$

(contractivity)

The lemma now follows inductively from the fact that $A(\rho_0, \rho'_0) = 0$. \hfill $\square$

**Lemma 7.** For any $\varepsilon > 0$ the bound in Lemma \ref{lem:trace-bound} can be written in terms of the trace norm as

$$\frac{1}{2} \| \rho_a - \rho'_a \|_\text{Tr} < \varepsilon + \frac{a}{\varepsilon} \sum_{i=0}^{a-1} \frac{1}{2} \| \text{Tr}_C(\rho_{i+1}) - \text{Tr}_C(V_i \rho_i V_i^*) \|_\text{Tr}$$

Proof. It follows immediately from Lemma \ref{lem:trace-bound} and Proposition \ref{prop:tr-bures}(Relationship between trace norm and Bures angle) that

$$\frac{1}{2} \| \rho_a - \rho'_a \|_\text{Tr} \leq \sum_{i=0}^{a-1} \sqrt{\frac{\pi}{2} \| \text{Tr}_C(\rho_{i+1}) - \text{Tr}_C(V_i \rho_i V_i^*) \|_\text{Tr}}.$$ 

The lemma then follows from the fact that $\sqrt{\frac{\pi}{2}} x < \frac{1}{2\delta} x + \delta$ for all $x \geq 0$ and all $\delta > 0$. \hfill $\square$

6 The MMW oracle-algorithm for double quantum interactive proofs

In this section we describe an efficient parallel oracle-algorithm that approximates $\lambda(R)$ to arbitrary precision. (It is a simple matter to modify this algorithm so as to also produce unitaries $A_1, \ldots, A_a$ for Alice and $B_1, \ldots, B_b$ for Bob that are arbitrarily close to optimal. See Section \ref{sec:contractibility}.)

We begin in Section \ref{subsec:preliminaries} with formal statements of the problem solved by our algorithm and the oracle it requires, as well as a brief review of the relevant facts concerning the MMW. Our algorithm and its analysis are provided in Section \ref{subsec:algorithm}. In Section \ref{subsec:applications} we note that our algorithm can be used to approximate the solution of a semidefinite program on consistent density matrices efficiently in parallel, from which we recover the SDP \ref{eq:sdp} and the direct proof of $\text{QIP} = \text{PSPACE}$ mentioned at the beginning of this paper.

6.1 Preliminaries: formal statement of the problem, review of the MMW

Precise statements of the problem solved by our algorithm and the oracle it requires are given below. For matrix inputs, each entry is written explicitly. The real and complex parts of all numbers are written as rational numbers in binary.

**Problem 1** (Approximation of $\lambda(R)$).

**Input:** A referee $R = (|\psi\rangle, V_1, \ldots, V_{a+b}, \Pi)$ and an accuracy parameter $\delta > 0$.

**Oracle:** Weak optimization for $P(R)$. (See Problem \ref{prob:optimization} below.)

**Output:** A number $\tilde{\lambda}$ with $|\tilde{\lambda} - \lambda(R)| < \delta$. 

15
Theorem 8 (Algorithmic form). Our statement of this theorem is somewhat nonstandard: the result is usually presented in the form of an algorithm, whereas our presentation is purely mathematical. However, a cursory examination of the literature—say, Kale’s thesis [Kal07, Chapter 3]—reveals that our mathematical formulation is equivalent to the more conventional algorithmic form.

Theorem 8 (Multiplicative weights update method—see Ref. [Kal07, Theorem 10]). Fix $\gamma \in (0, 1/2)$. Let $M^{(1)}, \ldots, M^{(T)}$ be arbitrary $D \times D$ “loss” matrices with $0 \preceq M^{(t)} \preceq \alpha I$. Let $W^{(1)}, \ldots, W^{(T)}$ be $D \times D$ “weight” matrices given by

$$W^{(1)} = I \quad \text{and} \quad W^{(t+1)} = \exp \left( -\gamma \left( M^{(1)} + \cdots + M^{(t)} \right) \right).$$

Let $\rho^{(1)}, \ldots, \rho^{(T)}$ be density operators obtained by normalizing each $W^{(1)}, \ldots, W^{(T)}$ so that $\rho^{(t)} = W^{(t)}/\text{Tr}(W^{(t)})$. For all density operators $\rho$ it holds that

$$\frac{1}{T} \sum_{t=1}^{T} \langle \rho^{(t)}, M^{(t)} \rangle \leq \left\langle \rho, \frac{1}{T} \sum_{t=1}^{T} M^{(t)} \right\rangle + \alpha \left( \gamma + \frac{\ln D}{\gamma T} \right).$$

Note that Theorem 8 holds for all choices of loss matrices $M^{(1)}, \ldots, M^{(T)}$, including those for which each $M^{(t)}$ is chosen adversarially based upon $W^{(1)}, \ldots, W^{(t)}$. This adaptive selection of loss matrices is typical in implementations of the MMW.

6.2 Statement and analysis of the MMW oracle-algorithm

Let $\epsilon > 0$ and consider the linear mapping

$$f_{R,\epsilon} : \langle \rho_1, \ldots, \rho_a \rangle \mapsto \left( \rho_a, \frac{a}{\epsilon} \left[ \text{Tr}_C(\rho_a) - \text{Tr}_C(V_{a-1}\rho_{a-1}V_{a-1}^*) \right], \ldots, \frac{a}{\epsilon} \left[ \text{Tr}_C(\rho_1) - \text{Tr}_C(V_1\rho_1V_1^*) \right], \frac{a}{\epsilon} \left[ \text{Tr}_C(\rho_1) - \text{Tr}(\rho_1) \text{Tr}(|\psi\rangle\langle\psi|) \right] \right).$$

It is clear that

$$\mu_\epsilon(R) = \min_{(\rho_1, \ldots, \rho_a)} \max_{P \in \mathcal{P}(R)} \langle f_{R,\epsilon}(\rho_1, \ldots, \rho_a), (P, \Pi_a, \ldots, \Pi_1) \rangle.$$

It is tedious but straightforward to compute the adjoint map $f_{R,\epsilon}^*$:

$$f_{R,\epsilon}^* : \langle P, \Pi_a, \ldots, \Pi_1 \rangle \mapsto \left( P + \frac{a}{\epsilon} \Pi_a \otimes I_C, \frac{a}{\epsilon} \left[ \Pi_{a-1} \otimes I_C - V_{a-1}^*(\Pi_a \otimes I_C)V_{a-1} \right], \ldots, \frac{a}{\epsilon} \left[ \Pi_2 \otimes I_C - V_2^*(\Pi_3 \otimes I_C)V_2 \right], \frac{a}{\epsilon} \left[ \Pi_1 \otimes I_C - V_1^*(\Pi_2 \otimes I_C)V_1 - \langle \psi|\Pi_1|\psi\rangle I_{CV} \right] \right).$$
The statement of our MMW algorithm in Figure 4 employs this formula for the adjoint.

**Proposition 9.** The oracle-algorithm presented in Figure 4 approximates $\lambda(R)$ to precision $\delta$ (Problem 1). Assuming unit cost for the oracle, this algorithm can be implemented in parallel with run time bounded by a polynomial in $a + b, 1/\delta$, and $\log(\dim(CV))$.

**Proof.** First, we note the fact that each loss matrix $M_i^{(t)}$ satisfies $0 \leq M_i^{(t)} \leq \frac{1}{a} I$ follows immediately from its definition in step 2 and the observation that the adjoint mapping $f_{R,\varepsilon}$ satisfies

$$
\left(0, -\frac{a}{\varepsilon} I, \ldots, -\frac{a}{\varepsilon} I, -\frac{2a}{\varepsilon} I \right) \preceq f_{R,\varepsilon}(P, \Pi_a, \ldots, \Pi_1) \preceq \left((1 + \frac{a}{\varepsilon}) I, \frac{a}{\varepsilon} I, \ldots, \frac{a}{\varepsilon} I \right).
$$

For each $i = 1, \ldots, a$ it is clear that the construction of the density operators $\rho_i^{(t)}$ in terms of the loss matrices $M_i^{(t)}$ presented in Figure 4 obeys the condition of Theorem 8. It therefore follows that for any density operator $\rho_i \in \text{Dens}(CV)$ we have

$$
\frac{1}{T} \sum_{t=1}^{T} \langle \rho_i^{(t)}, M_i^{(t)} \rangle \leq \left\langle \rho_i, \frac{1}{T} \sum_{t=1}^{T} M_i^{(t)} \right\rangle + \frac{1}{a} \left(\gamma + \frac{\ln D}{\gamma T}\right).
$$

Summing these inequalities over all $i$ we find that for any density operators $(\rho_1, \ldots, \rho_a)$ it holds that

$$
\frac{1}{T} \sum_{t=1}^{T} \left\langle \left(\rho_1^{(t)}, \ldots, \rho_a^{(t)}\right), \left(M_1^{(t)}, \ldots, M_a^{(t)}\right) \right\rangle \leq \left\langle (\rho_1, \ldots, \rho_a), \frac{1}{T} \sum_{t=1}^{T} \left(M_1^{(t)}, \ldots, M_a^{(t)}\right) \right\rangle + \left(\gamma + \frac{\ln D}{\gamma T}\right).
$$

Substituting the definition of the loss matrices $M_i^{(t)}$ from step 2 and simplifying, we obtain

$$
\bar{\lambda} = \frac{1}{T} \sum_{t=1}^{T} \left\langle \left(\rho_1^{(t)}, \ldots, \rho_a^{(t)}\right), f_{R,\varepsilon}^{(t)}(P^{(t)}, \Pi_a^{(t)}, \ldots, \Pi_1^{(t)}) \right\rangle
\leq \left\langle (\rho_1, \ldots, \rho_a), \frac{1}{T} \sum_{t=1}^{T} f_{R,\varepsilon}^{(t)}(P^{(t)}, \Pi_a^{(t)}, \ldots, \Pi_1^{(t)}) \right\rangle + \frac{4a^2}{\varepsilon} \left(\gamma + \frac{\ln D}{\gamma T}\right).
$$

Substituting the choice of $\gamma, T$ from step 1 we see that the error term on the right side is at most $\delta/2$. Since this inequality holds for any choice of $(\rho_1, \ldots, \rho_a)$ it certainly holds for the optimal choice, from which it follows that the right side is at most $\mu_\varepsilon(R) + \delta/2$. By construction each $(P^{(t)}, \Pi_a^{(t)}, \ldots, \Pi_1^{(t)})$ is a $\delta/2$-best response to $(\rho_1^{(t)}, \ldots, \rho_a^{(t)})$ so it must be that the left side of this inequality is at most $\mu_\varepsilon(R) - \delta/2$. It then follows from item 1 of Theorem 5 (Rounding theorem) and the choice $\varepsilon = \delta/2$ that $|\bar{\lambda} - \lambda(R)| < \delta$ as desired.

Next we argue that the density operator $\tilde{\rho}^\alpha$ returned in step 4 is $3\delta/2$-optimal for $\lambda(R)$. By item 5 of Theorem 5 it suffices to argue that $(\tilde{\rho}_1, \ldots, \tilde{\rho}_a)$ are $\delta$-optimal for $\mu_\varepsilon(R)$. To this end, choose any $(P, \Pi_1, \ldots, \Pi_a)$. Since each $(P^{(t)}, \Pi_a^{(t)}, \ldots, \Pi_1^{(t)})$ is a $\delta/2$-best response to $(\rho_1^{(t)}, \ldots, \rho_a^{(t)})$ it holds that the inner product

$$
\left\langle \left(\rho_1^{(t)}, \ldots, \rho_a^{(t)}\right), f_{R,\varepsilon}^{(t)}(P^{(t)}, \Pi_a^{(t)}, \ldots, \Pi_1^{(t)}) \right\rangle
$$
1. Let $\varepsilon = \delta/2$, let $\gamma = \frac{\varepsilon}{16a^2}$, and let $T = \left\lceil \frac{\ln(\dim(CV))}{\gamma^2} \right\rceil$. Let $W_i^{(1)} = I_{CV}$ for each $i = 1, \ldots, a$.

2. Repeat for each $t = 1, \ldots, T$:
   
   (a) For $i = 1, \ldots, a$: Compute the updated density operators $\rho_i^{(t)} = W_i^{(t)} / \text{Tr}(W_i^{(t)})$.
   (b) For $i = 0, \ldots, a - 1$: Compute the projection $\Pi_{i+1}^{(t)}$ onto the positive eigenspace of $\text{Tr}_C(\rho_{i+1}^{(t)}) - \text{Tr}_C(V_i^{(t)} \rho_0^{(t)} V_i^{(t)\ast})$.
   (c) Use the oracle to obtain a $\delta/2$-optimal solution $P^{(t)}(\cdot)$ to the Weak optimization problem for $P^{(t)}(\cdot)$ on input $\rho_0^{(t)}$.
   (d) Compute the loss matrices $\left( M_1^{(t)}, \ldots, M_a^{(t)} \right) = \frac{\varepsilon}{4a^2} \left[ f_{R,\varepsilon} \left( P^{(t)}, \Pi_0^{(t)}, \ldots, \Pi_1^{(t)} \right) + \frac{2a}{\varepsilon} \left( I_{CV}, \ldots, I_{CV} \right) \right]$ so that each loss matrix $M_i^{(t)}$ satisfies $0 \preceq M_i^{(t)} \preceq \frac{1}{a} I$.
   (e) Update each weight matrix according to the standard MMW update rule:
   $$W_i^{(t+1)} = \exp \left( -\gamma \left( M_1^{(t)} + \cdots + M_i^{(t)} \right) \right).$$

3. Return
   $$\tilde{\lambda} = \frac{1}{T} \sum_{t=1}^{T} \left\langle f_{R,\varepsilon} \left( P^{(t)}, \Pi_0^{(t)}, \ldots, \Pi_1^{(t)} \right), \left( \rho_1^{(t)}, \ldots, \rho_a^{(t)} \right) \right\rangle$$
   as the $\delta$-approximation to $\lambda(R)$.

4. If optimal strategies are desired then compute
   $$(\tilde{\rho}_1, \ldots, \tilde{\rho}_a) = \frac{1}{T} \sum_{t=1}^{T} \left( \rho_1^{(t)}, \ldots, \rho_a^{(t)} \right)$$
   and
   $$(\tilde{P}, \tilde{\Pi}_a, \ldots, \tilde{\Pi}_1) = \frac{1}{T} \sum_{t=1}^{T} \left( P^{(t)}, \Pi_a^{(t)}, \ldots, \Pi_1^{(t)} \right),$$
   both of which are $\delta$-optimal for $\mu_{\varepsilon}(R)$.

   Compute $(\tilde{\rho}_1, \ldots, \tilde{\rho}_a)$ from $(\tilde{\rho}_1, \ldots, \tilde{\rho}_a)$ as described in item 4 of Theorem 5.
   
   Return $\tilde{\rho}_a$ and $\tilde{P}$, both of which are $3\delta/2$-optimal for $\lambda(R)$.

Figure 4: An efficient parallel oracle-algorithm for approximating $\lambda(R)$ (Problem II).
can increase by no more than $\frac{\delta}{2}$ when $(P, \Pi_1, \ldots, \Pi_a)$ is substituted for $(P^{(t)}, \Pi_1^{(t)}, \ldots, \Pi_a^{(t)})$. It then follows from the above expression for $\tilde{\lambda}$ that

$$\langle \frac{1}{T} \sum_{t=1}^{T} \left( \rho_1^{(t)}, \ldots, \rho_a^{(t)} \right), f_{R,\varepsilon}^* (P, \Pi_1, \ldots, \Pi_1) \rangle \leq \tilde{\lambda} + \frac{\delta}{2} \leq \mu_\varepsilon(R) + \delta$$

and hence $(\tilde{\rho}_1, \ldots, \tilde{\rho}_a)$ is $\delta$-optimal for $\mu_\varepsilon(R)$ as desired.

Next we argue that the operator $\tilde{P}$ returned in step 4 is $3\frac{\delta}{2}$-optimal for $\lambda(R)$. By item 2 of Theorem 5 it suffices to argue that $(\tilde{P}, \tilde{\Pi}_1, \ldots, \tilde{\Pi}_1)$ are $\delta$-optimal for $\mu_\varepsilon(R)$. To this end, choose any $(\rho_1, \ldots, \rho_a)$. It follows from the above expression for $\tilde{\lambda}$ that

$$\langle (\rho_1, \ldots, \rho_a), f_{R,\varepsilon}^* (\tilde{P}, \tilde{\Pi}_1, \ldots, \tilde{\Pi}_1) \rangle \geq \tilde{\lambda} - \frac{\delta}{2} \geq \mu_\varepsilon(R) - \delta$$

and hence $(\tilde{P}, \tilde{\Pi}_1, \ldots, \tilde{\Pi}_1)$ is $\delta$-optimal for $\mu_\varepsilon(R)$ as desired.

The efficiency of this algorithm is not difficult to argue. Each individual step consists only of matrix operations that are known to admit an efficient parallel implementation. Efficiency then follows from the observation that the number $T$ of iterations is polynomial in $a + b, 1/\delta$, and $\log(\dim(CV))$.

6.3 Special case: semidefinite programs on consistent density operators, a direct simulation of QIP

Consider a special case of the problem of approximating $\lambda(R)$ (Problem 1) in which $b = 0$. Since there is no interaction with Bob, this scenario corresponds to an ordinary, single-prover quantum interactive proof. In this case, $P(R) = \{V_a^*IV_a\}$ is a singleton set and the expression (7) for $\lambda(R)$ simplifies to

$$\lambda(R) = \min_{(\rho_1, \ldots, \rho_a) \text{ consistent with } R} \langle \rho_a, V_a^*IV_a \rangle,$$

which is a semidefinite program whose feasible region consists of density operators $\rho_1, \ldots, \rho_a$ consistent with $R$. The SDP (1) from the beginning of this paper is recovered by substituting the explicit conditions for consistency with referee $R$ listed in Proposition 4.

Since $P(R)$ is a singleton set, the oracle is trivial to implement so it follows immediately from Proposition 9 that the algorithm presented in Figure 4 can be used to solve SDPs of this form efficiently in parallel and thus prove QIP = PSPACE via direct simulation of a multi-message quantum interactive proof.

Later we will see that the oracle for weak optimization for $P(R)$ (Problem 2) required for general instances of $\lambda(R)$ can be reduced to an instance of this SDP special case of Problem 1 plus some post-processing.

7 Implementation of the oracle

In Section 6 we presented a parallel oracle-algorithm (Figure 4) for the problem of approximating $\lambda(R)$ (Problem 1) and proved its correctness and efficiency (Proposition 9). In order to complete the description of our algorithm for double quantum interactive proofs it remains only to describe the implementation of the oracle for weak optimization for $P(R)$ (Problem 2). In this section we establish the following.
**Proposition 10.** The weak optimization problem (Problem 2) for the set $P(R)$ specified in Eq. (6) admits a parallel algorithm with run time bounded by a polynomial in $a + b, 1/\delta$, and $\log(\dim(CV))$.

It follows that the algorithm of Figure 4 is an unconditionally efficient parallel algorithm for approximating $\lambda(R)$ (Problem 7).

As mentioned earlier, this instance of Problem 2 will be rephrased as a new instance of Problem 1 (plus some post-processing) so that the algorithm of Section 6 can be reused in the implementation of our oracle. Incidentally, we shall see that this new instance of Problem 1 has the special SDP form described in Section 6.3.

Choose any state $\rho$ and suppose that a (possibly cheating) Alice was somehow able to make it so that the state of the registers $(C, V)$ after the interaction with Alice is in state $\rho$. Let $A$ be a register large enough to admit a purification of $\rho$ and let $|\varphi\rangle \in ACV$ be any such purification. If Bob acts according to $(B_1, \ldots, B_b)$ then (similar to Eq. (3)) his probability of victory is

$$\Pr[\text{Bob wins} | \rho, (B_1, \ldots, B_b)] = \| \Pi V_{a+b} B_b V_{a+b-1} B_{b-1} \cdots B_1 V_a |\varphi\rangle \|^2.$$ 

Notice that this quantity also represents the probability of victory in a different, one-player game with a referee $R'$ whose initial state is $V_a |\varphi\rangle$. (Formally, the referee $R'$ exchanges $b$ rounds of messages with one of the players and zero messages with the other.) The unitaries $B_1, \ldots, B_b$ could specify actions for either Alice or Bob—a choice that depends only upon how we label the components of the referee $R'$.

Since our goal is to reduce Problem 2 to an instance of the SDP special case of Problem 1 it befits us to view $B_1, \ldots, B_b$ as actions for Alice in the game with referee $R'$. Let us write

$$R' = (V_a |\varphi\rangle, V_1', \ldots, V_b', \Pi')$$

where $V_i' = V_{a+i} \otimes I_A$ for each $i = 1, \ldots, b$ and $\Pi' = (I - \Pi) \otimes I_A$. The private memory register $V'$ of the new referee $R'$ is identified with the registers $(V, A)$ and the message register $C'$ of the new referee is identified with $C$. In this case, the set $P(R') = \{ Q \}$ is a singleton set with $Q = V_b' \Pi' V_1'$. Each choice of unitaries $B_1, \ldots, B_b$ induces both a measurement operator $P \in P(R)$ and a state $\xi \in A(R')$ with

$$\langle \rho, P \rangle = \| \Pi V_{a+b} B_b V_{a+b-1} B_{b-1} \cdots B_1 V_a |\varphi\rangle \|^2 = 1 - \langle \xi, Q \rangle.$$ 

and therefore

$$\max_{P \in P(R)} \langle \rho, P \rangle = 1 - \lambda(R') = 1 - \min_{\xi \in A(R')} \langle \xi, Q \rangle.$$ 

Moreover, if $P \in P(R)$ achieves the maximum on the left side then the unitaries $B_1, \ldots, B_b$ that induce $P$ also induce a state $\xi \in A(R')$ that achieves the minimum on the right side. As the right side is an instance of the SDP special case of Problem 1, a solution to Problem 2 presents itself:

1. Use the algorithm of Figure 4 to find $\xi \in A(R')$ minimizing $\langle \xi, Q \rangle$.

2. Find the unitaries $B_1, \ldots, B_b$ that induce $\xi$. These unitaries also induce a measurement operator $P \in P(R)$ maximizing $\langle \rho, P \rangle$. Compute $P$ using $B_1, \ldots, B_b$ via standard matrix multiplication.

We already saw in Section 6 how the algorithm of Figure 4 can be used to accomplish step 1. In the remainder of this section we fill in the details for step 2. Recall that the algorithm of Figure 4 finds near-optimal density operators $\xi_1, \ldots, \xi_b$ consistent with the referee $R' = (V_0 |\varphi\rangle, V_1', \ldots, V_b', \Pi')$, meaning that $\text{Tr}_C(\xi_{i+1}) = \text{Tr}_C(V_i' \xi_i V_i''\Pi')$ for each $i = 0, \ldots, b-1$ where $V_0' = V_a \otimes I_A$ and $\xi_0 = |\varphi\rangle \langle \varphi |$ for convenience. The following algorithm finds the unitaries $B_1, \ldots, B_b$. 

20
1. Let $B$ be a space large enough to admit purifications of $\xi_1, \ldots, \xi_b$ and write $|\alpha_0\rangle = |\varphi\rangle|0_B\rangle$.

2. For each $i = 1, \ldots, b$:
   
   (a) Compute a purification $|\alpha_i\rangle \in AC\forall B$ of $\xi_i$.
   
   (b) Compute a unitary $B_i \in U(CB)$ that maps $V_{i-1}'|\alpha_{i-1}\rangle$ to $|\alpha_i\rangle$.

3. Return the desired unitaries $(B_1, \ldots, B_b)$.

It is straightforward to verify the correctness of the above algorithm for step 2, and hence the whole algorithm when one compose the two steps together. It remains to show the above algorithm admits an efficient parallel implementation. Again such efficiency comes from the efficiency of each step and the number of steps is polynomial in $a + b$. The only non-standard matrix operations involved are computing purifications and computing a unitary that maps one purification to another. These operations as well as possible precision issues are handled explicitly in the previous version of this paper [GW10, Lemma 3-4, Appendix B].

8 Containment of DQIP inside PSPACE

In this section we explain how our parallel algorithm for double quantum interactive proofs implies the complexity theoretic equality $DQIP = DIP = \text{PSPACE}$. Formally, a decision problem $L$ is said to admit a double quantum interactive proof with completeness $c(|x|)$ and soundness $s(|x|)$ if there exists a polynomial-time uniform quantum referee $R_x$ such that

$$
\begin{align*}
&x \text{ is a no-instance of } L \implies \lambda(R_x) \geq c(|x|) \\
&x \text{ is a yes-instance of } L \implies \lambda(R_x) \leq s(|x|).
\end{align*}
$$

The complexity class $DQIP$ consists of all decision problems $L$ that admit double quantum interactive proofs with completeness $c(|x|)$ and soundness $s(|x|)$ for which there exists a polynomial-bounded function $p(|x|)$ such that $c - s \geq 1/p$. The class $DIP$ is defined similarly except that the referee is classical. By definition it holds that $DIP \subseteq DQIP$.

**Proposition 11.** $DQIP \subseteq \text{PSPACE}$.  

*Proof.* Let $L$ be any decision problem in $DQIP$ and let $R_x$ denote the referee witnessing this fact. Let $x$ be any input string and consider the following algorithm for deciding whether $x$ is a yes-instance or a no-instance of $L$.

1. Compute an explicit description of the referee $R_x = (|\psi\rangle, V_1, \ldots, V_{a+b}, \Pi)$.

2. Compute a $\delta$-approximation of the quantity $\lambda(R_x)$ by running an efficient parallel implementation of the algorithm of Figure 4 for the choice $\delta = (c - s)/3$ and accept or reject accordingly.

---

6The roles of no-instances and yes-instances in this definition are the opposite of convention, an artifact of our decision to define the quantity $\lambda(R)$ in terms of the probability of victory for Bob, as opposed to Alice. This cosmetic choice was made so as to better facilitate the use of the MMW.
The first step requires only simple matrix multiplication and can therefore be implemented by standard parallel algorithms with run time bounded by a polynomial in \(\log(\dim(CV))\). Proposition 10 establishes the same for the second step given the promise \(c - s \geq 1/p\). The entire process can then be simulated in polynomial space by standard methods (\(\text{NC}(\text{poly}) = \text{PSPACE} \quad \text{[Bor77]}\)), from which it follows that \(L \in \text{PSPACE}\) and hence \(\text{DQIP} \subseteq \text{PSPACE}\). \(\square\)

The characterization \[\text{DQIP} = \text{DIP} = \text{PSPACE}\]

now follows immediately from the well known fact that \(\text{IP} = \text{PSPACE}\) and from the trivial containment \(\text{IP} \subseteq \text{DIP}\).

Often in the study of interactive proofs the precise values of the completeness and soundness parameters \(c, s\) are immaterial because sequential repetition (or sometimes parallel repetition) can be used to transform any interactive proof with \(c - s \geq 1/p\) into another interactive proof in which \(c\) tends toward one and \(s\) tends toward zero exponentially quickly in the bit length of the input string \(x\). For this reason, it is typical to assume without loss of generality that \(c, s\) are constants such as \(2/3\) and \(1/3\) or that \(1 - c\) and \(s\) are exponentially small whenever it is convenient to do so. However, it is not immediately clear from their definition that double quantum interactive proofs are robust with respect to the choice of \(c, s\) so we must be as inclusive as possible when defining the classes \(\text{DIP}\) and \(\text{DQIP}\). Fortunately, our algorithm for double quantum interactive proofs does not require any extra promise on \(c, s\) beyond the standard condition \(c - s \geq 1/p\).

A fortunate corollary of the collapse of \(\text{DQIP}\) and \(\text{DIP}\) to \(\text{PSPACE}\) is that these classes are fully robust with respect to the choice of \(c, s\). That is, if a decision problem \(L\) admits a double quantum interactive proof \(c - s \geq 1/p\) then \(L\) also admits a double quantum interactive proof with \(c = 1\) and \(s \leq 2^{-q}\) for any desired polynomial-bounded function \(q(x)\). However, the method by which the original game is transformed into the low-error game is very circuitous: the original game must be solved in polynomial space according to Proposition 11 and then that polynomial-space computation must be converted back into an interactive proof with perfect completeness and exponentially small soundness according to proofs of \(\text{IP} = \text{PSPACE}\). It would be nice to know whether a more straightforward transformation such as parallel repetition followed by a majority vote could be used to reduce error for double quantum interactive proofs and other bounded-turn quantum games.

9 Extensions

9.1 Finding near-optimal strategies

Thus far we have concerned ourselves primarily with the problem of approximating the value \(\lambda(R)\) of a double quantum interactive proof. But it is not difficult to extend our result so as to solve the related search problem of finding near-optimal strategies for the players. Indeed, step 4 of the algorithm of Figure 4 returns a transcript \(\tilde{\rho}_1', \ldots, \tilde{\rho}_a'\) and a measurement operator \(\tilde{P} \in \mathcal{P}(R)\), both of which are \(3\delta/2\)-optimal for the formulation of \(\lambda(R)\) given in Eq. (7). The unitaries \(A_1, \ldots, A_a\) for Alice can be recovered from the transcript \(\tilde{\rho}_1', \ldots, \tilde{\rho}_a'\) via the method described in Section 7 with no additional complication.

It is only slightly more difficult to recover Bob’s unitaries \(B_1, \ldots, B_b\) from \(\tilde{P}\). Our definition of Problem 2 (Weak optimization for \(\mathcal{P}(R)\)) specifies only that a solution produce a near-optimal measurement operator \(P^{(t)}\) for a given state \(\rho_a^{(t)}\). But the solution to Problem 2 described in Section 7 produces \(P^{(t)}\) by first
constructing the associated unitaries $B_1^{(t)}, \ldots, B_b^{(t)}$. It is a simple matter to modify our definition of Problem 2 so as to also return those unitaries in addition to the desired measurement operator $P(t)$.

The near-optimal measurement operator $\tilde{P}$ returned in step 4 of the algorithm of Figure 4 is given by

$$\tilde{P} = \frac{1}{T} \sum_{t=1}^{T} P(t),$$

which indicates a strategy for Bob that selects $t \in \{1, \ldots, T\}$ uniformly at random and then acts according to $B_1^{(t)}, \ldots, B_b^{(t)}$. It is a simple matter to construct unitaries $B_1, \ldots, B_b$ that implement this probabilistic strategy by sampling the integer $t$ during the first round, recording that integer in Bob’s private memory (which must be enlarged slightly to make room for it), and controlling the operation in subsequent turns on the contents of that integer. All of the matrix operations required to construct $B_1, \ldots, B_b$ from each $B_1^{(t)}, \ldots, B_b^{(t)}$ in this way can be implemented efficiently in parallel.

### 9.2 Arbitrary payoff observables

In this paper we restricted attention to win-lose zero-sum games wherein the referee’s measurement $\{\Pi, I - \Pi\}$ at the end of the game indicates only a winner without specifying payouts to the players. In general, the referee’s final measurement $\{\Pi_a\}_{a \in \Sigma}$ could have outcomes belonging to some arbitrary finite set $\Sigma$. In this case, the referee awards payouts to the players according to a payout function $v: \Sigma \to \mathbb{R}$ where $v(a)$ denotes the payout to Alice in the event of outcome $a$. (Since the game is zero-sum, Bob’s payout must be $-v(a)$.) Jain and Watrous describe a simple transformation by which their algorithm for one-turn games can be used to compute the expected payout in this more general setting [JW09]. Their transformation extends without complication to our games.

In our case, the expected payout to Alice when she and Bob play according to $(A_1, \ldots, A_a)$ and $(B_1, \ldots, B_b)$, respectively, is given by

$$\sum_{a \in \Sigma} v(a) \langle \phi | \Pi_a | \phi \rangle = \langle \phi | \Pi_\Sigma | \phi \rangle$$

where

$$|\phi\rangle = V_{a+b} B_b V_{a+b-1} B_{b-1} \cdots B_1 V_a A_a V_{a-1} A_{a-1} \cdots A_2 V_1 A_1 |\psi\rangle$$

is the final state of the game and the Hermitian operator $\Pi_\Sigma = \sum_{a \in \Sigma} v(a) \Pi_a$ denotes the payout observable induced by the referee. The expected payout of this game can be computed simply by translating and rescaling $\Pi_\Sigma$ so as to obtain a measurement operator $0 \leq \Pi \leq I$ and then running our algorithm for double quantum interactive proofs with referee $R = (|\psi\rangle, V_1, \ldots, V_{a+b}, \Pi_\Sigma)$. The expected payout of the original game is then obtained by inverting the scaling and translation operations by which $\Pi$ was obtained from $\Pi_\Sigma$. As noted by Jain and Watrous, this transformation has the effect of inflating the additive approximation error $\delta$ by a factor of $\|\Pi_\Sigma\|$, which is the maximum absolute value of any given payout.

### Acknowledgements

The authors are grateful to Tsuyoshi Ito, Rahul Jain, Zhengfeng Ji, Yaoyun Shi, Sarvagya Upadhyay, and John Watrous for helpful discussions. Particularly, the alternative formulation of the strategies by density operators and measurements is inspired during the discussion with John Watrous. XW also wants to thank
the hospitality and invaluable guidance of John Watrous when he was visiting the Institute for Quantum Computing, University of Waterloo. The research was partially conducted during this visit and was supported by the Canadian Institute for Advanced Research (CIFAR). XW’s research is also supported by NSF grant 1017335. GG’s research is supported by the Government of Canada through Industry Canada, the Province of Ontario through the Ministry of Research and Innovation, NSERC, DTO-ARO, CIFAR, and QuantumWorks.

References

[AHK05] Sanjeev Arora, Elad Hazan, and Satyen Kale. The multiplicative weights update method: a meta algorithm and applications. Submitted, 2005.

[AK07] S. Arora and S. Kale. A combinatorial, primal-dual approach to semidefinite programs. In Proceedings of the Thirty-Ninth Annual ACM Symposium on Theory of Computing, pages 227–236, 2007.

[Bor77] Allan Borodin. On relating time and space to size and depth. SIAM Journal on Computing, 6(4):733–744, 1977.

[FIKU08] Lance Fortnow, Russell Impagliazzo, Valentine Kabanets, and Christopher Umans. On the complexity of succinct zero-sum games. Computational Complexity, 17(3):353–376, 2008.

[FK97] Uriel Feige and Joe Kilian. Making games short. In Proceedings of the 29th ACM Symposium on Theory of Computing (STOC 1997), pages 506–516, 1997.

[FKS95] Joan Feigenbaum, Daphne Koller, and Peter Shor. A game-theoretic classification of interactive complexity classes. In Proceedings of the 10th Conference on Structure in Complexity Theory, pages 227–237, 1995.

[FvdG99] Christopher Fuchs and Jeroen van de Graaf. Cryptographic distinguishability measures for quantum mechanical states. IEEE Transactions on Information Theory, 45(4):1216–1227, 1999. arXiv:quant-ph/9712042v2.

[GS89] Shafi Goldwasser and Michael Sipser. Private coins versus public coins in interactive proof systems. In Silvio Micali, editor, Randomness and Computation, volume 5 of Advances in Computing Research, pages 73–90. JAI Press, 1989.

[Gut05] Gus Gutoski. Upper bounds for quantum interactive proofs with competing provers. In Proceedings of the 20th IEEE Conference on Computational Complexity (CCC’05), pages 334–343, 2005.

[GW05] Gus Gutoski and John Watrous. Quantum interactive proofs with competing provers. In Proceedings of the 22nd Symposium on Theoretical Aspects of Computer Science (STACS’05), volume 3404 of Lecture Notes in Computer Science, pages 605–616. Springer, 2005. arXiv:cs/0412102v1 [cs.CC].

[GW07] Gus Gutoski and John Watrous. Toward a general theory of quantum games. In Proceedings of the 39th ACM Symposium on Theory of Computing (STOC 2007), pages 565–574, 2007. arXiv:quant-ph/0611234v2.
[GW10] Gus Gutoski and Xiaodi Wu. Short quantum games characterize PSPACE. arXiv:1011.2787v1.

[JJUW10] Rahul Jain, Zhengfeng Ji, Sarvagya Upadhyay, and John Watrous. QIP = PSPACE. In Proceedings of the 42nd ACM Symposium on Theory of Computing (STOC 2010), pages 573–582, 2010. arXiv:0907.4737v2 [quant-ph].

[JJUW09] Rahul Jain, Sarvagya Upadhyay, and John Watrous. Two-message quantum interactive proofs are in PSPACE. In Proceedings of the 50th IEEE Symposium on Foundations of Computer Science (FOCS 2009), pages 534–543, 2009. arXiv:0905.1300v1 [quant-ph].

[JW09] Rahul Jain and John Watrous. Parallel approximation of non-interactive zero-sum quantum games. In Proceedings of the 24th IEEE Conference on Computational Complexity (CCC 2009), pages 243–253, 2009. arXiv:0808.2775v1 [quant-ph].

[Kal07] Satyen Kale. Efficient algorithms using the multiplicative weights update method. PhD thesis, Princeton University, 2007.

[Kit02] Alexei Kitaev. Quantum coin-flipping. Presentation at the 6th Workshop on Quantum Information Processing (QIP 2003), 2002.

[KM92] Daphne Koller and Nimrod Megiddo. The complexity of two-person zero-sum games in extensive form. Games and Economic Behavior, 4:528–552, 1992.

[KMvS94] Daphne Koller, Nimrod Megiddo, and Bernhard von Stengel. Fast algorithms for finding randomized strategies in game trees. In Proceedings of the 26th ACM Symposium on Theory of Computing (STOC 1994), pages 750–759, 1994.

[KW00] Alexei Kitaev and John Watrous. Parallelization, amplification, and exponential time simulation of quantum interactive proof system. In Proceedings of the 32nd ACM Symposium on Theory of Computing, pages 608–617, 2000.

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

[LN93] Michael Luby and Noam Nisan. A parallel approximation algorithm for positive linear programming. In Proceedings of the 25th ACM Symposium on Theory of Computing (STOC 1993), pages 448–457, 1993.

[Meg92] Nimrod Megiddo. A note on approximate linear programming. Information Processing Letters, 42(1):53, 1992.

[MW05] Chris Marriott and John Watrous. Quantum Arthur-Merlin games. Computational Complexity, 14(2):122–152, 2005. arXiv:cs/0506068v1 [cs.CC].

[NC00] Michael Nielsen and Issac Chuang. Quantum Computation and Quantum Information. Cambridge University Press, 2000.

[Ser91] Maria Serna. Approximating linear programming is log-space complete for P. Information Processing Letters, 37(4):233–236, 1991.

[Sha92] Adi Shamir. IP = PSPACE. Journal of the ACM, 39(4):869–877, 1992.
[TX98] Luca Trevisan and Fatos Xhafa. The parallel complexity of positive linear programming. *Parallel Processing Letters*, 8(4):527–533, 1998.

[vzG93] J. von zur Gathen. Parallel linear algebra. In J. Reif, editor, *Synthesis of Parallel Algorithms*, 1993.

[Wat08] John Watrous. *Lecture Notes on Theory of Quantum Information*. 2008.

[Wu10a] Xiaodi Wu. Equilibrium value method for the proof of QIP=PSPACE. [arXiv:1004.0264v4 [quant-ph]], 2010.

[Wu10b] Xiaodi Wu. Parallized solutions to semidefinite programings in quantum complexity theory. [arXiv:1009.2211v1 [quant-ph]], 2010.