DIPLOMARBEIT

Äquivalenz von Klassischen und Quantencomputern in Interaktiven Beweisen und Spielen mit Schiedsrichter

(Equivalence of Classical and Quantum Computation in Interactive Proofs and Refereed Games)

Angefertigt am
Mathematischen Institut

Vorgelegt der
Mathematisch-Naturwissenschaftlichen Fakultät der
Rheinischen Friedrich-Wilhelms-Universität Bonn

August 2012

Von

Leonhard Schneider

geboren am 13. Januar 1984

in Regensburg

Contents

1 Einleitung

Diese Diplomarbeit beschäftigt sich mit Quantenspielen und den neuesten Errungenschaften bezüglich interaktiver Beweissysteme, deren Teilnehmer Quantencomputer benutzen dürfen (QIP). Im Hauptteil werden vor allem zwei Beweise behandelt, der wissenschaftlich anerkannte Beweis von Jain, Ji Upadhyay und Watrous für QIP = PSPACE [JJUW09] und der Beweis von Gutoski und Wu für DQIP = PSPACE [GW11]. Die Komplexitätsklasse DQIP beinhaltet alle Entscheidungsprobleme, die in Quantenspielen mit Schiedsrichter in einer konstanten Anzahl von Kommunikationsrunden gelöst werden können. PSPACE bezeichnet, wie üblich die Klasse von Problemen, die von klassischen Computern in exponentieller Zeit unter Nutzung von polynomiell viel Speicherplatz entscheidbar sind.
Von besonderem Interesse ist die Einordnung dieser Quanten-Komplexitätsklassen, QIP und DQIP, in PSPACE, da diese die Äquivalenz von klassischen und Quantencomputern in bestimmten komplexitätstheoretischen Situationen zur Folge hat: QIP = IP und DIP = DQIP. Wobei die Klassen IP und DIP Interaktive Beweissysteme und doppelte Interaktive Beweissysteme bezeichnen, welche die klassischen Varianten von QIP und DQIP darstellen. Die beiden Gleichungen folgen aus PSPACE = IP und DQIP = PSPACE, wobei die erstere von Shamir [Sha92] bewiesen wurden und die zweite im Hauptteil dieser Diplomarbeit behandelt wird.
Probleme in DIP können von einem polynomiell beschränkten klassischen Schiedsrichter nach einer polynomiellen Anzahl von Kommunikationsrunden mit zwei omnipotenten Spielern, Alice und Bob, entschieden werden. Die Einordnung der Klasse DQIP in PSPACE stellt eine Verallgemeinerung des ursprünglichen Beweises von Gutoski und Wu für SQG = PSPACE [GW10] dar. Der einzige Unterschied zwischen SQG, der Klasse von kurzen Quantenspielen, und DQIP besteht darin, dass jeweils nur eine Frage an die Spieler gestellt wird. Beiden Klassen ist jedoch gemein, dass der Schiedsrichter die Fragen an Bob von Alices Anwort abhängig machen kann.
Anhand dieser Formulierungen kann man bereits erkennen, dass deutsche Begrifflichkeiten in diesem Feld entweder noch nicht existieren oder auf einem rudimentären Level des Eindeutschens englischer Fachbegriffe beruhen. Darüberhinaus soll vor allem der Beweis für DQIP = PSPACE detailiert dargestellt werden und möglichst vielen Lesern einen Zugang bieten. Aus diesen Gründen ist es sinnvoll die Diplomarbeit auf Englisch zu verfassen. Nichtsdestotrotz sind für die Einführung und die Darstellung der Ergebnisse dieser Arbeit einige deutsche Begrifflichkeiten zu klären. Die Anzahl an unschönen Neologismen soll jedoch auf ein Minimum beschränkt werden. Deshalb werden die allgemein üblichen Bezeichnungen für Komplexitätsklassen verwendet, obwohl sich diese auf die englischen Fachbegriffe beziehen. Interaktive Beweissysteme mit Quantencomputern soll zum Beispiel als Äquivalent zu dem Begriff “Quantum Interactive Proof systems” (QIP)

verwendet werden.
Die Beweise für QIP = PSPACE und DQIP = PSPACE beruhen auf ähnlichen Ideen. Die Problemstellungen in den Quanten-Komplexitätsklassen QIP und DQIP können als semidefinite Programme formuliert werden und mit NC-Algorithmen gelöst werden. Diese NC-Algorithmen beruhen auf der Anwendung boolscher Kreise, welche logarithmische Speichernutzung und polylogarithmische Tiefe in der Eingabegröße haben dürfen. Des Weiteren benutzen die Algorithmen eine von Kale entwickelte Subroutine, die multiplikative Gewichts-Erneuerungs-Methode für Matrizen [Kal07]. Diese wird im Englischen "matrix multiplicative weight update method" (MMW) genannt.
Diese Arbeit erweitert Gus Gutoskis und Xiaodi Wus Abhandlung [GW11] um zahlreiche Details, die zum Nachvollziehen ihrer Beweise notwendig sind. Des Weiteren wird die rudimentäre Fehlerkorrektur, der ersten Version ihrer Arbeit [GW10] präzisiert und auf eine konstante Anzahl von Kommunikationsrunden verallgemeinert. Die Allgemeinheit des Beweises von Gutoski und Wu lässt sicherlich eine Veröffentlichung in mathematischen Journalen und nicht nur in Online-Archiven zu . Zumal die beiden oben genannten Gleichungen, QIP = IP und DIP = DQIP, die ersten allgemeinen Ergebnisse darstellen, die eine Äquivalenz von klassischen Computern und Quantencomputern unter gewissen komplexitätstheoretischen Annahmen beweisen.
Im Folgenden wird die Entwicklung der mathematischen Theoreme, die zu den oben genannten Meilensteinen der Quanten-Komplexitäts-Theorie geführt haben, skizziert. 1989 definierten Silvio Micali und Charles Rackhoff interaktive Beweissysteme zum ersten Mal [GMR89]. Die Komplexitätsklasse IP beinhaltet alle Probleme die von einem Verifizierer in polynomieller Zeit mit hoher Wahrscheinlichkeit entschieden werden können, wobei dieser einen klassischen probabilistischen Computer benutzen darf und mit einem omnipotenten Beweiser in polynomiell vielen Runden Nachrichten austauschen kann. Der Beweiser kann in diesem Fall die zufälligen Münzwürfe, die der Verifizierer ausführt, nicht einsehen. Man spricht deshalb von privaten Münzen. Im Gegensatz dazu bezeichnet die Komplexitätsklasse AM ("Arthur-Merlin class") Beweisysteme in denen die Münzwürfe öffentlich sind. Man benutzt die Bezeichnung Arthur-Merlin, da Merlin ein Zauberer ist, und damit die Möglichkeit besitzt an Informationen zu gelangen, die selbst einem omnipotenten Beweiser verborgen bleiben. Die auf zwei Runden beschränkte Version dieser Klasse, AM(2), wurde bereits 1985 von Lazlo Babai eingeführt [Bab85]. Die analogen Quantenkomplexitätsklassen, QIP und QAM, unterscheiden sich von ihren klassischen Versionen, IP und AM, nur dadurch, dass der Verifizierer beziehugsweise Arthur einen Quantencomputer benutzen darf und die Teilnehmer Quanteninformationen austauschen können. Die Quantenkomplexitätsklassen QIP und QMAM werden ausführlich im Hauptteil behandelt, wobei in QMAM die Kommunikation von Merlin initiiert wird und insgesamt drei Nachrichten verschickt werden. Im Gegensatz zur klassischen Komplexitätstheorie kann in QIP die Anzahl der Runden auf drei beschränkt werden ohne die Klasse zu verkleinern. Der Beweis von Alexei Kitaev und John Watrous für QIP = QIP(3) [KW00] und der Beweis von Chris Marriott und Watrous für QIP(3) = QMAM [MW05] werden im Hauptteil genau erklärt, um den Stand der Forschung zu verdeutlichen, welcher zum Beweis der beiden Gleichungen QIP = PSPACE und DQIP = PSPACE führte. Eine solche Beschränkung auf eine kon-

stante Anzahl von Runden ist klassisch kaum vorstellbar, zumal dies den Kollaps der polynomiellen Hierarchie (PH) auf dem zweiten Level zur Folge hätte.
Als die Forschung für diese Diplomarbeit 2010 begann, war nicht bekannt ob QRG(2) = PSPACE gilt. Bis heute wurde kein direkter Beweis für diese Fragestellung gefunden, jedoch wurde sie durch die Arbeit von Gutoski und Wu [GW10] gelöst, zumal QRG(2) $\subseteq$ SQG trivialer Weise gilt. Angesichts der Tatsache, dass dieser Beweis bis heute nicht über jeden Zweifel erhaben ist, schien es wichtiger diesen ausführlich zu erklären und seine Schwächen zu beheben als einen direkten Beweis für QRG(2) = PSPACE zu finden. Zumal nach wie vor wertvolle Forschungsressourcen diesem Problem gewidmet werden, obwohl es bereits gelöst wurde, ist die ausführliche und detaillierte Darlegung dieses Beweises notwendig.
Um eingängiges Verständnis zu garantieren, werden alle grundlegenden mathematischen Konzepte in den Vorbereitungen (Kapitel 4) dargelegt. Des Weiteren werden zu Beginn sowohl komplexitätstheoretische wie auch quantenmechanische und spieltheoretische Grundlagen erklärt. Es folgen eine ausführliche Erklärung der MMW-Methode sowie eine kurze Beschreibung semidefiniter Programme. Im Hauptteil wird das Ergebnis QIP = QIP(3) von Kitaev und Watrous [KW00] präsentiert, da es eine wichtige Rolle im genau ausgeführten Beweis von QIP = PSPACE [JJUW09] spielt. Dieser Beweis ist wiederum in der Struktur ähnlich zu dem Ergebnis DQIP = PSPACE, welches detailliert und in sich abgeschlossen beschrieben wird. Neben der Korrektur zahlreicher kleinerer Fehler sowie der Behebung einiger Ungenauigkeiten, wird in Abschnitt 6.2.6 eine Entscheidungsregel entworfen und erläutert, welche die von Gutoski und Wu gestellten Kriterien erfüllt. Ebenso wird in den Abschnitten 5.3.2, 6.2.3 und 6.2.4 eine genau Analyse der Beweise für die Gleichungen QIP = PSPACE und DQIP = PSPACE vorgenommen. Außerdem wird im Abschnitt 5.3.3 die Fehlerkorrektur verbessert und detailliter dargestellt. Zuletzt wird diese in Abschnitt 6.2.5 zusätzlich auf eine kostante Anzahl an Kommunikationsrunden verallgemeinert.

2 Introduction

2.1 Abstract

This thesis emphasizes the latest advances in quantum interactive proof systems (QIP) and double quantum interactive proofs (DQIP), a generalization of quantum refereed games (QRG), focusing on two major proofs. On the one hand the scientifically acknowledged proof of QIP = PSPACE from Jain, Ji, Upadhyay and Watrous [JJUW10], and on the other hand the proof of DQIP = PSPACE from Gutoski and Wu [GW11]. Hereby PSPACE stands for deterministic polynomial bounded space. The above equations were the first distinct results on the boundaries of quantum computation in terms of classical complexity classes. Moreover, these equations imply DQIP = DIP (Double Interactive Proofs) and QIP = IP (Interactive Proof systems), stating the equivalence of classical and quantum computation in terms of complexity classes. Both proofs share similar ideas and structures. In fact both problems can be formulated as semidefinite programs (SDP) and be solved by NC algorithms, relying upon the matrix multiplicative weight update method (MMW) from Kale [Kal07]. However, only Watrous' QIP proof is published in a mathematical journal.
To guarantee complete insight into the complex proofs of the main part the preliminaries in Chapter 4 are quite resourceful, including complexity theoretic and quantum mechanical principles, as well as a result from game theory and a complete description of the MMW method. In Chapter 5 the proofs for QIP = QIP(3) from Kitaev and Watrous [KW00] and QIP(3) = QMAM from Marriott and Watrous [MW05] are discussed since their results are essential for the proof of QIP = PSPACE [JJUW09]. Besides the correction of several errors and the addition of explanations especially in the proof of DQIP = PSPACE in Chapter 6, the precision issues are discussed in greater detail in Section 5.3.3 and 6.2.5. Moreover, the accuracy in the proof of DQIP = PSPACE is generalized to multiple rounds of interactions. In addition an explicit description of the oracle algorithm and a decision rule are presented and proven in Section 6.2.4 and 6.2.6, respectively.

2.2 Motivation and history

In 2010 my interest was drawn upon a paper [JJUW10], which accomplished the celebrated proof of QIP = PSPACE. Studying this significant achievement, I became aware of the previously unsolved problem whether or not QRG(2)=PSPACE. Since Kale's MMW method seemed a promising tool in solving this problem the possibility to contribute to scientific advances was exciting. In order to follow the proofs discussed in

this thesis, knowledge in different fields of mathematics, physics and computer science is needed. Hopefully my background enables me to describe quantum computation at its recent state of the art in an understandable and precise fashion. To achieve this goal repetitions of important calculations and principles are necessary to some extend. The fact that SQG = PSPACE is still stated as a hypothesis emphasizes the strong need for further explanations. The following calculations presented in this thesis clarify the subject matter and resolve this issue, as well as the generalization DQIP = PSPACE.
Even though quantum computation might seem totally irrational, which is mathematically backed, it is a stringent logical generalization of classical computers. There is even the possibility that every physically realizable computing device is essentially a quantum computer, like every classical one is essentially a Turing machine according to the Church-Turing thesis. Even though the advances in experimental quantum computation are quite small so far, the above hypothesis emphasizes, that the chances in this field might be the best we get. The theoretical part seems even more interesting than the practical one as the development of physical quantum computers is on a poor level from a computational point of view. Recently a firm claimed to have built a quantum computer with 84 qubits. Even if such a machine exists, one can see that it might take decades to realize helpful physical quantum computers. One needs to take into consideration that the proven limits of quantum computation, also the ones presented in this thesis are only on levels far above today's classical computers. Comparing classical and quantum computation the only significant result was proven by Peter Shor as he provided a polynomial-time quantum algorithm for prime factorization and discrete logarithms [Sho97]. Since the problem of prime factorization is known for quite some time, it is widely believed that a polynomial time classical algorithm does not exist. These statements suggest that further investigation is needed in order to fully understand the chances and limits of quantum computation. More time and energy should be invested into quantum computational matters, as they are still widely unexplored and new theorems and hypothesis are published continuously.
Before the explanations regarding this thesis continue the history of interactive proofs and the Arthur Merlin class is stated briefly. In 1989 Shafi Goldwasser, Silvio Micali and Charles Rackhoff introduced the concept of interactive proof systems [GMR89]. A probabilistic polynomial time verifier decides in polynomially many rounds of interaction with an omnipotent prover, if the statement the prover suggests is true or false. The verifier has to answer correctly significantly more than half of the times. He uses private random coins, separating IP from AM. In any Arthur-Merlin class, like AM for example, the prover sees the verifier‘s random choices, and is therefore called Merlin, the magician. Lazlo Babai introduced the two round version AM(2) in 1985 [Bab85]. In the quantum versions of the above classes, QIP and QAM, the verifier is allowed to use quantum computers and to exchange quantum information. Moreover, problems in the class QMAM can be solved by three turns of communication. Arthur responds to Merlins first message by sending random bits and uses a quantum computer to process the information after he received the second and final message from Merlin.
In the large field of game theory, this thesis solely focuses on competitive two player games with a limited number of rounds. The polynomially bounded referee poses ques-

tions to the omnipotent players, receives their answers, and declares a winner. One player supports a certain statement, while his opponent denies it. Hence the referee decides to approve or discard a statement, as in every decision problem, by declaring a winner. Therefore, a language belongs to what this thesis calls an interactive class, if the verifier can distinguish yes- and no-instances after the communication with the prover. DQIP includes all languages, which can be decided by a quantum refereed game with a constant number of rounds and separated private communication. Hereby no player is able to see the messages the referee exchanges with the opponent.
However, after QIP = PSPACE was proven the question remained, how quantum refereed games behave compared to classical complexity classes. The research for this thesis started in 2010, when the proof for QRG(2) $\subseteq$ PSPACE was yet to be found. Opposed to DQIP the communication in QRG(2) is limited to two rounds and the players are asked at the same time. As mentioned above the proof of QIP = PSPACE relies upon the matrix multiplicative weight update method (MMW) from Kale [Kal07]. This method is a promising approximation algorithm. It can be used to solve certain semidefinite programs. Therefore, it seemed obvious that the MMW method can help resolving QRG(2) $\subseteq$ PSPACE, since any problem in QRG(2) can be expressed as a semidefinite program.
Even though this thesis solved several problems, important ones remained, when Gutoski and Wu published their proof of SQG = PSPACE. Since QRG(2) $\subseteq$ SQG their paper solved the question, whether QRG(2) = PSPACE holds. Of course one could still have tried to prove this equation directly, however, concretizing the proof of Gutoski and Wu is of higher importance. Gutoski and Wu had the simple but great idea of separating the different interactions of the referee with each player in a two round quantum refereed game. Furthermore, they introduced an interesting algorithm design, including a smartly chosen subroutine for a weak approximation, and a main algorithm, heavily relying upon the MMW method. This subroutine is implemented by a special case of the main algorithm. Moreover, they managed to generalize their methods to multiple rounds of interaction proving DQIP = PSPACE. This result even implies QIP = PSPACE. The goal of this thesis is therefore to explain the mathematical approach to the problem QRG(2) = PSPACE as well as to present the mentioned results in greater detail than it has ever been done. To my knowledge neither books nor scripts have been published on these matters yet. Hopefully this work will explain their tremendous achievement in a coherent way. To my best knowledge Gutoski's and Wu's proofs are correct, even though they omitted important explanations and an error correction. Nevertheless, they should receive credit for their work.
Before the mathematical elaboration starts this thesis emphasizes, what is known about interactive proof systems and refereed games in a classical setting. The first significant result on classical interactive proof systems, PSPACE $\subseteq$ IP, was introduced by Shamir in 1992 [Sha92]. He used arithmetization to convert quantified Boolean formulas into polynomials, whose value depend on the formulas being true or false. The transformation of these polynomials eased their calculation, while their size increased only polynomially. This enables a polynomial time verifier to check whether or not the prover lied, with an interactive protocol in polynomially many rounds. Another impor-

tant classical result was found by Uriel Feige and Joe Killian in 1997 [FK97]. In order to prove RG(2) = PSPACE, they pointed out how the #P-complete problem #3-SAT can be solved using arithmetization. Combining this procedure and Shamir's interactive proof for the value of an arithmetic simple QBF (Quantified Boolean Formula) provides PSPACE $\subseteq$ RG(2). For the reverse containment they argued, that approximately good strategies (almost always winning) can be found in polynomial space by applying Savitch's theorem, which is standard in complexity theory, see for example [AB09]. The equation at hand only holds, because the referee can distribute his information through private communication channels. Otherwise, when each player sees all the information the referee shares, one needs polynomial many rounds of communication to solve PSPACE-complete problems with a high probability. Moreover, Feige and Killian also provided the best known bound for refereed games with polynomially many rounds of communication: RG = EXP [FK97].
The organization of this thesis focuses on clarifying recent advances in Quantum Complexity theory. Therefore, the preliminaries are quite resourceful, taking many different mathematical concepts into account. Moreover, this part explains basic quantum mechanical principles needed to understand quantum computers. In addition the preliminaries include important notational issues as well as some core algorithms and subroutines, especially Kale's MMW method and the fidelity function. These two concepts are extensively used in the main part of the thesis, which provides a complete, self-contained description of the two major advances in quantum complexity theory: QIP = PSPACE and DQIP = PSPACE. To this end the proofs of QIP = QIP(3) and QIP(3) = QMAM are discussed before.

3 Notation

$\mathcal{L}(\mathcal{X})$	set of linear operators acting on $\mathcal{X}$
$Herm(\mathcal{X})$	set of Hermitian operators acting on $\mathcal{X}$
$Pos(\mathcal{X})$	set of positive semidefinite operators acting on $\mathcal{X}$
$\mathcal{U}(\mathcal{X})$	set of unitary operators acting on $\mathcal{X}$
$\mathcal{D}(\mathcal{X})$	set of density operators acting on $\mathcal{X}$
$Pro(\mathcal{X})$	set of projection operators acting on $\mathcal{X}$
$Meas(\mathcal{X})$	set of projective measurement operators acting on $\mathcal{X}$
$T(\mathcal{X})$	set of superoperators acting on $\mathcal{L}(\mathcal{X})$
$\dim(\cdot)$	dimension of a space
$\mathrm{tr}(\cdot)$	trace of an operator
$\mathrm{tr}_{\mathcal{X}}(\cdot)$	partial trace of an operator ($\mathcal{X}$ is traced out)
$\mathrm{vec}(\cdot)$	vector mapping transforming matrices into vectors
$\exp(\cdot)$	exponential function for complex numbers and matrices
$\mathrm{e}^{\cdot}$	exponential function for real numbers
$\log(\cdot)$	logarithm with base two or logarithmic time, meaning $\mathrm{O}(\log n)$
$\mathrm{polylog}(\cdot)$	polylogarithmic time, meaning $\mathrm{O}((\log n)^k)$ for some constant k
poly	space of polynomial time computable functions
$\langle \cdot, \cdot \rangle$	scalar product or Hilbert-Schmidt inner product for linear operators
$\| \cdot \|$	euclidean norm or matrix norm induced by the l_2-norm
$\| \cdot \|_p$	Schatten p-norm for linear operators
$\| \cdot \|_{\mathrm{tr}}$	trace norm for linear operators and superoperators
$\| \cdot \|_{\diamond}$	diamond norm for superoperators
$\mathcal{XYZ}$	tensor product of the spaces $\mathcal{X}, \mathcal{Y}$ and$\mathcal{Z}$ (short notation for $\mathcal{X} \otimes \mathcal{Y} \otimes \mathcal{Z}$)
$\angle(\cdot, \cdot)$	Bures angle between density operators
$\mathbb{E}[\cdot]$	expected value of a random variable

4 Preliminaries

Since we want to describe the state of art in quantum computation, we will presume, that the reader is familiar with basic mathematical structures like vectors or matrices over complex spaces, as well as linear programming and complexity theory. In order to understand the complicated proofs of the theorems, all lemmas are stated and the lemmas, which are not standard, are also proven.

4.1 Basic mathematical structure and notation

To describe quantum computational phenomena, operators on finite complex vector spaces will prove suitable. Therefore, we define certain subsets of linear operators and superoperators, as well as the Dirac notation and the vector mapping in this section. Moreover, we examine some decompositions and lemmas, which we need at later chapters. Readers who are familiar with the definitions used in quantum computation and information, can skip this part, since most of the notation is either standard or more or less self-explanatory. The following definitions can be found in every sophisticated book on linear algebra. Most ideas for this section are from the book [NC00] and two papers discussed in the main part, namely [JJUW10] and [KW00]. Nevertheless, some easy proofs and details in the cited proofs were added to guarantee complete insight.

4.1.1 Linear operators

In general vectors are expressed in small Latin $x, y, v, w \ldots$, and matrices $A, B, U, P \ldots$ in capital Latin. In order to highlight the spaces we will use Calligraphic symbols, like $\mathcal{A}, \mathcal{B}, \mathcal{X}, \mathcal{Y}$, to describe them. Moreover, small Greek letters $\alpha, \beta, \gamma, \delta, \ldots$ represent real numbers and as usually in quantum information also density operators ρ, σ or τ.
First recall that a Hilbert space $\mathcal{X}$ is a complete metric space with an inner product $\langle \cdot, \cdot \rangle$. For two given Hilbert spaces $\mathcal{X}, \mathcal{Y}$ the set of linear operators mapping $\mathcal{X}$ to $\mathcal{Y}$ is referred to as $\mathcal{L}(\mathcal{X}, \mathcal{Y})$. To shorten notation $\mathcal{L}(\mathcal{X})$ will be used instead of $\mathcal{L}(\mathcal{X}, \mathcal{X})$. For any Hilbert space $\mathcal{X}$ the linear operators in $\mathcal{L}(\mathcal{X})$ form an algebra $\mathcal{B}(\mathcal{X})$. If $\mathcal{X}$ is finite dimensional, namely $\dim(\mathcal{X}) = n$, each linear operator $A \in \mathcal{B}(\mathcal{X})$ can be identified with a matrix acting on $\mathbb{C}^n$. To this end we just fix a basis $\{e_1, \ldots, e_n\}$ of $\mathcal{X}$ to observe

$$A_{ij} = \langle e_i, Ae_j \rangle \quad \forall i, j \in \{1, \ldots n\}.$$

Notice that A_{ij} refers to the matrix, while A on the right hand side represents the linear operator. Therefore, we have an algebraic isomorphism from $\mathcal{B}(\mathcal{H})$ to the algebra of $n \times n$ matrices with complex entries. Thus, linear algebraic terms can be applied to

linear operators on finite-dimensional Hilbert spaces, which means we can extend a lot of well known facts from linear algebra to these operators. This correspondence will not be mentioned every time, even though one symbol describes both the operator and the matrix. Since all Hilbert spaces under consideration are finite dimensional it will suffice throughout most parts of the thesis to think of a complex Euclidean space $\mathbb{C}^n$ instead.
If a linear operator $A \in \mathcal{L}(\mathcal{X})$ satisfies $A = A^*$ it is called hermitian. Here the notation A^* refers to the adjoint or conjugate transpose of A. Furthermore, let $Herm(\mathcal{X})$ be the set of all Hermitian operators acting on $\mathcal{X}$. Note that the sum of Hermitian operators is also Hermitian, whereas the product of two Hermitian operators is only Hermitian if they commute. A Hermitian operator $A : \mathcal{X} \to \mathcal{X}$ is positive semidefinite, if and only if $xAx \geq 0$ for all $x \in \mathcal{X}$. For $A, B \in Herm(\mathcal{X})$ the notation $A \leq B$ or $B \geq A$ will be used to describe the positive semidefiniteness of $B - A$. Moreover, $Pos(\mathcal{X})$ denotes the set of positive semidefinite operators acting on $\mathcal{X}$. Sometimes positive semidefiniteness is defined even more general as $Pos(\mathcal{X}) = \{A \in \mathcal{L}(\mathcal{X}) : \exists B \in \mathcal{L}(\mathcal{X}) \text{ s.t. } A = BB^*\}$. Furthermore, the set of unitary operators $U(\mathcal{X})$ includes all $U \in \mathcal{L}(\mathcal{X})$ satisfying $UU^* = \mathbb{1}_{\mathcal{X}}$. The notation $\mathbb{1}_{\mathcal{X}}$ refers to the neutral element of multiplication $\mathcal{L}(\mathcal{X})$. Equivalently we can characterize unitary operators as norm preserving, i.e.

$$U \in U(\mathcal{X}) \Leftrightarrow \forall x \in \mathcal{X} : \|Ux\| = \|x\|.$$

Here $\|\cdot\|$ represents the Euclidean vector norm as usual. But once additional notation is introduced, we apply the same notation to more complex operations without explicit mentioning, even if it coincides with standard notation. This means the notation $\|\cdot\|$ might refer to the Euclidean vector norm as well as an operator norm later on. If we would proceed in a different way either the explanations would repeat often or the notation would be abused, both complicating the understanding of the complex proofs at later chapters. Therefore, we have to reflect upon the mathematical object under consideration carefully.
Coming back to the basic definitions, an eigenvalue of a linear operator $A \in \mathcal{L}(\mathcal{X})$ is a scalar $\lambda \in \mathbb{K}$ such that

$$\exists v \in \mathcal{X}, v \neq 0 : Av = \lambda v,$$

where $\mathbb{K}$ refers to the body, the Hilbert space $\mathcal{X}$ is built upon. The set of all eigenvalues of some operator A is called the spectrum. It is referred to as $\text{spec}(A)$. We will restrict our analysis to Hermitian operators, since they only have real eigenvalues. Complex eigenvalues would not be meaningful from a physicist's point of view. This will be explained in detail later. Moreover, positive semidefinite operators have non-negative eigenvalues. An operator $\Pi \in Pos(\mathcal{X})$ is called a projection if and only if all its eigenvalues are either 0 or 1. $Pro(\mathcal{X})$ denotes the set of all projections acting on $\mathcal{X}$.

4.1.2 Decompositions

All Hermitian operators admit a spectral decompositions, which is also called eigendecomposition. Such a decomposition generally exists for normal operators, $A \in \mathcal{L}(\mathcal{X})$:

$AA^* = A^*A$, on complex Euclidean spaces. Let $\text{spec}(A) = \{\lambda_1, \ldots, \lambda_n\}$ and denote by $\lambda'_1, \ldots, \lambda'_k$ the distinct eigenvalues of A, then there exist an orthonormal basis $\{x_1, \ldots, x_n\}$ and orthogonal projection operators $P_1, \ldots, P_k$ such that

$$A = \sum_{j=1}^{n} \lambda_j x_j x_j^* = \sum_{j=1}^{k} \lambda'_j P_j, \tag{4.1}$$

where the P_j satisfy $\sum_j P_j = \mathbb{1}_{\mathcal{X}}$. In matrix notation (4.1) can be formulate equivalently as $A = UDU^*$, where D is a diagonal matrix with the eigenvalues as entries and U is the unitary matrix, which consist of the eigenvectors.
Another important decomposition is the singular value decomposition. Let $\mathcal{X}, \mathcal{Y}$ be complex Euclidean spaces and $A \in \mathcal{L}(\mathcal{X}, \mathcal{Y})$ a non-zero operator of rank r, then there exist real positive numbers $s_1 \ldots, s_r$ and orthonormal sets $\{x_1 \ldots x_r\} \subset \mathcal{X}$,$\{y_1, \ldots, y_r\} \subset \mathcal{Y}$ satisfying

$$A = \sum_{j=1}^{r} s_j x_j y_j^*. \tag{4.2}$$

In matrix notation (4.2) can be written equivalently as $A = UDV^*$. Here D is a diagonal matrix with the singular values (zeros are included now) as entries. And U, V are unitary matrices, since their column vectors form orthonormal bases. The spectral decomposition is related to the singular value decomposition as

$$s_j(A) = \sqrt{\lambda_j(A^*A)} = \sqrt{\lambda_j(AA^*)},$$

where $s_j(A)$ and $\lambda_j(A)$ refer to the singular values and eigenvalues of some matrix A, respectively. The right singular vectors $\{x_1, \ldots, x_r\}$ of A are eigenvectors of A^*A, whereas the left singular vectors $\{y_1, \ldots, y_r\}$ are eigenvectors of AA^*.
These decompositions are useful for various reasons. Using the eigenvalue decomposition one can see, that eigenvalues of Hermitian operators are real, even though the underlying Hilbert space is complex. Moreover, the spectral decomposition is useful in proving the following lemma:

Lemma 1. For a linear operator $A \in \mathcal{L}(\mathcal{X})$, with $0 \leq A \leq \mathbb{1}_{\mathcal{X}}$, and every $\beta \in \mathbb{R}$ the following statements hold:

$$\exp(\beta A) \leq \mathbb{1} + \beta \exp(\beta) A, \tag{4.3}$$
$$\exp(-\beta A) \leq \mathbb{1} - \beta \exp(-\beta) A. \tag{4.4}$$

Note that (4.3) and (4.4) are no inequalities in the usual sense but rather statements about the positive semidefiniteness of matrices. This will not be mentioned every time throughout the thesis, since this notation is used frequently. Sometimes we will also call these hermiticity relations matrix inequalities or even inequalities. Since actual matrix inequalities, which are inequalities for the individual entries, will not be used in this thesis at all this terminology should not cause any confusion.

Proof. The exponential function is extended to linear operators in the usual way: $\exp(A) = \sum_{i=1}^{\infty} A^i/i!$. Due to the restrictions on A, this sum converges, as $A^i \leq \mathbb{1}_{\mathcal{X}}, \forall i \in \mathbb{N}$ and $\sum_{i=0}^{\infty} 1/i! = \mathrm{e}$. Since any positive semidefinite operator is normal, the spectral decomposition of A implies

$$\exp(A) = \exp(UDU^*) = \sum_{i=1}^{\infty} \frac{(UDU^{-1})^i}{i!} = U\left(\sum_{i=1}^{\infty} \frac{D^i}{i!}\right)U^{-1}.$$

Once we plug this result into (4.3), we just have to prove the statement for a scalar $\alpha \in [0,1]$ instead of the linear operator A. Furthermore, we can assume $\alpha > 0$ since $\alpha = 0$ satisfies both inequalities with equality. According to the mean value theorem there exists $\alpha_0 \in [0,1]$ such that

$$\frac{\exp(\beta\alpha) - 1}{\alpha} = \beta\exp(\beta\alpha_0) \leq \beta\exp(\beta).$$

This proves the first inequality, the second one is proven analogously. □

Lemma 1 was stated and proven in [JJUW09]. The following lemma is also due to the spectral decomposition:

Lemma 2. Let $P \in Pos(\mathcal{X})$ be non zero and $A, B \in \mathcal{L}(\mathcal{X}, \mathcal{Y})$, s.t. $A^*A = B^*B = P$, then there exists a unitary operator $U \in \mathcal{L}(\mathcal{Y})$, such that $AU = B$.

Proof. Consider the spectral decomposition of P

$$P = \sum_{j=1}^{r} \lambda_j x_j^* x_j,$$

where r is the rank of P, $\lambda_1 \ldots, \lambda_r > 0$ are the non-zero eigenvalues and the corresponding eigenvectors $\{x_1, \ldots, x_r\} \subset \mathcal{X}$ form an orthonormal set. Then orthonormal sets $\{y_1, \ldots, y_r\} \subset \mathcal{Y}$ and $\{z_1, \ldots, z_r\} \subset \mathcal{Y}$ exist such that

$$A = \sum_{j=1}^{r} \sqrt{\lambda_j} x_j^* y_j \qquad \text{and} \qquad B = \sum_{j=1}^{r} \sqrt{\lambda_j} x_j^* z_j.$$

If you extend $\{y_1, \ldots, y_r\}$ and $\{z_1, \ldots, z_r\}$ to bases of $\mathcal{Y}$, the unitary operator U, satisfying $U = \sum_{j=1}^{n} y_j^* z_j$, will meet our needs. □

Dividing $P \in Pos(\mathcal{X})$ in this way is referred to as the square root decomposition. In general this decomposition exists if and only if $\dim(\mathcal{Y}) \geq \mathrm{rank}(P)$. This statement can be derived from the singular value decomposition as well. Therefore, Lemma 2 justifies the use of the square root for matrices $A = \sqrt{P}$ as the different solutions are unitary equivalent. Actually Lemma 2 is mainly proven to guarantee insight into square roots of matrices, since they are often used at later chapters.

Furthermore, the spectral decomposition ensures the existence of the polar decomposition:

$$\forall A \in \mathcal{L}(\mathcal{X},\mathcal{Y}) \exists B \in \mathcal{L}(\mathcal{X},\mathcal{Y}), P \in Pos(\mathcal{X},\mathcal{X}) : A = BP,$$

with $P = \sqrt{A^*A}$ and $BB^* = \Pi_{\mathrm{Im}(P)}$, where $\Pi_{\mathrm{Im}(P)}$ is a projection on the image of P. This version is sometimes also called the right polar decomposition. There is also a left polar decomposition: $A = P'B$. Either one is unique if and only if A is reversible. Its existence can be proven by the singular value decomposition, namely $A = VDW^*$. If we define $P = WDW^*$ and $B = VW^*$ we get the right polar decomposition, whereas $P' = VDV^*$ gives the left one. Notice that the dimension of $\mathcal{Y}$ provides an upper bound on the rank of A^*A. And if $A \in \mathcal{L}(\mathcal{X})$ then $B \in U(\mathcal{X})$.

4.1.3 The Dirac notation

Due to historic facts the Dirac notation is widely used in quantum physics. In this thesis it is used to describe complex vectors. There are bras $|\cdot\rangle$ and kets $\langle\cdot|$. They combine either to an inner product $\langle\cdot|\cdot\rangle : \mathbb{C}^n \otimes \mathbb{C}^n \to \mathbb{C}$ or an outer product $|\cdot\rangle\langle\cdot| : \mathbb{C}^n \otimes \mathbb{C}^n \to \mathbb{C}^{n\times n}$. But one can also "line them up" indicating a tensor product

$$|x\rangle|y\rangle = |x\rangle \otimes |y\rangle = |xy\rangle \in \mathbb{C}^{nm} \qquad \forall |x\rangle \in \mathbb{C}^n, |y\rangle \in \mathbb{C}^m.$$

Formally the Dirac notation is defined on finite dimensional Hilbert spaces in the following way. Let S denote some finite set and $\mathcal{H}_S$ the Hilbert space of dimension $|S|$, such that each $h \in \mathcal{H}_S$ is a mapping $h : S \to \mathbb{C}$. For each $s \in S$, $|s\rangle$ represents the unit vector corresponding to the map that satisfies

$$h(s) = 1 \qquad \text{and} \qquad h(s') = 0, \forall s' \neq s.$$

In this case arbitrary vectors like $|\psi\rangle$ can be written as linear combinations of the orthonormal basis $\{|s\rangle : s \in S\}$. Then the bra $\langle\psi|$ can be defined as linear functionals satisfying $\langle\psi| : |\phi\rangle \to \langle\psi|\phi\rangle, \forall \phi \in \mathcal{H}_S$, where $\langle\cdot|\cdot\rangle$ is the inner product of the Hilbert space $\mathcal{H}_S$.
Later binary numbers will be used in the Dirac notation. This refers to the j-th vector of the standard basis, if they are ordered lexicographically. For example we can describe an 8-dimensional complex vector space by a binary number with three figures. Therefore, $|000\rangle, |001\rangle, |010\rangle|011\rangle \ldots |111\rangle$ correspond to $e_1, e_2, e_3, e_4 \ldots e_8 \in \mathbb{C}^8$, respectively. Moreover, we will also use the notation $|j\rangle_{\mathcal{X}}$ to describe the j-th standard basis vector of the Hilbert space $\mathcal{X}$.

4.1.4 Norms and superoperators

With the matrix operations addition and scalar multiplication $\mathcal{L}(\mathcal{X})$ is a linear space. This can be illustrated by the trace $\mathrm{tr} : \mathcal{L}(\mathcal{X}) \to \mathbb{K}$, which is known from linear algebra

but generalized to linear operators on Hilbert spaces over the body $\mathbb{K}$. Let A be a matrix representation of some linear operator in $\mathcal{L}(\mathcal{X})$, then the trace is defined as

$$\mathrm{tr}(A) = \sum_{j=1}^{n} a_{jj} = \sum_{j=1}^{n} \lambda_j(A).$$

The trace is invariant under orthonormal basis changes. This definition allows us to characterize the last subset of linear operators of interest to this thesis, i.e. the set of density operators $\mathcal{D}(\mathcal{X})$. It includes all positive semidefinite operators with unit trace. With the notation of the previous section an operator norm can be derived from the l_2-norm $\|\cdot\|$ on $\mathcal{X}$ for all $A \in \mathcal{L}(\mathcal{X})$

$$\|A\| = \sup_{|\psi\rangle \in \mathcal{X}\setminus\{0\}} \frac{\|A|\psi\rangle\|}{\||\psi\rangle\|}.$$

When this norm is applied to a matrix A it is also called the spectral norm since $\|A\| = \sqrt{\lambda_{max}(A * A)}$, where $\lambda_{max}(\cdot)$ refers to the maximum eigenvalue of its argument. In the line of exploring the linear space $\mathcal{L}(\mathcal{X})$, we construct a conjugate symmetric positive definite sesquilinear form: $\langle A, B\rangle = \mathrm{tr}(A^*B)$, for $A, B \in \mathcal{L}(\mathcal{X})$. This is called the Hilbert-Schmidt inner product, it induces the Frobenius norm, $\|A\|_{\mathrm{Fr}} = \sqrt{\mathrm{tr}(A^*A)}$, which is a Schatten 2-norm. In general the Schatten p-norms are defined for $A \in \mathcal{L}(\mathcal{X}, \mathcal{Y})$ as

$$\|A\|_p = \left(\mathrm{tr}\left((A^*A)^{p/2}\right)\right)^{1/p}.$$

The Schatten 1-norm for $\mathcal{L}(\mathcal{X})$ is the trace norm $\|A\|_{\mathrm{tr}} = \mathrm{tr}(\sqrt{A^*A})$. The trace norm will often serve our future needs better than other Schatten norms. In general, a bounded linear operator is an element of the trace class if its trace norm is finite. However, this is always the case here, since the Hilbert spaces under consideration are finite dimensional anyway. Taking A^*A's positive semidefiniteness into account the trace of the square root is positive and there exist unitary operators transforming them into each other due to Lemma 2. Therefore, the trace norm is well defined. It is also easy to verify the definiteness, the homogeneity and the triangle inequality for the trace norm.
In the following lemma we will characterize the trace norm by unitary operators:

Lemma 3. For all $A \in \mathcal{L}(\mathcal{X})$ we have

$$\|A\|_{\mathrm{tr}} = \max_{U \in \mathcal{U}(\mathcal{X})} |\mathrm{tr}(UA)|. \tag{4.5}$$

Proof. Due to the left polar decomposition of A we can conclude

$$\begin{aligned} |\mathrm{tr}(AU)| &= |\mathrm{tr}(PVU)| = |\mathrm{tr}(P^{1/2}P^{1/2}VU)| \\ &\leq \sqrt{\mathrm{tr}(P)\mathrm{tr}(U^*V^*PVU)} = \mathrm{tr}(P) = \mathrm{tr}(\sqrt{A^*A}). \end{aligned}$$

Here the inequality is the Cauchy-Schwarz inequality ($|\langle A, B\rangle|^2 \leq \langle A, A\rangle\langle B, B\rangle$) for the Hilbert-Schmidt inner product. Once we choose $U = V^*$ equality is achieved. □

Extending the norm to a metric $d(x,y) = \|y - x\|_{\mathrm{tr}}$ gives insight into $\mathcal{L}(\mathcal{X})$ being a Hilbert space. In order to describe the quantum algorithms in the main part operators will not suffice. We need superoperators, mappings acting on operator spaces. The notation $T(\mathcal{X},\mathcal{Y})$ refers to the set of all linear superoperators $\Psi : \mathcal{L}(\mathcal{X}) \to \mathcal{L}(\mathcal{Y})$, again $T(\mathcal{X})$ denotes $T(\mathcal{X},\mathcal{X})$. Keep in mind that the superoperators under consideration, can be thought of as high dimensional matrices. If $\Psi \in T(\mathcal{X})$, then $\Psi \in \mathbb{K}^{\dim(\mathcal{X})^2 \times \dim(\mathcal{X})^2}$. On first sight it might seem odd to come up with a new name (superoperator) since it is just a linear operator. But its entirely different usage justifies this terminology which is standard in quantum computation.
The trace norm for such a superoperator Ψ can be derived from the one for operators in the following way:

$$\|\Psi\|_{\mathrm{tr}} = \sup_{A \in \mathcal{L}(\mathcal{X}) \setminus \{0\}} \frac{\|T(A)\|_{\mathrm{tr}}}{\|A\|_{\mathrm{tr}}}.$$

In order to define a more useful norm for superoperators we need tensor products. In general a tensor product space $\mathcal{X} \otimes \mathcal{Y}$ of two vector spaces $\mathcal{X}, \mathcal{Y}$ needs a universal bilinear function $\Phi_u : \mathcal{X} \times \mathcal{Y} \to \mathcal{X} \otimes \mathcal{Y}$. This provides a minimal space $\mathcal{X} \otimes \mathcal{Y}$, and for any choice of a vector space V and a bilinear function $\Phi : \mathcal{X} \times \mathcal{Y} \to V$ a linear mapping $A : V \to \mathcal{X} \otimes \mathcal{Y}$, such that $\Phi_u = A \circ \Phi$. Analogously, tensor products of three or more spaces relate to multilinear functions. For the description of quantum information finite dimensional complex spaces suffice. A tensor product operating on them is the Kronecker product

$$\begin{aligned} \otimes : \ & \mathcal{X} \times \mathcal{Y} \to \mathcal{X} \otimes \mathcal{Y} \\ & (x \otimes y)_{i,j} = x_i y_j, \end{aligned}$$

for all $x \in \mathcal{X}, y \in \mathcal{Y}$ and all elements i, j of some index sets I, J, respectively. Distributive laws hold for the scalar multiplication and vector addition of Kronecker products. They also extend to spaces of linear operators

$$\begin{aligned} \otimes : \ & \mathcal{L}(\mathcal{X}_1,\mathcal{Y}_1) \times \mathcal{L}(\mathcal{X}_2,\mathcal{Y}_2) \to \mathcal{L}(\mathcal{X}_1,\mathcal{X}_2) \otimes \mathcal{L}(\mathcal{Y}_1,\mathcal{Y}_2) \\ & (A \otimes B)_{(i,j)(k,l)} = A_{i,j} B_{k,l}, \end{aligned}$$

for all $A \in \mathcal{L}(\mathcal{X}_1,\mathcal{Y}_1), B \in \mathcal{L}(\mathcal{X}_2,\mathcal{Y}_2)$, and all i, j, k, l from the respective index sets. We define the linear operator $A \otimes B$ to be the unique linear mapping satisfying

$$A \otimes B(x \otimes y) = (Ax) \otimes (By),$$

for all $x \in \mathcal{X}, y \in \mathcal{Y}$. This gives an universal bilinear form and therefore an identification of the tensor product space $\mathcal{L}(\mathcal{X}_1,\mathcal{X}_2) \otimes \mathcal{L}(\mathcal{Y}_1,\mathcal{Y}_2)$ with $\mathcal{L}(\mathcal{X}_1 \otimes \mathcal{X}_2, \mathcal{Y}_1 \otimes \mathcal{Y}_2)$. This product can be exemplified as

$$A \otimes B = \begin{pmatrix} a_{11}B & a_{12}B & \cdots & a_{1n}B \\ a_{21}B & a_{22}B & \cdots & a_{2n}B \\ \vdots & & \ddots & \vdots \\ a_{m1}B & \cdots & & a_{mn}B \end{pmatrix},$$

for $A \in \mathcal{L}(\mathcal{X}, \mathcal{Y})$, with $\dim(\mathcal{X}) = n$ and $\dim(\mathcal{Y}) = m$.
Dealing with superoperators on tensored operator spaces is not entirely trivial. But luckily we can restrict our analysis to admissible superoperators: An admissible transformation $\Phi : \mathcal{D}(\mathcal{X}) \to D(\mathcal{Y})$ is a map, for which a collection $\{A_1, \ldots, A_k\} \subseteq \mathcal{L}(\mathcal{X}, \mathcal{Y})$ exists that satisfies the conditions

1. $\Phi(\rho) = \sum_{j=1}^{k} A_j \rho A_j^*$ for all $\rho \in \mathcal{D}(\mathcal{X})$,
2. $\sum_{j=1}^{k} A_j^* A_j = \mathbb{1}_{\mathcal{X}}$.

Admissible transformations can be identified with elements of $T(\mathcal{X}, \mathcal{Y})$ as follows: $\Phi(X) = \sum_{j=1}^{k} A_j X A_j^*$ for all $X \in \mathcal{L}(\mathcal{X})$. The trace of a linear operator is the only admissible transformation in $T(\mathcal{X}, \mathbb{C})$ for any Hilbert space $\mathcal{X}$. The collection $\{A_1, \ldots, A_k\}$ are the matrices having a single non-zero entry on the diagonal of value one.
Another admissible transformation, which is widely used in quantum computation is the partial trace. For $\rho \in \mathcal{D}(\mathcal{X} \otimes \mathcal{Y})$ the partial trace $\mathrm{tr}_{\mathcal{Y}} : \mathcal{D}(\mathcal{X} \otimes \mathcal{Y}) \to \mathcal{D}(\mathcal{X})$ is defined for any orthonormal basis $\{|e_1\rangle, \ldots, |e_n\rangle\}$ of $\mathcal{Y}$ as

$$\mathrm{tr}_{\mathcal{Y}}(\rho) = \sum_{j=1}^{n} (\mathbb{1}_{\mathcal{X}} \otimes \langle e_j|)\rho(\mathbb{1}_{\mathcal{X}} \otimes |e_j\rangle).$$

Since this operation is essential to most of the quantum algorithms studied later, the following statement will be used repeatedly

$$\forall A \in L(\mathcal{X} \otimes \mathcal{Y}) : \|\mathrm{tr}_{\mathcal{Y}}(A)\|_{\mathrm{tr}} \leq \|A\|_{\mathrm{tr}}. \tag{4.6}$$

It immediately follows from Lemma 3, since the maximum on the right-hand side is taken over a bigger space, as the partial trace drops away all the components belonging to $\mathcal{Y}$. This is sometimes called to "trace out".
In the following lemma an upper bound is provided by a tensor product of a constant operator and a partial trace. Moreover, it relies upon the Pauli matrices, well known in quantum physics.

Lemma 4. Let $A \in Pos(\mathcal{X} \otimes \mathcal{Y})$ with $dim(\mathcal{X}) = 2$, then $A \leq 2\mathbb{1}_{\mathcal{X}} \otimes \mathrm{tr}_{\mathcal{X}}(A)$.

Proof. The Pauli matrices are defined as

$$\sigma_x = \begin{pmatrix} 0 & 1 \\ 1 & 0 \end{pmatrix}, \qquad \sigma_y = \begin{pmatrix} 0 & -i \\ i & 0 \end{pmatrix}, \qquad \sigma_z = \begin{pmatrix} 1 & 0 \\ 0 & -1 \end{pmatrix}.$$

All of them are Hermitian and therefore

$$(\sigma_x \otimes \mathbb{1}_{\mathcal{Y}})A(\sigma_x \otimes \mathbb{1}_{\mathcal{Y}})$$

is positive semidefinite. for all $i \in \{1,2,3\}$. Moreover, we divide A into four square matrices A_{11}, A_{12}, A_{21} and A_{22} in the obvious way to conclude

$$\begin{aligned}
&2\mathbb{1}_{\mathcal{X}} \otimes \mathrm{tr}_{\mathcal{X}}(A) = 2\begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix} \otimes \left((\mathbb{1}_{\mathcal{Y}} \otimes \langle 0|)\, A((\mathbb{1}_{\mathcal{Y}} \otimes |0\rangle) + (\mathbb{1}_{\mathcal{Y}} \otimes \langle 1|)\, A((\mathbb{1}_{\mathcal{Y}} \otimes |1\rangle)\right) \\
&= \begin{pmatrix} 2 & 0 \\ 0 & 2 \end{pmatrix} \otimes (A_{11} + A_{22}) = A + \begin{pmatrix} A_{22} & A_{21} \\ A_{12} & A_{11} \end{pmatrix} + \begin{pmatrix} A_{22} & -A_{21} \\ -A_{12} & A_{11} \end{pmatrix} + \begin{pmatrix} A_{11} & -A_{12} \\ -A_{21} & A_{22} \end{pmatrix} \\
&= A + \sum_{i=1}^{3} (\sigma_i \otimes \mathbb{1}_{\mathcal{Y}}) A (\sigma_i \otimes \mathbb{1}_{\mathcal{Y}}) \geq A.
\end{aligned}$$

□

The partial trace can also be used to define the diamond norm $\|\cdot\|_\diamond : T(\mathcal{X}, \mathcal{Y}) \to \mathbb{C}$ by

$$\|\Phi\|_\diamond = \inf_{A,B \in L(\mathcal{X}, \mathcal{Y} \otimes \mathcal{Z})} \left\{ \|A\| \|B\| : \Phi = \mathrm{tr}_{\mathcal{Z}}(A \cdot B^*) \right\}, \tag{4.7}$$

for an arbitrary Hilbert spaces $\mathcal{Z}$, such that $\dim(\mathcal{X})\dim(\mathcal{Y}) \leq \dim(\mathcal{Z})$. Since this definition is not very intuitive we will use an equivalent definition relying upon the trace norm for superoperators. For $\Phi \in T(\mathcal{X}, \mathcal{Y})$ we define

$$\|\Phi\|_\diamond = \|\Phi \otimes \mathbb{1}_{\mathcal{L}(\mathcal{X})}\|_{\mathrm{tr}} = \max_{X \in \mathcal{L}(\mathcal{X} \otimes \mathcal{X})} \left\{ \|(\Phi \otimes \mathbb{1}_{\mathcal{L}(\mathcal{X})}(X)\|_{\mathrm{tr}} : \|X\|_{\mathrm{tr}} = 1 \right\}. \tag{4.8}$$

For the proof of QIP = QIP(3), which is explained in the main part, it is important to note that the diamond norm is multiplicative with respect to tensor products:

Lemma 5. Let $\Phi_1 \Phi_2 \in T(\mathcal{X}, \mathcal{Y})$ be any superoperators, then $\|\Phi_1 \otimes \Phi_2\|_\diamond = \|\Phi_1\|_\diamond \|\Phi_2\|_\diamond$.

Proof. Let X_1 and X_2 be linear operators acting on the tensor product $\mathcal{X} \otimes \mathcal{X}$, such that $\|X_1\|_{\mathrm{tr}} = \|X_2\|_{\mathrm{tr}} = 1$, $\|\Phi_1\|_\diamond = \| \left(\Phi_1 \otimes \mathbb{1}_{\mathcal{L}(\mathcal{X})}\right)(X_1)\|_{\mathrm{tr}}$ and $\|\Phi_2\|_\diamond = \| \left(\Phi_2 \otimes \mathbb{1}_{\mathcal{L}(\mathcal{X})}\right)(X_2)\|_{\mathrm{tr}}$, then $\|X_1 \otimes X_2\|_{\mathrm{tr}} = 1$ and therefore

$$\begin{aligned}
\|\Phi_1 \otimes \Phi_2\|_\diamond &= \|\Phi_1 \otimes \Phi_2 \otimes \mathbb{1}_{\mathcal{L}(\mathcal{X})} \otimes \mathbb{1}_{\mathcal{L}(\mathcal{X})}\|_{\mathrm{tr}} \\
&\geq \| \left(\Phi_1 \otimes \Phi_2 \otimes \mathbb{1}_{\mathcal{L}(\mathcal{X})} \otimes \mathbb{1}_{\mathcal{L}(\mathcal{X})}\right)(X_1 \otimes X_2)\|_{\mathrm{tr}} \\
&= \| \left(\Phi_1 \otimes \mathbb{1}_{\mathcal{L}(\mathcal{X})}\right)(X_1) \otimes \left(\Phi_2 \otimes \mathbb{1}_{\mathcal{L}(\mathcal{X})}\right)(X_2)\|_{\mathrm{tr}} \\
&= \| \left(\Phi_1 \otimes \mathbb{1}_{\mathcal{L}(\mathcal{X})}\right)(X_1)\|_{\mathrm{tr}} \| \left(\Phi_2 \otimes \mathbb{1}_{\mathcal{L}(\mathcal{X})}\right)(X_2)\|_{\mathrm{tr}} = \|\Phi_1\|_\diamond \|\Phi_2\|_\diamond.
\end{aligned}$$

We proof the reverse inequality using the original definition of the diamond norm, (4.7), as

$$\begin{aligned}
\|\Phi_1 \otimes \Phi_2\|_\diamond &= \inf \left\{ \|A\| \|B\| : \Phi_1 \otimes \Phi_2 = \mathrm{tr}_{\mathcal{Z}}(A \cdot B^*) \right\} \\
&\leq \inf \left\{ \|A_1\| \|B_1\| : \Phi_1 = \mathrm{tr}_{\mathcal{Z}_1}(A_1 \cdot B_1^*) \right\} \cdot \inf \left\{ \|A_2\| \|B_2\| : \Phi_2 = \mathrm{tr}_{\mathcal{Z}_2}(A_2 \cdot B_2^*) \right\} \\
&= \|\Phi_1\|_\diamond \|\Phi_2\|_\diamond,
\end{aligned}$$

where the first infimum is taken over all $A, B \in \mathcal{L}(\mathcal{X} \otimes \mathcal{X}, \mathcal{Y} \otimes \mathcal{Y} \otimes \mathcal{Z})$. The other infima are taken over all $A_1, B_1 \in \mathcal{L}(\mathcal{X}, \mathcal{Y} \otimes \mathcal{Z}_1)$ and $A_2, B_2 \in \mathcal{L}(\mathcal{X}, \mathcal{Y} \otimes \mathcal{Z}_2)$, respectively. Notice that $\dim(\mathcal{Z}_1)$ and $\dim(\mathcal{Z}_2)$ are at least $\dim(\mathcal{X})\dim(\mathcal{Y})$ while $\dim(\mathcal{Z}) \geq$

$\dim^2(\mathcal{X})\dim^2(\mathcal{Y})$. Therefore, choosing $\mathcal{Z} = \mathcal{Z}_1 \otimes \mathcal{Z}_2$ respects the original definition, (4.7). Thus, the above inequality is due to the fact that the infimum over the tensor product space is at most the product over the individual infima. □

A proof of Lemma 5 was published by Dorit Aharonov, Alexei Kitaev and Noam Nisan [AKN98]. Since they only relied upon the original definition of the diamond norm a proof for the equivalence of (4.7) and 4.8 can be found in their paper as well.

4.1.5 The vector mapping

In the later chapters we will often rely upon the linear mapping $\mathrm{vec} : \mathcal{L}(\mathcal{X}, \mathcal{Y}) \to \mathcal{X} \otimes \mathcal{Y}$, which is called the vector mapping in this thesis. It transforms matrices into vectors by lining the entries up row wise in increasing order. Here we state some properties of the vector mapping:

1. For all appropriate choices of linear operators A, B and X such that AXB^t exists, the following holds
$$(A \otimes B)\mathrm{vec}(X) = \mathrm{vec}(AXB^t).$$

2. The vector mapping transforms kets into bras: For all $|\psi\rangle \in \mathcal{X}$, and $|\phi\rangle \in \mathcal{Y}$ we have
$$\mathrm{vec}(|\psi\rangle\langle\phi|) = |\psi\rangle|\phi\rangle.$$

3. For all finite Hilbert spaces $\mathcal{X}, \mathcal{Y}$ and all linear operators $A, B \in \mathcal{L}(\mathcal{X}, \mathcal{Y})$ we get
$$\begin{aligned} \mathrm{tr}_{\mathcal{X}}(\mathrm{vec}(A)\mathrm{vec}(B)^*) &= AB^*, \text{ and} \\ \mathrm{tr}_{\mathcal{Y}}(\mathrm{vec}(A)\mathrm{vec}(B)^*) &= (B^*A)^t. \end{aligned}$$

Properties 1 and 2 can be proven by straightforward calculations. Let $|e_1\rangle, \ldots, |e_n\rangle$ be an orthonormal basis of $\mathcal{X}$, then the first equation of property (3) follows from

$$\begin{aligned} \mathrm{tr}_{\mathcal{X}}(\mathrm{vec}(A)\mathrm{vec}(B)^*) &= \sum_{j=1}^{n} (\mathbb{1}_{\mathcal{Y}} \otimes \langle e_j|)\,\mathrm{vec}(A)\mathrm{vec}(B)^*\,(\mathbb{1}_{\mathcal{Y}} \otimes |e_j\rangle) \\ &= \sum_{j=1}^{n} \mathrm{vec}(A|e_j\rangle)\mathrm{vec}(B|e_j\rangle)^* = \sum_{j=1}^{n} A|e_j\rangle\langle e_j|B^* = AB^*, \end{aligned}$$

where the second equality is due to property 1. The remaining equation in property 3 is proven analogously. This completes the pure mathematical structure needed to describe quantum computation and information. Of course, there is a lot more to come before we can discuss the recent developments in quantum interactive proofs and quantum refereed games.

4.2 Classical computation

Throughout this thesis a robust definition for polynomial time computable functions is required. To this end we use the following definition, which was also used by Kitaev and Watrous [KW00], for instance.

Definition 1. A function $f : \mathbb{Z}^+ \to \mathbb{N}$ is in the class *poly* if

1. $\exists$ polynomial $p, \forall n \in \mathbb{Z}^+ : f(n) \leq p(n)$,
2. $f(n)$ is computable in polynomial time.

This is a robust definition since it allows all kind of sub-polynomial functions and guarantees their polynomial time computability as well. Moreover, we will utilize the Landaus symbol $\Omega(\cdot)$, specifying an asymptotical lower bound. For functions $f, g : \mathbb{R} \to \mathbb{R}$ we define

$$f \in \Omega(g) \Leftrightarrow 0 < \liminf_{x \to a} \frac{f(x)}{g(x)} \leq \infty,$$

where $a \in \mathbb{R} \cup \{-\infty, \infty\}$. It is also common in complexity theory to insert classes of functions instead of individual functions. For example $\mathcal{O}(poly)$ refers to the class of functions, which are polynomially bounded, using the standard big O notation. We will need the Landaus symbol Ω to guarantee that a parameter δ is at least inverse polylogarithmic in the size of the input $n \in \mathbb{N}$:

$$\delta = \Omega\left(\frac{1}{\text{polylog}(n)}\right),$$

where polylog(n) is the class of functions, which are polynomials of logarithms. Therefore, all functions $f \in \text{polylog}(n)$ obey $f \leq (\log n)^k$ for some $k \in \mathbb{N}$.

4.2.1 Classical complexity classes

We will have to deal with a couple of classical complexity classes like interactive proofs (IP), deterministic polynomial bounded space (PSPACE), or NC, describing functions, which can be computed by polynomially sized Boolean circuits of polylogarithmic depth. The name NC stands for Nick's class in honor of professor Nick Pippenger, who proved several results in parallel computation. Moreover, we deal with two classes of classical refereed games RG and RG(2). The class RG consists of refereed zero-sum games for two players, while problems in RG(2) can be solved by the two round version of such a game. Furthermore, AM and MA refer to the Arthur-Merlin and Merlin-Arthur class, respectively. MA is the randomized version of NP, whereas the more powerful class AM even contains statistically zero-knowledge proofs (SZP).
The exact definitions of all these complexity classes are not provided since they are essentially the same as their quantum counterparts. Except that the participants are only allowed to use classical computation, but neither quantum computers nor quantum information. The quantum counterparts will be defined in detail at later chapters.

Furthermore, the standard notation is used for problems solvable in polynomial time (P), in nondeterministic polynomial time (NP), and in exponential time (EXP). Detailed information on classical complexity classes can be found in the book of Sanjeev Arora and Boaz Barak [AB09], for instance. The following relations between the stated classical complexity classes are known:

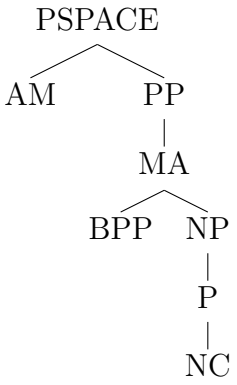
$$\mathrm{NC} \subseteq \mathrm{P} \subseteq \mathrm{NP} \subseteq \mathrm{MA} \subseteq \mathrm{AM} \subseteq \mathrm{PSPACE} \subseteq \mathrm{EXP}.$$

Unfortunately only one subset relation is provably strict: $\mathrm{P} \subset \mathrm{EXP}$. We do not even know if $\mathrm{P} \subset \mathrm{PSPACE}$ holds. Furthermore, we have the following classical characterizations of PSPACE

$$\mathrm{PSPACE} = \mathrm{IP} = \mathrm{NC(poly)} = \mathrm{RG}(2).$$

Here NC(poly) refers to the class of problems, which can be decided by Boolean circuits of polynomial depth. Since all of these equalities are essential to the proofs in the main part, we have to take a closer look.
The containment $\mathrm{PSPACE} \subseteq \mathrm{IP}$, was proven by Shamir in 1992 [Sha92]. He used a technique called arithmetization, which relates quantified Boolean formulas (QBF) to polynomials. The values of the polynomials decide whether the QBF under consideration are true or false. Furthermore, he transformed the polynomials in a way, that made them easy to compute, but only increased their size polynomially. Therefore, the values of these new polynomials can be checked using an interactive protocol with a polynomial number of rounds. These considerations proved $\mathrm{PSPACE} \subseteq \mathrm{IP}$. But some of the credit given to Shamir should be shared with Lance Fortnow, Carsten Lund, Howard Karloff and Noam Nisan. They used very similar ideas to prove the containment of the whole polynomial hierarchy PH in IP, as they found a one-prover protocol for the permanent [FLKN92]. The opposite containment, $\mathrm{IP} \subseteq \mathrm{PSPACE}$, has already been proven before by Christos Papadimitriou [Pap83]. He showed how a verifier can simulate an optimal prover in polynomial space by traversing the tree of all possible interactions and calculating the probabilities recursively.
Moreover, the proof of $\mathrm{PSPACE} \subseteq \mathrm{RG}(2)$ is similar to Shamir's proof of $\mathrm{PSPACE} \subseteq \mathrm{IP}$. For the reverse containment one has to find a near-optimal strategy for one of the players. The implementation in PSPACE can be proven by Savitch's theorem, which is standard in complexity theory and can be found in [AB09], for example. The whole equality $\mathrm{PSPACE} = \mathrm{RG}(2)$ was proven by Uriel Feige and Joe Kilian in 1997 [FK97]. Notice that they used a different notation relying upon rounds of communication instead of turns. Moreover, they included the privacy of the coin tosses, naming RG(2) RG(private,1). Actually we also know that refereed games with a polynomial number of rounds (RG) are equivalent to EXP. First $\mathrm{RG} \subseteq \mathrm{EXP}$ was proven by fast algorithms for game trees [KM90], [KMvS94], before the reverse containment was established in 1997 [FK97].
The equality NC(poly) = PSPACE was proven by Allan Borodin in 1977 [Bor77]. He connected the space a Turing machine uses to the depth of a circuit: $\mathrm{Depth}(s(n)) \subseteq \mathrm{DSPACE}(s(n)) \subseteq \mathrm{Depth}(s(n)^2)$.
These results can be summarized in the following diagram. The classes at the bottom

are included in the upper ones. The lines connecting the classes stand for these subset relations. The diagrams at hand are standard in complexity theory and therefore explanations can probably be found in any scientific book regarding this topic.



Actually, this form of presentation is only partly meaningful here. Nevertheless it is quite useful in order to compare these classes to their quantum analogues. Moreover, AM also includes MA since the multiple rounds of communication do not increase the computational power of AM [BM88] and therefore a protocol for AM[3] can simulate one for MA.
Since we mentioned the class PP it has to be discussed shortly. To this end keep in mind that the counting class #P can basically count the number of accepting paths from a nondeterministic Turing machine. The class PP is the decision class analogue to #P. It computes the most significant bit of a function in #P. The class PP was introduced by Gill in 1977 [Gil77]. More information on PP and #P can be found in a paper of Lance Fortnow [For97], for instance. Furthermore, the class PP can be defined analogously to the standard class BPP, which stands for bounded-error probabilistic polynomial time. The only difference is that both the soundness and completeness are exactly $1/2$ instead of arbitrarily close to $1/2$. These considerations explain why a soundness and a completeness of exactly $1/2$ dramatically increase the computational power of a complexity class.
In the main part we will utilize a few more results from classical computation. Since this thesis is mainly concerned with quantum computers, the proofs of such well known classical results will be skipped.

4.2.2 Efficient parallel algorithms

To prove the implementation of various algorithms in PSPACE we need a couple of results on parallel computing. Note that NC can be defined equivalently as the class of problems that can be decided in polylogarithmic time by polynomially many processors. These processors are connected in parallel. We state that matrix exponentials, spectral decompositions and projections onto the positive eigenspaces can be found in NC up to

any rational accuracy. Consider the following problems:

Matrix exponentials

Input: An $n \times n$ matrix $A, \eta \in \mathbb{Q}^+$ and $k \in \mathbb{N}$ in unary notation.
Promise: $\|A\| \leq \eta$
Output: An $n \times n$ matrix X, such that $\|\exp(A) - X\| < \eta$.

Spectral decomposition

Input: An $n \times n$ Hermitian matrix A and $\eta \in \mathbb{Q}^+$.
Output: An $n \times n$ unitary matrix U and a $n \times n$ real diagonal matrix D such that $\|A - UDU^*\| < \eta$.

Positive eigenspace projection

Input: An $n \times n$ Hermitian matrix $A, \eta \in \mathbb{Q}^+$.
Output: An $n \times n$ positive semidefinite matrix $P \leq \mathbb{1}$ such that $\|P - \Pi\| < \eta$,where Π is the projection operator onto positive eigenspace of A.

All three problems can be efficiently computed in parallel. This means NC algorithms exist for these problems. In addition exact NC algorithms for elementary matrix operations, such as sums, products, tensor products and inverses can be found in the survey of Joachim von zur Gathen [vzG93]. Moreover, von zur Gathen also states NC implementations for iterated sums, as well as the trace and the partial trace. The existence of such NC implementations is key to the algorithms discussed in the main part.

4.3 Quantum computation

In this section a simple mathematical introduction to quantum computation is provided at first. Secondly, we briefly discuss essential quantum mechanical properties and examine a slightly more complex approach using density operators. To handle the density operators we define purifications, the fidelity function, and Bures angle. In order to guarantee complete insight into the matter at hand several lemmas and theorems are proven. Moreover, we discuss a couple of additional NC implementations, which we need at later chapters. Finally, a suitable quantum computational model based on small quantum gates is stated.

4.3.1 Basics in quantum computation and information

Initially, we introduce some basic definitions and notations from quantum computation. These can be found in the book of Michael Nielsen and Isaac Chuang [NC00], for instance. Unlike classical computers quantum computers use qubits instead of bits. A qubit can be thought of as an elementary particle, which has two basic states 0 and 1 referring

to some physical property such as energy values or spin. Unlike classical bits, qubits do not only exist in the two basic states. Superposition of these basic or pure states can occur, leaving a qubit in a mixed state $\alpha_0|0\rangle + \alpha_1|1\rangle$, where $\alpha_0, \alpha_1 \in \mathbb{C}$ such that $|\alpha_0|^2 + |\alpha_1|^2 = 1$. Once we measure this qubit it will be in state $|0\rangle$ with probability $|\alpha_0|^2$, and in state $|1\rangle$ with probability $|\alpha_1|^2$. Quantum computers operate on registers of qubits, which are vectors in finite complex Hilbert spaces. As stated before Calligraphic symbols like $\mathcal{X}, \mathcal{Y}$ will be used to describe those Hilbert spaces.
Let us examine an n-qubit register. It has 2^n pure states $|0^n\rangle, |10^{n-1}\rangle, \ldots |0^{n-1}1\rangle, \ldots |1^n\rangle$. Now let $|\psi\rangle$ be some mixed state such that

$$|\psi\rangle = \sum_{j=0}^{2^n} \alpha_j |\phi_j\rangle,$$

where $|\phi_j\rangle$ is the j-th pure state and α_j are complex coefficients for all j, such that $\sum_j |\alpha_j|^2 = 1$. Basically there are two actions we can perform on a quantum register:

1. transformation by unitary operators $U : \mathbb{C}^{2^n} \to \mathbb{C}^{2^n}$
2. quantum measurement by a projection $\Pi : \mathbb{C}^{2^n} \to \mathbb{C}^{2^n}$ that collapses the quantum state and outputs state $|\phi_j\rangle$ with probability $|\alpha_j|^2$.

One can only apply unitary operators because those are the only linear operators that preserve $\sum |\alpha_j|^2 = 1$.
From a physicist's point of view unitary operators are precisely the ones conserving the energy of a quantum state. The fact that one can only use linear operators is due to quantum mechanics. In particular, both properties are due to the Schrödinger equation, which describes the time evolution of a quantum state $|\psi(t)\rangle$ as

$$\hat{H}|\psi(t)\rangle = i\hbar \frac{\partial}{\partial t}|\psi(t)\rangle, \tag{4.9}$$

where $\hbar$ is the reduced Planck constant and $\hat{H}$ is the Hamiltonian operator, which is the sum of the operators corresponding to the kinetic and potential energy of the system. Its spectrum describes all the energy levels of the system and those ought to be real. Therefore, $\hat{H}$ is Hermitian, the eigenvalues $\lambda_1; \ldots, \lambda_n$ of $\hat{H}$ are real, and the corresponding eigenvectors $|\phi_1\rangle, \ldots, |\phi_n\rangle$ can be chosen orthonormal. Once we describe $\psi(t)$ in this eigenbasis

$$|\psi(t)\rangle = \sum_{j=1}^{n} \alpha_j(t)|\phi_j\rangle, \tag{4.10}$$

and plug the result into the Schrödinger equation

$$i\hbar \frac{\partial \sum_{j=1}^{n} \alpha_j(t)|\phi_j\rangle}{\partial t} = \hat{H} \sum_{j=1}^{n} \alpha_j(t)\lambda_j|\phi_j\rangle = \sum_{j=1}^{n} \alpha_j(t)|\phi_j\rangle,$$

we derive the following differential equation

$$i\hbar\frac{\partial\alpha_j(t)}{\partial t} = \lambda_j\alpha_j(t) \Rightarrow \alpha_j(t) = e^{-i\lambda_j t/\hbar}\alpha_j(0), \qquad \forall j \in \{1,\dots,n\}.$$

Reinserting into (4.10) yields

$$|\psi(t)\rangle = \sum_{j=1}^{n} e^{-i\lambda_j t/\hbar}\alpha_j(0)|\phi_j\rangle = \begin{pmatrix} e^{-i\lambda_1 t/\hbar} & & 0 \\ & \ddots & \\ 0 & & e^{-i\lambda_n t/\hbar} \end{pmatrix} \begin{pmatrix} \alpha_1(0) \\ \vdots \\ \alpha_n(0) \end{pmatrix} = U(t)\psi(0)\rangle,$$

where $U(t)$ is obviously a diagonalized unitary operator. Note that the state $|\psi(t)\rangle$ of a quantum system is often the representation for a wave function $\varphi(x,t)$, where x describes the location of a particle and t the time. This is actually one of the reasons why physicists have to think in terms of linear operators instead of matrices. Moreover, a physical quantum computer can not be separated from its environment completely. Therefore, the environment interacts with the quantum computer causing even saved data to vanish over time. Since the above calculation is standard in quantum mechanics, it can probably be found in any scientific book on this topic. The notation used for the derivative is not a perfect solution from a pure mathematical point of view but standard in theoretical physics.
Actually we already applied the three postulates of quantum mechanics. The first postulate says every isolated physical system is associated to a Hilbert space and it is completely described by the unit vectors of this space. The second one describes the time evolution of a closed system, which is given by the Schrödinger equation, (4.9). The third postulate of quantum mechanics is concerned with measurements. It will be explained later. A measurement is the only physically realizable way to get information out of quantum states. Unlike classical computers, quantum computers are probabilistic machines by nature. But the reason for this probabilistic nature is not discovered yet. The term quantum decoherence refers to the loss of coherence, the ordering of the phase angle of the elements in a quantum superposition. A measurement, which coincides with the collapse of the wave function is thought of as an interaction with the nature in an irreversible way. The state that evolves in this process lies in a high dimensional space, depending on the degrees of freedom of the measuring device. If we choose the expansion in a way that the interaction is element specific, almost no interference of the original particle will occur. Therefore, they will be separated from each other with high probability, as they take their own independent paths due to their natural unitary evolution. This consideration also reveals the fact that decoherence is concerned with the transition from quantum physics to classical physics, where untangling leads to a single macroscopic reality.
Nevertheless, the probabilistic nature of quantum states cannot be derived. Today we can only interpret the experimental data. One example is the multi-universe interpretation, claiming that each outcome of a measurement is realized in a different universe. Under certain circumstances they even have to merge in order to rule out the possibility of one observer seeing different universes. But there are arguable weaknesses, as for the

conservation of energy to hold universes have to be weighted according to the probability they "occur" with. Measurements are observer independent in this theory, as it is based upon quantum decoherence. Therefore, it can be used to resolve problems like the EPR paradox or Schrödinger's cat experiment. For further information about the multi-universe interpretation and its connections to quantum computers, see for example [Deu85].

In order to generalize the notion of pure and mixed states, remember the definition of the Dirac notation in Section 4.1.3. S was some finite set while $\mathcal{H}_S$ denoted the Hilbert space of dimension $|S|$, with each $h \in \mathcal{H}_S$ being a mapping $h : S \to \mathbb{C}$. Pure states are unit vectors in $\mathcal{H}_S$. A mixed state of a quantum system is a distribution on not necessarily orthogonal pure states. Moreover, a mixture is defined to be a collection $\{(p_k, |\psi_k\rangle)\}$ with $\sum p_k = 1$. Here each $p_k \geq 0$ and each $|\psi_k\rangle$ is a pure state. The quantum system is with probability p_k in state $|\psi_k\rangle$. Such a mixture is described by a density operator $\rho = \sum_k p_k |\psi_k\rangle\langle\psi_k|$. Obviously $\rho : \mathcal{L}(\mathcal{H}_S) \to \mathcal{L}(\mathcal{H}_S)$ is a linear operator. It has unit trace because $\sum p_k = 1$ and it is positive semidefinite since each $p_k \geq 0$. Different mixtures might yield identical states, in the sense that no measurement can distinguish between them even statistically. But if two mixtures yield different density operators, they can be distinguished statistically, since their probability distributions are different. Therefore, describing mixtures with density operators makes sense.

Before $\mathcal{D}(\mathcal{X})$ was the set of all density operators for some Hilbert space $\mathcal{X}$. For measuring the distance between density operators the trace norm defined in the previous section is suitable. Since transformations of mixed states are examined later, one needs superoperators between spaces of density operators. This is the reason why we can restrict our view to admissible superoperators. The only physically realizable transformations that map $\mathcal{D}(\mathcal{X})$ to $\mathcal{D}(\mathcal{Y})$ are admissible transformations. In the main part we will often use the partial trace, $\mathrm{tr}_{\mathcal{Y}} : \mathcal{D}(\mathcal{X} \times \mathcal{Y}) \to \mathcal{D}(\mathcal{X})$, defined in Section 4.1.4. Sometimes it is also called the "trace out" operation, since the part of the register corresponding to $\mathcal{Y}$ is eliminated. Observe the following equivalent definition for clarification. The partial trace $\mathrm{tr}_{\mathcal{Y}} : \mathcal{L}(\mathcal{X}, \mathcal{Y}) \to \mathcal{L}(\mathcal{X})$ is precisely the mapping that satisfies $\mathrm{tr}_{\mathcal{Y}}(A \otimes B) = \mathrm{tr}(B)A$ for all $A \in \mathcal{L}(\mathcal{X})$ and $B \in \mathcal{L}(\mathcal{Y})$. This can be extended to tensor products of multiple spaces by linearity. The above information is available in the paper of Kitaev and Watrous [KW00].

Note that we can also describe the first and second postulate of quantum mechanics in terms of density operators. The state space is $\mathcal{D}(\mathcal{X})$ instead of $\mathcal{X}$ in this case. Moreover, the time evolution of any density operator ρ to a new density operator ρ' can be described by

$$\rho' = U(t)\rho U(t)^*,$$

where $U(t)$ is the unitary operator derived from the Schrödinger equation, (4.9). Since there is no theory describing measurements in a way, which is useful for quantum computation, besides the collapsing interpretation, we are just going the brute-force way and define measurements, as functions $\mu : I \to Pos(\mathcal{X})$, for some finite nonempty set of measurement outcomes I. For each $a \in I$ there exists a measurement operator Π_a that

corresponds to the outcome a, such that

$$\sum_{a \in I} \Pi_a = \mathbb{1}_{\mathcal{X}} \tag{4.11}$$

holds. If we apply a measurement $\{\Pi_a : a \in I\}$ to a quantum system in the mixed state $\rho \in \mathcal{D}(\mathcal{X})$ the following happens:

1. One element $a \in I$ is chosen. The probability of a being selected is

$$\Pr(a) = \langle \Pi_a, \rho \rangle, \qquad \forall a \in I.$$

2. The register collapses, which means it drops away, resulting in a single state:

$$\frac{\Pi_a \rho \Pi_a^*}{\langle \Pi_a, \rho \rangle}$$

Notice that this definition respects the third postulate of quantum mechanics, which can be described as follows:
A quantum measurement is a set of measurement operators $\{\Pi_a\}$, where a are the possible outcome of the measurement. If the quantum system under consideration is in state $|\psi\rangle$ the probability that outcome a occurs is

$$\Pr(a) = \langle \psi | \Pi_a^* \Pi_a | \psi \rangle = \langle \psi | \Pi_a | \psi \rangle.$$

The second equality is due to the fact that we can restrict our view to projective measurements. This means all the Π_a are projections. After the measurement the system is in state

$$\frac{\Pi_a |\psi\rangle}{\sqrt{\langle \psi | \Pi_a | \psi \rangle}}.$$

Finally, the third postulate also requires the completeness condition, (4.11).
The only difference between these definitions is the use of the density matrix opposed to the direct description in terms of state vectors. Moreover, both definitions can be used to prove the fact that quantum states can only be distinguished if they are orthogonal.
Later on we will also measure just some part of a quantum register. To describe such a measurement let $\rho \in \mathcal{D}(\mathcal{X} \otimes \mathcal{Y})$ be the state of some pair of registers $(\mathbf{X}, \mathbf{Y})$. Furthermore, let $\{\Pi_{a_1} : a_1 \in I_1\} \subset Pos(\mathcal{X})$ be a measurement, that only acts on $\mathbf{X}$ and $\{\Pi_{a_2} : a_2 \in I_2\} \subset Pos(\mathcal{Y})$ a measurement, that only acts on $\mathbf{Y}$. Both measurements are interpreted as functions $\mu_1 : I_1 \to [0,1]$ and $\mu_2 : I_2 \to [0,1]$. First observe the probability vector, corresponding to the product measurement $\mu = (\mu_1, \mu_2)$

$$p(a_1, a_2) = \langle \Pi_{a_1} \otimes \Pi_{a_2}, \rho \rangle,$$

and the probability vector, corresponding to the measurement μ_1

$$p_1(a_1) = \sum_{a_2 \in I_2} p(a_1, a_2) = \langle \mu_1(a_1) \otimes \mathbb{1}_{\mathcal{Y}}, \rho \rangle = \langle \mu_1(a_1), \mathrm{tr}_{\mathcal{Y}}(\rho) \rangle.$$

If we condition the second measurement on μ_1 having outcome a_1, we end up with the following probability vector

$$p_2^{(a_1)}(a_2) = \frac{p(a_1, a_2)}{p_1(a_1)} = \frac{\langle \Pi_{a_1} \otimes \Pi_{a_2}, \rho \rangle}{\langle \Pi_{a_1} \otimes \mathbb{1}_{\mathcal{Y}}, \rho \rangle} = \langle \Pi_{a_2}, \rho_2^{(a_1)} \rangle, \tag{4.12}$$

where

$$\rho_2^{(a_1)} = \frac{\mathrm{tr}_{\mathcal{X}}(\Pi_{a_1} \otimes \mathbb{1}_{\mathcal{Y}})\rho}{\langle \Pi_{a_1} \otimes \mathbb{1}_{\mathcal{Y}}, \rho \rangle}.$$

The last equality of (4.12) is due to $\mathrm{tr}((A \otimes B)C) = \mathrm{tr}(B(\mathrm{tr}_{\mathcal{X}}(A \otimes \mathbb{1}))C)$, which holds for all A, B and C, where $(A \otimes B)C$ exists.
After performing the measurement μ_1 on ρ with outcome a_1 the remaining system is in state $\rho_2^{(a_1)}$. This is a standard way to describe product measurements, it can be found in Nielsen's and Chuang's book [NC00] for example.
In the main part $Meas(\mathcal{X})$ will be used to describe the set of measurement operators within $\mathcal{L}(\mathcal{X})$. Note that this set is compact and convex, since all measurement operators $\Pi \in Meas(\mathcal{X})$ are positive semidefinite and obey $\Pi \leq \mathbb{1}_{\mathcal{X}}$.
This completes the introduction into quantum computation, which was supposed to point out different perspectives on quantum states and how they can be manipulated. It will enable us to immerge further into the subject matter in the following sections.

4.3.2 Purification

For every quantum state described by a density operator $\rho \in \mathcal{D}(\mathcal{X})$ there exist pure states $|\psi\rangle \in \mathcal{D}(\mathcal{X} \otimes \mathcal{Y})$, called purifications of ρ, such that

$$\mathrm{tr}_{\mathcal{Y}}|\psi\rangle\langle\psi| = \rho. \tag{4.13}$$

The existence of many different purifications for one state is due to the fact that the square root decomposition, discussed in Section 4.1.2, is not unique. For every density operator $\rho \in \mathcal{D}(\mathcal{X})$

$$\exists A \in \mathcal{L}(\mathcal{X}, \mathcal{Y}) \text{ and an orthonormal basis } \{|e_1\rangle, \ldots, |e_n\rangle\} \subseteq \mathcal{X} : \rho = AA^* = \sum_{k=1}^{n} |e_k\rangle\langle e_k|.$$

If $\dim(\mathcal{Y}) = m$ is sufficiently large, $n \leq m$, an orthonormal basis $\{e'_1, \ldots, e'_m\}$ exists such that: $|\psi\rangle = \sum_{k=1}^{n} |e_k\rangle \otimes |e'_k\rangle$ satisfies (4.13). This provides the basis for a principle of enormous importance in quantum information, the unitary equivalence of purifications.

Theorem 1. If two quantum states $|\phi\rangle|\psi\rangle \in \mathcal{X} \otimes \mathcal{Y}$ are purifications of the same operator $X \in Pos(\mathcal{X})$. $\mathrm{tr}_{\mathcal{Y}}|\phi\rangle\langle\phi| = X = \mathrm{tr}_{\mathcal{Y}}|\psi\rangle\langle\psi|$ then a unitary operator $U \in \mathcal{U}(\mathcal{Y})$ exists, which transforms them into each other

$$(\mathbb{1}_{\mathcal{X}} \otimes U)|\phi\rangle = |\psi\rangle.$$

Moreover, U can be computed in NC.

Proof. Lemma 2 states the existence of a unitary operator $V \in \mathcal{U}(\mathcal{Y})$ transforming A into B: $AV = B$ for matrices A, B with $AA^* = BB^*$. It is just the change of the orthonormal basis providing their singular value decomposition. Once you choose $A, B \in \mathcal{L}(\mathcal{X}, \mathcal{Y})$, such that $\text{vec}(A) = |\psi\rangle, \text{vec}(B) = |\phi\rangle$, applying property 3 about the vector mapping (see Section 4.1.5) gives

$$AA^* = \text{tr}_{\mathcal{Y}}|\psi\rangle\langle\psi| = X = \text{tr}_{\mathcal{Y}}|\phi\rangle\langle\phi| = BB^*.$$

Using Lemma 2 and property 1 about the vector mapping, we end up with

$$\text{vec}(B) = \text{vec}(AV) = \text{vec}(\mathbb{1}_{\mathcal{X}}AV) = (\mathbb{1}_{\mathcal{X}} \otimes V^t)\text{vec}(A),$$

therefore $U = V^t$ achieves the task, proving the unitary equivalence of purifications.
In order to compute U in NC observe the singular value decompositions of A and B, which can be computed approximately in NC according to Section 4.2:

$$A = S_1 D_1 T_1 \text{ and } B = S_2 D_2 T_2,$$

where $S_1, S_2 \in U(\mathcal{X})$, $T_1, T_2 \in U(\mathcal{Y})$ and $D_1, D_2 \in \mathcal{L}(\mathcal{Y}, \mathcal{X})$ are diagonal matrices. If we could compute the singular value decomposition exactly we would get $S_1 = S_2$ and $D_1 = D_2$ immediately. Therefore, we could choose $U^t = T_1^* T_2$ to end up with $AU^t = B$ and thus $(\mathbb{1}_{\mathcal{X}} \otimes U)|\phi\rangle = |\psi\rangle$ as desired. But since the singular value decomposition is not exact we have to choose $U^t = T_1^* S_1^* V' S_2 T_2$, with $V' \in U(\mathcal{Y})$, such that $S_2 D_2^* S_2^* S_1 D_1 S_1^* V$ is positive semidefinite. Such a choice will enable us to prove that $(\mathbb{1}_{\mathcal{X}} \otimes U)|\phi\rangle$ and $|\psi\rangle$ are close to each other in terms of the trace norm. But the closeness we desire highly depends on the situation, in which the unitary operator U is used. Thus, we will do this adjustment for the case at hand, when we need it. □

Since the unitary equivalence of purifications is a standard theorem in quantum computation it can probably be found in any scientific book on this topic, see for example [NC00].

4.3.3 The fidelity function

The fidelity function is used to describe the distance between density operators. It was first mentioned by Richard Jozsa in 1994 [Joz94]. Roots of positive semidefinite operators acting on Hilbert spaces are not unique, but unitary equivalent and hence norm invariant. Therefore, we can define the fidelity function $F(\rho, \sigma)$ of two density operators $\rho, \sigma \in \mathcal{D}(\mathcal{X})$ as

$$F(\rho, \sigma) = \left\|\sqrt{\rho}\sqrt{\sigma}\right\|_{\text{tr}} = \text{tr}\left(\sqrt{(\sqrt{\sigma})^*(\sqrt{\rho})^*\sqrt{\rho}\sqrt{\sigma}}\right) = \text{tr}\left(\sqrt{(\sqrt{\sigma})^*\rho\sqrt{\sigma}}\right),$$

where $\|\cdot\|_{\text{tr}}$ denotes the trace norm defined in Section 4.1.4. Obviously the fidelity is non-negative and we have $F(\rho, \sigma) = 1$ if and only if $\rho = \sigma$.
With the fidelity function at hand we are able to state Uhlmann‘s theorem, which correlates the fidelity to the trace of some purifications.

Theorem 2. Let $\mathcal{X}, \mathcal{Y}$ be Hilbert spaces such that $\dim(\mathcal{X}) \leq \dim(\mathcal{Y})$. Let $\rho, \sigma \in \mathcal{D}(\mathcal{X})$ be density operators and $|\phi\rangle \in \mathcal{XY}$ any purification of ρ. Then

$$F(\rho, \sigma) = \max\{|\langle \phi, \psi \rangle| : |\psi\rangle \in \mathcal{XY} \text{ is a purification of } \sigma\}. \tag{4.14}$$

This theorem was first proven by Uhlmann in 1976 [Uhl76]. Back then the notation in quantum information was quite different, the fidelity was called transition probability. Therefore, we will not examine the original proof, but instead a proof based on the modern terminology and methods, inspired by a paper from Jozsa [Joz94].

Proof. Observe the polar decomposition of the first density operator $\rho = \sqrt{\rho}A^*$, where $A \in \mathcal{L}(\mathcal{X}, \mathcal{Y}) : AA^* = \Pi_{\mathrm{Im}(\rho)}$. Then a purification $|\phi\rangle \in \mathcal{X} \otimes \mathcal{Y}$ of ρ exists such that $|\phi\rangle = \mathrm{vec}(\sqrt{\rho}A^*)$. Similarly, $\exists B \in \mathcal{L}(\mathcal{X}, \mathcal{Y}) : BB^* = \Pi_{\mathrm{Im}(\sigma)}$ such that $|\psi\rangle = \mathrm{vec}(\sqrt{\sigma}B^*)$ purifies σ. Due to Theorem 1 every purification of σ can be described as

$$|\zeta\rangle = \mathrm{vec}(\sqrt{\sigma}B^*U^*) \text{ for some } U \in \mathcal{U}(\mathcal{Y}).$$

Therefore, we can convert the right hand side of (4.14):

$$\begin{aligned}
&\max_{|\zeta\rangle \in \mathcal{X}\otimes\mathcal{Y}} \{|\langle \phi, \zeta \rangle| : \mathrm{tr}_{\mathcal{Y}} |\zeta\rangle\langle\zeta| = \sigma\} = \max_{U \in \mathcal{U}(\mathcal{Y})} \left\{ \left| \langle \sqrt{\rho}A^*, \sqrt{\sigma}B^*U^* \rangle \right| \right\} \\
&= \max_{U \in \mathcal{U}(\mathcal{Y})} \left\{ \left| \mathrm{tr}\left(A\sqrt{\rho}\sqrt{\sigma}B^*U^*\right) \right| \right\} = \left\| A\sqrt{\rho}\sqrt{\sigma}B^* \right\|_{\mathrm{tr}}.
\end{aligned}$$

Since $\|A\| \leq 1$ and $\|B\| \leq 1$ hold, we find a lower bound for the fidelity,

$$\left\| A\sqrt{\rho}\sqrt{\sigma}B^* \right\|_{\mathrm{tr}} \leq \|A\| \left\| \sqrt{\rho}\sqrt{\sigma} \right\|_{\mathrm{tr}} \|B^*\| \leq \left\| \sqrt{\rho}\sqrt{\sigma} \right\|_{\mathrm{tr}} = F(\rho, \sigma).$$

On the other hand

$$\begin{aligned}
F(\rho, \sigma) &= \left\| \sqrt{\rho}\sqrt{\sigma} \right\|_{\mathrm{tr}} = \left\| A^*A\sqrt{\rho}\sqrt{\sigma}B^*B \right\|_{\mathrm{tr}} \\
&\leq \|A^*\| \left\| A\sqrt{\rho}\sqrt{\sigma}B^* \right\|_{\mathrm{tr}} \|B\| \leq \left\| A\sqrt{\rho}\sqrt{\sigma}B^* \right\|_{\mathrm{tr}}
\end{aligned}$$

provides the same upper bound, completing the proof. □

Uhlmann's theorem can also be stated differently:
Let $\mathcal{X}, \mathcal{Y}$ be Hilbert spaces such that $\dim(\mathcal{X}) \leq \dim(\mathcal{Y})$ and let $\rho, \sigma \in \mathcal{D}(\mathcal{X})$ be any density operators, then

$$F(\rho, \sigma) = \max_{|\phi\rangle, |\psi\rangle} |\langle \phi | \psi \rangle|,$$

where the maximum is taken over all purifications $|\phi\rangle \in \mathcal{XY}$ of ρ and all purifications $|\psi\rangle \in \mathcal{XY}$ of σ.
From this characterization of the fidelity its symmetry, $F(\rho, \sigma) = F(\sigma, \rho)$, is obvious. Moreover, both formulations of Uhlmann's theorem can be used for further characterization of the fidelity. Due to the Cauchy-Schwarz inequality the fidelity is bounded by one as pure states are unit vectors, leading to $0 \leq F(\rho, \sigma) \leq 1$ for all $\rho, \sigma \in \mathcal{D}(\mathcal{X})$.

Uhlmann's theorem also implies the monotonicity of the fidelity function, which was proven by Barnum, Caves, Fuchs, Josza and Schumacher [BCF+96]:

$$\forall \rho, \sigma \in \mathcal{D}(\mathcal{X}): \qquad F(\Phi(\rho), \Phi(\sigma)) \leq F(\rho, \sigma), \tag{4.15}$$

where $\Phi \in T(\mathcal{X})$ is any admissible superoperator.
In order to prove the monotonicity choose purifications $|\phi\rangle|\psi\rangle \in \mathcal{XY}$ of ρ and σ, respectively, such that $F(\rho, \sigma) = \langle\phi|\psi\rangle$. As stated in Section 4.3.1 there exists a unitary operator U on a bigger space simulating Φ. Therefore, $U|\phi\rangle|0\rangle$ is a purification of $\Phi(\rho)$ and $U|\psi\rangle|0\rangle$ is a purification of $\Phi(\sigma)$. Applying Uhlmann's theorem gives

$$F(\Phi(\rho), \Phi(\sigma)) \geq \langle\phi|\langle 0|U^* U|\psi\rangle|0\rangle| = |\langle\phi|\psi\rangle| = F(\rho, \sigma).$$

Later on we need the following lemma, introducing bounds on the fidelity function:

Lemma 6.

$$\forall \rho, \sigma, \tau \in \mathcal{D}(\mathcal{X}): \qquad F(\rho, \sigma)^2 + F(\sigma, \tau)^2 \leq 1 + F(\rho, \tau) \tag{4.16}$$

$$\forall \rho, \sigma \in \mathcal{D}(\mathcal{X}): \qquad 2 - 2F(\sigma, \rho) \leq \|\sigma - \rho\|_{\mathrm{tr}} \leq 2\sqrt{1 - F(\sigma, \rho)^2} \tag{4.17}$$

Proof. In order to prove (4.16) let $|\psi\rangle \in \mathcal{XY}$ and $|\zeta\rangle \in \mathcal{XY}$ be purifications of ρ and τ, respectively. Due to Uhlmann's theorem there exist purifications $|\phi_1\rangle \in \mathcal{XY}$ and $|\phi_2\rangle \in \mathcal{XY}$ of σ, such that

$$F(\rho, \sigma) = |\langle\psi|\phi_1\rangle| \qquad \text{and} \qquad F(\sigma, \tau) = |\langle\phi_2|\zeta\rangle|.$$

Of course, $\mathcal{Y}$ has to large enough to admit these purifications. Moreover, the unitary equivalence of purifications (Theorem 1) guarantees the existence of a unitary operator $U \in \mathcal{U}(\mathcal{XY})$, for which $U|\phi_2\rangle = |\phi_1\rangle$ holds. Therefore, we can estimate

$$\begin{aligned} F(\rho, \sigma)^2 + F(\sigma, \tau)^2 &= |\langle\psi|\phi_1\rangle|^2 + |\langle\zeta|\phi_2\rangle|^2 = |\langle\psi|U|\phi_2\rangle|^2 + |\langle\zeta|\phi_2\rangle|^2 \\ &\leq \max_{\phi \in \mathcal{XY}} \left(|\langle\psi|U|\phi\rangle|^2 + |\langle\zeta|\phi\rangle|^2\right) = |\langle\psi|U|\zeta\rangle| + 1 \\ &\leq \max_{U} \left(|\langle\psi|U|\zeta\rangle|\right) + 1, \end{aligned}$$

where the maximum is taken over all unitary operators, which obey $\mathrm{tr}_y(U|\zeta\rangle\langle\zeta|U^*) = \tau$. This choice of U guarantees that $U|\zeta\rangle$ is a purification of τ. The equality in the second line is due to the fact that $|\langle\psi|U|\zeta\rangle| + 1$ is the maximal eigenvalue of $|\zeta\rangle\langle\zeta| + U|\psi\rangle\langle\psi|U^*$. Finally, Uhlmann's theorem yields

$$1 + \max_{U} \left(|\langle\psi|U|\zeta\rangle|\right) \leq 1 + F(\rho, \tau),$$

as the maximum runs through all purifications $U|\zeta\rangle$ of τ due to the unitary equivalence of purifications. This proof is similar to a proof from Ashwin Nayak and Peter Shor [NS03]. A proof for the presented formulation of (4.16) can also be found in a paper from Robert W. Spekkens and Terry Rudolph [SR02].

In order to prove the second inequality of (4.17) let $\mathcal{Y}$ be a Hilbert space with the same dimension as $\mathcal{X}$. According to Uhlmann's theorem purifications $|\psi\rangle, |\phi\rangle \in \mathcal{X}\mathcal{Y}$ of ρ and σ exist such that $F(\sigma,\rho) = |\langle\phi,\psi\rangle|$. Therefore, we get

$$\|\sigma-\rho\|_{\mathrm{tr}} \leq \||\phi\rangle\langle\phi| - |\psi\rangle\langle\psi|\|_{\mathrm{tr}} = 2\sqrt{1-|\langle\phi,\psi\rangle|^2} = 2\sqrt{1-F(\rho,\sigma)^2}$$

where the inequality follows from (4.6). The equality holds for any unit vectors, because the non-zero eigenvalues of $A = |\phi\rangle\langle\phi| - |\psi\rangle\langle\psi|$ are $\pm\sqrt{1-|\langle\phi,\psi\rangle|^2}$. Observe first $\mathrm{rank}(A) \leq 2$ and $\mathrm{tr}(A) = 0$. Therefore, zero is an eigenvalue with multiplicity $n-2$ and there are only two non-zero eigenvalues $\pm\lambda$. Moreover, the eigenvalues can be calculated as follows

$$\begin{aligned} 2\lambda^2 &= \mathrm{tr}(A^2) = \mathrm{tr}\left((|\phi\rangle\langle\phi| - |\psi\rangle\langle\psi|)^2\right) \\ &= \mathrm{tr}(|\phi\rangle\langle\phi| + |\psi\rangle\langle\psi| - |\phi\rangle\langle\phi||\psi\rangle\langle\psi| - |\psi\rangle\langle\psi||\phi\rangle\langle\phi|) = 2 - 2|\langle\phi|\psi\rangle|^2. \end{aligned}$$

In order to prove the first inequality of (4.17) observe

$$\|\sqrt{\sigma}-\sqrt{\rho}\|_{\mathrm{Fr}}^2 = \mathrm{tr}\left((\sqrt{\sigma}-\sqrt{\rho})^2\right) = \mathrm{tr}(\sigma) + \mathrm{tr}(\rho) - 2\mathrm{tr}\left(\sqrt{\sigma}\sqrt{\rho}\right) \geq 2 - 2F(\sigma,\rho).$$

Therefore, it is sufficient to prove that the trace norm is an upper bound on the Frobenius norm in the following sense

$$\|\sqrt{\sigma}-\sqrt{\rho}\|_{\mathrm{Fr}}^2 \leq \|\sigma-\rho\|_{\mathrm{tr}},$$

which generally holds for positive semidefinite operators. This can be proven by applying the spectral decomposition to $\sqrt{\sigma}-\sqrt{\rho}$ first. Moreover, we have to choose a unitary operator $U = \sum \mathrm{sgn}(\lambda_i) u_i u_i^*$ and take Lemma 2 into account. These consideration lead to the above inequality by the triangle inequality for real numbers, as well as an easy operator identity, namely

$$A^2 - B^2 = \frac{1}{2}\left((A-B)(A+B)\right) + \frac{1}{2}\left((A+B)(A-B)\right).$$

□

For a complete but different proof see for example [NC00]. Sometimes (4.17) and the following equivalent formulation are called the Fuchs-van de Graaf inequalities

$$1 - \frac{1}{2}\|\rho-\sigma\|_{\mathrm{tr}} \leq F(\rho,\sigma) \leq \sqrt{1-\frac{1}{4}\|\rho-\sigma\|_{\mathrm{tr}}^2}.$$

These have been proven initially by Christopher Fuchs and Jeroen van de Graaf [FvdG99]. We apply the Fuchs-van de Graaf inequalities to prove the following lemma relating the distances of a state to the one after a partial measurement.

Lemma 7. Let $\sigma, \sigma' \in \mathcal{D}(\mathcal{X})$, and $\rho \in \mathcal{D}(\mathcal{X}\otimes\mathcal{Y})$ be density operators such that $\mathrm{tr}_{\mathcal{Y}}(\rho) = \sigma$. Then there exists a density operator $\rho' \in \mathcal{D}(\mathcal{X}\otimes\mathcal{Y})$, such that $\mathrm{tr}_{\mathcal{Y}}(\rho') = \sigma'$. Moreover, if

$$\alpha = \frac{1}{2}\|\sigma-\sigma'\|_{\mathrm{tr}} \qquad \beta = \frac{1}{2}\|\rho-\rho'\|_{\mathrm{tr}},$$

then

$$\alpha \geq 1 - \sqrt{1-\beta^2} \qquad \beta \geq 1 - \sqrt{1-\alpha^2},$$

and ρ' can be computed efficiently in parallel.

The lemma is mainly used in the proof DQIP = PSPACE [GW11] from Gutoski and Wu, who called it the fidelity trick. Moreover, the first part of Lemma 7 is also called the preservation of subsystem fidelity, since the construction of ρ' will imply $F(\sigma, \sigma') = F(\rho, \rho')$.

Proof. It is sufficient to prove $F(\sigma, \sigma') = F(\rho, \rho')$ since the Fuchs-van de Graaf inequalities of these fidelity functions imply

$$\begin{aligned} 1 - \alpha \leq F(\sigma, \sigma') = F(\rho, \rho') \leq \sqrt{1-\beta^2}, \text{and} \\ 1 - \beta \leq F(\rho, \rho') = F(\sigma, \sigma') \leq \sqrt{1-\alpha^2}. \end{aligned}$$

Due to the monotonicity of the fidelity, (4.15), one inequality follows immediately:

$$F(\sigma, \sigma') = F(\mathrm{tr}_{\mathcal{Y}}(\rho), \mathrm{tr}_{\mathcal{Y}}(\rho') \geq F(\rho, \rho').$$

To prove the reverse inequality choose $U \in U(\mathcal{X})$ such that $\sqrt{\sigma}\sqrt{\sigma'}U$ is positive semidefinite. Then $F(\sigma, \sigma') = \mathrm{tr}\left(\sqrt{\sigma}\sqrt{\sigma'}U\right)$ holds. Now define $\mathcal{Z} = \mathcal{X} \otimes \mathcal{Y}$ and choose the purification $|\psi\rangle \in \mathcal{X} \otimes \mathcal{Y} \otimes \mathcal{Z}$ of ρ, such that

$$|\psi\rangle = \mathrm{vec}(\sqrt{\rho}).$$

By reordering the coefficients we can get an operator $A : \mathcal{Y} \otimes \mathcal{Z} \to \mathcal{X}$, which obeys $\mathrm{vec}(A) = |\psi\rangle$. Since $|\psi\rangle$ is a purification of ρ and ρ is a purification of σ, also $|\psi\rangle$ is a purification of σ, just on a bigger space than ρ. We notice here that the relation "is a purification of" is transitive. Due to the unitary equivalence of purifications there exists a linear isometry $V : \mathcal{X} \to \mathcal{Y} \otimes \mathcal{Z}$, such that

$$A = \sqrt{\sigma}V^*.$$

Then $|\phi\rangle = \mathrm{vec}\left(\sqrt{\sigma'}UV^*\right) \in \mathcal{X} \otimes \mathcal{Y} \otimes \mathcal{Z}$ is a purification of σ'. Therefore, we can choose $\rho' = \mathrm{tr}_{\mathcal{Z}}(|\phi\rangle\langle\phi|)$ to finally conclude

$$\begin{aligned} F(\rho, \rho') &\geq \left|\langle \mathrm{vec}\left(\sqrt{\sigma}V^*\right), \mathrm{vec}\left(\sqrt{\sigma'}UV^*\right)\rangle\right| \\ &= \left|\langle \sqrt{\sigma}V^*, \sqrt{\sigma'}UV^*\rangle\right| = \mathrm{tr}(\sqrt{\sigma}\sqrt{\sigma'}U) = F(\sigma, \sigma'). \end{aligned}$$

The initial inequality is due to Uhlmann's theorem. Notice that the first inner product is the standard scalar product, but the second one is the Hilbert-Schmidt inner product. The equality between them holds in general for all hermitian matrices:

$$|\langle \mathrm{vec}(A), \mathrm{vec}(B)\rangle| = |\sum_{i,j} a_{i,j} b_{i,j}| = \mathrm{tr}(A^*B).$$

Moreover, we have to prove the computability of ρ' in NC. First the computation of U can be done in NC, since we can find a singular value decomposition of $\sqrt{\sigma}\sqrt{\sigma'}$ in NC due to Section 4.2. Let SDT be such a singular value decomposition then the choice $U = T^*S$ guarantees the positive semidefiniteness of $\sqrt{\sigma}\sqrt{\sigma'}U$. Moreover, the calculation of A is easy as we just have to rearrange the coefficients in the entries of $\sqrt{\rho}$ to guarantee $\mathrm{vec}(A) = |\psi\rangle = \mathrm{vec}(\sqrt{\rho})$. To be able to compute the linear isometry V we compute the singular value decomposition of $\sqrt{\sigma}$, namely $S_1 D_1 T_1$ and the inverse or the pseudo-inverse of $\sqrt{\sigma}$, namely $T_1^* D_1^{-1} S_1^*$. Notice that D_1 is not necessarily invertible. In this case D_1^{-1} refers to the inversion of the diagonal entries, which are not zero. Once we choose $V = A^*((\sqrt{\sigma})^{-1})^*$ it only remains to calculate ρ':

$$\rho' = \mathrm{tr}_{\mathcal{Z}} \left(\mathrm{vec}\left(\sqrt{\sigma'} U V^*\right) \mathrm{vec}\left(\sqrt{\sigma'} U V^*\right)^* \right).$$

Since standard matrix operations like multiplication and partial trace can be done in NC according to Section 4.2, we can compute ρ' efficiently in parallel from the classical representations of σ, σ' and ρ. □

Furthermore, we need the following corollary of Uhlmann's theorem.

Corollary 1. Let ρ, σ be as before. Define $\epsilon = \min\{\||\phi\rangle - |\psi\rangle\|\}$, where the minimum is taken over all purifications $|\phi\rangle$ and $|\psi\rangle$ of ρ and σ, respectively. Then the fidelity can be characterized as

$$F(\rho, \sigma) = \left(1 - \frac{\epsilon^2}{2}\right)^2.$$

This is a direct consequence of Uhlmann's theorem, since

$$\begin{aligned} \min \||\phi\rangle - |\psi\rangle\|^2 &= \min\left(|\langle\phi|\phi\rangle| - 2|\langle\phi|\psi\rangle| + |\langle\psi|\psi\rangle|\right) = \min\left(2 - 2|\langle\phi|\psi\rangle|\right) \\ &= 2 - 2\max|\langle\phi|\psi\rangle| \end{aligned}$$

holds, when the minima and the maximum are taken over purifications $|\phi\rangle$ and $|\psi\rangle$. Therefore, we can prove the corollary by

$$\left(1 - \frac{\epsilon^2}{2}\right)^2 = \max_{|\phi\rangle, |\psi\rangle} |\langle\phi|\psi\rangle| = F(\rho, \sigma).$$

Now we stated all the theorems, lemmas and corollaries about the fidelity function, which we will need in the main part. In the next section we will examine a metric relying upon the fidelity function.

4.3.4 Bures angle

The Bures angle is a tool to measure the angle between two quantum states. The name goes back to a paper of Donald Bures [Bur69], in which he studied a metric. Eventually this metric was called the Bures metric. Uhlmann pointed out its application to quantum states in 1992 [Uhl92]. Nevertheless, the following section is mostly due to Nielsen's and

Chuang's book [NC00] as well as Gutoski's and Wu's paper [GW11]. For any two density operators $\rho, \sigma \in \mathcal{D}(\mathcal{X})$ the Bures angle is defined as

$$\angle(\rho, \sigma) = \arccos F(\rho, \sigma).$$

This is a quite unusual notation, because the Bures angle is referred to as $A(\cdot, \cdot)$ in the literature. But the extensive use of the letter A for the player Alice in a game justifies this notation avoiding double occupancy.
Since the range of the fidelity function is $[0, 1]$, Bures angle is well defined and non-negative as $0 \leq \arccos x \leq \pi/2$ holds for all $x \in [0, 1]$. Moreover, the statement ,$F(\rho, \sigma)$=1 if and only if $\rho = \sigma$, implies $\angle(\rho, \sigma) = 0$ if and only if $\rho = \sigma$. Since the Bures angle is also symmetric and obeys the triangle inequality, $\angle(\rho, \sigma) \leq \angle(\rho, \tau) + \angle(\tau, \sigma)$ for all density operators ρ, σ, τ, it is a metric on quantum states.
In order to prove the triangle inequality let $|\zeta\rangle$ be a purification of τ. Now we choose purifications $|\psi\rangle$ of σ and $|\phi\rangle$ of ρ such that

$$F(\rho, \tau) = \langle\phi|\zeta\rangle \qquad F(\tau, \sigma) = \langle\zeta|\psi\rangle,$$

and $\langle\phi|\psi\rangle$ is a positive real number. Such a choice is always possible since we can multiply the purifications with suitable phase factors to force $\langle\phi|\psi\rangle$ into being real and positive. As any purification is a unit vector,

$$\arccos\langle\phi|\psi\rangle \leq \arccos\langle\phi|\zeta\rangle + \arccos\langle\zeta|\psi\rangle \tag{4.18}$$

is the triangle inequality for the angle between points on the surface of the unit sphere. Combining a consequence of Uhlmann's theorem (Theorem 2), namely $F(\rho, \sigma) \geq \langle\phi|\psi\rangle$, the monotone decrease of the arccos function, and (4.18) we conclude

$$\angle(\rho, \sigma) \leq \arccos\langle\phi|\psi\rangle \leq \angle(\rho, \tau) + \angle(\tau, \sigma).$$

Additionally the Bures angle is contractive. This means

$$\angle(\Phi(\rho), \Phi(\sigma)) \leq \angle(\rho, \sigma),$$

for any quantum channel $\Phi \in \mathcal{T}(\mathcal{X}, \mathcal{Y})$ and all density operators $\rho, \sigma \in \mathcal{D}(\mathcal{X})$. The contractiveness of the Bures angle is a consequence of the monotonicity of the fidelity, which was discussed in the previous section. Since $F(\Phi(\rho), \Phi(\sigma)) \geq F(\rho, \sigma)$ holds for all density operators $\rho, \sigma \in \mathcal{D}(\mathcal{X})$ and all quantum channels $\Phi \in \mathcal{T}(\mathcal{X}, \mathcal{Y})$ the contractiveness follows from the monotone decrease of the arccos function.
Furthermore, the Fuchs-van de Graaf inequalities can be used to find bounds on the Bures angle in terms of the trace norm.

Lemma 8. For all density operators $\rho, \sigma \in \mathcal{D}(\mathcal{X})$

$$\frac{1}{2}\|\rho - \sigma\|_{\mathrm{tr}} \leq \angle(\rho, \sigma) \leq \sqrt{\frac{\pi}{2}\|\rho - \sigma\|_{\mathrm{tr}}} \qquad .$$

Proof. The upper bound is a direct consequence of (4.17):

$$\frac{1}{2}\|\rho-\sigma\|_{\mathrm{tr}} \leq \sqrt{1-\cos(\angle(\rho,\sigma))^2} = \sin(\angle(\rho,\sigma)) \leq \angle(\rho,\sigma).$$

Here we applied the equality $\sqrt{1-(\cos x)^2} = \sin x$ and the inequality $\sin x \leq x$, which holds for all $x \geq 0$ due to basic calculus.
Also the lower bound can be proven by applying (4.17):

$$\frac{1}{2}\|\rho-\sigma\|_{\mathrm{tr}} \geq 1-\cos(\angle(\rho,\sigma)) \geq \frac{\angle(\rho,\sigma)^2}{\pi},$$

where the second inequality follows from the inequality $\cos x \leq 1 - x^2/\pi$, which holds for all $x \in [0, \pi/2]$ due to basic calculus. □

After this theoretical approach, we will discuss a computational model for quantum computers.

4.3.5 Quantum circuits

We will rely upon the quantum circuit model of computation, a computational model from David Deutsch [Deu89]. Despite the fact that there exist many different equivalent models, such as, for example, the quantum Turing machine, which was also introduced by Deutsch [Deu85], the quantum circuit model best suits our purpose. Actually, we will use a refined version by Yao [Yao93], who also established the equivalence of these models. Transformations on quantum registers ought to be unitary and hence invertible. Therefore, it is easy to describe unitary operators, which reorder qubits or flip them around. But it is harder to define simple operations like copying a qubit. Actually, copying a qubit can only be accomplished by increasing the register's size.
Basic unitary operations are called gates. Quantum circuits process quantum information through quantum gates. Since matrices are used to describe these gates, a basis has to be selected. Here the standard basis will be chosen. For single qubit operations this means

$$|0\rangle \text{ corresponds to } e_1 = \begin{pmatrix}1\\0\end{pmatrix} \text{ and } |1\rangle \text{ corresponds to } e_2 = \begin{pmatrix}0\\1\end{pmatrix}.$$

In general the lexicographical order of the basis in the dirac notation corresponds to the standard basis as explained in Section 4.1.3.
Initially, we will examine three quantum gates, which will produce any others up to sufficient precision:

1. The Hadamard gate is probably the most frequently used gate in quantum algorithms. It operates on a single qubit and has the following effect on the standard basis: $|0\rangle \to (1/\sqrt{2})(|0\rangle + |1\rangle), |1\rangle \to (1/\sqrt{2})(|0\rangle - |1\rangle)$. The matrix representation of the Hadamard gate on one qubit is

$$H = \frac{1}{\sqrt{2}}\begin{pmatrix}1 & 1\\ -1 & 1\end{pmatrix}.$$

2. The $\sqrt{\sigma_3}$-gate or phase shift gate also operates on one qubit, it has the following matrix representation

$$\sqrt{\sigma_3} = \begin{pmatrix} 1 & 0 \\ 0 & i \end{pmatrix}.$$

The notation $\sqrt{\sigma_3}$ refers to the Pauli matrices utilized in the proof of Lemma 4. Alternatively, the notation $\sqrt{\sigma_z}$-gate is used in the literature. Notice that this gate does not change the outcome of a measurement performed on the resulting qubit, but instead it only shifts the phase: $|0\rangle$ remains unchanged and $|1\rangle \to e^{i\pi/2}|1\rangle$.

3. The Toffoli gate operates on 3 qubit. In terms of Boolean variables it has the following effect on pure states: $|xyz\rangle \to |xy(z \oplus x \wedge y)\rangle$. This leads to the matrix representation

$$\text{CCNOT} = \begin{pmatrix} 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 \end{pmatrix}.$$

The Toffoli gate is sometimes also called the CCNOT gate, since it is a doubly controlled not operation: the last qubit is negated if and only if the first two qubits are in state $|1\rangle$.

These three gates are called Shor's basis as they generate a set of gates dense in the set of 3-qubit gates. Moreover, Shor pointed out, how to construct a polynomial size quantum circuit that tolerates $\mathcal{O}(1/\log^c m)$ amounts of decoherence and inaccuracy for any quantum computation with m gates [Sho96]. We will rely upon an improved theorem, which can be found in the book of Arora and Barak [AB09].

Theorem 3. For every $n \in \mathbb{N} : n \geq 3$ and $\epsilon \in \mathbb{R} : \epsilon > 0$, there exists an $m \leq 100\left(\text{nlog}\left(\frac{1}{\epsilon}\right)\right)^3$ such that for all $i, j \leq n$ the entry U_{ij} of any unitary $n \times n$ matrix U can approximated by unitary operators $U_1 \ldots, U_m$ in the following sense:

$$|U_{ij} - (U_1 \cdots U_m)_{ij}| \leq \epsilon,$$

where each U_k for $k \in \{1, 2, \ldots, m\}$ is either a Hadamard, a phase shift or a Toffoli gate, tensored with the identity operator on the remaining qubits.

Initially, the stated accuracy was published by Kitaev for arbitrary dense sets [Kit97] and independently proven by Solovay, resulting in the Kitaev-Solovay theorem. Since the proof of Theorem 3 is known for a long time and quite elaborate it will not be discussed here. Details can also be found in the book of Nielsen and Chuang [NC00], for instance. In the main part we will also use two more quantum gates:

1. The controlled not or CNOT gate acts on 2 qubits. It is determined by its actions on the standard basis, $|0x\rangle \to |0x\rangle$ and $|1x\rangle \to |1\neg x\rangle$, for $x \in \{0, 1\}$. Therefore, the matrix representing a CNOT gate is

$$\text{CNOT} = \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 1 & 0 \end{pmatrix}.$$

   It is called a CNOT gate, since the first qubit controls, whether or not the second qubit is flipped.

2. The swap gate acts on two qubits, it exchanges the values of the two qubits in the following way: $|01\rangle \to |10\rangle, |10\rangle \to |01\rangle$, whereas $|00\rangle$ and $|11\rangle$ remain unchanged. The matrix representing the swap gate is:

$$\text{SWAP} = \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix}.$$

Keep in mind that all these matrix representations are with respect to the standard basis which is related to the lexicographical order of the binary numbers in the Dirac notation, as mentioned in Section 4.1.3.
This concludes the part of the preliminaries concerned with quantum phenomena. In this thesis a theoretical approach to quantum computation through unitary matrices on Hilbert spaces was already presented. Moreover, this section explains a practical approach utilizing gates acting on few qubits. But there is still some work to do before we can tackle the most recent advances in quantum complexity theory.

4.4 Game theory

In the wide field of game theory this thesis is only concerned with two-player zero-sum games. This section is mainly due to a paper from Maurice Sion [Sio58]. Even though most of the game theory we apply, is easy or even self-explanatory, a few results on two-player zero-sum games have to be considered. At first John von Neumann proved the existence of a saddle point in 1928 [vN28] for finite dimensional simplices $\mathcal{M}$ and $\mathcal{N}$, and a bilinear function f acting on $\mathcal{M} \otimes \mathcal{N}$. Analytically formulated this means

$$\max_{\mu \in \mathcal{M}} \min_{\nu \in \mathcal{N}} f(\mu, \nu) = \min_{\nu \in \mathcal{N}} \max_{\mu \in \mathcal{M}} f(\mu, \nu). \tag{4.19}$$

This equation relates to two-player games, as $\mathcal{M}$ and $\mathcal{N}$ can be interpreted as sets of strategies available to the players. In this case the function f specifies the payout for the player, who chooses his strategy from $\mathcal{M}$. He tries to maximize the value of f, but his

opponent tries to minimize it. Moreover, on the left side of (4.19), the player maximizes his minimal win (Maximin strategy), opposed to the right side, where his opponent minimizes his maximal loss (Minimax strategy). Therefore, we can immediately conclude

$$\max_{\mu\in\mathcal{M}}\min_{\nu\in\mathcal{N}} f(\mu,\nu) \leq \min_{\nu\in\mathcal{N}}\max_{\mu\in\mathcal{M}} f(\mu,\nu).$$

All points (x', y'), which satisfy a min-max theorem, such as (4.19), are called equilibrium points.
Sion generalized von Neumann's theorem in 1958 to semi-continuous, quasi-concave-convex functions on compact convex subsets [Sio58]. Actually, Fan did this generalization using fixpoint arguments [Fan53], but Sion unified the proofs of the existing theories.

Theorem 4. For compact convex sets $\mathcal{X}, \mathcal{Y}$, any real valued, quasi-concave-convex function f on $\mathcal{X}\otimes\mathcal{Y}$, which is upper semi-continuous on $\mathcal{X}$ and lower semi-continuous on $\mathcal{Y}$ satisfies

$$\sup_{y\in\mathcal{Y}}\inf_{x\in\mathcal{X}} f(x,y) = \inf_{x\in\mathcal{X}}\sup_{y\in\mathcal{Y}} f(x,y).$$

In order to apply this to quantum refereed games it will suffice to understand, that bilinear functions are quasi-concave-convex. This is obvious, since a function f acting on $\mathcal{X}\mathcal{Y}$ is called quasi-concave-convex if

1. for all $y\in\mathcal{Y}$, $c\in\mathbb{R}$, the set $\{x : f(x,y)\geq c\}$ is convex, (quasi-concave on $\mathcal{Y}$) and
2. for all $x\in\mathcal{X}$, $c\in\mathbb{R}$, the set $\{y : f(x,y)\leq c\}$ is convex (quasi-convex on $\mathcal{X}$).

Therefore, Theorem 4 can be applied to bilinear functions. Further details are presented in the original paper from Sion [Sio58]. We will consider a scaled down version of his proof for the special case of bilinear functions.
Like in all equilibrium point problems one inequality follows immediately, $\sup\inf f(x,y) \leq \inf\sup f(x,y)$. Since the supremum is always taken over $\mathcal{Y}$, while the infimum is always taken over $\mathcal{X}$, this shortened notation is used. For the reverse inequality, observe that bilinear functions over compact sets, make the use of the infima and suprema obsolete, because they are always achieved. Therefore, we can use minima and maxima instead. Sion's arguments are outlined without including the proofs of the following lemmas, since these statements have been known for a long time.

Lemma 9. For an n-dimensional simplex $\mathcal{S}$ with vertices $a_0,\ldots,a_n$, let $\mathcal{A}_0,\ldots\mathcal{A}_n$ be open sets, such that $\mathcal{S}\subset\bigcup_{j=0}^n\mathcal{A}_j$, $\mathcal{S}-\mathcal{A}_j$ is convex ,and $a_j\notin\mathcal{A}_k$, for all $j,k=0,\ldots,n : j\neq k$, then $\bigcap_{j=0}^n\mathcal{A}_j=\emptyset$.

Lemma 10. Let $\mathcal{U}=\{a_0\ldots a_n\}$ be a set of $n+1$ points in a linear space of dimension $k<n$, then $\bigcap_{j=0}^n \mathrm{conv}(\mathcal{U}-a_j)\neq\emptyset$.

The notation $\mathrm{conv}(\cdot)$ denotes the convex hull, which is the set of all convex combinations. Let $\mathcal{X}$ be any subset of a vector space, then

$$\mathrm{conv}(\mathcal{X}) = \left\{\sum_{j=1}^n \alpha_j x_j : x_j\in\mathcal{X}, n\in\mathbb{N}, \sum_{j=1}^n\alpha_j = 1, \alpha_j\geq 0\right\}.$$

Lemma 9 and Lemma 10 are only used to prove the following lemma:

Lemma 11. Let $\mathcal{X}$ be a convex set, $\mathcal{N}$ a finite set and f a bilinear function on $\mathcal{X} \otimes \mathcal{N}$. If $\mathcal{N}$ is minimal with respect to the property: $\forall x \in \mathcal{X} \exists \nu \in \mathcal{N} : f(x, \nu) < c$, then there exists x_0, such that $f(x_0, \nu) < c, \forall \nu \in \mathcal{N}$.

Originally this lemma was designed for quasi-concave and upper semi-continuous functions, and thus there was an analogous formulation for lower semi-continuous quasi-convex functions. Since we only need the statement for bilinear functions and both sets $\mathcal{X}$ and $\mathcal{Y}$ are convex and compact, our situation is completely symmetric. Therefore, the statement holds, if we exchange the role of $\mathcal{X}$ and $\mathcal{Y}$. Although the proof would be simpler for bilinear functions, it is usefuI to present this more complex proof since it unites the different concepts in two-player zero-sum games.

Proof. In order to prove Lemma 11 name the elements of $\mathcal{N}$ $\nu_0, \ldots, \nu_n$ and define

$$\mathcal{A}_j = \{x : f(x, \nu_j) < c\} \qquad \text{for } j = 0, \ldots, n.$$

Since f is bilinear, A_j is open and $\mathcal{X} - \mathcal{A}_j$ is convex for all j. The minimality condition on $\mathcal{N}$ implies for each j the existence of a $a_j \in \mathcal{X}$, such that $a_j \in \mathcal{X} - \mathcal{A}_k$ for all $k \neq j$. Moreover, we define $\mathcal{U} = \{a_0, \ldots, a_n\}$ in order to achieve

$$\operatorname{conv}(\mathcal{U} - \{a_j\}) \subset \operatorname{conv}(\mathcal{X} - \mathcal{A}_j) = \mathcal{X} - \mathcal{A}_j,$$

which is due to $\mathcal{U} \subset \mathcal{X}$ and the fact that the convex hull of a convex set is the set itself. Furthermore, due to minimality criterion on $\mathcal{N}$ we conclude $\mathcal{X} \subset \bigcup_{j=1}^{n} \mathcal{A}_j$ and therefore we have

$$\bigcap_{j=0}^{n} \operatorname{conv}(\mathcal{U} - \{a_j\}) = \emptyset.$$

Now the negation of Lemma 10 implies that $\mathcal{U}$ is a n-dimensional simplex. Thus, we can apply Lemma 9 to find a $x_0 \in \bigcap_{j=0}^{n} \mathcal{A}_j$ satisfying $f(x_0, \nu) < c, \forall \nu \in \mathcal{N}$. □

Note that due to the bilinearity of f the roles of $\mathcal{X}$ and $\mathcal{Y}$ can be exchanged, as well as the inequality in the minimality condition can be reversed, to formulate an analogous lemma. Lemma 11 was originally designed to take the different assumptions on the convexity of f into account. Therefore, it is not important to this proof and we show Theorem 4 by contradiction.

Proof. Suppose $\max\min f < c < \min\max f$ and define

$$\mathcal{A}_x = \{y : f(x, y) > c\} \quad \text{and} \quad \mathcal{B}_y = \{x : f(x, y) < c\}.$$

Since f is bilinear all $\mathcal{A}_x$ are open and their union covers $\mathcal{Y}$. Due to the compactness of $\mathcal{Y}$ some finite subset of $\{\mathcal{A}_x : x \in \mathcal{X}\}$ also covers $\mathcal{Y}$. Analogously, some finite subset of $\{\mathcal{B}_y : y \in \mathcal{Y}\}$ covers $\mathcal{X}$. Now we can choose finite subsets $\mathcal{M}_1 \subset \mathcal{X}$ and $\mathcal{N}_1 \subset \mathcal{Y}$, such that for each $y \in \mathcal{Y}$, and therefore for each $y \in \operatorname{conv}(\mathcal{N}_1)$, there exists a $\mu \in \mathcal{M}_1$, with $f(\mu, y) > c$. Analogously, for each $x \in \mathcal{X}$ and therefore for each $x \in \operatorname{conv}(\mathcal{M}_1)$, there exists a $\nu \in \mathcal{M}_1$ with $f(x, \nu) < c$. Furthermore, let $\mathcal{M}_2$ be a minimal subset of $\mathcal{M}_1$,

such that for each $y \in \mathcal{N}_1$ there exists a $\mu \in \mathcal{M}_2$, with $f(\mu, y) > c$. Moreover, let $\mathcal{N}_2$ be the set, which is defined by an analogous construction with $\mathcal{N}_2$ in place of $\mathcal{M}_2$.
Once we iterate this alternating process we end up with two finite subsets $\mathcal{M} \subset \mathcal{X}$ and $\mathcal{N} \subset \mathcal{Y}$, such that $\mathcal{N}$ is minimal with respect to the property:

$$\forall \nu \in \text{conv}(\mathcal{N}), \exists \mu \in \mathcal{M} : f(\mu, \nu) > c.$$

Analogously, $\mathcal{M}$ is minimal with respect to the property:

$$\forall \mu \in \text{conv}(\mathcal{M}), \exists \nu \in \mathcal{N} : f(\nu, \mu) < c.$$

Therefore, Lemma 11 ensures the existence of a $x_0 \in \text{conv}(\mathcal{M})$ such that $f(x_0, y) < c$ for all $y \in \text{conv}(\mathcal{N})$. The analogous version of Lemma 11 guarantees the existence of a $y_0 \in \text{conv}(\mathcal{N})$ such that $f(x, y_0) > c$ for all $x \in \text{conv}(\mathcal{M})$. But then $c < f(x_0, y_0) < c$, which is obviously a contradiction. □

In the main part we will use Theorem 4 to construct a suitable SDP, representing the game value of a quantum refereed game. This SDP can be solved by a version of the matrix multiplicative weight update algorithm, which we will examine in the next section.

4.5 Matrix multiplicative weight update method

In 2007 Sayten Kale published in his thesis [Kal07] a generalization of the multiplicative weight update method onto matrices. It is called the MMW method and relies upon a survey by Sanjeev Arora, Elad Hazan and Kale [AHK12], in which the multiplicative weight update method and some applications are discussed. Actually this survey exists since 2005, but it was just recently published in a journal. The MMW method was originally designed to solve the following problem:
In each round $k \in \{1, 2, \ldots, N\}$ a player selects randomly one of n experts, which are each represented by a unit vector in $\mathbb{C}^n$. The player has to carry the advice of the chosen expert into execution for example to buy or sell a certain stock. The monetary consequences of choosing a certain expert are represented by a loss matrix. The letter $M \in \mathbb{C}^{n \times n}$ serves as the description of a quadratic form quantifying the loss, caused by expert $v \in \mathbb{S}^{n-1}$ as $v^t M v$, where $\mathbb{S}^{n-1}$ is the unit sphere in $\mathbb{C}^n$. There is only the following restriction on M: $0 \leq M \leq \mathbb{1}_n$. Here $\mathbb{1}_n$ is the $n \times n$ dimensional identity matrix. Moreover, Kale restricted his proof to vectors and matrices with real entries rather than complex ones, only mentioning the possibility of this generalization. Since we discuss the generalization it is important to keep in mind that all matrices under consideration are hermitian.
Coming back to the original problem, the objective is to choose the experts, such that the best expert does not control his losses significantly better than the player. The opinion of the player about the experts changes over the course of the experiment. In the beginning a rational player should not favor any expert. When the experts give good or bad advice, the player has to judge their success or failure to improve future

decisions. Therefore, the player has to keep track of a probability distribution $D^{(k)}$ after which he chooses the experts at random. This can be interpreted as assigning a weight to each expert, justifying the name of this method. The expected loss of the player, after choosing his expert in round k according to $D^{(k)}$ is

$$\mathbb{E}\left[v^t M^{(k)} v\right] = \mathbb{E}\left[\langle M^{(k)}, vv^t\rangle\right] = \langle M^{(k)}, \mathbb{E}\left[vv^t\right]\rangle,$$

where $\mathbb{E}$ is the expected value ranging over all experts $v \in D^{(k)}$ of a certain distribution. Observe that $\rho = \mathbb{E}\left[vv^t\right]$ is positive semidefinite, since $\forall v \in \mathbb{S}^{n-1} : vv^t \in Pos(\mathbb{R}^n)$ and the expectation is just a convex combination of these matrices. Furthermore, notice that $\mathrm{tr}(vv^t) = \|v\|^2 = 1$ implies $\rho \in \mathcal{D}(\mathbb{C}^n)$. Moreover, keeping the spectral decomposition, $\rho = \sum_j \lambda_j(\rho) v_j v_j^t$, in mind, the density matrix can be associated with the probability distribution, which chooses vector v_j with probability λ_j. In general $\lambda_j(A)$ refers to the j-th largest eigenvalue of a hermitian matrix A. Since we are interested in the expected loss, the density matrix suffices to calculate this value as

$$\mathbb{E}\left[v^t M^{(k)} v\right] = \langle M^{(k)}, \rho\rangle.$$

The expected loss over N rounds is therefore

$$\sum_{k=1}^{N} \mathbb{E}\left[v^t M^{(k)} v\right] = \sum_{k=1}^{N} \langle M^{(k)}, \rho\rangle.$$

The above considerations enable us to state the matrix multiplicative weight update algorithm

1. Initialize $\gamma \in \mathbb{R} : 0 \leq \gamma \leq \frac{1}{2}$ and $W^{(1)} = \mathbb{1}_n$
2. For $k = 1, 2, \ldots, N$
   a) Update $\rho^{(k)} = W^{(k)}/\mathrm{tr}(W^{(k)})$, and
   b) Observe the loss matrix $M^{(k)}$ and update the weight matrix

$$W^{(k+1)} = \exp\left(-\gamma \sum_{i=1}^{k} M^{(i)}\right).$$

The next theorem provides an upper bound on the overall expected loss if the player applies the MMW algorithm.

Theorem 5. The MMW algorithm guarantees, that after N rounds, every density matrix $\rho \in \mathcal{D}(\mathbb{C}^n)$ satisfies

$$(1-\gamma)\sum_{k=1}^{N} \mathrm{tr}\left(M^{(k)}\rho^{(k)}\right) \leq \mathrm{tr}\left(\sum_{k=1}^{N} M^{(k)}, \rho\right) + \frac{\ln(n)}{\gamma}.$$

Before we are able to prove this theorem some basic considerations are necessary: First note that the Golden-Thompson inequality, $\mathrm{tr}(\exp(A+B)) \leq \mathrm{tr}(\exp(A)\exp(B)$, holds for all hermitian matrices A and B. Secondly if two real valued functions f and g satisfy $f(x) \geq g(x)$ for all x in some domain, then $f(A) \geq g(A)$ for any hermitian matrix A, whose eigenvalues lie in this domain. In order understand this statement observe that the application of a function to a diagonalizable matrix can be understood as the application of the function to the eigenvalues in the diagonal matrix: $f(A) = Uf(D)U^*$. Utilizing this notation the positive semidefiniteness of $f(A) - g(A)$ is simply due to the non-negativity of the eigenvalues of $f(D) - g(D)$ as $f(A) - g(A) = U(f(D) - g(D))U^*$. Moreover, we apply this consideration to the following inequality

$$\exp(-\gamma x) \leq 1 - (1 - \exp(-\gamma))x,$$

which holds for all $\gamma \leq 1$ and all $x \in [0,1]$ due to the convexity of the exponential function. This leads to the matrix inequality

$$\exp(-\gamma A) \leq \mathbb{1}_n - (1 - \exp(-\gamma))A, \tag{4.20}$$

which holds for all hermitian $n \times n$ matrices A with eigenvalues in $[0,1]$. These considerations enable us to prove Theorem 5.

Proof. Observe the following recursive inequality:

$$\begin{aligned}
\mathrm{tr}\left(W^{(k+1)}\right) &= \mathrm{tr}\left(\exp\left(-\gamma \sum_{j=1}^{k} M^{(j)}\right)\right) \\
&\leq \mathrm{tr}\left(\exp\left(-\gamma \sum_{j=1}^{k-1} M^{(j)}\right) \exp\left(-\gamma M^{(k)}\right)\right) = \mathrm{tr}\left(W^{(k)} \exp\left(-\gamma M^{(k)}\right)\right) \\
&\leq \mathrm{tr}\left(W^{(k)}\left(\mathbb{1}_n - (1 - e^{-\gamma})M^{(k)}\right)\right) = \mathrm{tr}\left(W^{(k)}\right)\left(1 - \mathrm{tr}\left[(1 - e^{-\gamma})M^{(k)}\rho^{(k)}\right]\right) \\
&\leq \mathrm{tr}\left(W^{(k)}\right) \exp\left(\mathrm{tr}\left[(e^{-\gamma} - 1)M^{(k)}\rho^{(k)}\right]\right),
\end{aligned}$$

where we used the shortened notation for the exponential function on real values. The second line is due to the Golden-Thompson inequality. The third line follows from (4.20) and the definition of $\rho^{(k)}$ in the algorithm. Moreover, the last inequality is due to $1 + x \leq e^x$, which holds for all real x, and therefore by the above consideration $1 + A \leq \exp(A)$ holds for any hermitian matrix A.
Iterating the resulting inequality leads to

$$\mathrm{tr}\left(W^{(N+1)}\right) \leq \mathrm{tr}\left(W^{(1)}\right) \exp\left((e^{-\gamma} - 1)\mathrm{tr}\left[\sum_{k=1}^{N} M^{(k)}\rho^{(k)}\right]\right) \tag{4.21}$$

by induction. Due to the initialization of the weight matrix in the algorithm we have $\mathrm{tr}(W^{(1)}) = \mathrm{tr}(\mathbb{1}_n) = n$.On the other hand observe

$$\mathrm{tr}\left(W^{(N+1)}\right) = \mathrm{tr}\left(\exp\left(-\gamma \sum_{k=1}^{N} M^{(k)}\right)\right) \geq \exp\left(-\gamma \lambda_{min}\left(\sum_{k=1}^{N} M^{(k)}\right)\right).$$

In general $\lambda_{min}(A)$ refers to smallest eigenvalue of a normal matrix A. The last inequality follows from $\mathrm{tr}(\exp(A)) = \sum_j \lambda_j(\exp(A)) = \sum_j \exp(\lambda_j(A)) \geq \exp(\lambda_{min}(A))$. Combining the above inequalities for $\mathrm{tr}(W^{(N+1)})$ leads to

$$\exp\left(-\gamma\lambda_{min}\left(\sum_{k=1}^{N} M^{(k)}\right)\right) \leq n \exp\left((e^{-\gamma}-1)\mathrm{tr}\left[\sum_{k=1}^{N} M^{(k)}\rho^{(k)}\right]\right). \tag{4.22}$$

Moreover, any unit vector v obeys $\sum_{k=1}^{N} v^t M^{(k)} v \geq \lambda_{min}\left(\sum_{k=1}^{N} M^{(k)}\right)$. However, since the theorem is stated in terms of density matrices instead of unit vectors, we need the eigenvalue decomposition, $\rho = \sum_{j=1}^{n} \lambda_j(\rho) v_j v_j^t$, which exists for any $\rho \in \mathcal{D}(\mathbb{C}^n)$ to conclude

$$\begin{aligned}\sum_{k=1}^{N} \mathrm{tr}(M^{(k)}\rho) &= \sum_{k=1}^{N} \mathrm{tr}\left(M^{(k)} \sum_{j=1}^{n} \lambda_j(\rho) v_j v_j^t\right) = \sum_{k=1}^{N}\sum_{j=1}^{n} v_j^t M^{(k)} v_j \lambda_j(\rho) \\ &\geq \sum_{j=1}^{n} \lambda_j(\rho)\lambda_{min}\left(\sum_{k=1}^{N} M^{(k)}\right) = \lambda_{min}\left(\sum_{k=1}^{N} M^{(k)}\right).\end{aligned}$$

The last inequality is due to $\sum \lambda_i(\rho) = \mathrm{tr}(\rho) = 1$. Since $\exp(-\gamma x)$ is strictly monotonically decreasing we can conclude

$$\exp\left(-\gamma \sum_{k=1}^{N} \mathrm{tr}(M^{(k)}\rho)\right) \leq \exp\left(-\gamma\lambda_{min}\left(\sum_{k=1}^{N} M^{(k)}\right)\right).$$

Plugging this result into (4.22) and taking logarithms, we end up with

$$-\gamma \sum_{k=1}^{N} \mathrm{tr}(M^{(k)}\rho) \leq \ln n + (e^{-\gamma}-1)\mathrm{tr}\left[\sum_{k=1}^{N} M^{(k)} P^{(k)}\right].$$

Furthermore, we use $e^{-\gamma} - 1 \leq -\gamma(1-\gamma)$ for $\gamma \leq 1/2$ to rephrase this inequality:

$$-\gamma \sum_{k=1}^{N} \mathrm{tr}(M^{(k)}\rho) \leq \ln n - \gamma(1-\gamma)\mathrm{tr}\left[\sum_{k=1}^{N} M^{(k)} \rho^{(k)}\right].$$

Dividing by γ and rearranging the terms completes the proof. □

In the main part we will need a slightly different variant of Theorem 5. The difference will regard the upper bound on the loss matrices: $0 \leq M^{(k)} \leq \alpha \mathbb{1}_n$. This implies

$$(1-\gamma)\sum_{k=1}^{N} \mathrm{tr}\left(M^{(k)}\rho^{(k)}\right) \leq \mathrm{tr}\left(\sum_{k=1}^{N} M^{(k)}, \rho\right) + \alpha\frac{\ln(n)}{\gamma}, \tag{4.23}$$

since all other terms in the inequality besides $\ln(n/\gamma)$ are lowered by the factor α. Furthermore, we can use the eigenvalue decomposition to conclude

$$\mathrm{tr}(M^{(k)}\rho) = \mathrm{tr}\left(M^{(k)} \sum_{j=1}^{n} \lambda_j(\rho) v_j v_j^t\right) = \sum_{j=1}^{n} v_j^t M^{(k)} v_j \lambda_j(\rho) \leq \sum_{j=1}^{n} \lambda_j(\rho)\lambda_{max}(M^{(k)}) \leq \alpha$$

for any density operator ρ and all $k \in \{1, \ldots, N\}$. Therefore, adding $\gamma \sum_k \mathrm{tr}(M^{(k)}\rho^{(k)})$ to both sides of inequality (4.18) and using the Hilbert-Schmidt inner product gives

$$\sum_{k=1}^{N} \left\langle \rho^{(k)}, M^{(k)} \right\rangle \leq \left\langle \rho, \sum_{k=1}^{N} M^{(k)} \right\rangle + \alpha\gamma N + \frac{\alpha \ln(n)}{\gamma}$$

Thus, dividing by N provides the desired extended version of Theorem 5:

$$\frac{1}{N} \sum_{k=1}^{N} \left\langle \rho^{(k)}, M^{(k)} \right\rangle \leq \left\langle \rho, \frac{1}{N} \sum_{k=1}^{N} M^{(k)} \right\rangle + \alpha \left(\gamma + \frac{\ln(n)}{\gamma N} \right). \tag{4.24}$$

This completes the description of the MMW method. The further adjustments we need are postponed to the main part, where we will apply the MMW method to semidefinite programs. The next section is dedicated to this generalization of linear programs.

4.6 Semidefinite programs

The formulation of quantum interactive proofs and quantum refereed games as semidefinite programs (SDPs) is important to the new polynomial space algorithms. Therefore, this generalization of linear programs ought to be examined. Since the following description of SDPs is completely standard similar presentations can be found in most books regarding this topic.

The previously defined superoperators enable a compact description of semidefinite programs as triples (Ψ, A, B) of a hermiticity preserving superoperator $\Psi \in \mathcal{T}(\mathcal{X}, \mathcal{Y})$ and two Hermitian operators $A \in Herm(\mathcal{X})$ and $B \in Herm(\mathcal{Y})$ such that

Primal	**Dual**
minimize : $\langle A, X \rangle$	maximize : $\langle B, Y \rangle$
subject to : $\Psi(X) \geq B$	subject to : $\Psi^*(Y) \leq A$
$X \in Pos(\mathcal{X})$	$Y \in Pos(\mathcal{Y})$.

Note that the restrictions to positive semidefinite matrices causes nonlinearities, as a Hermitian operator $X \in Herm(\mathcal{X})$ is positive semidefinite if and only if $v^t X v \geq 0$ for all $v \in \mathcal{X}$. Obviously, this condition is quadratic. Therefore SDPs are a generalization of linear programs, which is also explained in detail in a survey of Lieven Vanderberghe and Stephen Boyd [VB96].

Analogously to linear programs, semidefinite programs have a primal and a dual formulation, with primal and dual feasible sets

$$\mathcal{P} = \{X \in Pos(\mathcal{X}) : \Psi(X) \geq B\} \quad \text{and} \quad \mathcal{D} = \{Y \in Pos(\mathcal{Y}) : \Psi^*(Y) \leq A\}$$

and optimal primal and dual values $p, d \in \mathbb{R}$. The two formulations obey the following weak duality theorem:

Theorem 6. For any Euclidean vector spaces $\mathcal{X}, \mathcal{Y}$ and a semidefinite program (Ψ, A, B) the primal optimum value is at least as big as the dual optimum value:

$$p = \inf_{X \in Pos(\mathcal{X})} \{\langle A, X\rangle : \Psi(X) \geq B\} \geq \sup_{Y \in Pos(\mathcal{Y})} \{\langle B, Y\rangle : \Psi^*(Y) \leq A\} = d. \tag{4.25}$$

Notice that the above equations define the optimal values of the primal and dual SDP appropriately.

Proof. Suppose the primal feasible set and dual feasible sets are not empty and let $X \in \mathcal{P}, Y \in \mathcal{D}$ be elements of these sets. Then

$$\langle A, X\rangle \geq \langle \Psi^*(Y), X\rangle = \langle Y, \Psi(X)\rangle \geq \langle B, Y\rangle,$$

due to the restrictions on X and Y in the above SDP. If either $\mathcal{P} = \emptyset$ or $\mathcal{D} = \emptyset$ holds, (4.25) would follow immediately, because p or d have infinite values in these cases. If $\mathcal{P}$ is empty $p = \infty$, if $\mathcal{D}$ is empty $d = -\infty$. □

Remember that linear programs, obey strong duality ($\exists p, d \in \mathbb{R} : p = d$) if there exists an optimal solution to the primal problem, for instance. Additionally optimal solutions to semidefinite programs have to obey a statement concerned with positive definiteness for strong duality to hold. Positive definiteness can be defined analogously to positive semidefiniteness. A matrix $X \in Herm(\mathcal{X})$ is called positive definite if $v^t X v > 0$ for all $v \in \mathcal{X}\backslash\{0\}$. Moreover, the notation $A > B$ or $B < A$ is used to describe the positive definiteness of $A - B$ for $A, B \in Herm(\mathcal{X})$. This notation allows a compact presentation of the strong duality theorem for semidefinite programs:

Theorem 7. Let (Ψ, A, B) be a semidefinite program as above then the following two implications hold:

1. If p is finite and $\exists Y \in Pos(\mathcal{Y})$, such that Y is positive definite and $\Psi^*(Y) < A$, then $p = d$ and $\exists X \in \mathcal{P}$, s.t. $\langle A, X\rangle = p$.

2. If d is finite and $\exists X \in Pos(\mathcal{X})$, such that X is positive definite and $\Psi(X) > B$, then $p = d$ and $\exists Y \in \mathcal{D}$, s.t. $\langle B, Y\rangle = d$.

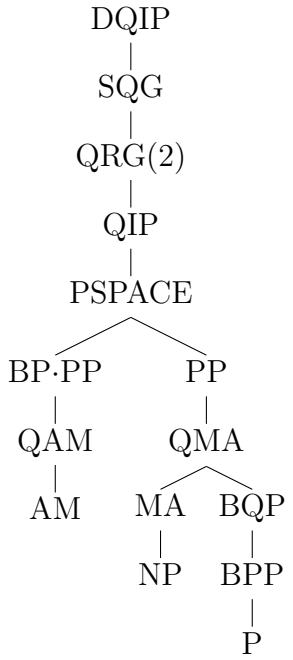
Since this theorem is standard knowledge in Convex optimization and the focus of the thesis lies upon quantum phenomena, a proof is not presented. Moreover, the survey from Vanderberghe and Boyd [VB96] discusses many different ways of tackling sample SDPs at the state of art. At start the equivalence of different formulations of one SDP might cause similar questions on first sight as in linear programming. But these problems vanish as one gets used to the application of SDPs. Moreover, since SDPs are solvable by generalizations of interior point methods polynomial-time SDP algorithms exist. Therefore, all polynomial sized SDPs can be solved in PSPACE. In the main part the problem will be the pure size of the input, which is $O(2^n)$ for n-qubit systems. Here interior point methods would need exponential time, but might possibly yield an exponential usage of space as well. Therefore, these methods cannot be used. Instead we

will rely upon the MMW method, which was introduced in the previous section. More information on convex optimization is available in the book from Boyd and Vanderberghe [BV04]. The above thoughts conclude the preliminaries. In the next step the thesis explores the path to the latest advances in complexity theory of quantum computation concerned with interactive protocols.

5 Quantum interactive proofs

Before we start to examine the results leading to the proof of QIP=PSPACE by Jain, Ji, Upadhyay and Watrous [JJUW09], we give an overview of some quantum complexity classes and the previously known relations between these classes. The complexity theoretic boundaries of quantum computation are not known yet. Even though it is widely believed that BQP, the quantum analogue to P, and P are if not the same at least close to each other. Only few relations between quantum and classical complexity classes were known before 2010.
Some of the quantum complexity classes we are about to examine are not defined formally yet. But their classical counterparts are standard knowledge in complexity theory and the generalization to quantum complexity classes can be achieved by allowing quantum computation and messages instead of classical computation and information. The only classes not derived from standard classical ones are SQG and DQIP. SQG stands for short quantum games. In these refereed two-player games each player is asked a question separately, opposed to this the players in a standard quantum refereed game with two turns (QRG(2)) are questioned at the same time. DQIP stands for double quantum interactive proofs. It is a generalization of SQG, in which the referee can exchange a constant number of messages with each player. But the communication of the two players with the referee is separated just like in SQG.
Examining Figure 5.1 we observe that most relations are either trivial or rely upon well known classical facts. Exceptions are the subset relations QAM $\subseteq$ BP·PP and QMA $\subseteq$ PP, which were proven in 2005 [MW05]. Notice that many relations between the presented quantum complexity classes are uncertain. There is for example an oracle O for which AM^O is not a subset of QMA^O. On the other hand the upper part of the diagram will collapse to PSPACE, as a proof for DQIP = PSPACE [GW11] will be presented in the next chapter of this thesis. Prior to this the celebrated proof of QIP = PSPACE [JJUW10] will be explained completely in section 5.3. This was one of the first general results on the boundaries of quantum computation in terms of classical complexity classes. Combined with IP = PSPACE it even establishes an equivalence of quantum and classical computation in interactive proofs: IP = QIP.
In order to understand the state of the discussion, before QIP = PSPACE was proven, we will start to examine the proof of QIP = QIP(3) from Kitaev and Watrous [KW00], as well as the one of QMAM = QIP(3) from Marriott and Watrous [MW05]. The number in brackets refers to the number of rounds a game is played, as in classical complexity theory involving round-based games. An intermediate results will not be discussed in detail, but at least mentioned here. In 2009 Jain and Watrous presented a proof for QIP(2) $\subseteq$ PSPACE.

Figure 5.1 Relations between quantum complexity classes



5.1 QIP = QIP(3)

Most of this section is due to Kitaev and Watrous, who surprisingly proved, that quantum interactive proofs can be reduced to three rounds of communication, without decreasing its computational potential at all [KW00]. In order to present a complete proof QIP is defined initially. Afterwards the completeness is increased and the number of rounds is reduced by parallelization. Finally, the soundness error is decreased appropriately. Especially in the proofs of Theorem 8 and Theorem 9 important details were added. Moreover, notice that an equivalent classical statement is unlikely to hold, since it would collapse the polynomial hierarchy at the second level. Therefore, one of the few key differences between classical and quantum computation is pointed out here.

5.1.1 Definition of QIP

Following the classical interactive proofs, the omniscient prover P tries to persuade a computationally bounded verifier V of some statement. Unlike in the classical version messages contain quantum information and V may use a quantum computer to verify or discard the statement. In order to formalize the notion of quantum interactive proofs let $\Sigma^* = \{0,1\}^*$ be the standard language set. Then a quantum verifier is a polynomial-time computable mapping V such that

$$\forall x \in \Sigma^*, \exists k \in poly: \quad \mathrm{V}(x) = \big(V_1(x), \ldots, V_{k(|x|)}(x)\big)\,.$$

Each $V_i(x)$ represents a quantum circuit acting on $q_{\mathrm{V}}(x) + q_{\mathrm{M}}(x)$ qubits, where $q_{\mathrm{V}}(x)$ and $q_{\mathrm{M}}(x)$ are polynomially bounded functions referring to the qubits of the verifier and the qubits of the message, respectively.
Analogously, we denote by $\mathrm{P}(x) = (P_1(x), \ldots, P_{l(|x|)}(x))$ a prover. Here each $P_i(x)$ acts on $q_{\mathrm{P}}(x)+q_{\mathrm{M}}(x)$ qubits, where $q_{\mathrm{P}}(x)$ refers to the private qubits of the prover. Sometimes we will drop the index or the variable x for simplicity. As in a classical interactive proof V and P exchange m messages. Of course, they have to agree on the number of messages and their size to be compatible as quantum circuits are always constructed for a certain number of qubits. In this case V and P are called m-message verifier and m-message prover, respectively. Furthermore, both apply unitary operations to their memories and the messages. In the end V measures the initial part of his memory and accepts if the output is one.
If the prover starts to send messages the total number of messages is odd and one has to choose $l = (m+1)/2$ and $k = (m-1)/2$. On the other hand if the verifier starts to send messages the total number of messages is even and one has to choose $l = k = m/2$. Notice that an interactive proof with an even number of messages can be simulated by one with an odd number of messages. The verifier and the prover simply agree on the first message and V rejects immediately if it differs. Generally the output is denoted by the function $\mathrm{out}_{\mathrm{V,P}} : \Sigma^* \to \{0,1\}$. In a quantum setting all the circuits describing V and P are applied to $q_{\mathrm{V}}(x) + q_{\mathrm{M}}(x) + q_{\mathrm{P}}(x)$ qubits, which are in state $|0\rangle$, initially. Therefore, the operators the verifier applies are tensored with the identity on the qubits of the prover's workspace and vice versa.
These definitions enable us to describe the complexity class, $\mathrm{QIP}(m, c, s)$, where m represents the number of messages, c the completeness probability, and s the soundness probability.

Definition 2. If $m : \mathbb{Z}^+ \to \mathbb{N}$ and $c, s : \mathbb{Z} \to [0,1]$, then a language $L \subseteq \Sigma^*$ is in $\mathrm{QIP}(m, c, s)$ if $\exists$ a m-message verifier V, such that

$$\begin{aligned} &\forall x \in L : \exists \text{ m-message prover P } : \Pr[\mathrm{out}_{\mathrm{V,P}}(x) = 1] \geq c(|x|) \qquad \text{(completeness)} \\ &\forall x \notin L : \forall \text{ m-message prover P } : \Pr[\mathrm{out}_{\mathrm{V,P}}(x) = 1] \leq s(|x|) \qquad \text{(soundness).} \end{aligned}$$

Furthermore, we will use the following notation

$$\mathrm{QIP}(poly, c, s) = \bigcup_{m \in poly} \mathrm{QIP}(m, c, s).$$

Analogously to its classical counterpart, $\mathrm{IP} = \mathrm{IP}(poly, c, s)$, the equation, $\mathrm{QIP} = \mathrm{QIP}(poly, c, s)$, holds, if c and s are separated polynomially, meaning $\exists p \in poly : (c-s) < 1/p$. Despite the fact that quantum information is shared, Definition 2 is analogous to the classical one for $\mathrm{IP}(m, c, s)$. But the boundaries on the size of the registers are very tight, since both V and P have to know the exact size of the others register. Otherwise their unitary operators could not be applied. Classically the size of the message is of little importance as both participants have to able to process the information consecutively. Therefore, the size of the workspaces is not essential for classical interactive proofs.

5.1.2 Perfect completeness

This section describes how perfect completeness can be achieved by only two additional rounds of communication. Kitaev and Watrous found a suitable postprocessing protocol for the verifier in case he rejects, even though the input was a yes-instance. Perfect completeness will be guaranteed on cost of a high soundness error:

Theorem 8. Let $m \in poly$. Observe for $a : \mathbb{Z} \to [0,1]$ the family of polynomial-time uniformly generated quantum circuits, whose members perform a unitary transformation $\Phi_{c(n)}$ with the following effect on pure states:

$$\Phi_{c(n)}(|0\rangle) = \sqrt{c(n)}|0\rangle - \sqrt{1-c(n)}|1\rangle,$$

$$\Phi_{c(n)}(|1\rangle) = \sqrt{1-c(n)}|0\rangle - \sqrt{c(n)}|1\rangle.$$

Let $b : \mathbb{Z}^+ \to [0,1]$ satisfy $s(n) < c(n) \forall n \in \mathbb{N}$ then
$QIP(m,c,s) \subseteq QIP(m+2, 1, 1-(c-s)^2)$.

Notice that the transformations $\Phi_{c(n)}$ can perform a reflection among the y-axis for $c(n) = 1$ and even a $3/2\pi$-rotation for $c(n) = 0$. But this second extreme situation can be ruled out, as $1/2 < c(n)$.

Proof. Observe a protocol which verifies $L \in \mathrm{QIP}(m,c,s)$. Without loss of generality one can assume, that the completeness condition in Definition 2 is satisfied with equality by such a protocol. In order to prove $L \in QIP(m+2, 1, 1-(c-s)^2)$ the original protocol is modified in the following way:

1. The original protocol is executed, but the verifier does not accept or reject. Instead he holds two additional one-qubit registers $\mathbf{B}$ and $\mathbf{B}'$, both initially zero. Let $\mathbf{R}$ be the verifiers register after the original protocol was executed. Now if and only if V would reject originally he increments both $\mathbf{B}$ and $\mathbf{B}'$.

2. V$\to$ P: Register $(\mathbf{B}', \mathbf{R})$ and the prover performs U on $(\mathbf{B}', \mathbf{R})$.

3. P$\to$ V: Register $\mathbf{B}'$. V applies a CNOT operation to $(\mathbf{B}, \mathbf{B}')$ and performs $T_{c(n)}$ on $\mathbf{B}$. Finally, V accepts if and only if $\mathbf{B}$ is in the zero state, $|0\rangle$.

The notation V $\to$ P : $\mathbf{B}'$ symbolizes V sending the message $\mathbf{B}'$ to P. Let $|\psi\rangle$ be the state of register $\mathbf{R}$ together with any of the prover's private registers. Then after step 1 the entire system is in the mixed state

$$\alpha_{acc}|00\rangle|\psi_{acc}\rangle + \alpha_{rej}|11\rangle|\psi_{rej}\rangle,$$

where $\alpha_{acc}, \alpha_{rej} \in [0,1]$ and $|\psi\rangle = \alpha_{acc}|\psi_{acc}\rangle + \alpha_{rej}|\psi_{rej}\rangle$ Here $|\psi_{acc}\rangle$ and $|\psi_{rej}\rangle$ are the normalized projections of $|\psi\rangle$ onto accepting and rejecting states, respectively. Therefore, right before the application of $\Phi_{a(n)}$ in step 3 the system's state is

$$\alpha_{acc}|0\rangle|\phi_{acc}\rangle + \alpha_{rej}|1\rangle|\phi_{rej}\rangle,$$

where $|\phi_{acc}\rangle = U(|0\rangle|\psi_{acc}\rangle$ and $|\phi_{rej}\rangle$ is equivalent to $U(|1\rangle|\psi_{rej}\rangle$, but with $\mathbf{B}'$ flipped. The flip is due to the application of the CNOT gate as $\mathbf{B}$ is in state $|1\rangle$ in this case. Finally, after the verifier applied $\Phi_{c(n)}$ the probability of acceptance is

$$\left\| \alpha_{acc}\sqrt{c(n)}|\phi_{acc}\rangle + \alpha_{rej}\sqrt{1-c(n)}|\phi_{rej}\rangle \right\|^2.$$

In case of $x \in L$ we have $\alpha_{acc} = \sqrt{c(n)}$ and $\alpha_{rej} = \sqrt{1-c(n)}$. Now if the prover chooses U, such that $U(|0\rangle|\psi_{acc}\rangle = |0\rangle|\gamma\rangle$ and $U(|1\rangle|\psi_{rej}\rangle = |1\rangle|\gamma\rangle$, for arbitrary $|\gamma\rangle$, we get $|\phi_{acc}\rangle = |\phi_{rej}\rangle$. Since $|0\rangle|\psi_{acc}\rangle$ and $|1\rangle|\psi_{rej}\rangle$ are orthonormal, and U is norm preserving the probability of acceptance is

$$\|c(n)|0\rangle|\gamma\rangle + (1-c(n))|0\rangle|\gamma\rangle\|^2 = 1.$$

In case of $x \notin L$ the probability of acceptance is bounded by

$$\left(\alpha_{acc}\sqrt{c(n)} + \alpha_{rej}\sqrt{1-c(n)}\right)^2, \tag{5.1}$$

since $\|U|\zeta\rangle\| = \|\zeta\rangle\| = 1$ holds for every pure state $|\zeta\rangle$. Furthermore, substituting $\alpha_{rej} = \sqrt{1-\alpha_{acc}}$ and leaving away the index of α_{acc} and the argument of c, implies the following equivalent formulation of (5.1):

$$1 + 2\alpha^2 c - \alpha^2 - c + 2\alpha\sqrt{c(1-\alpha^2)(1-c)}.$$

In order to proof an upper bound of $1-(c-\alpha^2)^2$ for this term it is sufficient to proof the following inequality for all α and c under consideration

$$1 + 2\alpha^2 c - \alpha^2 - c + 2\alpha\sqrt{c(1-\alpha^2)(1-c)} \leq 1 - (c-\alpha^2)^2.$$

This can be formulated equivalently as

$$2\alpha\sqrt{c(1-\alpha^2)(1-c)} \leq \alpha^2 + c - c^2\alpha^4.$$

By squaring the whole inequality and subtracting the left side we end up with

$$0 \leq (-\alpha^2 + c - c^2 + \alpha^4)^2,$$

which holds for all $\alpha, c \in \mathbb{R}$ in general. The reverse direction holds because $0 \leq \alpha, c \leq 1$ and squaring is monotone in this case. This argument completes the proof, since the soundness condition implies

$$1 - (c(n) - \alpha_{acc}^2)^2 \leq 1 - (c(n) - s(n))^2,$$

as required. □

If $c'(n) \leq c(n), \forall n \in \mathbb{N}$ the subset relation, $QIP(m,c,s) \subseteq QIP(m,c',s)$, holds. Therefore, the remaining problem is to find a suitable $c'(n)$. This would allow us to come up with a family of polynomial-time uniformly generated quantum circuits in the Shor basis, performing the transformation $T_{c'(n)}$ exactly. One can find such a function c', obeying $s(n) < c'(n) \leq c(n)\ \forall n \in \mathbb{N}$. It can even be chosen exponentially close to c in a pointwise sense, as all unitary transformations can be approximated appropriately according to Theorem 3. Therefore,

$$\mathrm{QIP}(m,c,s) \subseteq \mathrm{QIP}(m+2,1,1-poly^{-1})$$

holds for $c-s \in poly^{-1}$. This means only two additional messages are needed to achieve perfect completeness. Moreover, the completeness and the soundness are still separated polynomially.

5.1.3 Parallelization

This section is concerned with the reduction of quantum interactive proofs to three rounds. A reduction to a constant number of rounds is unlikely to hold classically, as mentioned before. The following theorem reduces the number of rounds, such that the completeness remains perfect and the soundness does not increase to much.

Theorem 9. For $m \in poly$ and any function $\epsilon : \mathbb{Z} \to [0,1]$ the following holds

$$\mathrm{QIP}(m,1,1-\epsilon) \subseteq \mathrm{QIP}\left(3,1,1-\frac{\epsilon^2}{4m^2}\right).$$

The proof will just cover the case of m being odd, since a protocol where m is even can be simulated by one with an odd number of messages, as discussed before in Section 5.1.1. Choose $k=(m+1)/2$ and let $P_1,\ldots,P_k \in \mathcal{U}(\mathcal{M}\otimes\mathcal{P})$ and $V_1,\ldots,V_k \in \mathcal{U}(\mathcal{V}\otimes\mathcal{M})$ be the circuits the prover P and the verifier V apply to the initial state $|\psi_{init}\rangle \in \mathcal{V}\otimes\mathcal{M}\otimes\mathcal{P}$. Since the input x is fixed we can drop the arguments of P_i and V_i for notational ease. If Π_0 denotes the projection on zero states, concerning the verifiers register, and Π_{acc} denotes the projection on accepting states we define the maximum acceptance probability as

$$\mathrm{MAP}(V_1,\ldots,V_k) = \max_{P_1\ldots P_k \in \mathcal{U}(\mathcal{M}\otimes\mathcal{P})} \left\|\Pi_{acc}V_kP_k\ldots V_1P_1|\psi_{init}\rangle\right\|^2.$$

In order to prove Theorem 9 we need the following lemma:

Lemma 12. Let $\rho_1,\ldots\rho_k \in \mathcal{D}(\mathcal{V}\otimes\mathcal{M})$ satisfy $\rho_k = (V_k^\dagger\Pi_{acc}V_k)\rho_k(V_k^\dagger\Pi_{acc}V_k)$, $\rho_1 = \Pi_0\rho_1\Pi_0$, and $\mathrm{MAP}(V_1,\ldots,V_k) < 1-\epsilon$. Then

$$\sum_{j=1}^{k-1}\sqrt{F\left(\mathrm{tr}_{\mathcal{M}}V_j\rho_jV_j^\dagger, \mathrm{tr}_{\mathcal{M}}\rho_{j+1}\right)} \leq (k-1) - \frac{\epsilon^2}{8(k-1)}.$$

The condition on ρ_k implies that it represents an accepting state, the condition on ρ_1 implies that it represents the first unit vector in $\mathcal{V}\otimes\mathcal{M}$. Therefore, ρ_1 is the matrix E_1, where only the first entry is one all others are zero.

Proof. In order to proof Lemma 12 let $|\psi_1\rangle, \ldots, |\psi_k\rangle \in \mathcal{V} \otimes \mathcal{M} \otimes \mathcal{P}$ be purifications of $\rho_1, \ldots, \rho_k$, meaning $\mathrm{tr}_{\mathcal{P}}|\psi_j\rangle\langle\psi_j| = \rho_j$ for all $j \in \{1, \ldots, k\}$. According to Corollary 1, which is a version of Uhlmann's theorem, there exist unit vectors $|\zeta_j\rangle, |\phi_j\rangle \in \mathcal{V} \otimes \mathcal{M} \otimes \mathcal{P}$ and numbers $\eta_j \in \mathbb{R}$, satisfying $\mathrm{tr}_{\mathcal{M}\otimes\mathcal{P}}|\zeta_j\rangle\langle\zeta_j| = \mathrm{tr}_{\mathcal{M}} V_j \rho_j V_j^\dagger$, $\mathrm{tr}_{\mathcal{M}\otimes\mathcal{P}}|\phi_j\rangle\langle\phi_j| = \mathrm{tr}_{\mathcal{M}}\rho_{j+1}$, and $\||\zeta_j\rangle - |\phi_j\rangle\| \leq \eta_j$, such that

$$F\left(\mathrm{tr}_{\mathcal{M}} V_j \rho_j V_j^\dagger, \mathrm{tr}_{\mathcal{M}}\rho_{j+1}\right) = \left(1 - \frac{\eta_j^2}{2}\right)^2,$$

for each $j \in \{1, \ldots, k-1\}$. The following consideration holds for each $j \in \{1, \ldots, k-1\}$: Since $\mathrm{tr}_{\mathcal{M}\otimes\mathcal{P}}|\phi_j\rangle\langle\phi_j| = \mathrm{tr}_{\mathcal{M}\otimes\mathcal{P}}|\psi_{j+1}\rangle\langle\psi_{j+1}|$, the unitary equivalence of purifications stated in Theorem 1 ensures the existence of operators $Q_{j+1} \in \mathcal{U}(\mathcal{M} \otimes \mathcal{P})$, which satisfy $Q_{j+1}|\phi_j\rangle = |\psi_{j+1}\rangle$. Analogously, operators $R_{j+1} \in \mathcal{U}(\mathcal{M} \otimes \mathcal{P})$ exist such that $R_{j+1}(V_j|\psi_j\rangle) = |\zeta_j\rangle$. Moreover, $P_{j+1} = Q_{j+1}R_{j+1}$ and choose P_1 such that $P_1|\psi_{init}\rangle = |\psi_1\rangle$. Now, we can conclude the following inequality for each $j \in \{1, \ldots, k-1\}$

$$\|P_{j+1}V_j|\psi_j\rangle - |\psi_{j+1}\rangle\| = \||\zeta_j\rangle - |\phi_j\rangle\| \leq \eta_j,$$

which is due to the fact that Q_{j+1} is norm preserving. Therefore, the difference between $|\psi_k\rangle$ and the execution of the protocol on the initial state is bounded as

$$\|P_k V_{k-1} \cdots P_1|\psi_{init}\rangle - |\psi_k\rangle\| = \|P_k V_{k-1} \cdots P_2 V_1|\psi_1\rangle - |\psi_k\rangle\| \leq \sum_{j=1}^{k-1} \eta_j. \tag{5.2}$$

Together with the initial property of ρ_k, which implies $\|\Pi_{acc} V_k|\psi_k\rangle\| = 1$, one concludes

$$\|\Pi_{acc} V_k P_k \ldots V_1 P_1|\psi_{init}\rangle\| \geq 1 - \sum_{j=1}^{k-1} \eta_j.$$

By plugging in the provided bound on the maximum acceptance probability, we end up with

$$\sum_{j=1}^{k-1} \eta_j \geq 1 - \sqrt{1-\epsilon} \geq \epsilon/2.$$

Maximizing $\sum_j \eta_j^2$ under this condition yields

$$\sum_{j=1}^{k-1}\left(1 - \frac{\eta_j^2}{2}\right) = (k-1) - \frac{1}{2}\sum_{j=1}^{k-1}\eta_j^2 \leq (k-1) - \frac{\epsilon^2}{8(k-1)},$$

since every summands in $\sum_j \eta_j$ are bounded by $\epsilon/(2(k-1))$ in average and one factor $(k-1)$ cancels out as the sum is estimated. □

Now we utilize Lemma 12 for a proof of Theorem 9. To this end we consider the following verification procedure:

1. P $\to$ V : Registers $\mathbf{V}_1, \ldots, \mathbf{V}_k$ and $\mathbf{M}_1, \ldots, \mathbf{M}_k$.
   The verifier V rejects if $\mathbf{V}_1$ does not contain all zeros. Otherwise he applies V_k to $(\mathbf{V}_k, \mathbf{M}_k)$ and rejects if $(\mathbf{V}_k, \mathbf{M}_k)$ does not contain an accepting state. If no rejection occurs he performs $V_k^\dagger$ on $(\mathbf{V}_k, \mathbf{M}_k)$.

2. V prepares the register $(\mathbf{B}, \mathbf{B}')$ in state $|\phi^+\rangle = \frac{1}{\sqrt{2}}(|0\rangle|0\rangle + |1\rangle|1\rangle)$ and chooses $r \in_R \{1, \ldots, k-1\}$. Then he applies V_r to $(\mathbf{V}_r, \mathbf{M}_r)$ and performs a controlled swap between $\mathbf{V}_r$ and $\mathbf{V}_{r+1}$ with control bit $\mathbf{B}$.
   V $\to$ P : Registers $\mathbf{M}_r, \mathbf{M}_{r+1}, \mathbf{B}'$ and r.

3. P $\to$ V : Register $\mathbf{B}'$.
   V performs a controlled not operation (CNOT) on $(\mathbf{B}, \mathbf{B}')$ and a Hadamard transformation on $\mathbf{B}$. If $\mathbf{B}$ is in state $|0\rangle$ he accepts, else he rejects.

Observe that the verifier measures $\mathbf{V_1}$ with respect to Π_0 and $(\mathbf{V}_k, \mathbf{M}_k)$ with respect to $\Pi_{acc} V_k$ in step 1. Let $\rho_j \in \mathcal{D}(\mathcal{V} \otimes \mathcal{M})$, such that ρ_j is the state of $(\mathbf{V}_j, \mathbf{M}_j)$. We describe a prover P, who would cause a verifier following the above protocol, to accept with certainty, assuming $\mathrm{MAP}(V_1, \ldots, V_k) = 1$. The prover prepares $(\mathbf{V}_1, \mathbf{M}_1, \mathbf{P}_1)$ in state $P_1|\psi_{init}\rangle$ and $(\mathbf{V}_{j+1}, \mathbf{M}_{j+1}, \mathbf{P}_{j+1})$ in state $P_{j+1}P_jV_j \cdots V_1P_1|\psi_{init}\rangle$ for $j \geq 1$. In step 2 the prover applies P_{r+1} to $(\mathbf{M}_r, \mathbf{P}_r)$ and performs a controlled swap on $(\mathbf{M}_r, \mathbf{P}_r)$ and $(\mathbf{M}_{r+1}, \mathbf{P}_{r+1})$, using control bit $\mathbf{B}'$. Such a prover would also cause a 3-message verifier to accept with certainty in case $\mathrm{MAP}(V_1, \ldots, V_k) = 1$. In the part concerning the register $(\mathbf{B}, \mathbf{B}')$ all operators under consideration are tensored with identity, except the CNOT gate which applied in step 3. Therefore, only a Hadamard gate is applied to $\mathbf{B}$, resulting in an output of $|0\rangle$ before measuring as

$$H \otimes \mathbb{1}_{\mathbf{B}'}(\mathrm{CNOT}|\phi^+\rangle) = \frac{1}{\sqrt{2}} \begin{pmatrix} 1 & 0 & 1 & 0 \\ 0 & 1 & 0 & 0 \\ 1 & 0 & -1 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix} \frac{1}{\sqrt{2}} (|0\rangle|0\rangle + |1\rangle|0\rangle) = |0\rangle|0\rangle.$$

Remember the entries of the matrix are due to lexicographical order of the basis as discussed in Section 4.1.3. Therefore, the verifier accepts with certainty.
Now we consider the general case $\mathrm{MAP}(V_1, \ldots, V_k) = 1 - \epsilon$. Let $(|0\rangle|\phi_0\rangle + |1\rangle|\phi_1\rangle)/\sqrt{2}$, with $|\phi_0\rangle, |\phi_1\rangle \in \mathcal{V} \otimes \mathcal{K}$ be the state of the entire system after the CNOT gate was applied in step 3. Here $|\phi_0\rangle$ and $|\phi_1\rangle$ are unit vectors, whose $\mathcal{V}$ component describes the state of $\mathbf{V}_r$ and whose $\mathcal{K}$ component includes the other parts of the system besides $\mathbf{B}$ and $\mathbf{V}_r$. In the end the verifier performs a Hadamard transformation on $\mathbf{B}$ and accepts if the resulting qubit is in state $|0\rangle$. Therefore, the probability of acceptance is

$$\left\| \frac{1}{2}|0\rangle \left(|\phi_0\rangle + |\phi_1\rangle\right) \right\|^2 = \left(\frac{1}{2} \sqrt{\left(\langle 0|0\rangle\right) \left(\langle \phi_0|\phi_0\rangle + 2\langle \phi_0|\phi_1\rangle + \langle \phi_1|\phi_1\rangle\right)} \right)^2 = \frac{1}{2} + \frac{1}{2}|\langle \phi_0|\phi_1\rangle|.$$

Now because the equations $\mathrm{tr}_{\mathcal{K}}|\phi_1\rangle\langle\phi_1| = \mathrm{tr}_{\mathcal{M}}\rho_{r+1}$ and $\mathrm{tr}_{\mathcal{K}}|\phi_0\rangle\langle\phi_0| = \mathrm{tr}_{\mathcal{M}} V_r \rho_r V_r^\dagger$ hold, Uhlmann's theorem implies

$$|\langle \phi_0|\phi_1\rangle|^2 \leq F\left(\mathrm{tr}_{\mathcal{M}} V_r \rho_r V_r^\dagger, \mathrm{tr}_{\mathcal{M}} \rho_{r+1}\right).$$

Therefore, the right hand side of the above equality is bounded by the following probability

$$p_r = \frac{1}{2} + \frac{1}{2}\sqrt{F(\mathrm{tr}_{\mathcal{M}} V_r \rho_r V_r^\dagger, \mathrm{tr}_{\mathcal{M}} \rho_{r+1})}.$$

Moreover, using Lemma 12 we can calculate the total probability

$$\sum_{r=1}^{k-1} \frac{p_r}{k-1} \leq 1 - \frac{\epsilon^2}{16(k-1)^2} = 1 - \frac{\epsilon^2}{4(m-1)^2}.$$

Taking into account that m was assumed to be odd, one has to increase m by one if it is even. Finally, the required soundness is achieved.
This completes the reduction of QIP to three rounds. We only have to decrease the soundness error appropriately to achieve QIP = QIP(3).

5.1.4 Soundness error reduction

In order to reduce the soundness error, one needs the following lemma connecting the diamond norm of a superoperator to the maximum acceptance probability of a 3-message protocol:

Lemma 13. For V_1, V_2, Π_{acc} and Π_0 as before, let $\Phi \in \mathcal{T}(\mathcal{V} \otimes \mathcal{M}, \mathcal{M})$ respect $\Phi(X) = \mathrm{tr}_{\mathcal{V}}(V_1 \Pi_0 X \Pi_{acc} V_2)$. Then $\mathrm{MAP}(V_1, V_2) = \|\Phi\|_\diamond^2$.

Proof. Let $|\psi\rangle, |\phi\rangle \in \mathcal{V} \otimes \mathcal{M} \otimes \mathcal{P}$ be unit vectors then

$$\mathrm{MAP}(V_1, V_2) = \max_{\substack{|\psi\rangle, |\phi\rangle \in \mathcal{V}\otimes\mathcal{M}\otimes\mathcal{P} \\ U \in \mathcal{U}(\mathcal{M}\otimes\mathcal{P})}} \left\{ |\langle\phi|\Pi_{acc} V_2 U V_1 \Pi_0 |\psi\rangle|^2 \right\}.$$

The definition of the diamond applied to the case at hand gives

$$\|\Phi\|_\diamond = \max_{Y \in \mathcal{L}(\mathcal{V}\otimes\mathcal{M}\otimes\mathcal{P})} \left\{ \left\| \Phi \otimes \mathbb{1}_{\mathcal{L}(\mathcal{P})}(Y) \right\|_{\mathrm{tr}} : \|Y\|_{\mathrm{tr}} = 1 \right\},$$

if the prover holds enough private qubits, meaning $\mathcal{P}$ is large enough. Moreover, for all $Y \in \mathcal{L}(\mathcal{V} \otimes \mathcal{M} \otimes \mathcal{P})$ with $\|Y\|_{\mathrm{tr}} = 1$ there exist $\alpha_j \in \mathbb{R}$, satisfying $\sum_j |\alpha_j| = 1$, and unit vectors $|\psi_j\rangle, |\Pi^{(j)}\rangle \in \mathcal{V} \otimes \mathcal{M} \otimes \mathcal{P}$, such that $Y = \sum_j \alpha_j |\psi_j\rangle\langle\Pi^{(j)}|$. Therefore, the maximum is achieved by an operator of the form $|\psi\rangle\langle\phi|$ and we can conclude

$$\begin{aligned} \|\Phi\|_\diamond &= \max_{|\psi\rangle, |\phi\rangle \in \mathcal{V}\otimes\mathcal{M}\otimes\mathcal{P}} \left\{ \left\| \Phi \otimes \mathbb{1}_{\mathcal{L}(\mathcal{P})}(|\psi\rangle\langle\phi|) \right\|_{\mathrm{tr}} \right\} \\ &= \max_{|\psi\rangle, |\phi\rangle \in \mathcal{V}\otimes\mathcal{M}\otimes\mathcal{P}} \left\{ \|\mathrm{tr}_{\mathcal{V}}(V_1 \Pi_0 |\psi\rangle\langle\phi| \Pi_{acc} V_2)\|_{\mathrm{tr}} \right\}. \end{aligned}$$

The last equality is due to the condition on $\Phi(X)$ the lemma demands. Since unitary

operators can simulate the trace norm according to Lemma 3 we end up with

$$\begin{aligned}\|\Phi\|_\diamond^2 &= \max_{\substack{|\psi\rangle,|\phi\rangle\in\mathcal{V}\otimes\mathcal{M}\otimes\mathcal{P}\\ U\in\mathcal{U}(\mathcal{M}\otimes\mathcal{P})}} \left\{\left|\mathrm{tr}\left(U\mathrm{tr}_{\mathcal{V}}(V_1\Pi_0|\psi\rangle\langle\phi|\Pi_{acc}V_2)\right)\right|\right\}^2 \\ &= \max_{\substack{|\psi\rangle,|\phi\rangle\in\mathcal{V}\otimes\mathcal{M}\otimes\mathcal{P}\\ U\in\mathcal{U}(\mathcal{M}\otimes\mathcal{P})}} \left\{\left|\mathrm{tr}\left(U(V_1\Pi_0|\psi\rangle\langle\phi|\Pi_{acc}V_2)\right)\right|\right\}^2 \\ &= \max_{\substack{|\psi\rangle,\phi\rangle\in\mathcal{V}\otimes\mathcal{M}\otimes\mathcal{P}\\ U\in\mathcal{U}(\mathcal{M}\otimes\mathcal{P})}} \left\{\left|\langle\phi|\Pi_{acc}V_2UV_1\Pi_0|\psi\rangle\right|\right\}^2 = MAP(V_1,V_2).\end{aligned}$$

Here the second inequality utilizes the fact that the repeated application of the trace does not change the value: $\mathrm{tr}(\mathrm{tr}_{\mathcal{X}}A) = \mathrm{tr}(A), \forall A \in \mathcal{L}(\mathcal{X}\otimes\mathcal{Y})$. The third equality is due to the fact that $\mathrm{tr}(A|\psi\rangle\langle\phi|B) = \langle\phi|BA|\psi\rangle$ holds for any linear operators A, B and states $|\phi\rangle, |\psi\rangle$, where these products exist. □

The idea is now to run several protocols in parallel and to accept only if all individual protocols accepted. Moreover, Lemma 13 and Lemma 5 enable the description of the maximum acceptance probability in terms of diamond norms, if we choose suitable superoperators.

Theorem 10. Let $p \in poly$ and $s : \mathbb{Z}^+ \to [0,1]$ be any function, then $\mathrm{QIP}(3,1,s) \subseteq \mathrm{QIP}(3,1,s^p)$.

Proof. Let $L \in \mathrm{QIP}(3,1,s)$ and let V_1, V_2 be the description of a verifier V witnessing this fact. Then the choice of Φ, namely $\Phi(X) = \mathrm{tr}_{\mathcal{V}}(V_1\Pi_0 X\Pi_{acc}V_2), \forall X \in \mathcal{L}(\mathcal{V}\mathcal{M})$ implies

$$\begin{aligned}\|\Phi\|_\diamond^2 &= 1 \quad \text{for } x \in L \text{ and} \\ \|\Phi\|_\diamond^2 &\leq s \quad \text{for } x \notin L,\end{aligned} \tag{5.3}$$

according to Lemma 13 which states $\mathrm{MAP}(V_1,V_2) = \|\Phi\|_\diamond$. Now we define a new verifier V′, who runs p protocols of the original verifier V in parallel. He accepts if and only if all p single protocols accept. Moreover, we generalize V_1, V_2, Π_0 and Π_{acc} as follows

$$\begin{aligned}V_i' &= V_i \otimes \cdots \otimes V_i \quad \text{for } i \in \{1,2\} \\ \Pi_0' &= \Pi_0 \otimes \cdots \otimes \Pi_0 \\ \Pi_{acc}' &= \Pi_{acc} \otimes \cdots \otimes \Pi_{acc},\end{aligned}$$

where each tensor product has p factors. These definition allow a compact description of Φ':

$$\Phi'(X) = \mathrm{tr}_{\mathcal{V}}\left(V_1'\Pi_0'X\Pi_{acc}'V_2'\right) \quad \forall X \in \mathcal{L}(\mathcal{V}\mathcal{M}\otimes\cdots\otimes\mathcal{V}\mathcal{M}),$$

where the tensor product has p factors again. Thus, we have $\Phi' = \Phi\otimes\cdots\otimes\Phi$ and by Lemma 5, we can conclude

$$\|\Phi'\|_\diamond = \|\Phi\otimes\cdots\otimes\Phi\|_\diamond = \|\Phi\|_\diamond^p.$$

Finally, Lemma 13 implies $\mathrm{MAP}(V_1', V_2') = \|\Phi'\|_\diamond^2$ and by (5.3) we get

$$\mathrm{MAP}(V_1', V_2') = \|\Phi\|_\diamond^p = 1 \quad \text{for } x \in L \text{ and}$$
$$\mathrm{MAP}(V_1', V_2') = \|\Phi\|_\diamond^p \leq s^p \quad \text{for } x \notin L.$$

□

Combining the results of Theorem 8, Theorem 9, and Theorem 10), we finally conclude the following corollary:

Corollary 2. Let $p \in poly, \epsilon \in poly^{-1}$, and let $c, s : \mathbb{Z}^+ \to [0, 1]$ be functions, such that c and s satisfy $c(n) - s(n) \geq \epsilon(n)$ for every n. Then $\mathrm{QIP}(poly, c, s) \subseteq \mathrm{QIP}(3, 1, 2^{-p})$ holds.

Since it is one of the most important facts about quantum computation we can not stress this conclusion enough. The above statement, QIP = QIP(3), reveals a difference between quantum and classical computation. If classical interactive proofs could be reduced to a constant number of rounds, the polynomial hierarchy would collapse at the second level. Thus P/poly, the class of decision problems, which can be solved in polynomial time by a Turing machine using polynomial advice, might contain the class NP as the Karp-Lipton theorem would not suggest the opposite statement anymore. This would actually be quite strong evidence for P=NP. For detailed information on this classical topic see for example [AB09].

5.2 QIP(3) = QMAM

In order to prove QIP = PSPACE we have to express the class QIP(3) in terms of quantum Arthur-Merlin classes. Therefore, we have to examine the class QMAM first. Afterwards we utilize a suitable protocol to prove QIP ⊆ QMAM. This section is mostly due to a paper from Marriott and Watrous [MW05]. Nevertheless, a couple of details were added, explaining Corollary 3, Corollary 4 and their usage.

5.2.1 Definition of QMAM

In the class QMAM the prover is the magician Merlin (M). Since he can see Arthur's actions they play a public coin game. Furthermore, Arthur can only send random bits as messages. Sending only one bit does not decrease the computational power of this game. In this interactive protocol Merlin sends the first message, Arthur responds with some random bit and then Merlin sends a second message. Finally, Arthur can apply quantum circuits and a measurement.
In order to define $\mathrm{QMAM}(c, s)$ formally let $m_1, m_2 \in poly$ be the functions that specify the number of qubits in Merlin's first and second message register, respectively. In addition to the message space $\mathcal{M}_1$ and $\mathcal{M}_2$ Merlin also holds a private work space $\mathcal{P}$, which is not presented to Arthur at any point of the game. Moreover, denote by $s \in poly$ the number of random bits Arthur sends. Additionally, Arthur holds a uniformly

polynomial-time generated family of quantum circuits,whose elements are called $A_{x,y}$, where $x \in \Sigma^*$ is the input and $y \in \Sigma^s$ are the coin tosses. These elements are the actions Arthur performs conditioned on the input and the outcome of the coin tosses.
Note that Merlin can not change the information he sent to Arthur in the first message, after he received the coin tosses. Therefore, his only freedom lies in selecting the initial state $|\psi\rangle \in \mathcal{M}_1 \otimes \mathcal{M}_2 \otimes \mathcal{P}$ and applying unitary operators of the form

$$\mathbb{1}_{\mathcal{M}_1} \otimes U_y,$$

which are conditioned on the results of the coin tosses. Obviously he is also free to choose the dimension of his workspace, defined by $p = \dim(\mathcal{P})$. Therefore, the number p the collection of unitary operators $\{U_y : y \in \Sigma^s\}$ and the initial state $|\psi\rangle$ completely specify the prover Merlin. These definitions and considerations enable a compact definition of the class QMAM. Analogously to other interactive classes we have to introduce soundness and completeness.

Definition 3. A language $L \subseteq \Sigma^*$ belongs to QMAM(c, s) if the following conditions hold:

1. If $x \in L$ then a prover Merlin exists, who can convince Arthur with a probability of at least c, meaning $\exists p \in \mathbb{N}, |\psi\rangle \in \mathcal{M}_1 \otimes \mathcal{M}_2 \otimes \mathcal{P}, \{U_y : y \in \Sigma^s\} \subseteq U(\mathcal{M}_2 \otimes \mathcal{P})$:

$$\frac{1}{2^s} \sum_{y \in \Sigma^s} \Pr[\mathrm{out}_{A_{x,y}}((\mathbb{1}_{\mathcal{M}_1} \otimes U_y)|\psi\rangle) = 1] \geq c.$$

2. If $x \notin L$ then no prover Merlin can convince Arthur with a probability higher than s, meaning $\forall p \in \mathbb{N}, |\psi\rangle \in \mathcal{M}_1 \otimes \mathcal{M}_2 \otimes \mathcal{P}, \{U_y : y \in \Sigma^s\} \subseteq U(\mathcal{M}_2 \otimes \mathcal{P})$:

$$\frac{1}{2^s} \sum_{y \in \Sigma^s} \Pr[\mathrm{out}_{A_{x,y}}((\mathbb{1}_{\mathcal{M}_1} \otimes U_y)|\psi\rangle) = 1] \leq s.$$

Moreover, Arthur acts on $m_1 + m_2 + p$ qubits. But since Merlin keeps the last p qubits of $|\psi\rangle$, Arthur cannot change them. Therefore, he applies the identity on these qubits.

Notice that the sum is taken over all possible coin tosses. Furthermore, the left sides of both inequalities are bounded by one, since the probability of acceptance is at most one for each outcome of the coin tosses, whereas 2^s is the number of possible outcomes for s coin tosses and therefore the amount of summands.

5.2.2 QIP(3) $\subseteq$ QMAM

In order to prove QMAM = QIP(3), we only need to discuss QIP(3) $\subseteq$ QMAM. The other containment is trivial, since quantum Arthur-Merlin-Games are a restricted form of quantum interactive proof systems. The following theorem nearly suffices to show QIP(3) $\subseteq$ QMAM. It only remains to reduce the soundness error.

Theorem 11. Let $L \in \mathrm{QIP}(3, 1, 2^{-2p})$ and $p \in poly$. Then $L \in \mathrm{QMAM}(1, 1/2 + 2^{-p})$ with $s = 1$, meaning Arthur only sends one random bit.

Proof. Let $\mathcal{P}$, $\mathcal{M}$, $\mathcal{V}$ be the prover's work space, the message space and the verifier's work space respectively with dimensions, p, m and k, and let $\mathbf{P}, \mathbf{M}, \mathbf{V}$ be the corresponding registers. Furthermore, let $|\psi\rangle$ be the first message of the prover, $\Pi_{acc} \in Meas(\mathcal{VM})$ the projection onto accepting states, $U \in U(\mathcal{MP})$ the unitary operator the prover applies, and $V_1, V_2 \in U(\mathcal{VM})$ the unitary operators the verifier applies, then the maximum probability a verifier can be forced to accept in a QIP(3)-protocol is

$$\left\|(\Pi_{acc} \otimes \mathbb{1}_{\mathcal{P}})(V_2 \otimes \mathbb{1}_{\mathcal{P}})(\mathbb{1}_{\mathcal{V}} \otimes U)(V_1 \otimes \mathbb{1}_{\mathcal{P}})(|0^k\rangle|\psi\rangle)\right\|^2.$$

Suppose Arthur acts the following way in a QMAM-protocol:

1. M $\to$ A: Register $\mathbf{V}$
2. Arthur flips a coin. A $\to$ M: The result of the coin.
3. M $\to$ A: Register $\mathbf{M}$. If the coin shows heads Arthur applies V_2 to $(\mathbf{V}, \mathbf{M})$ and accepts if the first qubit of $\mathbf{V}$ is in state $|1\rangle$, else he rejects. If the result of the coin is tails he applies V_1^* to $(\mathbf{V}, \mathbf{M})$ and accepts if all qubits of $\mathbf{V}$ are in state $|0\rangle$, otherwise he rejects.

If $x \in L$, there exists a prover, Merlin, who can convince Arthur to accept always. Let the state $|\psi\rangle$ and the unitary operator U describe the prover. Then Merlin can force Arthur to accept with certainty as follows:

1. Prepare state $|0^k\rangle$ in register $\mathbf{V}$ and state $|\psi\rangle$ in register $(\mathbf{M}, \mathbf{P})$. Apply V_1 to $(\mathbf{V}, \mathbf{M})$. M $\to$ A: Register $\mathbf{V}$.
2. If the coin shows heads apply U to $(\mathbf{M}, \mathbf{P})$: M $\to$ A: Register $\mathbf{M}$, otherwise M $\to$ A: Register $\mathbf{M}$.

In step 2 Merlin applies U only if the coin shows heads, in case of tails $\mathbf{M}$ is sent unchanged. Thus, we get exactly the same accepting probability as in the QIP(3)-protocol if the coin shows heads

$$\left\|(\Pi_{acc} \otimes \mathbb{1}_{\mathcal{P}})(V_2 \otimes \mathbb{1}_{\mathcal{P}})(\mathbb{1}_{\mathcal{V}} \otimes U)(V_1 \otimes \mathbb{1}_{\mathcal{P}})(|0^k\rangle|\psi\rangle)\right\|^2. \tag{5.4}$$

On the other hand if the coin shows tails one gets $V_1^*(V_1(|0^k\rangle, \mathbf{M})) = (|0^k\rangle, \mathbf{M})$. Since the hole register $\mathcal{V}$ is in the state $|0\rangle$ Arthur also accepts with certainty in this case.
For the soundness ($x \notin L$) we need some more definitions and lemmas. The following notation is not really standard in quantum computation, even though it is used in some papers. But it enables a compact description of accepting probabilities and will therefore be explained in detail. Notice that the paper of Kitaev and Watrous used a similar description without adding the notation, we are about to examine.

Let $S(\Pi)$ denote the set of all mixed states $\rho \in \mathcal{D}(\mathcal{VM})$, satisfying $\rho = \Pi\rho\Pi$ for some projection $\Pi \in Pro(\mathcal{VM})$, and define

$$S_V(\Pi) = \{\mathrm{tr}_M \rho : \rho \in S(\Pi)\}.$$

Notice that $S_V(\Pi)$ is the collection of states whose support is contained in the space onto which Π projects. Moreover, we can connect such sets to the accepting probability by a corollary, which follows from Uhlmann's theorem (Theorem 2).

Corollary 3. Let the register $(\mathbf{V}, \mathbf{M})$ be in a mixed state τ, such that $\mathrm{tr}_M \tau = \sigma$. If $(\mathbf{V}, \mathbf{M})$ is measured with respect to the binary-valued measurement $\{\Pi_{rej}, \Pi_{acc}\}$, the probability of acceptance is less than $F(\rho, \sigma)^2$ for some $\rho \in S_V(\Pi_{acc})$.

First note that $S_V(\Pi_{acc})$ is the set of accepting states. Furthermore, let $|\zeta\rangle \in \mathcal{XVM}$ be a purification of τ. Since $|\zeta\rangle$ is also a purification of σ Uhlmann's theorem implies

$$F(\sigma, \rho)^2 = \max_{|\phi\rangle} |\langle\zeta|\phi\rangle|^2, \tag{5.5}$$

where the maximum is taken over purifications $|\phi\rangle$ of ρ. Moreover, due to the monotonicity of the trace we can conclude the following for the probability of acceptance:

$$\langle \Pi_{acc}, \tau\rangle = \mathrm{tr}(\Pi_{acc}\mathrm{tr}_{\mathcal{X}}(|\zeta\rangle\langle\zeta|)) = \langle\zeta|\Pi_{acc}|\zeta\rangle.$$

This term is clearly bounded by the right hand side of (5.5). The next corollary is concerned with the acceptance probability in a QIP(3)-protocol. It is also due to Uhlmann's theorem.

Corollary 4. A verifier, who applies V_1 and V_2, can be forced to accept with a maximum probability of

$$\max\{F(\rho, \zeta)^2 : \rho \in S_V(V_1\Pi_0 V_1^*), \zeta \in S_V(V_2^*\Pi_{acc}V_2)\}.$$

Here we use $\Pi_0 = |0^k\rangle\langle 0^k| \otimes \mathbb{1}_{\mathcal{M}}$ and $\Pi_{acc} = |1\rangle\langle 1| \otimes \mathbb{1}_{\mathcal{V}'\mathcal{M}}$, where $\mathcal{V}'$ is the space $\mathcal{V}$ without the first qubit.

In order to understand this corollary observe that the maximum acceptance probability can be formulated in the following way:

$$\max_{|\phi\rangle,|\psi\rangle,U} |\langle\phi|\Pi_{acc}V_2UV_1\Pi_0|\psi\rangle|^2,$$

where the maximum is taken over all unit vectors $|\phi\rangle, |\psi\rangle \in \mathcal{VMP}$ and all $U \in U(\mathcal{MP})$. Notice that the tensored identities are skipped to present a simple notation. Now, we utilize the fact that $\mathrm{tr}(\mathrm{tr}_{\mathcal{X}}A) = \mathrm{tr}(A)$ as we did in the proof of Lemma 13 and Uhlmann's theorem to end up with the desired formulation in terms of the fidelity.
Furthermore, the fidelity function satisfies (4.16):

$$\forall \rho, \zeta, \sigma \in \mathcal{D}(\mathcal{X}): \quad F(\rho, \sigma)^2 + F(\sigma, \zeta)^2 \leq 1 + F(\rho, \zeta).$$

Suppose $x \notin L$, then no prover can convince the verifier with a higher probability than $\epsilon = 2^{-2p}$. Let σ be the reduced density matrix of the register $\mathbf{V}$ Merlin sends. By Corollary 3 the probability that Arthur accepts is less than

$$\frac{1}{2}F(\rho,\sigma)^2 + \frac{1}{2}F(\zeta,\sigma)^2 \leq \frac{1}{2} + \frac{1}{2}F(\rho,\zeta), \tag{5.6}$$

maximized over $\rho \in S_V(V_1\Pi_0 V_1^\dagger)$ and $\zeta \in S_V(V_2^\dagger \Pi_{acc} V_2)$. The inequality holds according to (4.16), while the factor 1/2 is due to the coin flip. Moreover, the reduced state ρ corresponds to the case where the coin shows tails as Arthur applies V_1^* and accepts if and only if all qubits of register $\mathbf{V}$ are in state $|0\rangle$. On the other hand the reduced state ζ corresponds to the case where the coin shows heads as Arthur applies V_2 and accepts if the first qubit is in state $|1\rangle$. By Corollary 4 we find the following upper bound on the right side of (5.6):

$$\frac{1}{2} + \frac{\sqrt{\epsilon}}{2} \leq \frac{1}{2} + 2^{-p}.$$

□

This implies for any $p \in poly$ QIP $\subseteq$ QMAM$(1, 1/2 + 2^{-p})$ due to QIP $\subseteq$ QIP(3), stated in Corollary 2.
In order to reduce the soundness error, we just repeat the protocol n times in parallel and accept if and only if all n repetitions accept. Merlin gains no advantage over playing the repetitions independently, since this was already proven for a general 3-message quantum interactive proof system.
Finally, we can conclude QIP $\subseteq$ QMAM$(1, 2^{-p})$ for any $p \in poly$, which implies QIP = QMAM$(1, 2^{-p})$. Actually, the weaker equality QIP = QMAM$(1, 1/2+2^{-p})$, which follows directly from Theorem 11, will enable us to establish the equivalence of classical and quantum computation in interactive proof systems.

5.3 QMAM = PSPACE

In this section we consider the proof of QIP = PSPACE, which was published in 2009 by Jain, Ji, Upadhyay and Watrous [JJUW09]. They reviewed their paper and did another version in 2010 [JJUW10]. Even though they skipped some details, their proof was accepted and celebrated by the audience. Initially, a SDP formulation for problems in QMAM$(1, 1/2 + \epsilon)$ is provided. Afterwards we use a parallel algorithm based upon the multiplicative weight update method (see Section 4.5) to solve this SDP. Finally, we discuss the NC implementation and the precision issues. Concretizing the stated papers a lot of details were added.

5.3.1 SDP formulation for QMAM

First we explain how a single-coin quantum Arthur-Merlin game can be expressed by a semidefinite program. Let $\mathbf{X}, \mathbf{Y}$ be the quantum registers Merlin sends in his first and

second message and let $\mathcal{X}$, $\mathcal{Y}$ be the corresponding complex vector spaces. Furthermore, $\{\Pi^a_{acc}, \Pi^a_{rej} : a \in \{0,1\}\} \subset \mathrm{Pos}(\mathcal{X} \otimes \mathcal{Y})$ are projection operators on $\mathcal{X} \otimes \mathcal{Y}$ representing Arthur's binary-valued measurement for each outcome of the coin a. After Arthur received the first message from Merlin he sends the result of the coin toss to Merlin. In the end he measures the register $(\mathbf{X}, \mathbf{Y})$ with respect to $\{\Pi^a_{rej}, \Pi^a_{acc}\}$. In the previous section we proved, that perfectly complete QMAM-games with a soundness error, which is bigger than $1/2$ by an exponentially small amount are powerful enough to decide any language, which belongs to QIP: $\mathrm{QIP} \subseteq \mathrm{QMAM}(1, 1/2 + \epsilon)$. Therefore, the maximum probability a verifier can be forced to accept in such a game, is the optimal value of the following SDP:

$$\begin{aligned} \text{maximize}: \quad & \frac{1}{2}\langle \Pi^0_{acc}, \rho_0 \rangle + \frac{1}{2}\langle \Pi^1_{acc}, \rho_1 \rangle \\ \text{subject to}: \quad & \mathrm{tr}_{\mathcal{Y}}(\rho_0) = \mathrm{tr}_{\mathcal{Y}}(\rho_1) \\ & \rho_0, \rho_1 \in \mathcal{D}(\mathcal{X} \otimes \mathcal{Y}), \end{aligned}$$

where ρ_0 and ρ_1 are density operators representing the possible states of the register-pair $(\mathbf{X}, \mathbf{Y})$ conditioned on the outcome of the coin toss. Moreover, the factor $1/2$ is due to the coin toss, while the description of the acceptance probability in terms of the Hilbert-Schmidt inner product was initially mentioned in Section 4.3.1. Therefore, it only remains to explain the constraints. The necessity of the condition

$$\mathrm{tr}_{\mathcal{Y}}\rho_0 = \sigma = \mathrm{tr}_{\mathcal{Y}}\rho_1 \tag{5.7}$$

is clear, since Merlin does not know the outcome of the coin-flip, when he sends $\mathbf{X}$ to Arthur. Therefore, ρ_0 and ρ_1 have to agree on $\mathbf{X}$. The sufficiency follows from the unitary equivalence of purifications: If Merlin holds a purification of state σ he is free to set the state of $(\mathbf{X}, \mathbf{Y})$ to any choice of ρ_0 and ρ_1, satisfying (5.7), without having access to $\mathbf{X}$. Let $\mathcal{A}$ be the 2 dimensional complex vector space corresponding to Arthur's random bit a. Once we define

$$\rho = \begin{pmatrix} \frac{1}{2}\rho_0 & 0 \\ 0 & \frac{1}{2}\rho_1 \end{pmatrix} \in \mathcal{D}(\mathcal{A} \otimes \mathcal{X} \otimes \mathcal{Y}),$$

we can conclude

$$\mathrm{tr}_{\mathcal{Y}}(\rho) = \frac{1}{2}\mathbb{1}_{\mathcal{A}} \otimes \sigma \tag{5.8}$$

from (5.7). The other direction follows from the fact that for any $\rho \in \mathrm{Pos}(\mathcal{A} \otimes \mathcal{X} \otimes \mathcal{Y})$ satisfying (5.8), the operators $\rho_a = (\langle a| \otimes \mathbb{1}_{\mathcal{X}\otimes\mathcal{Y}})2\rho(|a\rangle \otimes \mathbb{1}_{\mathcal{X}\otimes\mathcal{Y}}), a \in \{0,1\}$ are in $\mathrm{Pos}(\mathcal{X} \otimes \mathcal{Y})$ and satisfy (5.7). Therefore, the following relaxation of the above SDP can be considered:

$$\begin{aligned} \text{maximize}: \quad & \langle P_\alpha^{-2}, \rho \rangle \\ \text{subject to}: \quad & \mathrm{tr}_{\mathcal{Y}}(\rho) \leq \frac{1}{2}\mathbb{1}_{\mathcal{A}} \otimes \sigma, \\ & \rho \in \mathrm{Pos}(\mathcal{A} \otimes \mathcal{X} \otimes \mathcal{Y}), \\ & \sigma \in \mathcal{D}(\mathcal{X}), \end{aligned}$$

where P_α satisfies

$$P_\alpha = \begin{pmatrix} \Pi^0_{acc} + \alpha\Pi^0_{rej} & 0 \\ 0 & \Pi^1_{acc} + \alpha\Pi^1_{rej} \end{pmatrix}.$$

The optimal value of this SDP is at least the optimal value of the original SDP but at most $1/\alpha^2$ plus that value. Now an equivalent SDP that is suitable for a parallel algorithm can be expressed in the following way:

	Primal problem		Dual problem
maximize :	$\mathrm{tr}(\rho)$	minimize :	$\frac{1}{2}\|\mathrm{tr}_{\mathcal{A}}(Q)\|$
subject to :	$\mathrm{tr}_{\mathcal{Y}}(P_\alpha \rho P_\alpha) \leq \frac{1}{2}\mathbb{1}_{\mathcal{A}} \otimes \sigma$,	subject to :	$P_\alpha(Q \otimes \mathbb{1}_{\mathcal{Y}})P_\alpha \geq \mathbb{1}_{\mathcal{A}\otimes\mathcal{X}\otimes\mathcal{Y}}$,
	$\mathrm{tr}(\sigma) \leq 1$,		$Q \in \mathrm{Pos}(\mathcal{A} \otimes \mathcal{X})$.
	$\rho \in \mathrm{Pos}(\mathcal{A} \otimes \mathcal{X} \otimes \mathcal{Y})$,		
	$\sigma \in \mathrm{Pos}(\mathcal{X})$.		

The primal SDP differs from the relaxation of the initial SDP in the sense that the multiplication with P_α is performed in the constraints, rather than the objective. However, the solution remains unchanged. Furthermore, the equality $\mathrm{tr}(\sigma) = 1$, which was hidden in $\sigma \in \mathcal{D}(\mathcal{X})$ before, is changed to an inequality. This does not affect the value of the maximum, because it is attained for $\mathrm{tr}(\sigma) = 1$ in any case as the first "inequality" in the primal constraints allows a wider choice of ρ And this wider choice can only increase the optimal value of the primal SDP. We used quotation marks to highlight the fact that we do not consider an inequality in the usual sense but rather a statement about the hermiticity of a matrix. Due to the extensive use of such statements, this thesis will continue with the above terminology without using quotation marks anymore.
In order to get insight into the formulation of the dual SDP we compare it to the definition of the dual SDP in Section 4.6: First notice that we could have defined the primal and dual SDP in Section 4.6 equivalently by exchanging the minimum with the maximum and reversing the inequalities, to end up with $\Psi(X) \leq B$ and $\Psi^*(Y) \geq A$. Moreover, the hermiticity preserving superoperator $\Psi(\cdot)$ corresponds to $\mathrm{tr}_{\mathcal{Y}}(P_\alpha \cdot P_\alpha)$ in the case at hand. The matrices A and B correspond to $\mathbb{1}_{\mathcal{A}\otimes\mathcal{X}\otimes\mathcal{Y}}$ and $1/2\mathbb{1}_{\mathcal{A}} \otimes \sigma$, respectively. Carefully observing the above SDP one notices that neither the inequality $\mathrm{tr}(\sigma) \leq 1$ in the primal SDP nor the supremum, which is due to the application of the matrix norm in the objective of the dual SDP, have been explained yet. But since we only need to understand that weak duality holds an extensive discussion will be skipped here.
The weak duality can be verified as follows. Once primal feasible operators $\sigma \in Pos(\mathcal{X})$, $\rho \in Pos(\mathcal{AXY})$ and a dual feasible operator $Q \in Pos(\mathcal{AX})$ exist we have

$$\begin{aligned} \mathrm{tr}(\rho) &\leq \langle P_\alpha(Q \otimes \mathbb{1}_{\mathcal{Y}})P_\alpha, \rho\rangle = \langle Q, \mathrm{tr}_{\mathcal{Y}}(P_\alpha \rho P_\alpha)\rangle \leq \frac{1}{2}\langle Q, \mathbb{1}_{\mathcal{A}} \otimes \sigma\rangle = \frac{1}{2}\langle \mathrm{tr}_{\mathcal{A}}(Q), \sigma\rangle \\ &\leq \frac{1}{2}\|\mathrm{tr}_{\mathcal{A}}(Q)\|, \end{aligned}$$

where the inequalities are due to the constraints on Q, ρ and σ, respectively. These considerations complete the SDP formulation for problems in QMAM$(1, 1/2+\epsilon)$. In the next section we will provide a discussion of the algorithm solving the above SDP up to sufficient accuracy.

5.3.2 Parallel SDP algorithm for QMAM

The parallel algorithm relies upon Kale's matrix multiplicative weight update method [Kal07]. It is finally used to prove QMAM$(1, 1/2 + \epsilon) \subseteq$ PSPACE. Since the completeness is perfect and the soundness is close to 1/2 we can consider the following promise problem: The probability of acceptance and thus the value of the SDP is not in the interval (5/8,7/8). It is either $1/2 + \epsilon + 1/\alpha^2 < 5/8$, in case of rejection, or it is exactly $1 > 7/8$, in case of acceptance. The algorithm of Figure 5.2 solves the SDP from the previous section up to the desired accuracy. Its input consists of four projection operators, $\Pi^0_{rej}, \Pi^0_{acc}, \Pi^1_{rej}, \Pi^1_{acc} \in \mathrm{Pos}(\mathcal{X} \otimes \mathcal{Y})$. These operators satisfy $\Pi^0_{acc} + \Pi^0_{rej} = \mathbb{1}_{\mathcal{X}\otimes\mathcal{Y}} = \Pi^1_{rej} + \Pi^1_{acc}$, where $\mathcal{X}$ and $\mathcal{Y}$ are complex vector spaces with dimension N.

Observing the algorithm of Figure 5.2 it is clear that $X^{(t)}$ and $Y^{(t)}$ are positive semidefinite throughout the whole process, since the exponential function preserves this property. Therefore, $\rho^{(t)}$ and $\sigma^{(t)}$ remain density operators $\forall t \in \{1, \ldots, T-1\}$. In order to prove the correctness of this algorithm, we have to keep in mind that the algorithm should accept if the optimal value of the SDP is greater than 5/8, and it should reject if the optimal value is smaller than 7/8. Since we will examine several statements regarding the semidefiniteness of certain matrices we will call these statements inequalities as mentioned in the discussion of Lemma 1, for instance.

Proof. First we assume the algorithm of Figure 5.2 accepts during some iteration t in step 2. Define $\rho' \in \mathrm{Pos}(\mathcal{A} \otimes \mathcal{X} \otimes \mathcal{Y})$ and $\sigma' \in \mathrm{Pos}(\mathcal{X})$ as

$$\rho' = \frac{\rho^{(t)}}{\gamma + 4\beta^{(t)}}, \qquad \sigma' = \frac{\gamma\sigma^{(t)} + 4\mathrm{tr}_{\mathcal{A}}[\Pi^{(t)}\mathrm{tr}_{\mathcal{Y}}(P_\alpha\rho^{(t)}P_\alpha)\Pi^{(t)}]}{\gamma + 4\beta^{(t)}}.$$

Assume (ρ', σ') is a primal feasible pair of the SDP. Then its objective value is greater than 5/8 since $\mathrm{tr}(\rho^{(t)}) = 1$ implies

$$\mathrm{tr}(\rho') = \frac{1}{\gamma + 4\beta^{(t)}} \geq \frac{1}{\gamma + 4\epsilon} = \frac{48}{67} > \frac{5}{8}.$$

Therefore, remains to prove that (ρ', σ') is a primal feasible pair. By the definition of $\Pi^{(t)}$ and the positive semidefiniteness of $\sigma^{(t)}$ the following inequalities hold:

$$\begin{aligned}\mathrm{tr}_{\mathcal{Y}}(P_\alpha\rho^{(t)}P_\alpha) - \frac{\gamma}{2}\mathbb{1}_{\mathcal{A}} \otimes \sigma^{(t)} &\leq \Pi^{(t)}\left(\mathrm{tr}_{\mathcal{Y}}(P_\alpha\rho^{(t)}P_\alpha) - \frac{\gamma}{2}\mathbb{1}_{\mathcal{A}} \otimes \sigma^{(t)}\right)\Pi^{(t)} \\ &\leq \Pi^{(t)}\mathrm{tr}_{\mathcal{Y}}(P_\alpha\rho^{(t)}P_\alpha)\Pi^{(t)}.\end{aligned}$$

Figure 5.2 Parallel SDP algorithm for QMAM$(1, 1/2 + \epsilon)$

1. Initialize:

$$\gamma = \frac{4}{3}, \qquad \epsilon = \frac{1}{64}, \qquad \delta = \frac{\epsilon}{16}, \qquad T = \left\lceil \frac{8 \log N}{\epsilon^2 \delta} \right\rceil$$

$$X^{(0)} = \mathbb{1}_{\mathcal{A}\otimes\mathcal{X}\otimes\mathcal{Y}}, \qquad \rho^{(0)} = \frac{X^{(0)}}{(2N+2)}, \qquad Y^{(0)} = \mathbb{1}_{\mathcal{X}}, \qquad \text{and} \qquad \sigma^{(0)} = \frac{Y(0)}{N}.$$

2. Repeat the following steps for each $t = 0, \ldots, T-1$.

   a) Compute the projection $\Pi^{(t)}$ onto the positive eigenspaces of

$$\mathrm{tr}_{\mathcal{Y}}(P_\alpha \rho^{(t)} P_\alpha) - \frac{\gamma}{2} \mathbb{1}_{\mathcal{A}} \otimes \sigma^{(t)}.$$

   b) If

$$\beta^{(t)} = \langle P_\alpha (\Pi^{(t)} \otimes \mathbb{1}_{\mathcal{Y}}) P_\alpha, \rho^{(t)} \rangle \leq \epsilon$$

   holds, stop and accept.

   c) Prepare the following operators for the next iteration:

$$X^{(t+1)} = \exp\left(-\epsilon\delta \sum_{j=0}^{t} P_\alpha (\Pi^{(j)} \otimes \mathbb{1}_{\mathcal{Y}}) P_\alpha / \beta^{(j)} \right),$$

$$Y^{(t+1)} = \exp\left(\epsilon\delta \sum_{j=0}^{t} \mathrm{tr}_{\mathcal{A}}(\Pi^{(j)} / \beta^{(j)}) \right),$$

$$\rho^{(t+1)} = \frac{X^{(t+1)}}{\mathrm{tr}(X^{(t+1)})} \qquad \textit{and} \qquad \sigma^{(t+1)} = \frac{Y^{(t+1)}}{\mathrm{tr}(Y^{(t+1)})}.$$

3. If acceptance did not occur in step 2, stop and reject.

Thus, we utilize Lemma 4 to find an upper bound for the right hand side of the above inequalities as

$$\Pi^{(t)} \mathrm{tr}_{\mathcal{Y}}(P_\alpha \rho^{(t)} P_\alpha) \Pi^{(t)} \leq 2\mathbb{1}_{\mathcal{A}} \otimes \mathrm{tr}_{\mathcal{A}}[\Pi^{(t)} \mathrm{tr}_{\mathcal{Y}}(P_\alpha \rho^{(t)} P_\alpha) \Pi^{(t)}]. \tag{5.9}$$

Combining these inequalities one concludes

$$\mathrm{tr}_{\mathcal{Y}}(P_\alpha \rho^{(t)} P_\alpha) \leq \frac{1}{2} \mathbb{1}_{\mathcal{A}} \otimes (\gamma \sigma^{(t)} + 4 \mathrm{tr}_{\mathcal{A}}[\Pi^{(t)} \mathrm{tr}_{\mathcal{Y}}(P_\alpha \rho^{(t)} P_\alpha) \Pi^{(t)}]). \tag{5.10}$$

Finally, the primal feasibility,

$$\mathrm{tr}_{\mathcal{Y}}(P_\alpha \rho' P_\alpha) \leq \frac{1}{2} \mathbb{1}_{\mathcal{A}} \otimes \sigma',$$

is due to (5.10) and the definitions of ρ' and σ'. Since the optimal value of the SDP is close to one the element under consideration is in the language and the algorithm accepts as desired. On the other hand we assume the algorithm rejects. Define a dual feasible operator $Q \in \mathrm{Pos}(\mathcal{A} \otimes \mathcal{X})$, whose dual objective value is provably less than 7/8:

$$Q = \frac{1+4\epsilon}{T} \sum_{t=0}^{T-1} \Pi^{(t)}/\beta^{(t)}.$$

Since Q is positive semidefinite we have to prove $P_\alpha(Q \otimes \mathbb{1}_\mathcal{Y})P_\alpha \geq \mathbb{1}_{\mathcal{A}\otimes\mathcal{X}\otimes\mathcal{Y}}$ in order to claim dual feasibility. Observe that

$$\begin{aligned} \mathrm{tr}(X^{(T)}) &= \mathrm{tr}\left(\exp\left(-\epsilon\delta \sum_{t=0}^{T-1} P_\alpha(\Pi^{(j)} \otimes \mathbb{1}_\mathcal{Y})P_\alpha/\beta^{(j)}\right)\right) \\ &\geq \exp\left(-\epsilon\delta\lambda_{min}\left(\sum_{t=0}^{T-1} P_\alpha(\Pi^{(j)} \otimes \mathbb{1}_\mathcal{Y})P_\alpha/\beta^{(j)}\right)\right), \end{aligned} \tag{5.11}$$

where $\lambda_{min}(\cdot)$ denotes the smallest eigenvalue of its argument. Furthermore, for $\beta^{(t)} > \epsilon$ it suffices to require $\alpha = 4$ to achieve

$$\left\| \frac{\delta}{\beta^{(t)}} P_\alpha(\Pi^{(j)} \otimes \mathbb{1}_\mathcal{Y})P_\alpha \right\| \leq \frac{\alpha^2\delta}{\epsilon} \leq 1.$$

The first inequality is due to the submultiplicativity of the spectral norm, and the fact, that $\Pi^{(t)}$ is a projection. Therefore, we can apply (4.4) from Lemma 1 to end up with

$$\exp\left(-\frac{\epsilon\delta}{\beta^{(t)}} P_\alpha(\Pi^{(j)} \otimes \mathbb{1}_\mathcal{Y})P_\alpha\right) \leq \mathbb{1} - \frac{\epsilon\delta\exp(-\epsilon)}{\beta^{(t)}} P_\alpha(\Pi^{(j)} \otimes \mathbb{1}_\mathcal{Y})P_\alpha,$$

and thus,

$$\begin{aligned} \mathrm{tr}(X^{(t+1)}) &= \mathrm{tr}\left(\exp\left(-\epsilon\delta \sum_{j=0}^{t} P_\alpha(\Pi^{(j)} \otimes \mathbb{1}_\mathcal{Y} P_\alpha/\beta^{(j)})\right)\right) \\ &\leq \mathrm{tr}\left(X^{(t)}\left(\mathbb{1} - \frac{\epsilon\delta\exp(-\epsilon)}{\beta^{(t)}} P_\alpha(\Pi^{(j)} \otimes \mathbb{1}_\mathcal{Y})P_\alpha\right)\right) \\ &\leq \mathrm{tr}(X^{(t)})\left(1 - \frac{\epsilon\delta\exp(-\epsilon)}{\beta^{(t)}} \langle P_\alpha(\Pi^{(j)} \otimes \mathbb{1}_\mathcal{Y})P_\alpha, \rho^{(t)}\rangle\right) \\ &= \mathrm{tr}(X^{(t)})(1 - \epsilon\delta\exp(-\epsilon)) \\ &\leq \mathrm{tr}(X^{(t)})\exp(-\epsilon\delta\exp(-\epsilon)). \end{aligned}$$

Notice that the second line is due to the Golden-Thompson inequality, namely $\mathrm{tr}(\exp(A+B)) \leq \mathrm{tr}(\exp(A)\cdot\exp(B))$. In the third line we inserted the definition of $\rho^{(t)}$ from step 2c of the algorithm at hand. Moreover, the last inequality follows from the estimation $1 + x \leq \exp(x)$, which holds for all real numbers x. Therefore, we can conclude

$$\mathrm{tr}(X^{(t)}) \leq \mathrm{tr}(X^{(0)})\exp(-T\epsilon\delta\exp(-\epsilon)) = 2N^2\exp(-T\epsilon\delta\exp(-\epsilon)).$$

by induction. Notice that such a recursion was already provided in Section 4.5 and we are actually using the MMW method here. Plugging the result at hand into (5.11) implies

$$\exp\left(-\epsilon\delta\lambda_{min}\left(\sum_{t=0}^{T-1} P_\alpha(\Pi^{(j)}\otimes \mathbb{1}_{\mathcal{Y}} P_\alpha/\beta^{(j)})\right)\right) \leq 2N^2\exp(-T\epsilon\delta\exp(-\epsilon)).$$

Since applying the logarithm to both sides provides

$$\lambda_{min}\left(\sum_{t=0}^{T-1} P_\alpha(\Pi^{(j)}\otimes \mathbb{1}_{\mathcal{Y}} P_\alpha/\beta^{(j)})\right) \geq T\exp(-\epsilon) - \frac{\log(2N^2)}{\epsilon\delta},$$

we utilize the definition of Q to finally conclude

$$\lambda_{min}(P_\alpha(Q\otimes \mathbb{1}_{\mathcal{Y}})P_\alpha) \geq (1+4\epsilon)\left(\exp(-\epsilon) - \frac{\log(2N^2)}{T\epsilon\delta}\right) \geq 1.$$

As $N \geq 2$ the last inequality is due to

$$\left(1+\frac{1}{16}\right)\left(\exp(-\epsilon) - \frac{\epsilon(\log 2 + 2\log N)}{8\log N}\right) \geq \left(\frac{17}{16}\right)\left(1-\epsilon-\frac{\epsilon}{8}-\frac{\epsilon}{4}\right) \geq \left(\frac{17}{16}\right)\left(\frac{501}{512}\right).$$

Since every eigenvalue of the matrix $P_\alpha(Q\otimes\mathbb{1}_{\mathcal{Y}})P_\alpha$ is therefore greater than 1 we conclude $P_\alpha(Q\otimes\mathbb{1}_{\mathcal{Y}})P_\alpha \geq \mathbb{1}_{\mathcal{A}\otimes\mathcal{X}\otimes\mathcal{Y}}$. Therefore, both conditions of the dual SDP are satisfied. Hence Q is dual feasible and it remains to establish an upper bound on the objective value of Q. To this end we observe

$$\mathrm{tr}(Y^{(t)}) = \mathrm{tr}\left[\exp\left(\epsilon\delta\sum_{t=0}^{T-1}\mathrm{tr}_{\mathcal{A}}(\Pi^{(t)}/\beta^{(t)})\right)\right] \geq \exp\left(\epsilon\delta\left\|\sum_{t=0}^{T-1}\mathrm{tr}_{\mathcal{A}}(\Pi^{(t)}/\beta^{(t)})\right\|\right). \quad (5.12)$$

The facts that the trace is the sum of the eigenvalues and the spectral norm can be expressed in terms of the maximum eigenvalue, imply (5.12) once we utilize the spectral decomposition of $\sum_t \mathrm{tr}_{\mathcal{A}}(\Pi^{(t)}/\beta^{(t)})$. In order to find a recursion formula for the left hand side of the above inequality observe that

$$\left\|\mathrm{tr}_{\mathcal{A}}(\Pi^{(t)}/\beta^{(t)})\right\| = \frac{\epsilon}{16\beta^{(t)}}\left(\left\|(\langle 0|\otimes\mathbb{1}_{\mathcal{X}})\Pi^{(t)}(|0\rangle\otimes\mathbb{1}_{\mathcal{X}})\right\| + \left\|(\langle 1|\otimes\mathbb{1}_{\mathcal{X}})\Pi^{(t)}(|1\rangle\otimes\mathbb{1}_{\mathcal{X}})\right\|\right) < 1,$$

is due to the fact that $\Pi^{(t)}$ is a projection and $\beta^{(t)} > \epsilon$ holds, as the algorithm rejects. This inequality allows the application of (4.3), stated in Lemma 1 to end up with

$$\exp(\epsilon\delta\cdot\mathrm{tr}_{\mathcal{A}}(\Pi^{(t)}/\beta^{(t)})) \leq \mathbb{1} + \epsilon\delta\exp(\epsilon)\mathrm{tr}_{\mathcal{A}}(\Pi^{(t)}/\beta^{(t)}).$$

Therefore, we get the following multiplicative recursion formula:

$$\begin{aligned}\mathrm{tr}(Y^{(t+1)}) &\leq \mathrm{tr}(Y^{(t)})(1+\epsilon\delta\exp(\epsilon)\langle\mathrm{tr}_{\mathcal{A}}(\Pi^{(t)}/\beta^{(t)}),\sigma^{(t)}\rangle)\\ &= \mathrm{tr}(Y^{(t)})\left(1+\frac{\epsilon\delta\exp(\epsilon)}{\beta^{(t)}}\langle\Pi^{(t)},\mathbb{1}_{\mathcal{A}}\otimes\sigma^{(t)}\rangle\right).\end{aligned} \quad (5.13)$$

The definition of $\Pi^{(t)}$ implies

$$\left\langle \Pi^{(t)}, \mathrm{tr}_{\mathcal{Y}}(P_\alpha \rho^{(t)} P_\alpha) - \frac{\gamma}{2} \mathbb{1}_{\mathcal{A}} \otimes \sigma^{(t)} \right\rangle \geq 0,$$

and equivalently

$$\langle \Pi^{(t)}, \mathbb{1}_{\mathcal{A}} \otimes \sigma^{(t)} \rangle \leq \frac{2}{\gamma} \langle \Pi^{(t)}, \mathrm{tr}_{\mathcal{Y}}(P_\alpha \rho^{(t)} P_\alpha) \rangle = \frac{2\beta^{(t)}}{\gamma}.$$

by the definition of $\beta^{(t)}$. Therefore, we obtain a constant recursion factor

$$\mathrm{tr}(Y^{(t+1)}) \leq \mathrm{tr}(Y^{(t)}) \left(1 + \frac{2\epsilon\delta \exp(\epsilon)}{\gamma}\right) \leq \mathrm{tr}(Y^{(t)}) \exp\left(\frac{2\epsilon\delta \exp(\epsilon)}{\gamma}\right),$$

and thus,

$$\mathrm{tr}(Y^{(T)}) \leq N \exp\left(\frac{2\epsilon\delta \exp(\epsilon)}{\gamma}\right).$$

Combining this inequality with (5.12), taking logarithms, and dividing by $\epsilon\delta$ we end up with

$$\left\| \sum_{t=0}^{T-1} \mathrm{tr}_{\mathcal{A}}(\Pi^{(t)}/\beta^{(t)}) \right\| \leq \frac{2T \exp(\epsilon)}{\gamma} + \frac{\log N}{\epsilon\delta}.$$

This provides an upper bound on the value of the dual SDP

$$\frac{1}{2} \left\| \mathrm{tr}_{\mathcal{A}}(Q) \right\| \leq (1 + 4\epsilon) \left(\frac{\exp(\epsilon)}{\gamma} + \frac{\log N}{2T\epsilon\delta}\right) < \frac{27}{32} < \frac{7}{8}.$$

Plugging in the values defined in step 1 of the algorithm the second inequality is due to

$$\left(\frac{\exp(\epsilon)}{\gamma} + \frac{\epsilon}{16}\right) \leq \frac{102/100}{4/3} + \frac{1}{1024} < \frac{27}{32}.$$

Since the objective value is less than 7/8 the probability of acceptance is close to 1/2. Therefore, the element under consideration is not in the language and the algorithm rejects, as desired. □

Since the correctness of the algorithm of Figure 5.2 is proven, it remains to examine its running time and space usage. Note that the input are projection operators acting on the N^2-dimensional Hilbert space $\mathcal{X} \otimes \mathcal{Y}$. The steps 1, 2b and 3 can be performed exactly in NC, since these steps only require standard matrix operations. Moreover, the calculations of $\rho^{(t+1)}, \sigma^{(t+1)}$ and $\mathrm{tr}_{\mathcal{Y}}(P_\alpha \rho^{(t)} P_\alpha) - (\gamma/2) \mathbb{1}_{\mathcal{A}} \otimes \sigma^{(t)}$ can be performed in NC, as the trace and the partial trace can be implemented efficiently in parallel according to Section 4.2. Therefore, we only have to discuss the projection onto positive eigenspaces in step 2a and the matrix exponentials in step 2c. But since we assume both calculations to be exact, postponing precision issues to the next section, those steps can be implemented in NC as well according to Section 4.5. Since the maximal number of iterations, T, is

bounded by $\mathcal{O}(\log N)$ the whole algorithm can be implemented efficiently in parallel. Due to the fact that functions in NC compose well the result is a polynomial-sized circuit of polylogarithmic depth. It is perhaps easier to understand if we focus on the alternative definition of NC, which states that NC is the class of problems computable in polylogarithmic time by polynomially many parallel processors. For example we can calculated the matrix exponentials in each iteration on a independent set of parallel processors, which communicate appropriately to get access to the intermediate results the individual set of processors needs. This completes the analysis of the algorithm of Figure 5.2 under the assumption that all calculations are exact. The next section is dedicated to precision issues, which result from dropping this assumption.

5.3.3 Precision issues

In order to find an NC implementation of the above algorithm, keep in mind NC includes all functions computable by logarithmic-space uniform Boolean circuits of polylogarithmic depth. Therefore, NC circuits are polynomial in size and represent polynomial time computations. We can assume the entries of the matrices to be rational in both their real and imaginary part, which are encoded as pairs of integers in binary notation. Consequently elementary matrix operations such as iterated sums, products, tensor products and the inverse of matrices can be computed in NC as mentioned in Section 4.2. Moreover, the class NC(Poly) describes all functions computable by polynomial-space uniform Boolean circuits of polynomial depth. The polynomial many quantum gates, Arthur uses to measure, can be represented by their explicit matrix description of length polynomial in N. Since N is exponential in the size of an element x of the language, Arthur's measurement operators, $\Pi^0_{acc}, \Pi^1_{acc}, \Pi^0_{rej}$ and Π^1_{rej}, can be computed in NC(Poly) as elementary matrix operations produce these operators.
Since functions in NC(poly) compose well, it only remains to prove that the presented parallel SDP algorithm has an NC implementation. One has to consider the two parts of the algorithm, which cannot be implemented exactly in NC: the projection onto the positive eigenspaces $\Pi^{(t)}$, and the matrix exponentials defining $X^{(t+1)}$ and $Y^{(t+1)}$. But as stated in the preliminaries in Section 4.2 matrix exponentials, spectral decompositions, and projections onto the positive eigenspaces can be approximated up to high precision in NC. Therefore, only step 2 of the algorithm has to be adjusted to deal with these precision issues. We achieve sufficient accuracy in the following way, using the notation of the original algorithm of Figure 5.2.

1. Let $\sqrt{\Lambda^{(t)}}$ be a positive semidefinite operator approximating $\Pi^{(t)}$, such that $\Lambda^{(t)} \leq \mathbb{1}$ and
$$\left\| \sqrt{\Lambda^{(t)}} - \Pi^{(t)} \right\| < \frac{\epsilon^2}{8\alpha^2 N},$$
for each $t \in \{0, \dots T-1\}$.

2. Redefine the two density operators $\rho^{(t+1)}$ and $\sigma^{(t+1)}$ to achieve

$$\left\|\rho^{(t+1)} - \frac{X^{(t+1)}}{\mathrm{tr}(X^{(t+1)})}\right\| < \frac{\epsilon^2}{2\alpha^2 N^2},$$
$$\left\|\sigma^{(t+1)} - \frac{Y^{(t+1)}}{\mathrm{tr}(Y^{(t+1)})}\right\| < \frac{\epsilon^2}{4N}$$

for all $t \in \{0, \dots T-1\}$.

3. All the other variables and constants can be stored exactly. Therefore, $\beta^{(t)}$ as defined in the previous section in step 2b can be computed exactly for all $t \in \{0, \dots, T-1\}$.

In order to prove that these accuracy issues do not affect the algorithms capability of deciding the SDP, one concludes the following estimation from the definition of $\sqrt{\Lambda^{(t)}}$

$$\begin{aligned}&\sqrt{\Lambda^{(t)}}\left(\mathrm{tr}_{\mathcal{Y}}(P_\alpha \rho^{(t)} P_\alpha) - \frac{\gamma}{2}\mathbb{1}_{\mathcal{A}} \otimes \sigma^{(t)}\right)\sqrt{\Lambda^{(t)}} \\ \geq &\Pi^{(t)}\left(\mathrm{tr}_{\mathcal{Y}}(P_\alpha \rho^{(t)} P_\alpha) - \frac{\gamma}{2}\mathbb{1}_{\mathcal{A}} \otimes \sigma^{(t)}\right)\Pi^{(t)} - \frac{\epsilon^2}{4N}\mathbb{1}_{\mathcal{A}\otimes\mathcal{X}}.\end{aligned} \tag{5.14}$$

If the algorithm accepts we find a primal feasible pair (ρ', σ') by redefining

$$\rho' = \frac{\rho^{(t)}}{\gamma + 4\beta^{(t)} + \epsilon}, \qquad \sigma' = \frac{\gamma\sigma^{(t)} + 4\mathrm{tr}_{\mathcal{A}}[\sqrt{\Lambda^{(t)}}\mathrm{tr}_{\mathcal{Y}}(P_\alpha \rho^{(t)} P_\alpha)\sqrt{\Lambda^{(t)}}] + (\epsilon^2/N)\mathbb{1}_{\mathcal{X}}}{\gamma + 4\beta^{(t)} + \epsilon}.$$

Utilizing (5.9) we conclude the following inequality analogously to (5.10):

$$\mathrm{tr}_{\mathcal{Y}}(P_\alpha \rho^{(t)} P_\alpha) \leq \frac{1}{2}\mathbb{1}_{\mathcal{A}} \otimes (\gamma\sigma^{(t)} + 4\mathrm{tr}_{\mathcal{A}}[\sqrt{\Lambda^{(t)}}\mathrm{tr}_{\mathcal{Y}}(P_\alpha \rho^{(t)} P_\alpha)\sqrt{\Lambda^{(t)}}] + (\epsilon^2/N)\mathbb{1}_{\mathcal{X}}),$$

proving primal feasibility. Therefore, we can guarantee a primal objective value of at least

$$\frac{1}{\gamma + 5\epsilon} = \frac{192}{271} > \frac{5}{8}$$

In case the algorithm rejects, we use the bound on $\left\|\rho^{(t+1)} - X^{(t+1)}/\mathrm{tr}(X^{(t+1)})\right\|$ to obtain a slightly different recursion formula

$$\mathrm{tr}(X^{(t+1)}) \leq \mathrm{tr}(X^{(t)})\mathrm{exp}(-\epsilon\delta(1-\epsilon)\mathrm{exp}(-\epsilon)),$$

for all $t \in \{0, \dots, T-1\}$. The calculations utilized to find this inequality are completely analogous to the error-free case. Therefore, we can proceed as in the error-free case to observe

$$\begin{aligned}\lambda_{min}(P_\alpha(Q \otimes \mathbb{1}_{\mathcal{Y}})P_\alpha) &\geq (1+4\epsilon)\left((1-\epsilon)\mathrm{exp}(-\epsilon) - \frac{\log(2N^2)}{T\epsilon\delta}\right) \\ &\geq \left(1 + \frac{1}{16}\right)\left((1-\epsilon)^2 - \frac{\epsilon(\log 2 + 2\log N)}{8\log N}\right) \\ &\geq \left(\frac{17}{16}\right)\left(1 - 2\epsilon + \epsilon^2 - \frac{\epsilon}{8} - \frac{\epsilon}{4}\right) \geq \left(\frac{17}{16}\right)\left(\frac{493}{512}\right) \geq 1.\end{aligned}$$

This consideration proves Q's dual feasibility.
In order to find an upper bound on the objective value of the dual SDP observe that the error of $\left\|\sigma^{(t+1)} - Y^{(t+1)}/\mathrm{tr}(Y^{(t+1)})\right\|$ and (5.12) imply the following slightly different recursion formula

$$\mathrm{tr}(Y^{(t+1)}) \leq \mathrm{tr}(Y^{(t)})\exp\left(\frac{2\epsilon\delta(1+\epsilon)\exp(\epsilon)}{\gamma}\right).$$

The calculations leading to the above formula are analogous to the calculations in the error-free case, which imply (5.13). Therefore, we conclude

$$\frac{1}{2}\left\|\mathrm{tr}_{\mathcal{A}}(Q)\right\| \leq (1+4\epsilon)\left(\frac{(1+\epsilon)\exp(\epsilon)}{\gamma} + \frac{\log(N)}{2T\epsilon\delta}\right) \leq \frac{1+8\epsilon}{\gamma} < \frac{7}{8},$$

where the second inequality is due to

$$\left(\frac{(1+\epsilon)\exp(\epsilon)}{\gamma} + \frac{\epsilon}{16}\right) \leq \frac{(65/64)(102/100)}{4/3} + \frac{1}{1024} < \frac{27}{32}.$$

This completes the error-tolerant NC implementation of the algorithm of Figure 5.2. Keep in mind that the input to this algorithm consists of projections in $Pos(\mathcal{X}\otimes\mathcal{Y})$ on N-dimensional vector spaces. Since N is exponential in the size of the input x of the decision problems in QMAM, these problems can be decided by functions in NC(poly). Due to Borodin's result, namely NC(poly) = PSPACE, the above consideration is the last piece in the proof of QIP $\subseteq$ PSPACE

5.3.4 QIP = PSPACE

Finally, consider Corollary 2 and Corollary 5, which state QIP $\subseteq$ QIP(3) and QIP(3) $\subseteq$ QMAM$(1, 1/2 + 2^{-p})$, respectively. Together with the result of the previous section, QMAM$(1, 1/2 + \epsilon) \subseteq$ PSPACE, and with PSPACE $\subseteq$ QIP, which is a simple consequence of IP = PSPACE [Sha92], we conclude QIP = PSPACE. Therefore, in interactive proofs a quantum verifier has no advantage at all over a classical verifier. Both can only solve problems in PSPACE. Nevertheless it remains remarkable that using a quantum computer decreases the number of rounds of interaction from polynomial many to three. The next chapter is concerned with quantum refereed games, a generalization of quantum interactive proofs.

6 Quantum refereed games

This chapter is mainly concerned with a recent advancement in quantum refereed games, namely the paper from Gutoski and Wu [GW11] proving DQIP = PSPACE. Initially, we provide an explanation of the original problem this thesis tried to solve, whether QRG(2) = PSPACE holds. Later on we discuss the quantum complexity class SQG and its generalization DQIP, as well as the fact that any problem in these two classes admits a semidefinite programming formulation. Afterwards an approximation algorithm for such SDPs and a discussion of the precision issues is presented. Finally, we prove DQIP = PSPACE . Compared to the proof of Gutoski and Wu, several explanations are added and some errors are corrected. Moreover, the calculations, concerning the precision, were generalized and a decision rule is formulated and proven.
The simplest version of a refereed game involves one referee and two players. First the referee posts questions to each player. Then both players respond. Afterwards the referee processes their answers and declares a winner. Since classical refereed games with private communication and two turns were known to be equivalent to PSPACE due to Uriel Feige and Joe Killian [FK97], the chances seemed promising to prove the same for quantum refereed games. Moreover, quantum computers can simulate classical ones and therefore the class RG(2) is included in QRG(2). Thus, PSPACE $\subseteq$ QRG(2) was known due to RG(2) = PSPACE [FK97]. Note the different notation: Feige and Killian used RG(private,1) instead of RG(2). This indicates one round of communication instead of two turns.
Before (6.3), representing Quantum refereed with one round of communication, was solved efficiently an even stronger statement, namely SQG = PSPACE, was proven by Gutoski and Wu [GW10]. Since part of the scientific audience does not trust Gutoski and Wu they are still trying to solve the more specific problem concerning QRG(2). Therefore, prior to the explanation of their work an explanation is given why the multiplicative weight update method, which was used to prove QIP = PSPACE, does not work in this case directly. To this end we need the definition of QRG(2).
A referee R posts two questions to the players Alice and Bob. Let $\mathcal{A}_0$ and $\mathcal{B}_0$ denote their input spaces and $\mathcal{A}_1$ and $\mathcal{B}_1$ their output spaces, respectively. Furthermore, Alice plays strategy $A \in \mathcal{S}(\mathcal{A}_0, \mathcal{A}_1)$, whereas Bob plays strategy $B \in \mathcal{S}(\mathcal{B}_0, \mathcal{B}_1)$. The formulation $S(\mathcal{B}_0, \mathcal{B}_1)$ refers to a one turn non-measuring strategy. Such a strategy includes a memory space $\mathcal{Z}$, and an admissible superoperator $\Phi : \mathcal{L}(\mathcal{B}_0) \to \mathcal{L}(\mathcal{B}_1 \otimes \mathcal{Z})$ on $\mathcal{Z}$. In general an admissible superoperator $\Phi \in \mathcal{T}(\mathcal{X})$ are trace preserving, $\mathrm{tr}(\Phi(X)) = \mathrm{tr}(X), \forall X \in \mathcal{L}(\mathcal{X})$ and completely positive, meaning $\Phi \otimes \mathbb{1}_{\mathcal{L}(\mathcal{X})}(X)$ is positive semidefinite for every $X \in Pos(\mathcal{X})$.
On the other hand the referee is a measuring co-strategy in co-$\mathcal{S}(\mathcal{X}, \mathcal{Y})$. The role of input and output spaces is exchanged, since a co-strategy is designed to interact with

a strategy, meaning the output space of the co-strategy is the input space of the strategy and vice versa. Moreover, two memory spaces $\mathcal{V}_0, \mathcal{V}_1$, an admissible superoperator $\Psi : \mathcal{L}(\mathcal{Y} \otimes \mathcal{V}_0) \to \mathcal{L}(\mathcal{V}_1)$, a measurement $\{\Pi_a, \Pi_b\}$, and a initial density operator $\rho_0 \in \mathcal{D}(\mathcal{X} \otimes \mathcal{V}_0)$ are needed to achieve a co-strategy. Actually a measuring strategy or co-strategy allows any measurement, not just a binary valued one. But we can restrict our view to such measurements, as they suffice to describe QRG(2), SQG and DQIP. For more details on these topics see [GW07].
The interaction starts as the referee sends part of his first register, $\mathcal{X} = \mathcal{A}_0 \otimes \mathcal{B}_0$ to Alice and Bob, respectively. Then he combines their individual answers to $\mathcal{Y} = \mathcal{A}_1 \otimes \mathcal{B}_1$. Such a compact description is possible as the players are not allowed to share entangled states before the game starts. This assumption is backed by the fact that the players are competing provers who try to persuade the computationally bounded referee of their cause. After the communication the referee has to decide who won, by measuring his last memory space $\mathcal{V}_1$ according to $\{\Pi_a, \Pi_b\}$. Here a indicates a victory for Alice, and b indicates a victory for Bob. Alice wins with probability $\langle A \otimes B, \Pi_a \rangle$, whereas Bob wins with probability $\langle A \otimes B, \Pi_b \rangle$. Now we can formulated a min-max theorem for this zero-sum quantum game as

$$\max_{A \in \mathcal{S}(\mathcal{A}_0, \mathcal{A}_1)} \min_{B \in \mathcal{S}(\mathcal{B}_0, \mathcal{B}_1)} \langle A \otimes B, \Pi_a \rangle = \min_{B \in \mathcal{S}(\mathcal{B}_0, \mathcal{B}_1)} \max_{A \in \mathcal{S}(\mathcal{A}_0, \mathcal{A}_1)} \langle A \otimes B, \Pi_a \rangle. \tag{6.1}$$

Since Alice and Bob can choose their strategies from compact convex sets and $\langle A \otimes B, \Pi_a \rangle$ is linear in A and B, this min-max theorem is a direct consequence of Theorem 4. Furthermore, we define two linear functions, which extend to unique superoperators:

$$\Phi_a(A) = \mathrm{tr}_{\mathcal{A}_1 \otimes \mathcal{A}_0}((A \otimes \mathbb{1}_{\mathcal{B}_1 \otimes \mathcal{B}_0})\Pi_a),$$
$$\Phi_b(A) = \mathrm{tr}_{\mathcal{A}_1 \otimes \mathcal{A}_0}((A \otimes \mathbb{1}_{\mathcal{B}_1 \otimes \mathcal{B}_0})\Pi_b).$$

We should think of $\{\Phi_a(A), \Phi_b(A)\}$ as the co-strategy, which already includes Alice's strategy. Therefore, the probability for a victory of Bob can be formulated as

$$\max\{\langle B, \Phi_b(A) \rangle : B \in \mathcal{S}(\mathcal{B}_0, \mathcal{B}_1)\}.$$

Due to the theorem with number 9 in the paper of Gutoski and Watrous [GW07] the above maximum can be formulated equivalently as

$$\min\{p \geq 0 : \Phi_b(A) \leq pQ, Q \in \text{co-}\mathcal{S}(\mathcal{B}_0, \mathcal{B}_1)\}. \tag{6.2}$$

Therefore, a language L can be decided by a two-turn quantum refereed game (QRG(2)), if there exists a polynomial time referee, such that

1. $\forall x \in L \exists A \in \mathcal{S}(\mathcal{A}_0, \mathcal{A}_1) : \langle A \otimes B, \Pi_a \rangle \geq 2/3$
2. $\forall x \notin L \exists B \in \mathcal{S}(\mathcal{B}_0, \mathcal{B}_1) : \langle A \otimes B, \Pi_b \rangle \geq 2/3$

An algorithm solving the following SDP would prove the reverse containment QRG(2) $\subseteq$ PSPACE. Consider the SDP formulation of a language in QRG(2), which is due to (6.2)

$$\begin{aligned} \text{minimize: } & \operatorname{tr}(P) \\ \text{subject to: } & \Phi_b(A) \leq Q, \\ & \operatorname{tr}_{\mathcal{A}_1}(A) = \mathbb{1}_{\mathcal{A}_0}, \\ & Q = P \otimes \mathbb{1}_{\mathcal{B}_1}, \\ & A \in \mathcal{P}(\mathcal{A}_1 \otimes \mathcal{A}_0), \\ & Q \in \mathcal{P}(\mathcal{B}_1 \otimes \mathcal{B}_0), \\ & P \in \mathcal{P}(\mathcal{B}_0). \end{aligned} \tag{6.3}$$

A proof for the transformation of the strategy constraints in (6.2) to the linear and semidefinite constraints in (6.3) can also be found in [GW07]. The fact that (6.3) is not defined on a whole space but a subset of the density operators is due to the equality conditions. Therefore, the multiplicative weight update method has to be modified. To this end Gutoski and Wu smartly used the separation of the communication and an unusual algorithm design. The class SQG was under consideration but their multiplicative weight update algorithm calls an oracle, which is solved by some easy case of the algorithm itself. Perhaps it would have taken much longer to answer the question, whether or not QRG(2) = PSPACE holds, without these ideas. Moreover, they also generalized their proof to double quantum interactive proofs, referred to by the class DQIP.

6.1 Short quantum games

Before the process of adjusting the multiplicative weight update method from Kale to prove QRG(2) = PSPACE ended, the paper [GW10] from Gutoski and Wu was published. Their results go even further, as they proved SQG = PSPACE. SQG stands for short refereed quantum games. It is trivial to observe that this class contains QRG(2) and is contained in QRG(4). Therefore, the result , SQG = PSPACE implies QRG(2) $\subseteq$ PSPACE. Combining this subset relation with RG(2) = PSPACE [FK97] leads to the equivalence of two-turn quantum refereed games with classical two-turn refereed games, namely RG(2) = QRG(2).
In short quantum games the referee privately talks to Alice first, processes her response, and afterwards asks Bob a question. Therefore, opposed to QRG(2) the referee can condition the question to Bob on Alice's answer. Finally, when Bob answered the referee decides who won. In order to formalize the notion of short quantum games we need some definitions. Let $\mathcal{A}, \mathcal{B}, \mathcal{V}$ be the Hilbert spaces corresponding to $\mathbf{A}, \mathbf{B}, \mathbf{V}$, where $\mathbf{A}, \mathbf{B}$ are the message registers of the two players, Alice (A) and Bob (B), and $\mathbf{V}$ is the referee's workspace. The name of the referee's workspace emphasizes that the referee acts as a verifier. He is computationally bounded, whereas the players are computational unbounded competing provers, trying to persuade the referee of their cause. Moreover, let $|\psi\rangle \in \mathcal{A} \otimes \mathcal{B} \otimes \mathcal{V}$ be a pure state, $U \in U(\mathcal{A} \otimes \mathcal{B} \otimes \mathcal{U})$ a unitary operator and $\{\Pi_b, \Pi_a\} \subseteq Meas(\mathcal{A} \otimes \mathcal{B} \otimes \mathcal{V})$ a binary valued measurement. The outcomes a and

b indicate the winner, Alice or Bob, respectively. The players can apply the arbitrary channels, $\Phi : L(\mathcal{A}) \to L(\mathcal{A})$ and $\Psi : L(\mathcal{B}) \to L(\mathcal{B})$, to the registers they receive.
These definitions allow the following description of a protocol for a short quantum game:

1. R initializes registers $(\mathbf{A}, \mathbf{V}, \mathbf{B})$ in the state $|\psi\rangle$.

2. R $\to$ A : Register $\mathbf{A}$, Alice applies channel Φ to register $\mathbf{A}$ and
   A $\to$ R : Register $\mathbf{A}$.

3. R applies U to register $(\mathbf{A}, \mathbf{V}, \mathbf{B})$.

4. R $\to$ B : Register$\mathbf{B}$, Bob applies channel Ψ to register $\mathbf{B}$ and
   B $\to$ R : Register $\mathbf{B}$.

5. R measures $(\mathbf{A}, \mathbf{V}, \mathbf{B})$ with respect to the binary valued measurement $\{\Pi_b, \Pi_a\}$.

Therefore, the referee R of such a game is completely specified by the triple $(|\psi\rangle, U, \Pi_b)$. Now we are able to define the complexity class SQG. Analogously, to the previous definitions of QIP and QMAM we have to introduce soundness and completeness.

Definition 4. Let $c, s : \mathbb{Z} \to [0, 1]$, then a language $L \subseteq \{0, 1\}^*$ is in $SQG(c, s)$, if there exists a polynomially bounded function $f(|x|)$ and a polynomial-time uniform referee $R_x = (|\psi\rangle, U, \Pi_b)$, who acts according to the above protocol, such that $\forall x \in \{0, 1\}^* : c(|x|) - s(|x|) \geq 1/f(|x|)$ and:

$$x \notin L \Rightarrow \text{game-value}(\mathrm{R}_x) \geq c(|x|)$$
$$x \in L \Rightarrow \text{game-value}(\mathrm{R}_x) \leq s(|x|).$$

Compared to the standard notation in probabilistic classes the role of yes- and no-instances is exchanged due to the definition of the referee, which relies upon Π_b instead of Π_a. We will follow Gutoski and Wu with this construction, since it eases the application of the MMW method. Moreover, we examine the game-value in the next section.

6.1.1 SDP formulation for SQG

If Alice, Bob and the referee act according to the above protocol the probability that Bob wins is given by

$$\Pr[\mathrm{out}_{\mathrm{R},\Phi,\Psi}(x) = b] = \langle \Psi \left(U \Phi \left(|\psi\rangle\langle\psi| \right) U^* \right), \Pi_b \rangle. \tag{6.4}$$

Remember $|\psi\rangle\langle\psi|$ was the density matrix of the initial state. First it is transformed by Alice's channel Φ. Second the referee applies U and afterwards Bob uses channel Ψ. Finally, the Hilbert-Schmidt inner product gives the probability of the measurement result being b.
Since the set of channels acting on some register is compact and convex the set of strategies available to Alice and Bob is also compact and convex. Because the above

inner product is bilinear in (Φ, Ψ) we can apply Sion's Min-Max Theorem (Theorem 4) to prove the existence of an equilibrium value

$$\begin{aligned}\text{game-value(R)} &= \min_{\Phi\in T(\mathcal{A})} \max_{\Psi\in T(\mathcal{B})} \langle \Psi\left(U\Phi\left(|\psi\rangle\langle\psi|\right)U^*\right), \Pi\rangle \\ &= \max_{\Psi\in T(\mathcal{B})} \min_{\Phi\in T(\mathcal{A})} \langle \Psi\left(U\Phi\left(|\psi\rangle\langle\psi|\right)U^*\right), \Pi\rangle.\end{aligned}$$

The minimum is taken over all channels $\Phi : L(\mathcal{A}) \to L(\mathcal{A})$ and the maximum over all channels $\Psi : L(\mathcal{B}) \to L(\mathcal{B})$. Note that the index of Π_b was dropped. The first min-max-expression relates to Bob acting first. Of course he tries to maximize the probability for his victory, (6.4), opposed to Alice who tries to minimize this value. On the other hand in the max-min-expression Alice minimizes first , before Bob maximizes his chances of winning.
Although this definition of the game-value in terms of channels is intuitive, it does not meet our needs. The MMW method cannot be applied immediately. But we find an equivalent formulation using the following equality

$$\langle \Psi\left(U\Phi\left(|\psi\rangle\langle\psi|\right)U^*\right), \Pi\rangle = \langle \Phi\left(|\psi\rangle\langle\psi|\right), U^*\Psi^*\left(\Pi\right)U\rangle,$$

implying

$$\text{game-value(R)} = \min_{\rho\in D(\mathcal{AVB})} \max_{P\in Meas(\mathcal{AVB})} \langle \rho, U^*PU\rangle = \max_{P\in Meas(\mathcal{AVB})} \min_{\rho\in D(\mathcal{AVB})} \langle \rho, U^*PU\rangle,$$

where the minimum is over all ρ, such that a channel Φ exists with $\rho = \Phi(|\psi\rangle\langle\psi|)$. The maximum is over all measurement operators P such that a channel Ψ exists with $P = \Psi(\Pi)$.
This formulation would allow the usage of the MMW method. But due to the equality conditions the minimum and the maximum are taken over strict subsets of the set of density operators. Therefore, the MMW method has to be adjusted to this task, since it was originally designed to solve equilibrium problems on the whole set of density operators. To tackle this problem we will need further definitions and a rounding theorem. But this discussion is postponed to Section 6.2.2 since it is also necessary to provide a MMW-suitable SDP formulation of double quantum interactive proofs which will be explained in the following section.

6.2 Double quantum interactive proof

As Gutoski and Wu presented a paper in 2011 [GW11], they also included a generalization to a new complexity class DQIP. The class DQIP contains all problems solvable by a double quantum interactive proof. In opposition to SQG the referee is allowed to exchange a constant number of messages with Alice and Bob. But the communication with each player is still separated by a time line. This means the referee can not talk to Bob, before he finished his conversation with Alice.
Because of the multiple rounds of interaction the formulas get more complicated, as sums

over all rounds have to be included. But Gutoski and Wu did not need many new ideas to generalize their original paper [GW10]. Moreover, they skipped even more details in the new version [GW11], especially regarding the precision issues. Thus, they only present the pure complexity theoretic core of their proof. On first sight this might seem nice, but the whole complexity of the thought process one has to follow, to understand this proof, is not clear any more. The precision issues, arguably hardest to calculate not purely because of the vast size of the formulas, are a necessary part to take care about before new complexity theoretic theorems emerge. If the author leaves to much space for interpretation, the reader. who is not deeply concerned with the topic looses the confidence in himself fully understanding the presented matters.
As their new paper goes even further than the original, it seems uncertain that it actually helps Gutoski and Wu to establish the equivalence of PSPACE and SQG or DQIP, respectively. Therefore, a different approach was chosen for this thesis. Initially, DQIP is defined and a SDP formulation, which suits the MMW method is presented. Additionally, their new algorithm, which is used to prove DQIP = PSPACE, and all precision issues are discussed in detail. Moreover, this yields a different proof for QIP = PSPACE. As stated at the beginning of this chapter most of the remaining sections are due to [GW10] and [GW11]. Besides several additional explanations a couple of mistakes were corrected and the oracle algorithm is completely stated in a compact way. Moreover, the precision issues were generalized to multiple rounds of interaction and a decision rule is formulated and proven in the last subsection.

6.2.1 Definition of DQIP

In double quantum interactive proofs the referee R holds a message register $\mathbf{M}$ and a private memory register $\mathbf{V}$, with the corresponding spaces $\mathcal{M}$ and $\mathcal{V}$, respectively. Note that $\mathbf{V}$ is used since the computationally bounded referee acts as a verifier. Furthermore, he holds a pure state $|\psi\rangle \in \mathcal{M} \otimes \mathcal{V}$, unitary operators $V_1, \ldots, V_{a+b} \in U(\mathcal{CMV})$ and a projective measurement operator $\Pi \in Meas(\mathcal{MV})$. Here a is the number of rounds of communication between Alice (A) and the referee, whereas b is the number of rounds of communication between Bob (B) and the referee. The two players hold private memory registers $\mathbf{A}$ and $\mathbf{B}$, with the corresponding spaces $\mathcal{A}$ and $\mathcal{B}$, respectively. Their actions can be described by unitary operators $A_1 \ldots A_a \in U(\mathcal{AM})$ and $B_1 \ldots B_b \in U(\mathcal{MB})$. These definitions suffice to present a protocol for double quantum interactive proofs:

1. Initially, the referee prepares register $(\mathbf{M}, \mathbf{V})$ in state $|\psi\rangle$, whereas $\mathbf{A}$ and $\mathbf{B}$ are initially in state $|0\rangle$.

2. For $i = 1 \ldots a$: R $\rightarrow$ A : Register $\mathbf{M}$ and Alice applies A_i to the register $(\mathbf{A}, \mathbf{M})$. Afterwards A $\rightarrow$ R : Register $\mathbf{M}$ and the referee applies V_i to the register $(\mathbf{M}, \mathbf{V})$

3. For $i = 1 \ldots b$: R$\rightarrow$ B : Register $\mathbf{M}$ and Bob applies B_i to the register $(\mathbf{M}, \mathbf{B})$. Afterwards B $\rightarrow$ R : Register $\mathbf{M}$ and the referee applies V_{a+i} to the register $(\mathbf{M}, \mathbf{V})$

4. The referee applies the binary valued measurement $\{\Pi, \mathbb{1}_{\mathcal{MV}} - \Pi\}$ to the register $(\mathbf{M}, \mathbf{V})$.

Analogously to Section 6.1.1 Π indicates a victory for Bob. Therefore, the probability for a victory of Bob is given by

$$\|\Pi' V'_{a+b} B'_b V'_{a+b-1} B'_b - 1 \cdots B'_1 V'_a A'_a V'_{a-1} A'_{a-1} \cdots A'_1 |\psi\rangle|0\rangle|0\rangle\|^2, \tag{6.5}$$

where $A'_i = A_i \otimes \mathbb{1}_{\mathcal{MVB}}$ for $i = 1, \ldots a$, $B'_i = \mathbb{1}_{\mathcal{AMV}} \otimes B_i$ for $i = 1, \ldots b$, $\Pi' = \mathbb{1}_{\mathcal{A}} \otimes \Pi \otimes \mathbb{1}_{\mathcal{B}}$, and $V'_i = \mathbb{1}_{\mathcal{A}} \otimes V_i \otimes \mathbb{1}_{\mathcal{B}}$ for $i = 1, \ldots a+b$. The tensored identities will be skipped throughout this thesis to keep the notation as simple as possible. Moreover, maximizing (6.5) over the actions of Bob and minimizing it over the actions of Alice defines the game value, $\lambda(\mathrm{R}_x)$, which is explained in detail in the next section. Once we choose $a = b = 1$ the above protocol is identical to the protocol for short quantum games, except the fact that the actions of the players are specified by unitary operators here instead of admissible quantum channels. But this is only a notational matter as both mappings are trace-preserving and do not affect the positive semidefiniteness of their argument. Therefore, $a = b = 1$ yields equality between (6.5) and (6.4). The above protocol enables the formal definition of the quantum complexity class DQIP:

Definition 5. Let $c, s : \mathbb{Z} \to [0,1]$, then a language $L \subseteq \{0,1\}^*$ is in DQIP(c,s), if there exists a polynomially bounded function $p(|x|)$ and a polynomial-time uniform quantum referee $\mathrm{R}_x = (|\psi\rangle, V_1, \ldots, V_{a+b}, \Pi)$, who acts according to the above protocol, such that $\forall x \in \{0,1\}^* : c(|x|) - s(|x|) \geq 1/p(|x|)$ and:

$$x \notin L \Rightarrow \lambda(\mathrm{R}_x) \geq c(|x|)$$
$$x \in L \Rightarrow \lambda(\mathrm{R}_x) \leq s(|x|).$$

Just like in the definition of SQG the role of yes- and no-instances is exchanged. The reason for this is the definition of the referee in terms of the measurement Π, which indicates a victory for Bob. We follow the suggestion of Gutoski and Wu in this case, since it facilitates the use of the MMW method.

6.2.2 SDP formulation for DQIP

In this section we examine a key differences between the SDP formulation for DQIP and the SDP formulation for QMAM from Section 5.3.1. The reason for this difference is due to the primal dual approach Jain, Ji, Upadhyay and Watrous chose, whereas Gutoski and Wu used the equilibrium value approach. Actually, QIP = PSPACE was also proven with the equilibrium value approach by Wu [Wu10]. Since Bob tries to maximize the game value (6.5) while Alice tries to minimize it, we already obtained a min-max-expression, just like in the case of short quantum games. But the problem formulation does not suit the MMW method again. At first we have to use the Min-Max Theorem for zero-sum quantum games [GW07], which follows from Theorem 4 to conclude

$$\begin{aligned}\lambda(\mathrm{R}) &= \min_{(A_1,\ldots,A_a)} \max_{(B_1,\ldots B_b)} \|\Pi V_{a+b} B_b V_{a+b-1} B_b - 1 \cdots B_1 V_a A_a V_{a-1} A_{a-1} \cdots A_1 |\psi\rangle\|^2 \\ &= \max_{(B_1,\ldots,B_b)} \min_{(A_1,\ldots A_a)} \|\Pi V_{a+b} B_b V_{a+b-1} B_b - 1 \cdots B_1 V_a A_a V_{a-1} A_{a-1} \cdots A_1 |\psi\rangle\|^2,\end{aligned}$$

where the minimum is taken over all possible memory spaces $\mathcal{A}$ and actions $A_1, \dots A_a \in U(\mathcal{AM})$ from Alice, while the maximum is taken over all possible memory spaces $\mathcal{B}$ and actions $B_1, \dots B_b \in U(\mathcal{MB})$ from Bob. Notice that the tensored identities and the zero states are already skipped here. To enable the usage of the MMW method let $\rho \in \mathcal{D}(\mathcal{MV})$ be the reduced state of register $(\mathbf{M}, \mathbf{V})$ after the final unitary A_a was applied by Alice. Furthermore, we combine V_a, the whole conversation between Bob and the referee as well as the final measurement Π to a single measurement operator $P \in Meas(\mathcal{MV})$. This allows us to rewrite the game-value in the following way

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

where

$$\mathcal{C}(\mathrm{R}) = \{\mathrm{tr}_{\mathcal{A}}(|\alpha\rangle\langle\alpha|) : |\alpha\rangle = A_a V_{a-1} A_{a-1} \cdots A_1 |\psi\rangle \text{ for some } (A_1, \dots, A_a)\} \tag{6.6}$$

$$\mathcal{P}(\mathrm{R}) = \{U^* \Pi U : U = V_{a+b} B_b V_{a+b-1} B_b - 1 \cdots B_1 V_a \text{ for some } (B_1, \dots B_b)\}. \tag{6.7}$$

In order to understand this new formulation of the game-value, observe that for fixed choices $A_1, \dots, A_a$ and $B_1, \dots, B_b$ we can reformulate (6.5) in the following way:

$$\begin{aligned} \|\Pi U|\alpha\rangle\|^2 &= \langle\alpha|U^*\Pi^*\Pi U|\alpha\rangle = \mathrm{tr}(\Pi U|\alpha\rangle\langle\alpha|U^*) = \mathrm{tr}(\Pi U \mathrm{tr}_{\mathcal{A}}(|\alpha\rangle\langle\alpha|)U^*) \\ &= \langle U\rho U^*, \Pi\rangle = \langle\rho, P\rangle, \end{aligned} \tag{6.8}$$

where α and U are defined as in (6.6) and (6.7), respectively. Note that $\mathcal{C}(\mathrm{R})$ is the set of admissible states after the communication with Alice. Moreover, the whole communication between Bob and the referee takes place in the set $\mathcal{P}(\mathrm{R})$. Since the minimum is not taken over the whole set of density operators, one has to rely upon finding nearly optimal approximations on the game value. Therefore, we have to provide a suitable relaxation of $\lambda(\mathrm{R})$ and a rounding theorem. In order to achieve this task a couple of lemmas are needed:

Lemma 14. For a referee $\mathrm{R} = (|\psi\rangle, V_1, \dots V_{a+b}, \Pi)$ of a double quantum interactive proof, a given state $\rho \in \mathcal{D}(\mathcal{MV})$ and $\mathcal{C}(\mathrm{R})$ as above in (6.6), $\rho \in \mathcal{C}(\mathrm{R})$ holds if and only if $\exists \rho_a, \dots, \rho_1 \in \mathcal{D}(\mathcal{MV})$ with $\rho_a = \rho$ and

$$\mathrm{tr}_{\mathcal{M}}(\rho_{i+1}) = \mathrm{tr}_{\mathcal{M}}(V_i \rho_i V_i^*) \qquad \text{for } i = 0, \dots, a-1,$$

where $V_0 = \mathbb{1}_{\mathcal{MV}}$ and $\rho_0 = |\psi\rangle\langle\psi|$ is used to shorten the notation.

The states $\rho_1, \dots \rho_a$ are called consistent with R if they obey the equations in Lemma 14. The concept of consistent states was initially introduced by Kitaev [Kit02]. In order to understand this lemma note that the unitary matrices $A_1, \dots, A_a$ only act upon $\mathcal{AM}$ and are therefore tensored with the identity on $\mathcal{V}$. Thus, under the partial trace ($\mathrm{tr}_{\mathcal{M}}$) only the actions of $V_1, \dots, V_a$ have to be taken under consideration. This consideration implies the following choice of ρ_i :

$$\rho_i = V_i A_i \cdots V_1 A_1 |\psi\rangle\langle\psi| A_1^* V_1^* \cdots A_i^* V_i^*.$$

for $i \in \{1, \ldots, a-1\}$. Obviously, these density operators fulfill the consistency condition. On the other hand we assume the consistency equations hold on the whole space $\mathcal{MV}$ not just on $\mathcal{V}$. Then we would have

$$\rho_a = V_{a-1} \cdots V_1 \rho_0 V_1^* \cdots V_{a-1}^*$$

by induction. But since the consistency conditions only hold under the partial trace we have to consider purifications of the states at hand in a large enough space like $\mathcal{XMV}$, with $\dim(\mathcal{X}) \geq \dim(\mathcal{MV})$ for example. Due to the unitary equivalence of purifications there exist unitary operators, which map the purifications of $\mathrm{tr}_{\mathcal{M}}(\rho_i)$ to purifications of ρ_i. This works since $\mathrm{tr}_{\mathcal{M}}(\rho_i)$ is the reduced state of ρ_i. A formal proof of this statement can be found in a paper of Gutoski [Gut05]. But note that his proof is different from the one presented here.
Moreover, we have to examine the following lemma about the parallel computability of consistent states:

Lemma 15. For any referee $\mathrm{R} = (|\psi\rangle, V_1, \ldots V_{a+b}, \Pi)$ and any $\rho_a, \ldots, \rho_1 \in \mathcal{D}(\mathcal{MV})$ $\exists \rho_1^\dagger, \ldots, \rho_a^\dagger \in \mathcal{D}(\mathcal{MV})$ consistent with R, such that

$$\angle(\rho_a, \rho_a^\dagger) \leq \sum_{i=0}^{a-1} \angle\left(\mathrm{tr}_{\mathcal{M}}(\rho_{i+1}), \mathrm{tr}_{\mathcal{M}}\left(V_i \rho_i V_i^*\right)\right) \tag{6.9}$$

and $\rho_1^\dagger, \ldots, \rho_a^\dagger$ can be computed in parallel time $O(a\mathrm{polylog}(\dim(\mathcal{MV})))$. Furthermore, for any $\epsilon > 0$ the bound can be reformulated in terms of the trace norm

$$\frac{1}{2}\|\rho_a - \rho_a^\dagger\|_{\mathrm{tr}} < \epsilon + \frac{a}{\epsilon} \sum_{i=0}^{a-1} \frac{1}{2} \|\mathrm{tr}_{\mathcal{M}}(\rho_{i+1}) - \mathrm{tr}_{\mathcal{M}}\left(V_i \rho_i V_i^*\right)\|_{\mathrm{tr}}. \tag{6.10}$$

Proof. First let $\rho_0^\dagger = \rho_0$. By Lemma 7 there exists a $\rho_{i+1}^\dagger$ for each $i = 0, \ldots a-1$ such that $\mathrm{tr}_{\mathcal{M}}(\rho_{i+1}^\dagger) = \mathrm{tr}_{\mathcal{M}}\left(V_i \rho_i^\dagger V_i^*\right)$ and

$$\begin{aligned}
\angle(\rho_{i+1}, \rho_{i+1}^\dagger) &= \angle(\mathrm{tr}_{\mathcal{M}}(\rho_{i+1}), \mathrm{tr}_{\mathcal{M}}\left(V_i \rho_i^\dagger V_i^*\right)) \\
&\leq \angle(\mathrm{tr}_{\mathcal{M}}(\rho_{i+1}), \mathrm{tr}_{\mathcal{M}}\left(V_i \rho_i V_i^*\right)) + \angle(\mathrm{tr}_{\mathcal{M}}(V_i \rho_i V_i^*), \mathrm{tr}_{\mathcal{M}}\left(V_i \rho_i^\dagger V_i^*\right)) \\
&\leq \angle(\mathrm{tr}_{\mathcal{M}}(\rho_{i+1}), \mathrm{tr}_{\mathcal{M}}\left(V_i \rho_i V_i^*\right)) + \angle(\rho_i, \rho_i^\dagger)
\end{aligned}$$

The equality follows from the preservation of the fidelity. The first inequality is just the triangle inequality all metrics obey while the second inequality is due to the contractiveness of the Bures angle for any quantum channel. Since $\angle(\rho_0, \rho_0^\dagger) = 0$ holds trivially the first part of the lemma follows inductively from the non-negativity of the Bures angle. Moreover, the parallel implementability of $(\rho_1^\dagger, \ldots, \rho_a^\dagger)$ follows from the implementation of Lemma 7. In order to prove (6.10) observe the following consequence of Lemma 8 and (6.9)

$$\frac{1}{2}\|\rho_a - \rho_a^\dagger\|_{\mathrm{tr}} \leq \sum_{i=0}^{a-1} \sqrt{\frac{\pi}{2} \|\mathrm{tr}_{\mathcal{M}}(\rho_{i+1}) - \mathrm{tr}_{\mathcal{M}}\left(V_i \rho_i V_i^*\right)\|_{\mathrm{tr}}}.$$

Using the abbreviation $\alpha_i = \|\mathrm{tr}_{\mathcal{M}}(\rho_{i+1}) - \mathrm{tr}_{\mathcal{M}}\left(V_i\rho_i V_i^*\right)\|_{\mathrm{tr}}$ and the fact that $\sqrt{(\pi x)/2} < x/(2\delta) + \delta$ holds for all $x \geq 0$ and all $\delta > 0$ one gets

$$\sum_{i=0}^{a-1}\sqrt{\frac{\pi\alpha_i}{2}} < \sum_{i=0}^{a-1}\left(\frac{\alpha_i}{2\delta} + \delta\right) = a\delta + \frac{1}{\delta}\sum_{i=0}^{a-1}\frac{\alpha_i}{2}.$$

Therefore, choosing $\delta = \epsilon/a$ completes the proof of (6.10). □

Lemma 14 allows the following reformulation for the game value

$$\lambda(\mathrm{R}) = \min_{\substack{(\rho_a,\ldots,\rho_1)\\ \text{consistent with R}}} \max_{P\in\mathcal{P}(\mathrm{R})} \langle\rho_a, P\rangle. \tag{6.11}$$

Moreover, Lemma 15 enables a suitable relaxation $\mu_\epsilon(\mathrm{R})$ of the game value $\lambda(\mathrm{R})$

$$\begin{aligned}\mu_\epsilon(\mathrm{R}) &= \min_{(\rho_a,\ldots,\rho_1)} \max_{\substack{P\in\mathcal{P}(\mathrm{R})\\ (\Pi_1,\ldots,\Pi_a)}} \langle\rho_a, P\rangle + \frac{a}{\epsilon}\sum_{i=0}^{a-1}\langle \mathrm{tr}_{\mathcal{M}}(\rho_{i+1}) - \mathrm{tr}_{\mathcal{M}}\left(V_i\rho_i V_i^*\right), \Pi_{i+1}\rangle\\ &= \min_{(\rho_a,\ldots,\rho_1)} \max_{P\in\mathcal{P}(\mathrm{R})} \langle\rho_a, P\rangle + \frac{a}{\epsilon}\sum_{i=0}^{a-1}\frac{1}{2}\|\mathrm{tr}_{\mathcal{M}}(\rho_{i+1}) - \mathrm{tr}_{\mathcal{M}}\left(V_i\rho_i V_i^*\right)\|_{\mathrm{tr}}.\end{aligned}$$

The minimization over the density operators $\rho_a, \ldots, \rho_1 \in \mathcal{D}(\mathcal{M}\mathcal{V})$ is adjusted as not consistent operators are penalized by the terms in the summation. Furthermore, the factor a/ϵ increases this penalty and will finally guarantee the approximation of the game value

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

In the second equality the maximum over all $P \in \mathcal{P}(\mathrm{R})$ and over all measurement operators $\Pi_1, \ldots \Pi_a \in Meas(\mathcal{V})$ is reformulated using the equality

$$\frac{1}{2}\|\rho - \sigma\|_{tr} = \max_{0\leq\Pi\leq\mathbb{1}_{\mathcal{V}}}\langle\rho - \sigma, \Pi\rangle, \tag{6.12}$$

which holds for all density operators ρ, σ. For a proof of (6.12) let UDU^* be the diagonalization of $\rho - \sigma$. Then we can split up D into a diagonal matrix D^+, whose entries are the positive entries of D and a diagonal matrix D^- whose entries are the absolute values of the negative entries of D. Using the abbreviations, $\tau^+ = UD^+U^*$ and $\tau^- = UD^-U^*$ we can define the projectors Π^+ and Π^- onto the eigenspaces corresponding to τ^+ and τ^-, respectively. In terms of formulas this means

$$\Pi^+(\rho - \sigma)\Pi^+ = \tau^+ \quad \text{and} \quad \Pi^-(\rho - \sigma)\Pi^- = \tau^-.$$

Since τ^+ and τ^- have orthogonal supports we can conclude $|\rho - \sigma| = \mathrm{tr}(|\tau^+ - \tau^-|) = \tau^+ + \tau^-$. Therefore, we have

$$\begin{aligned}&\|\rho - \sigma\|_{\mathrm{tr}} = \mathrm{tr}(|\rho - \sigma|) = \mathrm{tr}(\tau^+ + \tau^-) = \mathrm{tr}(\tau^+) + \mathrm{tr}(\tau^-) \quad \text{and}\\ &\mathrm{tr}(\tau^+) + \mathrm{tr}(\tau^-) = \mathrm{tr}(\tau^+ - \tau^-) = \mathrm{tr}(\rho - \sigma) = \mathrm{tr}(\rho) - \mathrm{tr}(\sigma) = 0,\end{aligned}$$

implying $\mathrm{tr}(\tau^+) = \mathrm{tr}(\tau^-)$ and thus, $\|\rho - \sigma\|_{\mathrm{tr}} = \mathrm{tr}(\tau^+)$. Finally, we conclude that Π^+ achieves the maximum in (6.12) from

$$2\mathrm{tr}(\Pi^+(\rho-\sigma)) = 2\mathrm{tr}(\Pi(\tau^+ - \tau^-)) = \mathrm{tr}(\tau^+) = \|\rho-\sigma\|_{\mathrm{tr}} \quad \text{and}$$

$$2\max_{0\leq\Pi\leq\mathbb{1}_{\mathcal{V}}} \langle\rho-\sigma,\Pi\rangle = 2\max_{0\leq\Pi\leq\mathbb{1}_{\mathcal{V}}} \mathrm{tr}(\Pi(\tau^+-\tau^-)) \leq 2\max_{0\leq\Pi\leq\mathbb{1}_{\mathcal{V}}} \mathrm{tr}(\Pi(\tau^+)) \leq \mathrm{tr}(\tau^+) = \|\rho-\sigma\|_{\mathrm{tr}}.$$

Moreover, we will need some new terminology to formulate the rounding theorem concisely. For any equilibrium value λ satisfying a min-max theorem for some function $f(x,y)$, with $x \in \mathcal{X}$ and $y \in \mathcal{Y}$, a pair (x', y') is called δ-optimal for λ if

$$\max_{y\in\mathcal{Y}} f(x',y) \leq \lambda + \delta \qquad \text{and} \qquad \min_{x\in\mathcal{X}} f(x,y') \leq \lambda - \delta. \tag{6.13}$$

Similarly, any value λ' is δ-optimal for λ if $|\lambda' - \lambda| \leq \delta$ holds. In order to provide a intuitive notation δ-optimal elements are marked with prime. On the other hand optimal elements for λ are in general marked with the index λ displayed on top. For example if $\delta = 0$ the above x', which is optimal for λ in this case, is referred to as x^λ. Moreover, we generalize Gutoski's and Wu's definition to the single elements of the pair, namely x' and y'. We also call them δ-optimal for λ, once they satisfy the associated inequality in (6.13). Now we are able to formulate the rounding theorem in a concise fashion:

Theorem 12. Using the notation above the following statements hold for any referee R and any $\delta, \epsilon > 0$

1. $\lambda(\mathrm{R}) \geq \mu_\epsilon(\mathrm{R}) \geq \lambda(\mathrm{R}) - \epsilon$.
2. If $(P^\mu, \Pi_1^\mu, \ldots, \Pi_a^\mu)$ is δ-optimal for $\mu_\epsilon(\mathrm{R})$ then $P^\lambda \in \mathcal{P}(\mathrm{R})$ is also $(\delta+\epsilon)$-optimal for $\lambda(\mathrm{R})$.
3. If $(\rho_1^\mu, \ldots, \rho_a^\mu)$ is δ-optimal for $\mu_\epsilon(\mathrm{R})$ then $\exists(\rho_1^\lambda)^\dagger, \ldots (\rho_a^\lambda)^\dagger$ consistent with R, such that $(\rho_a^\lambda)^\dagger$ is $(\delta+\epsilon)$-optimal for $\lambda(\mathrm{R})$. Moreover $(\rho_a')^\dagger$ is computable in parallel in time $O(a \text{ polylog}(\dim(\mathcal{MV})))$

Proof. In order to prove the first inequality of item 1 let $(\rho_1^\lambda, \ldots, \rho_a^\lambda)$ achieve the minimum for $\lambda(\mathrm{R})$ in (6.11). Moreover, let $(P^\mu, \Pi_1^\mu, \ldots, \Pi_a^\mu)$ achieve the maximum for $\mu_\epsilon(\mathrm{R})$. We have to keep in mind that this notation should not suggest an exponent, but rather another index correlating the term below to the expression it optimizes. Thus, we get

$$\lambda(\mathrm{R}) \geq \left\langle \rho_a^\lambda, P^\mu \right\rangle = \left\langle \rho_a^\lambda, P^\mu \right\rangle + \frac{a}{\epsilon}\sum_{i=0}^{a-1} \left\langle \mathrm{tr}_{\mathcal{M}}(\rho_{i+1}^\lambda) - \mathrm{tr}_{\mathcal{M}}\left(V_i\rho_i^\lambda V_i^*\right), \Pi_{i+1}^\mu \right\rangle \geq \mu_\epsilon(\mathrm{R}),$$

where the equality is due to the consistency of $(\rho_1^\lambda, \ldots, \rho_a^\lambda)$ with R. The inequality on the right side follows from the optimal choice of $(P^\mu, \Pi_1^\mu, \ldots, \Pi_a^\mu)$. Using the notation

from Lemma 15 we can prove the second inequality of item 1. For any measurement operator P we have

$$\begin{aligned}\langle \rho_a, P\rangle &= \left\langle \rho_a^\dagger, P\right\rangle + \left\langle \rho_a - \rho_a^\dagger, P\right\rangle \geq \left\langle \rho_a^\dagger, P\right\rangle - \frac{1}{2}\|\rho_a - \rho_a^\dagger\|_{\mathrm{tr}} \\ &> \langle \rho_a^\dagger, P\rangle - \epsilon - \frac{a}{\epsilon}\sum_{i=0}^{a-1}\frac{1}{2}\|\mathrm{tr}_{\mathcal{M}}(\rho_{i+1}) - \mathrm{tr}_{\mathcal{M}}\left(V_i\rho_i V_i^*\right)\|_{\mathrm{tr}}.\end{aligned} \tag{6.14}$$

The first inequality is a direct consequence of (6.12), the second inequality follows from (6.10) in Lemma 15. Moreover, let $(\rho_1^\mu, \dots \rho_a^\mu)$ be optimal for $\mu_\epsilon(\mathrm{R})$ and P^λ be optimal for $\lambda(\mathrm{R})$ to conclude

$$\mu_\epsilon(\mathrm{R}) \geq \langle \rho_a^\mu, P^\lambda\rangle + \frac{a}{\epsilon}\sum_{i=0}^{a-1}\frac{1}{2}\|\mathrm{tr}_{\mathcal{M}}(\rho_{i+1}^\mu) - \mathrm{tr}_{\mathcal{M}}\left(V_i\rho_i^\mu V_i^*\right)\|_{\mathrm{tr}} > \left\langle (\rho_a^\mu)^\dagger, P^\lambda\right\rangle - \epsilon \geq \lambda(\mathrm{R}) - \epsilon.$$

Here the first inequality is due to the minimal choice of the density operators while the last inequality is due to the maximal choice of the measurement. The strict inequality in the middle is (6.14) with the substitutions $(\rho_1, \dots \rho_a) = (\rho_1^\mu, \dots \rho_a^\mu)$ and $P = P^\lambda$. This establishes the lower bound on the relaxation of the game value $\mu_\epsilon(\mathrm{R})$ completing the proof of item 1.
Now we can utilize item 1 to prove the remaining statements of Theorem 12. For a proof of item 2 consider the following inequality, which holds for any $\rho_a, \dots, \rho_1$ consistent with R

$$\lambda(\mathrm{R}) - \epsilon - \delta < \mu_\epsilon(\mathrm{R}) - \delta \leq \left\langle \rho_a, P^\lambda\right\rangle.$$

The first inequality follows directly from the second inequality of item 1. The second inequality is due to δ-optimality of $(P', \Pi_1^\mu, \dots, \Pi_a^\mu)$.
Item 3 follows from the same construction item 1 was proven with. In order to establish the $(\delta+\epsilon)$-optimality of $(\rho_a^\lambda)^\dagger$ observe for any $P \in \mathcal{P}(\mathrm{R})$

$$\lambda(\mathrm{R}) + \delta \geq \mu_\epsilon(\mathrm{R}) + \delta \geq \langle \rho_a^\mu, P\rangle + \frac{a}{\epsilon}\sum_{i=0}^{a-1}\frac{1}{2}\|\mathrm{tr}_{\mathcal{M}}(\rho_{i+1}^\mu) - \mathrm{tr}_{\mathcal{M}}\left(V_i\rho_i^\mu V_i^*\right)\|_{\mathrm{tr}} \geq \left\langle (\rho_a^\lambda)^\dagger, P\right\rangle - \epsilon,$$

where the first inequality is due to item 1. The second inequality is due to the δ-optimality of $(\rho_1^\mu, \dots, \rho_a^\mu)$ and the not necessarily optimal choice of P. Moreover the last inequality immediately follows from (6.14) by substituting $(\rho_1^\mu, \dots, \rho_a^\mu) = (\rho_a, \dots, \rho_1)$.
The parallel implementability of $\rho_a^\dagger$ is due to the implementability of Lemma 15 and the fact that standard matrix operations like the trace or the partial trace can be computed efficiently in parallel according to Section 4.2. □

Note again that the terminology of δ-optimality was defined less general by Gutoski and Wu [GW11]. Even though they used it in the same way their definition only included optimal pairs for a min-max expression. Since each inequality in (6.13) defines the δ-optimality of one part of the optimal pair, the individual entries of the tuple are also called δ-optimal, once they satisfy the associated inequality.

6.2.3 Algorithm for δ-optimal approximation on the game-value

input : A referee $\mathrm{R} = (|\psi\rangle, V_1, \dots V_{a+b})$, an oracle algorithm $\mathcal{O}$ providing a P', such that $\langle \rho, P' \rangle \geq \langle \rho, P \rangle - \delta/2, \forall P \in \mathcal{P}$, and an accuracy parameter $\delta = \Omega\left(1/\text{polylog}\left(\dim(\mathcal{M} \otimes \mathcal{V})\right)\right)$.

output : A δ-optimal approximation λ' for $\lambda(\mathrm{R})$, a density operator $\rho_a^\lambda \in \mathcal{D}(\mathcal{MV})$, and a measurement $P^\lambda \in Meas(\mathcal{MV})$, both $(3\delta/2)$-optimal for $\lambda(\mathrm{R})$.

The condition on the error parameter δ is essentially a promise on the value of δ. Due to Theorem 12 it suffices to compute a $\delta/2$-optimal solution for $\mu_\epsilon(\mathrm{R})$ in order to achieve δ-optimal solutions for $\lambda(\mathrm{R})$, once epsilon is chosen appropriately.

On the other hand the algorithm of Figure 6.1 $(\rho_a^\lambda)^\dagger$ and P^λ are both $3\delta/2$-optimal for $\lambda(\mathrm{R})$ since $(\rho_1^\lambda \dots \rho_a^\lambda)$ and $\left(P^\lambda, \Pi_a^\lambda, \dots \Pi_1^\lambda\right)$ are δ-optimal for $\mu_\epsilon(\mathrm{R})$. Furthermore, the parameter γ of the MMW method is $\epsilon\delta/16a^2$ in this case and each loss matrix $M_i^{(t)}$ satisfies $0 \leq M_i^{(t)} \leq (1/a)\mathbb{1}_{\mathcal{MV}}$ accounting for the multiple rounds of interaction.
Moreover, the following linear map will allow a compact description of the relaxation of the game value, $\mu_\epsilon(\mathrm{R})$. For $\epsilon > 0$ consider the corrected definition

$$\begin{aligned} f_{\mathrm{R},\epsilon} : (\rho_a, \dots, \rho_1) \to \Big(\rho_a, & \frac{a}{\epsilon}\left(\mathrm{tr}_{\mathcal{M}}(\rho_a) - \mathrm{tr}_{\mathcal{M}}(V_{a-1}\rho_{a-1}V_{a-1}^*)\right), \\ & \dots, \\ & \frac{a}{\epsilon}\left(\mathrm{tr}_{\mathcal{M}}(\rho_2) - \mathrm{tr}_{\mathcal{M}}(V_2\rho_2V_2^*)\right) \\ & \frac{a}{\epsilon}\left(\mathrm{tr}_{\mathcal{M}}(\rho_1) - \mathrm{tr}(\rho_1)\mathrm{tr}_{\mathcal{M}}(|\psi\rangle\langle\psi|)\right)\Big) \end{aligned}$$

Using the definition of $f_{\mathrm{R},\epsilon} : \mathcal{L}(\mathcal{MV}) \times \dots \times \mathcal{L}(\mathcal{MV}) \to \mathcal{L}(\mathcal{MV}) \times \mathcal{L}(\mathcal{V}) \times \dots \times \mathcal{L}(\mathcal{V})$ we restate the relaxation of the game value

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

Moreover, we define the adjoint map as

$$\begin{aligned} f_{\mathrm{R},\epsilon}^* : (P, \Pi_a, \dots, \Pi_1) \to \Big(P + & \frac{a}{\epsilon}\Pi_a \otimes \mathbb{1}_{\mathcal{M}}, \frac{a}{\epsilon}\left(\Pi_{a-1} \otimes \mathbb{1}_{\mathcal{M}} - V_{a-1}^*(\Pi_a \otimes \mathbb{1}_{\mathcal{M}})V_{a-1}\right), \\ & \dots, \\ & \frac{a}{\epsilon}\left(\Pi_2 \otimes \mathbb{1}_{\mathcal{M}} - V_2^*(\Pi_3 \otimes \mathbb{1}_{\mathcal{M}})V_2\right) \\ & \frac{a}{\epsilon}\left(\Pi_1 \otimes \mathbb{1}_{\mathcal{M}} - V_1^*(\Pi_2 \otimes \mathbb{1}_{\mathcal{M}})V_1 - \langle\psi|\Pi_1|\psi\rangle\mathbb{1}_{\mathcal{MV}}\right)\Big). \end{aligned}$$

Since the inner product in a Cartesian product of Hilbert spaces is the sum of the inner products in the individual Hilbert spaces we can prove the correctness of the above

Figure 6.1 Algorithm for a δ-optimal approximation on the game value for DQIP

1. Initialize

$$\epsilon = \delta/2 \qquad T = \left\lceil \frac{2^8 a^4 \ln(\dim(\mathcal{M}\otimes\mathcal{V}))}{\epsilon^2\delta^2} \right\rceil \qquad W_i^{(1)} = \mathbb{1}_{\mathcal{M}\otimes\mathcal{V}}.$$

2. For $t = 1, \ldots, T$
   a) For $i = 1 \ldots a$ update $\rho_i^{(t)} = W_i^{(t)}/\mathrm{tr}(W_i^{(t)})$.
   b) For $i = 0, \ldots a-1$ compute the projection $\Pi_{i+1}^{(t)}$ onto the positive eigenspaces of $\mathrm{tr}_{\mathcal{M}}(\rho_{i+1}^{(t)}) - \mathrm{tr}_{\mathcal{M}}(V_i\rho_i^{(t)}V_i^*)$
   c) Call the oracle $\mathcal{O}$ on input $\rho_a^{(t)}$ and δ, to get a $\delta/2$-optimal response $P^{(t)}$ to $\rho_a^{(t)}$.
   d) Compute the loss matrices

$$\left(M_1^{(t)}, \ldots, M_a^{(t)}\right) = \frac{\epsilon}{4a^2}\left(f^{\star}_{\mathrm{R},\epsilon}(P^{(t)}, \Pi_a^{(t)}, \ldots, \Pi_1^{(t)}) + \frac{2a}{\epsilon}(\mathbb{1}_{MV}, \ldots \mathbb{1}_{\mathcal{M}\mathcal{V}})\right).$$

   e) Update the weight matrices:

$$W_i^{(t+1)} = \exp\left(-\frac{\epsilon\delta}{16a^2}\left(M_i^{(1)} + \cdots + M_i^{(t)}\right)\right).$$

3. Compute

$$\lambda' = \frac{1}{T}\sum_{t=1}^{T}\left\langle f_{\mathrm{R},\epsilon}\left(\rho_1^{(t)}, \ldots \rho_a^{(t)}\right), \left(P^{(t)}, \Pi_a^{(t)}, \ldots, \Pi_1^{(t)}\right)\right\rangle,$$

$$(\rho_1^{\lambda} \ldots \rho_a^{\lambda}) = \frac{1}{T}\sum_{t=1}^{T}\left(\rho_1^{(t)}, \ldots, \rho_a^{(t)}\right), \quad \left(P^{\lambda}, \Pi_a^{\lambda}, \ldots \Pi_1^{\lambda}\right) = \frac{1}{T}\sum_{t=1}^{T}\left(P^{(t)}, \Pi_a^{(t)}, \ldots \Pi_1^{(t)}\right).$$

4. Compute $\left((\rho_1^{\lambda})^{\dagger}, \ldots (\rho_a^{\lambda})^{\dagger}\right)$ from $(\rho_1^{\lambda} \ldots \rho_a^{\lambda})$ according to the construction in item 3 of Theorem 14.
5. Return $\left((\rho_a^{\lambda})^{\dagger}, P^{\lambda}\right)$ and λ'.

---

adjoint map straightforward. Initially, we have to use the definition of $f_{\mathrm{R},\epsilon}$ to get

$$\begin{aligned}
&\langle f_{\mathrm{R},\epsilon}(\rho_a, \ldots, \rho_1), (P, \Pi_a, \ldots, \Pi_1)\rangle = \\
&\quad \langle \rho_a, P\rangle + \left\langle \frac{a}{\epsilon}\left(\mathrm{tr}_{\mathcal{M}}(\rho_a) - \mathrm{tr}_{\mathcal{M}}\left(V_{a-1}\rho_{a-1}V_{a-1}^*\right)\right), \Pi_a\right\rangle + \cdots \\
&\qquad + \left\langle \frac{a}{\epsilon}\left(\mathrm{tr}_{\mathcal{M}}(\rho_2) - \mathrm{tr}_{\mathcal{M}}\left(V_1\rho_1V_1^*\right)\right), \Pi_2\right\rangle + \left\langle \frac{a}{\epsilon}\left(\mathrm{tr}_{\mathcal{M}}(\rho_1) - \mathrm{tr}(\rho_1)\mathrm{tr}_{\mathcal{M}}\left(|\psi\rangle\langle\psi|\right)\right), \Pi_1\right\rangle.
\end{aligned}$$

Utilizing $\langle \mathrm{tr}_{\mathcal{M}}(\rho), \Pi\rangle = \langle \rho, \Pi \otimes \mathbb{1}_{\mathcal{M}}\rangle$ and $\langle \mathrm{tr}(\rho)\mathrm{tr}_{\mathcal{M}}(|\psi\rangle\langle\psi|), \Pi\rangle = \langle \rho, \langle\psi|\Pi|\psi\rangle \mathbb{1}_{\mathcal{MV}}\rangle$, which hold for all $\rho \in \mathcal{D}(\mathcal{MV})$, $|\psi\rangle \in \mathcal{MV}$, and $\Pi \in Meas(\mathcal{V})$, we can convert the right hand side of the above equality to

$$\begin{aligned}\langle \rho_a, P\rangle + \left\langle \rho_a - V_{a-1}\rho_{a-1}V_{a-1}^*, \frac{a}{\epsilon}\left(\Pi_a \otimes \mathbb{1}_{\mathcal{M}}\right)\right\rangle + \cdots \\ + \left\langle \rho_2 - V_1\rho_1 V_1^*, \frac{a}{\epsilon}\left(\Pi_2 \otimes \mathbb{1}_{\mathcal{M}}\right)\right\rangle + \left\langle \rho_1, \frac{a}{\epsilon}\left(\Pi_1 \otimes \mathbb{1}_{\mathcal{M}}\right)\right\rangle - \langle \rho_1, \langle\psi|\Pi_1|\psi\rangle \mathbb{1}_{\mathcal{MV}}\rangle .\end{aligned} \tag{6.15}$$

Therefore, rearranging the terms in (6.15) and plugging in the definition of $f^*_{\mathrm{R},\epsilon}$ gives

$$\begin{aligned}&\left\langle \rho_a, P + \frac{a}{\epsilon}\left(\Pi_a \otimes \mathbb{1}_{\mathcal{M}}\right)\right\rangle + \left\langle \rho_{a-1}, \frac{a}{\epsilon}\left(\Pi_{a-1} \otimes \mathbb{1}_{\mathcal{M}} - V_{a-1}^*(\Pi_a \otimes \mathbb{1}_{\mathcal{M}})V_{a-1}\right)\right\rangle + \cdots \\ &\quad + \left\langle \rho_2, \frac{a}{\epsilon}\left(\Pi_2 \otimes \mathbb{1}_{\mathcal{M}} - V_2^*(\Pi_3 \otimes \mathbb{1}_{\mathcal{M}})V_2\right)\right\rangle + \\ &\qquad \left\langle \rho_1, \frac{a}{\epsilon}\left(\Pi_1 \otimes \mathbb{1}_{\mathcal{M}} - V_1^*(\Pi_2 \otimes \mathbb{1}_{\mathcal{M}})V_1 - \langle\psi|\Pi_1|\psi\rangle \mathbb{1}_{\mathcal{MV}}\right)\right\rangle \\ &\qquad\quad = \left\langle (\rho_a, \ldots, \rho_1), f^*_{\mathrm{R},\epsilon}(P, \Pi_a, \ldots, \Pi_1)\right\rangle\end{aligned}$$

The above considerations enables us to prove the efficiency and correctness of the algorithm presented in Figure 6.1.

Theorem 13. Assuming unit cost for the oracle the algorithm in Figure 6.1 finds a δ-optimal approximation on the game value. Moreover it can be implemented in parallel with run time polynomial in $(a+b)$, $1/\delta$ and $\log(\dim(\mathcal{MV}))$.

Proof. Initially, we have to check the bounds on the loss matrices. We know that multiplying any matrix A with a unitary matrix U and its adjoint only changes the basis in the representation of A. In addition with the definition of $f^*_{\mathrm{R},\epsilon}$ and the general bounds on measurement operators: $\forall \Pi \in Meas(\mathcal{MV}) : 0 \leq \Pi \leq \mathbb{1}_{\mathcal{MV}}$ this implies

$$\begin{aligned}0 \leq P + \frac{a}{\epsilon}\Pi_a \otimes \mathbb{1}_{\mathcal{M}} \leq \left(1 + \frac{a}{\epsilon}\right)\mathbb{1}_{\mathcal{MV}} \\ -\frac{a}{\epsilon}\mathbb{1}_{\mathcal{MV}} \leq \frac{a}{\epsilon}\left(\Pi_i \otimes \mathbb{1}_{\mathcal{M}} - V_i^*(\Pi_{i+1} \otimes \mathbb{1}_{\mathcal{M}})V_i\right) \leq \frac{a}{\epsilon}\mathbb{1}_{\mathcal{MV}}, \qquad \forall i \in \{2, \ldots, a-1\} \\ -2\frac{a}{\epsilon}\mathbb{1}_{\mathcal{MV}} \leq \frac{a}{\epsilon}\left(\Pi_1 \otimes \mathbb{1}_{\mathcal{M}} - V_1^*(\Pi_2 \otimes \mathbb{1}_{\mathcal{M}})V_1 - \langle\psi|\Pi_1|\psi\rangle \mathbb{1}_{\mathcal{MV}}\right) \leq \frac{a}{\epsilon}\mathbb{1}_{\mathcal{MV}}\end{aligned}$$

Combining these inequalities leads to

$$\left(0, -\frac{a}{\epsilon}\mathbb{1}, \ldots, -\frac{a}{\epsilon}\mathbb{1}, -2\frac{a}{\epsilon}\mathbb{1}\right) \leq f^*_{\mathrm{R},\epsilon}(P, \Pi_a, \ldots, \Pi_1) \leq \left(\left(1 + \frac{a}{\epsilon}\right)\mathbb{1}, \frac{a}{\epsilon}\mathbb{1}, \ldots, \frac{a}{\epsilon}\mathbb{1}\right). \tag{6.16}$$

Here and from now on the identity operator $\mathbb{1}$ always refers to the neutral element of multiplication in $\mathcal{L}(\mathcal{MV})$. Together with the definition of the loss matrices in step 2d, (6.16) implies

$$0 \leq \left(\frac{1}{2a}\mathbb{1}, \frac{1}{4a}\mathbb{1}, \ldots, \frac{1}{4a}\mathbb{1}, 0\right) \leq \frac{\epsilon}{4a^2}\left[\left(0, -\frac{a}{\epsilon}\mathbb{1}, \ldots, -\frac{a}{\epsilon}\mathbb{1}, -2\frac{a}{\epsilon}\mathbb{1}\right) + \frac{2a}{\epsilon}\left(\mathbb{1}, \ldots, \mathbb{1}\right)\right]$$

$$\leq \left(M_1^{(t)}, \ldots, M_a^{(t)}\right) \leq \frac{\epsilon}{4a^2}\left[\left(\left(1+\frac{a}{\epsilon}\right)\mathbb{1}, \frac{a}{\epsilon}\mathbb{1}, \ldots, \frac{a}{\epsilon}\mathbb{1}\right) + \frac{2a}{\epsilon}\left(\mathbb{1}, \ldots, \mathbb{1}\right)\right]$$
$$\leq \left(\frac{\epsilon+3a}{4a^2}\mathbb{1}, \frac{3}{4a}\mathbb{1}, \ldots, \frac{3}{4a}\mathbb{1}\right) \leq \frac{1}{a}\left(\mathbb{1}, \ldots, \mathbb{1}\right)$$

for all $, t \in \{1, \ldots, T\}$. Furthermore, due to the definition of $W_i^{(t+1)}$ in step 2e and the normalization in step 2a all $\rho_i^{(t)}$ satisfy the conditions of the MMW method. Therefore, applying the adjusted version of Theorem 5, namely (4.24) any density operator $\rho_i \in \mathcal{D}(\mathcal{MV})$ obeys

$$\frac{1}{T}\sum_{t=1}^{T}\left\langle \rho_i^{(t)}, M_i^{(t)}\right\rangle \leq \left\langle \rho_i, \frac{1}{T}\sum_{t=1}^{T} M_i^{(t)}\right\rangle + \frac{1}{a}\left(\frac{\epsilon\delta}{16a^2} + \frac{16a^2 \ln(\dim(\mathcal{MV}))}{\epsilon\delta T}\right). \tag{6.17}$$

Notice that the factor γ in the MMW method is $\epsilon\delta/16a^2$ in this case. For notational purposes the last term will be abbreviated by $\beta = (\epsilon\delta)/(16a^2) + 16a^2\ln(\dim(\mathcal{MV}))/(\epsilon\delta T)$. Moreover, the summation of all these inequalities gives

$$\frac{1}{T}\sum_{t=1}^{T}\left\langle \left(\rho_a^{(t)}, \ldots, \rho_1^{(t)}\right), \left(M_1^{(t)}, \ldots M_a^{(t)}\right)\right\rangle \leq \left\langle (\rho_a, \ldots, \rho_1), \frac{1}{T}\sum_{t=1}^{T}\left(M_1^{(t)}, \ldots, M_a^{(t)}\right)\right\rangle + \beta, \tag{6.18}$$

for any $(\rho_a, \ldots, \rho_1) \in (\mathcal{D}(\mathcal{MV}))^a$. Due to the definition of the loss matrices the left side of (6.18) can be reformulated as

$$\frac{\epsilon}{4a^2 T}\sum_{t=1}^{T}\left\langle \left(\rho_a^{(t)}, \ldots, \rho_1^{(t)}\right), f_{\mathrm{R},\epsilon}^*\left(P^{(t)}, \Pi_a^{(t)}, \ldots \Pi_1^{(t)}\right) + \frac{2a}{\epsilon}(\mathbb{1}_{MV}, \ldots \mathbb{1}_{\mathcal{MV}})\right\rangle$$
$$= \frac{\epsilon}{4a^2}\left(\lambda' + \frac{1}{T}\sum_{t=1}^{T}\frac{2a}{\epsilon}\sum_{i=1}^{a}\mathrm{tr}(\rho_i^{(t)})\right) = \frac{\epsilon}{4a^2}\left(\lambda' + \frac{2a^2}{\epsilon}\right).$$

The last equality is a consequence of the fact that $tr(\rho_i^{(t)}) = 1$, which holds for all $i \in \{1, \ldots, a\}, t \in \{1, \ldots, T\}$. Analogously, we can simplify the right side of (6.18):

$$\frac{\epsilon}{4a^2}\left\langle (\rho_a, \ldots, \rho_1), \frac{1}{T}\sum_{t=1}^{T} f_{\mathrm{R},\epsilon}^*\left(P^{(t)}, \Pi_a^{(t)}, \ldots \Pi_1^{(t)}\right) + \frac{2a}{\epsilon}(\mathbb{1}_{MV}, \ldots \mathbb{1}_{\mathcal{MV}})\right\rangle + \beta$$
$$= \frac{\epsilon}{4a^2}\left(\left\langle (\rho_a, \ldots, \rho_1), \frac{1}{T}\sum_{t=1}^{T} f_{\mathrm{R},\epsilon}^*\left(P^{(t)}, \Pi_a^{(t)}, \ldots, \Pi_1^{(t)}\right)\right\rangle + \frac{2a^2}{\epsilon}\right) + \beta$$

Plugging these simplifications into (6.18), subtracting the $2a^2/\epsilon$-term and multiplying by $(4a^2)/\epsilon$ leads to

$$\begin{aligned}\lambda' &= \frac{1}{T}\sum_{t=1}^{T}\left\langle \left(\rho_a^{(t)}, \ldots, \rho_1^{(t)}\right), f_{\mathrm{R},\epsilon}^*\left(P^{(t)}, \Pi_a^{(t)}, \ldots \Pi_1^{(t)}\right)\right\rangle \\ &\leq \left\langle (\rho_a, \ldots, \rho_1), \frac{1}{T}\sum_{t=1}^{T} f_{\mathrm{R},\epsilon}^*\left(P^{(t)}, \Pi_a^{(t)}, \ldots, \Pi_1^{(t)}\right)\right\rangle + \frac{4a^2\beta}{\epsilon}.\end{aligned} \tag{6.19}$$

Due to the definition of β and the initialization of T the following estimation holds:

$$\frac{4a^2\beta}{\epsilon} = \frac{4a^2}{\epsilon}\left(\frac{\epsilon\delta}{16a^2} + \frac{16a^2\,\ln(\dim(\mathcal{MV}))}{\epsilon\delta T}\right) \leq \frac{\delta}{4} + \left(\frac{4a^2}{\epsilon}\right)\frac{\epsilon\delta}{16a^2} = \frac{\delta}{2}.$$

The inequality is necessary since T is rounded up. The parameter T is actually chosen in such a unnatural way to satisfy this inequality as sharply as possible. Since (6.19) holds for an arbitrary choice of $(\rho_a, \ldots, \rho_1)$ it also holds for an optimal choice $(\rho_1^\lambda, \ldots, \rho_a^\lambda)$. Thus, the right hand side of (6.19) is bounded by $\mu_\epsilon(\mathrm{R}) + \delta/2$. Moreover, the oracle call in step 2c guarantees that $(P^{(t)}, \Pi_a^{(t)}, \ldots, \Pi_1^{(t)})$ is a $\delta/2$-best response to $(\rho_1^{(t)}, \ldots, \rho_a^{(t)})$ and therefore $\lambda' \geq \mu_\epsilon(\mathrm{R}) + \delta/2$. Combining these results with the choice of $\epsilon = \delta/2$, and item 1 of Theorem 12 leads to

$$\lambda(\mathrm{R}) - \delta < \mu_\epsilon(\mathrm{R}) - \delta/2 \leq \lambda' \leq \mu_\epsilon(\mathrm{R}) + \delta/2 \leq \lambda(\mathrm{R}) + \delta/2 < \lambda(\mathrm{R}) + \delta.$$

Finally, these inequalities establish the δ-optimality of λ' for $\lambda(\mathrm{R})$ as $|\lambda' - \lambda(\mathrm{R})| < \delta$.
Next we prove the δ-optimality of $(\rho_1^\lambda, \ldots, \rho_a^\lambda)$ for $\mu_\epsilon(\mathrm{R})$ to establish the $3\delta/2$-optimality of $(\rho_a^\lambda)^\dagger$ for $\lambda(\mathrm{R})$ according to item 3 of Theorem 12. First choose any $(P, \Pi_1, \ldots, \Pi_a)$. Since each $(P^{(t)}, \Pi_1^{(t)}, \ldots, \Pi_a^{(t)})$ is a $\delta/2$ best response to $(\rho_a^{(t)}, \ldots, \rho_1^{(t)})$

$$\left\langle \left(\rho_a^{(t)}, \ldots, \rho_1^{(t)}\right), f^*_{\mathrm{R},\epsilon}\left(P, \Pi_1, \ldots, \Pi_a\right)\right\rangle \leq \left\langle \left(\rho_a^{(t)}, \ldots, \rho_1^{(t)}\right), f^*_{\mathrm{R},\epsilon}\left(P^{(t)}, \Pi_1^{(t)}, \ldots, \Pi_a^{(t)}\right)\right\rangle + \frac{\delta}{2}$$

holds for all $t \in \{1, \ldots, T\}$. Combining the expression for λ' in (6.19), the choice of ϵ and the linearity of the trace, the summation of these inequalities implies

$$\left\langle \frac{1}{T}\sum_{t=1}^{T}\left(\rho_a^{(t)}, \ldots, \rho_1^{(t)}\right), f^*_{\mathrm{R},\epsilon}\left(P, \Pi_1, \ldots, \Pi_a\right)\right\rangle \leq \lambda' + \frac{\delta}{2} \leq \mu_\epsilon(\mathrm{R}) + \delta,$$

leading to the δ-optimality of $(\rho_a^\mu, \ldots, \rho_1^\mu)$ for $\mu_\epsilon(\mathrm{R})$.
To establish the $3\delta/2$-optimality of P^λ for $\lambda(\mathrm{R})$ it suffices to prove the δ-optimality of $(P^\mu, \Pi_a^\mu, \ldots, \Pi_1^\mu)$ for $\mu_\epsilon(\mathrm{R})$ according to item 2 of Theorem 12. Again the expression for λ' in (6.19), the choice of ϵ and the linearity of the trace imply for any $(\rho_a, \ldots, \rho_1)$

$$\left\langle (\rho_a, \ldots, \rho_1), f^*_{\mathrm{R},\epsilon}\left(P^\mu, \Pi_a^\mu, \ldots, \Pi_1^\mu\right)\right\rangle \geq \lambda' - \frac{\delta}{2} \geq \mu_\epsilon(\mathrm{R}) - \delta.$$

This proves the δ-optimality of $(P^\mu, \Pi_a^\mu, \ldots, \Pi_1^\mu)$ for $\mu_\epsilon(\mathrm{R})$ as desired.
Therefore, it only remains to provide an efficient parallel implementation of the algorithm presented in Figure 6.1. First observe that the steps 1, 2a 2d 3 and 5 can be computed exactly in NC. Moreover, the steps 2b 2c and 2e can be approximated in NC up to sufficient accuracy according to Section 4.2. Step 4 uses the construction of Lemma 7 and can therefore be implemented in NC as well. This completes the proof of Theorem 13 as all non-oracle steps can be computed by circuits in NC and these circuits compose well. This analysis still assumes that real numbers can be stored exactly postponing precision issues to Section 6.2.5. □

Since the algorithm of Figure 6.1 uses an oracle we have to present a suitable algorithm which can be implemented efficiently. The next section is dedicated to this task.

6.2.4 Oracle algorithm

In this section we will construct an oracle algorithm using the original algorithm for the special case of $\mathcal{P}(\mathrm{R})$ being a singleton. Initially, we will provide an exact Problem formulation for the oracle:

Input: A referee $\mathrm{R} = (|\psi\rangle, V_1, \ldots, V_{a+b}, \Pi)$, a density operator $\rho \in \mathcal{D}(\mathcal{M} \otimes \mathcal{V})$, and an accuracy parameter $\delta = \Omega(1/\mathrm{polylog}(\dim(\mathcal{M} \otimes \mathcal{V}))) > 0$.

Output: A measurement operator $P' \in \mathcal{P}(\mathrm{R})$, such that $\langle \rho, P' \rangle \geq \langle \rho, P \rangle - \delta \ \forall P \in \mathcal{P}(\mathrm{R})$.

In order to motivate the oracle algorithm consider the previous problem, stated at the beginning of Section 6.2.3 for the special case $b = 0$. In this case no messages are exchanged between Bob and the referee and therefore the set $\mathcal{P}(\mathrm{R})$ becomes a singleton: $\mathcal{P}(\mathrm{R}) = \{V_a^* \Pi V_a\}$. Therefore, we can simplify (6.11) as follows:

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

Remember that the condition consistent with R can be substituted with the equalities of Lemma 14.
Moreover, Theorem 13 already guarantees an efficient computable solution of this SDP, because $\mathcal{P}(\mathrm{R})$ is a singleton and therefore the oracle is trivial to implement. Since $b = 0$ holds this SDP actually represents all single-prover quantum interactive proofs. Thus, our considerations are a direct proof of QIP $\subseteq$ PSPACE. As stated before we will reduce the above oracle problem to an instance of such a SDP leading to a general polynomial space algorithm for the original problem of double quantum interactive proofs.
In order to understand the algorithm for the oracle algorithm of Figure 6.2 remember the definition of ρ in Section 6.2.2: ρ is the reduced state of the register $(\mathbf{M}, \mathbf{V})$ after the last unitary A_a was applied by Alice. We also have to recall the definitions of $\mathcal{C}(\mathrm{R}) \subseteq \mathcal{D}(\mathcal{MV})$ and $\mathcal{P}(\mathrm{R}) \subseteq Meas(\mathcal{MV})$:

$$\mathcal{C}(\mathrm{R}) = \{\mathrm{tr}_{\mathcal{A}}(|\phi\rangle\langle\phi|) : \phi = A_a V_{a-1} A_{a-1} \cdots A_1 |\psi\rangle \text{ for some } (A_1, \ldots, A_a)\}$$
$$\mathcal{P}(\mathrm{R}) = \{U^* \Pi U : U = V_{a+b} B_b V_{a+b-1} B_b - 1 \cdots B_1 V_a \text{ for some } (B_1, \ldots B_b)\}.$$

Moreover, without loss of generality the player's memory spaces $\mathcal{A}$ and $\mathcal{B}$ can be chosen large enough to admit the purifications needed since the players are computationally unbounded. Alternatively we could introduce new state spaces $\mathcal{A}'$ and $\mathcal{B}'$. But we will skip this additional notation to reduce the complexity of the formulas.

In order to extend the considerations of Theorem 13 we drop the assumption on the costs of the oracle and calculate its actual costs in the following theorem.

Theorem 14. The algorithm of Figure 6.2 computes a δ-best response P' to ρ and it can be implemented in parallel with run time polynomial in $a + b$, $1/\delta$ and $\log(\dim(\mathcal{MV}))$.

Proof. First examine an expression for the probability that Bob wins, which is due to (6.5), using the purification $|\phi\rangle$ defined in step 1 of the algorithm:

$$\left\| \Pi' V'_{a+b} B'_b V'_{a+b-1} B'_b - 1 \cdots B'_1 V'_a |\phi\rangle |0_{\mathcal{B}}\rangle \right\|^2, \tag{6.20}$$

Figure 6.2 Algorithm for the oracle in a double quantum interactive proof

1. Define a new referee $\tilde{\mathrm{R}}$:
   a) Compute a purification $|\phi\rangle \in \mathcal{AMV}$ of ρ and a new measurement operator
   $$\tilde{\Pi} = \mathbb{1}_{\mathcal{A}} \otimes (\mathbb{1}_{\mathcal{MV}} - \Pi)$$
   b) For $i = 1, \ldots, b$ compute unitary operators $\tilde{V}_i = \mathbb{1}_{\mathcal{A}} \otimes V_{a+i}$
   c) Combine the results of the steps 1a and 1b
   $$\tilde{\mathrm{R}} = (V_a|\phi\rangle, \tilde{V}_1, \ldots, \tilde{V}_b, \tilde{\Pi})$$
2. Compute $Q = \tilde{V}_b^* \tilde{\Pi} \tilde{V}_b$ and use the algorithm of Figure 6.1 to solve the following SDP
   $$\min_{\zeta \in \mathcal{C}(\tilde{\mathrm{R}})} \langle \zeta, Q \rangle .$$
   The algorithm also provides $3\delta/2$-optimal density operators $(\zeta_1, \ldots, \zeta_b)$ for $\lambda(\tilde{\mathrm{R}})$.
3. To compute purifications of $(\zeta_1, \ldots, \zeta_b)$ in $\mathcal{AMVB}$ and optimal $B_1, \ldots, B_b$, define $|\sigma_0\rangle = |\phi\rangle|0_{\mathcal{B}}\rangle$ and $\tilde{V}_0 = V_a \otimes \mathbb{1}_{\mathcal{A}}$.
4. For each i=1,. . . ,b:
   a) Compute a purification $|\sigma_i\rangle \in \mathcal{AMVB}$ of ζ_i.
   b) Compute a unitary matrix $B_i \in \mathcal{U}(\mathcal{MB})$ that maps $\tilde{V}_{i-1}|\sigma_{i-1}\rangle$ to $|\sigma_i\rangle$.
5. Compute
   $$U = (\tilde{V}_b \otimes \mathbb{1}_{\mathcal{B}})(\mathbb{1}_{\mathcal{AV}} \otimes B_b)(\tilde{V}_{b-1} \otimes \mathbb{1}_{\mathcal{B}})(\mathbb{1}_{\mathcal{AV}} \otimes B_{b-1}) \cdots (\mathbb{1}_{\mathcal{AV}} \otimes B_1)(\tilde{V}_0 \otimes \mathbb{1}_{\mathcal{B}}).$$
   and return $P' = \mathrm{tr}_{\mathcal{AB}}(U^*(\mathbb{1}_{\mathcal{A}} \otimes \Pi \otimes \mathbb{1}_{\mathcal{B}})U)$.

where $B_i' = \mathbb{1}_{\mathcal{AV}} \otimes B_i$ for $i = 1, \ldots b$, $\Pi' = \mathbb{1}_{\mathcal{A}} \otimes \Pi$, and $V_i' = \mathbb{1}_{\mathcal{A}} \otimes V_i$ for $i = 1, \ldots a+b$. To simplify the notation the tensor products will be skipped from now on and the original notation is used again. The probability for Bob's victory, (6.20), just used the matrices B_i', Π' and V_i' to provide a complete description. But carrying on the index prime would only complicate the notation even more.
We will now interpret (6.20) as the probability of a victory in a one-player game with a different referee $\tilde{\mathrm{R}}$:

$$\tilde{\mathrm{R}} = (V_a|\phi\rangle, \mathbb{1}_{\mathcal{A}} \otimes V_{a+i}, \ldots, \mathbb{1}_{\mathcal{A}} \otimes V_{a+b}, \mathbb{1}_{\mathcal{A}} \otimes (\mathbb{1}_{\mathcal{MV}} - \Pi))$$

Note that in such a one-player game the referee exchanges b messages with one player and zero messages with the other player. Therefore the unitary matrices $B_1, \ldots, B_b$

could specify either Alice or Bob. This choice depends only upon how the components of the referee are labeled.
The memory register of the the new referee is extended to $(\mathbf{V}, \mathbf{A})$, the message register remains unchanged. Nevertheless, this construction does not affect the actions the referee performs on $\mathcal{A}$ as all the unitary matrices are tensored with the identity on this space. But the measurement outcomes are exchanged as $(\mathbb{1}_{\mathcal{MV}} - \Pi)$ is used instead of Π. Step 1 is only designated to define the desired referee $\tilde{\mathrm{R}}$.
Furthermore, the operator Q, defined in step 2 of the algorithm, is the only element of $\mathcal{P}(\tilde{\mathrm{R}})$. Note that only one player is left and his communication with the referee is done in $\mathcal{C}(\tilde{\mathrm{R}})$. Therefore, no B_i remains to choose from. Moreover, we can find a measurement operator $P \in \mathcal{P}(\mathrm{R})$ and a state $\zeta \in \mathcal{C}(\tilde{\mathrm{R}})$ for any actions of Bob, such that

$$\langle \rho, P \rangle = \|\Pi V_{a+b} B_b V_{a+b-1} B_b - 1 \cdots B_1 V_a |\phi\rangle |0_{\mathcal{B}}\rangle\|^2 = 1 - \langle \zeta, Q \rangle. \tag{6.21}$$

Here the first equality is due to (6.8) and the definition of $|\phi\rangle$ as a purification of $\rho \in \mathcal{C}(\mathrm{R})$. Moreover, the new space of admissible states after the communication with Bob is

$$\mathcal{C}(\tilde{\mathrm{R}}) = \left\{ \mathrm{tr}_{\mathcal{B}}(|\alpha\rangle\langle\alpha|) : |\alpha\rangle = B_b \tilde{V}_{b-1} B_{b-1} \cdots B_1 |\tilde{\phi}\rangle \text{ for some } (B_1, \ldots, B_b) \right\},$$

with $|\tilde{\phi}\rangle = V_a |\phi\rangle$. Combining the definition of Q from step 2 of the algorithm, namely $Q = V_{a+b}^*((\mathbb{1}_{\mathcal{MV}} - \Pi) \otimes \mathbb{1}_{\mathcal{A}}) V_{a+b}$, and

$$\|\Pi U |\phi\rangle\|^2 = 1 - \|(\mathbb{1}_{\mathcal{MV}} - \Pi) U |\phi\rangle\|^2$$

proves the second equality of (6.21). The above equality is due to the fact that any partial trace within a trace has no effect at all, which was already used in (6.8). Thus, we can conclude

$$\max_{P \in \mathcal{P}(\mathrm{R})} \langle \rho, P \rangle = 1 - \lambda(\tilde{\mathrm{R}}) = 1 - \min_{\zeta \in \mathcal{C}(\tilde{\mathrm{R}})} \langle \zeta, Q \rangle. \tag{6.22}$$

In step 2 the algorithm provides an optimal $(\zeta_b, \ldots, \zeta_1)$ correspond to $((\rho_a^\lambda)^\dagger), \ldots, (\rho_1^\lambda)^\dagger$ in the algorithm of Figure 6.1. Therefore, we can utilize (6.21) and (6.22) to compute an optimal P, maximizing the left hand side of (6.22) from ζ, which is the optimal choice on the right hand side of (6.22). In order to understand this, keep in mind that the B_i, defined in step 4b of the algorithm, are an intermediate result for this calculation. To this end observe that each B_i is chosen, such that each purification $|\sigma_i\rangle$ is the state of the system after B_i and $\tilde{V}_{i-1}$ were applied. This justifies the choice of P' in step 5 of the algorithm.
Since we proved the algorithms correctness we will now consider its run time. Remember from the preliminaries, Section 4.2, that standard operations like addition, multiplication and partial trace can be performed exactly in NC. On the other hand we only assume matrix exponentials, singular value decompositions and positive eigenspace projections to be exact, handling precision issues later on in Section 6.2.5. Additionally, we already proved that the matrices establishing the unitary equivalence of purifications can be computed in NC in Theorem 1. Moreover, the following lemma will provide suitable purifications of mixed states in NC:

Lemma 16. Let $\sigma \in \mathcal{D}(\mathcal{X})$ be any density operator, and let $\sigma = \sum_{i=1}^{m} \lambda_i |i\rangle_{\mathcal{X}} \langle i|_{\mathcal{X}}$ be the spectral decomposition of σ. Furthermore, define the canonical purification of σ as $\mu = \sum_{i=1}^{m} \sqrt{\lambda_i} |i\rangle_{\mathcal{X}} \otimes |i\rangle_{\mathcal{Y}} \in \mathcal{X} \otimes \mathcal{Y}$, where $\mathcal{Y}$ is an isomorphic copy of $\mathcal{X}$. Then, $\mu\mu^* \in \mathcal{D}(\mathcal{X} \otimes \mathcal{Y})$ can be computed in NC on input σ.

Proof. According to Section 4.2 the spectral decomposition, $\sigma = UDU^*$ can be computed in NC. Let $P = UD^{1/2}$ then the canonical purification of σ is $\mu = \mathrm{vec}(P)$. Finally, $\mu\mu^*$ can be computed directly from μ in NC. □

Now, we are able to provide NC implementations of every step for the algorithm of Figure 6.2:
Obviously the steps 1b, 1c, 3 and 5 can be implemented in NC as they only use standard matrix operations. Step 2 can be implemented efficiently in parallel according to Theorem 13 as it applies the algorithm of Figure 6.1. Moreover, the purifications in step 1a and 4a can be computed by circuits in NC due to Lemma 16. Finally, Theorem 1 guarantees the existence of an efficient parallel algorithm for step 4b. □

Combining Theorem 13 and Theorem 14 with the fact that circuits in NC compose well we can conclude the following corollary:

Corollary 5. The algorithm of Figure 6.1 admits an efficient parallel algorithm with run time polynomial in $a + b$, $1/\delta$ and $\log(\dim(\mathcal{MV}))$.

This corollary will finally enable us to prove DQIP = PSPACE. But we have to discuss the accurate implementation before, dropping the assumption that all calculations are exact.

6.2.5 Precision issues

Analogously to the discussion in Section 5.3.3, we truncate the real numbers at some point to end up with a storable rational number. Since we also have to handle complex values, we store two rational numbers referring to the real and imaginary part. For a decent notation the actual values will be marked with a bar. For example $\bar{\rho}_i^{(t)}$ is the actual density operator not its idealized value $\rho_i^{(t)}$.
Initially, the main algorithm of Figure 6.1 is considered. The steps 1, 2d, 2e, 3 and 5 can be computed exactly in NC. Actually $W_i^{(t)}$ is never stored in the memory but $\rho_i^{(t)}$ is stored instead, causing the first precision issue. According to Section 4.2 the exponential function can be approximated in NC. Thus, for a rational parameter δ_1 we can find a $\bar{\rho}_i^{(t)}$, such that

$$\left\| \bar{\rho}_i^{(t)} - \frac{W_i^{(t)}}{\mathrm{tr}(W_i^{(t)})} \right\|_{tr} < \delta_1.$$

Moreover, the projection onto positive eigenspaces can be computed in NC as stated in Section 4.2. Therefore, we can find a measurement operator $\bar{\Pi}_i^{(t)}$, such that

$$\left\| \bar{\Pi}_i^{(t)} - \Pi_i^{(t)} \right\|_{\mathrm{tr}} < \delta_1.$$

Both inequalities hold for all $i \in \{1, \ldots, a\}$ and $t \in \{1, \ldots, T\}$. In order to ensure the δ-optimality of λ it will suffice to choose $\delta_1 = \Omega(1/\text{polylog}(\dim(\mathcal{MV}))$. This choice guarantees the existence of NC circuits for both steps 2a and 2b. To extend the proof of Theorem 13 to the actual values observe that no idealized values for $M_i^{(t)}$ exist. But $0 \leq M_i^{(t)} \leq (1/a)\mathbb{1}$ still holds for all i and t, due to the assumption that the oracle outputs a measurement operator and the above fact: $\bar{\Pi}_i^{(t)} \in Meas(\mathcal{MV})$ for all i and t. Therefore, only the second version of Theorem 5, (4.24), has to be adjusted:

Theorem 15. Consider the MMW algorithm of Section 4.5 and let $\bar{\rho}_i^{(t)}$ and $\bar{M}_i^{(t)}$ be the actual values as above then

$$\frac{1}{T}\sum_{t=1}^{T}\left\langle \bar{\rho}_i^{(t)}, \bar{M}_i^{(t)} \right\rangle \leq \left\langle \rho_i, \sum_{t=1}^{T} \bar{M}_i^{(t)} \right\rangle + \frac{1}{a}\left(\gamma + \frac{\ln(\dim(\mathcal{MV}))}{\gamma T} + \frac{\delta_1}{2}\right),$$

where $\gamma = \epsilon\delta/(16a^2)$ in the case at hand.

The factor $(1/2)T$ is multiplied to δ_1, since the above accuracy issues do not allow the substitution $\rho^{(t)} = W^{(t)}/\text{tr}(W^{(t)})$ in the beginning of the proof of Theorem 5. Therefore, the recursion formula changes to

$$\text{tr}\left(W_i^{(t+1)}\right) \leq \text{tr}(W_i^{(t)}) \exp\left(\text{tr}\left((e^{-\gamma}-1)\bar{M}_i^{(t)}\bar{\rho}_i^{(t)}\right)\right) \exp\left(\frac{\delta_1\gamma}{2}\right),$$

leading to a new version of (4.21):

$$\text{tr}\left(W_i^{(T+1)}\right) \leq \text{tr}\left(W_i^{(1)}\right) \exp\left((e^{-\gamma}-1)\text{tr}\left(\sum_{t=1}^{T}\bar{M}_i^{(t)}\bar{\rho}_i^{(t)}\right)\right) \exp\left(\frac{\delta_1\gamma T}{2}\right)$$

Since the other considerations in the proof of Theorem 5, which are stated in Section 4.5, do not have to be adjusted, we can formulate analogously to (4.23):

$$(1-\gamma)\sum_{t=1}^{T}\text{tr}\left(\bar{M}_i^{(t)}\bar{\rho}_i^{(t)}\right) \leq \text{tr}\left(\sum_{t=1}^{T}\bar{M}_i^{(t)}\rho_i\right) + \frac{1}{a}\left(\frac{\ln(\dim(\mathcal{MV}))}{\gamma} + \frac{\delta_1 T}{2}\right). \qquad (6.23)$$

Therefore, the steps performed in Section 4.5, leading to (4.24) completes the proof of Theorem 15. Moreover, the summation of the inequalities in (6.23) gives

$$\frac{1}{T}\sum_{t=1}^{T}\left\langle \left(\bar{\rho}_a^{(t)}, \ldots, \bar{\rho}_1^{(t)}\right), \left(\bar{M}_1^{(t)}, \ldots, \bar{M}_a^{(t)}\right)\right\rangle \leq \left\langle \rho_i, \frac{1}{T}\sum_{t=1}^{T}\left(\bar{M}_1^{(t)}, \ldots, \bar{M}_a^{(t)}\right)\right\rangle + \beta',$$

where $\beta' = \gamma + \ln(\dim(\mathcal{MV}))/(\gamma T) + \delta_1/2$. We can take this new error term into account by choosing a smaller δ' instead of the original δ such that

$$\frac{4a^2\beta'}{\epsilon'} = \frac{4a^2}{\epsilon'}\left(\frac{\epsilon'\delta'}{16a^2} + \frac{16a^2\ln(\dim(\mathcal{MV}))}{\epsilon'\delta' T'} + \frac{\delta_1}{2}\right) \leq \frac{\delta'}{2} + \frac{2a^2\delta_1}{\epsilon'} = \frac{\delta}{2},$$

still holds. Here ϵ' and T' refer to the adjustment of the constants ϵ and T from the original algorithm of Figure 6.1. In their definitions δ is just replaced by δ'. To achieve the above task it suffices to choose

$$\delta' = \delta + \frac{a^2\delta_1}{\epsilon'}.$$

Since δ and δ_1 are inverse polylogarithmic in the dimension of $\mathcal{MV}$ the same holds for δ', namely $\delta' = \Omega(1/\text{polylog}(\dim(\mathcal{MV}))$. Therefore, λ' is computable in NC and its δ-optimality is still guaranteed according to the arguments in the proof of Theorem 13. The proof of the $3\delta/2$-optimality of $((\rho_1^\lambda)^\dagger, \ldots, (\rho_a^\lambda)^\dagger)$ and (P^λ) in this situation is postponed to the following discussion of the oracle, since the output of an optimal strategy is only required for the oracle.
It remains to prove the robustness of the oracle against precision issues. The goal is to show that the oracle algorithm of Figure 5.3 runs in NC up to sufficient accuracy on input R, $\rho = \rho_a^{(t)}$ and δ', namely

$$\forall P \in \mathcal{P} \left\langle \rho, \bar{P} \right\rangle \geq \langle \rho, P \rangle - \delta'.$$

Since most of the steps of the oracle algorithm involve precision issues, the analysis is quite elaborate. Fortunately, the errors of the individual steps are additive under the trace norm. Therefore, the individual errors sum up to the whole algorithms error. In order to extend Theorem 14 to inexact computation we have to extend Theorem 1, Lemma 7 and Lemma 16. All steps, which do not need the application of these considerations, can be implemented to any desired accuracy. Therefore, these steps do not cause any problems.
At first we will consider the precision issues in Lemma 16. According to Section 4.2 the spectral decomposition of $U\bar{\rho}_a^{(t)}U^*$ can be calculated in NC, such that

$$\|U\bar{\rho}_a^{(t)}U^* - U_1 D U_1^*\|_{\text{tr}} < 2\delta'/3. \tag{6.24}$$

And even an exact purification $|\phi\rangle$ of $U_1 D U_1^*$ can be found in NC. In order to state $U_1 D U_1^*$ in terms of $|\phi\rangle$ we have to split the players workspace into the original space $\mathcal{A}$ and an additional space $\mathcal{A}'$, which is large enough to admit purifications. The error for the whole oracle increases only by $\delta'/3$ because

$$\langle U\rho U^*, P\rangle = \langle \text{tr}_{\mathcal{A}'}(|\phi\rangle\langle\phi|), P\rangle + \langle (U\rho U^* - \text{tr}_{\mathcal{A}'}(|\phi\rangle\langle\phi|), P\rangle \geq \langle \text{tr}_{\mathcal{A}'}(|\phi\rangle\langle\phi|), P\rangle - \delta'/3,$$

for any $U^*PU \in \mathcal{P}(\text{R})$. Obviously $\text{tr}_{\mathcal{A}'}(|\phi\rangle\langle\phi|) = U_1 D U_1^*$ holds and from (6.12) we can easily conclude

$$\langle \sigma - \rho, \Pi\rangle \geq -\frac{1}{2}\|\sigma - \rho\|_{\text{tr}},$$

for all measurement operators Π and density operators ρ, σ, implying the above inequality by (6.24).
The second step can be performed in NC up to sufficient accuracy according to the discussion above regarding the main algorithm of Figure 6.1, since the oracle is trivial

as $\mathcal{P}(\tilde{\mathrm{R}})$ is a singleton. Step 3 can be done in NC according to Lemma 7. Observe that the analysis of Lemma 7 is very similar to the analysis of the oracle itself as the lemma only uses constructions, which are discussed here anyways, namely the unitary operator establishing the unitary equivalence of purifications and the spectral decomposition. Therefore, the combined additive error of step 2 and step 3 can be assumed to be less than $\delta'/3$. And thus, it remains to prove that step 4 can be implemented up to an error of $\delta'/3$ in NC.
Theorem 1 states the computability of the unitary matrices establishing the equivalence of purifications of the same state in NC. In the case at hand the states $|\phi\rangle|0\rangle_{\mathcal{B}}$ and $|\sigma_i\rangle$ of step 4 are not purifications of the same state, we can only make a statement about their image under the inverse vector mapping. We will restrict our analysis to the case $i = 1$, since all other iterations for $i \in \{2, \ldots, b\}$ can be proven analogously and we are free to choose a smaller $(\delta_2)'$, which accounts for the multiple rounds. To this end we define $A = \mathrm{vec}^{-1}(\mathbb{1}_{\mathcal{A}} \otimes V_a|\phi\rangle|0\rangle_{\mathcal{B}})$ and $B = \mathrm{vec}^{-1}(|\sigma_1\rangle)$. Due to the precision issues we have

$$\|AA^* - BB^*\|_{\mathrm{tr}} < \delta_1'/3 \text{ for some } \delta_1' = \Omega(1/\mathrm{poly}(|x|),$$

instead of $AA^* = BB^*$ as in Theorem 1. The error parameter δ_1' will be chosen appropriately later on. Moreover, A and B can be computed exactly in NC. According to Section 4.2 the spectral decompositions with diagonal matrices D_1 and D_2 of A and B respectively can be found in NC up to the following accuracy

$$\|A - S_1 D_1 T_1\|_{\mathrm{tr}} < \delta_1'/6 \text{ and } \|B - S_2 D_2 T_2\|_{\mathrm{tr}} < \delta_1'/6.$$

Thus, we get

$$\|S_1 D_1 D_1^* S_1^* - S_2 D_2 D_2^* S_2^*\|_{\mathrm{tr}} < \delta_1',$$

according to the triangle inequality all norms obey. Now the Fuchs-van de Graaf inequalities imply

$$F(S_1 D_1 D_1^* S_1^*, S_2 D_2 D_2^* S_2^*) \geq 1 - \frac{1}{2}\|S_1 D_1 D_1^* S_1^* - S_2 D_2 D_2^* S_2^*\|_{\mathrm{tr}} > 1 - \frac{1}{2}\delta_1',$$

Notice that the above fidelity is equal to $\|S_1\sqrt{D_1 D_1^*}S_1^* S_2\sqrt{D_2 D_2^*}S_2^*\|_{\mathrm{tr}}$. Remember from the NC implementation of Theorem 1 that V was supposed to guarantee the positive semidefiniteness of $S_2 D_2^* S_2^* S_1 D_1 S_1^* V$. Moreover, we have to split the players workspace in its original one $\mathcal{B}$ and a additional space $\mathcal{B}'$, which is only necessary to admit the purifications. Therefore, we can conclude from the choice of U_2, namely $(U_2)^t = T_1^* S_1^* V' S_2 T_2$,

$$\begin{aligned}\|(\mathbb{1}_{\mathcal{B}'} \otimes U_2)|\varphi\rangle\langle\varphi|(\mathbb{1}_{\mathcal{B}'} \otimes (U_2)^*) - |\sigma_1\rangle\langle\sigma_1|\|_{\mathrm{tr}} &= \|\mathrm{vec}(AU_2^t)\mathrm{vec}(AU_2^t)^* - \mathrm{vec}(B)\mathrm{vec}(B)^*\|_{\mathrm{tr}} \\ &= 2\sqrt{1 - |\langle \mathrm{vec}(B), \mathrm{vec}(AU_2^t)\rangle|^2},\end{aligned}$$

where we used the abbreviation $\mathrm{vec}(A) = |\varphi\rangle$. The last equality is due to the argument presented in the proof of Lemma 6 and the fact that unitary matrices do not affect the length of a vector. Therefore, we chose U' correctly to end up with

$$F(S_1 D_1 D_1^* S_1^*, S_2 D_2 D_2^* S_2^*) = |\langle \mathrm{vec}(B), \mathrm{vec}(AU^t)\rangle|.$$

Without loss of generality we can assume D_1 and D_2 to have only real entries, leading to

$$\|(\mathbb{1}_{\mathcal{B}'} \otimes U_2)|\varphi\rangle\langle\varphi|^*(\mathbb{1}_{\mathcal{B}'} \otimes U_2^*) - |\sigma_1\rangle\langle\sigma_1|\|_{tr} \leq 2\sqrt{1-(1-\frac{1}{2}\delta_1')^2}.$$

It remains to choose a $\delta_1' \in \mathbb{R}$ such that the right hand side of the last inequality is bounded by $\delta'/3$. Therefore, the following choice will suffice our needs:

$$\delta_1' \leq 2 - \sqrt{4 - \frac{(\delta')^2}{9}}.$$

Analogous considerations hold for the unitary matrices, which map $\tilde{V}_{i-1}|\sigma_{i-1}\rangle$ to $|\sigma_i\rangle$. Moreover, the sum of these errors is polynomial in δ_1' and linear in b. Therefore this sum can be combined in a new error parameter, which is still inverse polylogarithmic in the dimension of $\mathcal{MV}$.
Thus, the algorithms of Figure 6.2 and Figure 6.1 can be implemented in NC up to sufficient accuracy. Finally, it remains to explain, why this algorithm is good enough to solve double quantum interactive poofs in polynomial space.

6.2.6 DQIP = PSPACE

In this section we finally prove the result DQIP = DIP = PSPACE [GW11], a new complexity theoretic equality, which is a generalization of the previous results, SQG = PSPACE from Gutoski and Wu [GW10]. One direction, namely PSPACE $\subseteq$ DQIP, holds due to Shamir's result, PSPACE = IP, and the trivial fact that IP $\subseteq$ DQIP. Thus, it remains to prove the reverse containment:

Theorem 16. For c, s as stated in Definition 5, DQIP(c,s) $\subseteq$ PSPACE holds.

Proof. For any decision problem L in DQIP and any instance $x \in$ L the following short algorithm can distinguish between yes-instances and no-instances up to sufficient accuracy. Analogously to the definition of DQIP, we denote the referee by R_x.

1. Compute a referee $\mathrm{R}_x = (|\psi\rangle, V_1, \ldots, V_{a+b}, \Pi)$ from x.
2. Choose $\delta = (c-s)/3$ and run the algorithm of Figure 6.1 with the oracle implementation of Figure 6.2 to get a δ-optimal approximation $\lambda'(\mathrm{R}_x)$ for the game-value $\lambda(\mathrm{R}_x)$.
3. If $|\lambda'(\mathrm{R}_x) - c| < |\lambda'(\mathrm{R}_x) - s|$, then $x \notin L$ otherwise $x \in L$.

Initially, we have to check the correctness of this algorithm. Essentially the algorithm decides if the game-value is closer to c or to s in step 3. If we would have the exact game-value only two cases could occur, $\lambda(R_x) \geq c$ or $\lambda(R_x) \leq s$. Both cases could clearly be distinguished by the decision rule in step 3. Therefore, we just have to check that the approximation of the game-value does not cause any problems. Remember from Definition 5 that the completeness and soundness of any language L in DQIP obey

$c - s \geq 1/poly(|x|)$.
First notice that if $\lambda'(\mathrm{R}_x) \geq c$ holds, then $c > s$ implies $\lambda'(\mathrm{R}_x) - c < \lambda'(\mathrm{R}_x) - s$ and the algorithm answers no, $x \notin L$. But since δ can only make up for one third of the difference between c and s this also implies $\lambda(\mathrm{R}_x) - c < \lambda(\mathrm{R}_x) - s$. Therefore, the promise $\lambda(\mathrm{R}_x) \notin (s, c)$ leads to $\lambda(\mathrm{R}_x) \geq c$ implying that x really is a no-instance.
On the other hand if $\lambda'(\mathrm{R}_x) \leq s$, then $\lambda'(\mathrm{R}_x)$ is closer to s than it is to c, meaning $c - \lambda'(\mathrm{R}_x) > s - \lambda'(\mathrm{R}_x)$. Thus, the algorithm answers yes, $x \in L$. Analogously to the above consideration, δ can only make up for one third of the difference $c - s$ implying $c - \lambda(\mathrm{R}_x) > s - \lambda(\mathrm{R}_x)$. Therefore, the promise $\lambda(\mathrm{R}_x) \notin (s, c)$ leads to $\lambda(\mathrm{R}_x) \leq s$ and therefore x is a yes-instance.
Now, it only remains to prove the algorithm's correctness for the case $s < \lambda'(\mathrm{R}_x) < c$. In this case the inequality in step 3 is $c - \lambda'(\mathrm{R}_x) < \lambda'(\mathrm{R}_x) - s$. First we assume that $x \notin L$ but the algorithm does not answer correctly, meaning $c - \lambda'(\mathrm{R}_x) > \lambda'(\mathrm{R}_x) - s$. Then $c + s > 2\lambda'(\mathrm{R}_x)$ implies

$$c + s > 2\lambda'(\mathrm{R}_x) \geq 2(\lambda(\mathrm{R}_x) - \delta) \geq 2(c - \delta) = \frac{4}{3}c + \frac{2}{3}s, \tag{6.25}$$

where the second inequality is due to the δ-optimality of $\lambda'(\mathrm{R}_x)$ and the third inequality is due to $\lambda'(\mathrm{R}_x) \geq c$, which follows from the assumption $x \notin L$. But inequality (6.22) leads to the contradiction $s > c$.
Secondly we assume that $x \in L$ but the algorithm does not answer correctly, meaning $c - \lambda'(\mathrm{R}_x) < \lambda'(\mathrm{R}_x) - s$. Then $c + s < 2\lambda'(\mathrm{R}_x)$ implies

$$c + s < 2\lambda'(\mathrm{R}_x) \leq 2(\lambda(\mathrm{R}_x) + \delta) \leq 2(s + \delta) = \frac{4}{3}s + \frac{2}{3}c, \tag{6.26}$$

where the second inequality is due to the δ-optimality of $\lambda'(\mathrm{R}_x)$ and the third inequality is due to $\lambda'(\mathrm{R}_x) \leq s$, which follows from the assumption $x \in L$. But inequality (5.35) leads to the contradiction $c < s$.
Since we have proven the algorithms correctness we can examine the run time. The first step can be implemented by simple matrix multiplications. Therefore, it can be implemented by classical parallel algorithms with run time polynomial in the dimension of the matrices, $\log(\dim(\mathcal{MV}))$. Moreover, Corollary 5 gives an analogous bound for the second step, since a and b are constants, $\delta = \Omega(1/\mathrm{polylog}(\dim(\mathcal{MV})))$ and $c - s \geq 1/\mathrm{poly}$. Notice that the input size of the NC algorithms is exponentially bigger than the size of the input x of the decision problem in DQIP and step 3 is easy to implement. Therefore, the algorithm at hand can be implemented in NC(poly) and NC(poly) = PSPACE completes the proof. □

Now the characterization of DQIP in terms of PSPACE allows an equivalent definition of DQIP with $c = 2/3$ and $s = 1/3$ or $c = 1$ and $s = \epsilon$ just like it is known for IP. Here ϵ can be chosen exponentially small since problems in PSPACE are fully robust with respect to the choice of c and s. This leads to an open question regarding DQIP: Are there direct methods to decrease the soundness error. Since the field of quantum complexity theory is still being explored we will end the last section with this question.

7 Conclusions

The major goal of this thesis is to present the advances in quantum complexity theory in a complete fashion. Therefore, this work is a self-contained detailed analysis of the two major results QIP = PSPACE and DQIP = PSPACE. Hopefully this thesis will help to establish SQG = DQIP = PSPACE as a generally accepted theorem. To this end the arguments of Gutoski and Wu are concretized and some mistakes are corrected. The different versions of both papers, [JJUW10] and [GW11], underline the uncertainty of research in this field, especially how to present results, due to the variety of mathematical tools used. Therefore, this thesis tries to establish standards in scientific research presentation of quantum complexity theory. To fully understand the presented matters on first sight one needs experience in classical computation, especially approximation algorithms on convex programs, in game theory, and in quantum computation. Since only very few people are experts in all these fields it is of high importance to explain the results in full detail.

To achieve its major goal this thesis concretized the proofs of the main part in many ways. One important example is the explicit formulation of the decision rule in Section 6.2.6 and the explanation, why it works. Gutoski and Wu just stated the existence of such a rule. In addition the major proofs QIP(3) = QIP,QMAM = QIP,QIP = PSPACE and DQIP = PSPACE are discussed in detail in Section 5.1, 5.2.2, 5.3.2, 6.2.3 and 6.2.4, where many explanations are added. Moreover, the precision issues are concretized in Section 5.3.3 and generalized in Section 6.2.5. For example proofs of (4.24) and (6.23) are provided, whereas Gutoski and Wu just stated that a generalization of the precision issues discussed in their original paper [GW10] is possible.

To emphasize the need of this thesis observe that a paper where a careful reader is not able to understand all problems and ideas needed, does not help future researchers. But one should enable young researchers to tackle the most recent and seemingly difficult problems in quantum computation. As a matter of fact it is quite common to skip details, which the reader is supposed to know or understand without a hint. However, a self-contained and readable presentation is more adequate to guarantee complete insight into complex proofs. But such a detailed presentation is hardly realizable in a short research paper. Since the number of pages of a thesis is also limited, scripts and books should be published on the discussed matters, to enable students to understand the state of research in quantum computation, as several problems in this field need further investigation.

In fact we do not even know, whether quantum computers represent just a small intermediate step in securing cryptographic systems against potential threats, or if a “quantum jump”, enabling powerful physical quantum computers, might take place. Despite the fact, that this thesis is concerned with the limits of quantum computation, new quantum

algorithms should emerge, emphasizing the need for physical quantum computers. Using Shor's algorithm one could hack every RSA-based cryptosystem offering the wrong profit perhaps. Nevertheless, for the sake of humanity, hopefully no real quantum computer is built before at least the important secrets are protected in a more sophisticated way than storing prime factorizations of large numbers. Actually, the RSA cryptosystem was invented by Clifford Cocks but published by Ron Rivest, Adi Shamir and Leonard Adleman [RSA78].
Unfortunately, up to now the most significant result on quantum computation is an algorithm that allows quantum computers to factorize faster than classical ones. This was initially proven by Shor in 1997 [Sho97]. However, the quantum analogue of the Church Turing thesis suggests, that every physically realizable computing device might essentially be a quantum computer, like every classical computer is supposedly a Turing machine. Therefore, quantum computation might be the best chance we get in developing more powerful computers. Moreover, from a theoretical point of view quantum complexity theory might even help resolve P versus NP.
The issues mentioned above emphasize the importance of the discussed problems for the real world. It is important to theoretically understand quantum computers in order to make valid decisions in the development of physical quantum computing devices. One of today's biggest problems of quantum computers, the noisy environment, might be solved by theoretical methods namely better error correcting codes. Since noise occurs when any particle interacts with a computing device, causing the application of a CNOT gate, it is difficult to find an experimental solution for this physical problem.
Therefore, one can only encourage the audience to improve the theory of quantum computation, as this might yield huge benefits for mathematical research and for the real world.

Bibliography

[AB09] Sanjeev Arora and Boaz Barak, *Computational complexity: A modern approach*, Cambridge University Press, 2009.

[AHK12] Sanjeev Arora, Elad Hazan, and Sayten Kale, *The multiplicative weights update method: A meta-algorithm and applications*, Theory of Computing **8** (2012), 121–164.

[AKN98] Dorit Aharonov, Alexei Kitaev, and Noam Nisan, *Quantum circuits with mixed states*, Proceedings of the 13th ACM Symposium on Theory of Computing, 1998, pp. 20–30.

[Bab85] Lazlo Babai, *Trading group theory for randomness*, Proceedings of the 17th ACM Symposium on Theory of Computing, ACM, 1985.

[BCF+96] Howard Barnum, Carlton M. Caves, Christopher A. Fuchs, Richard Jozsa, and Benjamin Schumacher, *Noncommuting mixed states cannot be broadcast*, Physical ReviewLetters Physical Review Letters **76** (1996), no. 15, 2828–2821.

[BM88] Laszlo Babai and Shlomo Moran, *Arthur-merlin games: A randomized proof system, and a hierarchy of complexity classes*, Journal of Computer and Systems Sciences (1988), no. 36, 254–276.

[Bor77] Alan Borodin, *On relating time and space to size and depth*, SIAM Journal on Computing **6** (1977), no. 4, 733–744.

[Bur69] Donald Bures, *An extension of kakutani's theorem on infinite product measures to the tensor product of semifinite w*-algebras*, Transactions of the American Mathematical Society **135** (1969), 199–212.

[BV04] Stephen Boyd and Lieven Vandenberghe, *Convex optimization*, Cambridge University Press, 2004.

[Deu89] David Deutsch, *Quantum computational networks*, Proceedings of the Royal Society of London A, no. 425, 1889, pp. 73–90.

[Deu85] ———, *Quantum theory, the church-turing principle and the universal quantum computer*, Proceedings of the Royal Society of London A, no. 400, 1985, pp. 97–117.

[Fan53] Ky Fan, *Minimax theorems*, Proceedings of the National Academy of Sciences 39, 1953, pp. 42–47.

[FK97] Uriel Feige and Joe Killian, *Making games short*, Proceedings of the 29th ACM Symposium on Theory of Computing, 1997, pp. 506–516.

[FLKN92] Lance Fortnow, Carsten Lund, Howard Karloff, and Noam Nisan, *Algebraic methods for interactive proof systems*, Journal of the ACM **39** (1992), no. 4, 859–868.

[For97] Lance Fortnow, *Counting complexity*, Complexity Theory Retrospective II, Springer, 1997, pp. 81–107.

[FvdG99] Christopher A. Fuchs and Jeroen van de Graaf, *Cryptographic distinguishability measures of quantum-mechanical states*, IEEE Transactions on Information Theory **45** (1999), no. 4, 1216–1227.

[Gil77] John Gill, *Computational complexity of probibalistic Turing machines*, SIAM Journal on Computing **6** (1977), no. 4, 675–695.

[GMR89] Shafi Goldwasser, Silvio Micali, and Charles Rackhoff, *The knowledge complexity of interactive proof systems*, SIAM Journal on Computing **18** (1989), no. 1, 186–208.

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

[GW07] Gus Gutowski and John Watrous, *Toward a general theory of quantum games*, Proceedings of the 39th Annual ACM Symposium on Theory of Computing, 2007, pp. 565–574.

[GW10] Gus Gutowski and Xiaodi Wu, *Short quantum games characterize PSPACE*, http://arxiv.org/pdf/1011.2787v1.pdf, 2010.

[GW11] Gus Gutoski and Xiaodi Wu, *Parallel approximation of min-max problems with applications to classical and quantum zero-sum games*, http://arxiv.org/pdf/1011.2787v2.pdf, 2011.

[JJUW09] Rahul Jain, Zhengfeng Ji, Sarvagya Upadhyay, and John Watrous, *QIP=PSPACE*, http://arxiv.org/pdf/0907.4737v2.pdf, 2009.

[JJUW10] ———, *QIP=PSPACE*, Proceedings of the 42nd ACM Symposium on Theory of Computing, 2010, pp. 573–582.

[Joz94] Richard Jozsa, *Fidelity of mixed quantum states*, Journal of Modern Optics (1994), 2315–2324.

[Kal07] Sayten Kale, *Efficient algorithms using the multiplicative weights updates method*, Ph.D. thesis, Princeton University, 2007.

[Kit97] Alexei Kitaev, *Quantum computations: algorithms and error correction*, Russian Mathematical Survey **52** (1997), no. 6, 1191–1249.

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

[KM90] Daphine Koller and Nimrod Megiddo, *The complexity of two-person zero-sum games in extensive form*, Games and Economic Behavior **4** (1990), 528–552.

[KMvS94] Daphine Koller, Nimrod Megiddo, and Bernhard von Stengel, *Fast algorithms for finding randomized strategies in game trees*, Proceedings of the 26th ACM Symposium on Theory of Computing, 1994, pp. 750–759.

[KW00] Alexei Kitaev and John Watrous, *Parallelization, amplification, and exponential time simulation of quantum interactive proof systems*, Proceedings of the 32nd ACM Symposium on Theory of Computing, 2000, pp. 08–617.

[MW05] Chris Marriott and John Watrous, *Quantum Arthur-Merlin games*, Computational Complexity **14** (2005), no. 2, 122–152.

[NC00] Michael Nielsen and Isaac Chuang, *Quantum computation and quantum information*, Cambridge University Press, 2000.

[NS03] Ashwin Nayak and Peter Shor, *On bit-commitment based quantum coin flipping*, Physical Review A **67** (2003), no. 1, article no. 012304.

[Pap83] Christos Harilaos Papadimitriou, *Games against nature*, Proceedings of the 24th IEEE Symposium on the Foundation of Computer Science, 1983, pp. 446–450.

[RSA78] Ron Rivest, Adi Shamir, and Leonard Adleman, *A method fot obtaining digital signatures and public-key cryptosystems*, Communications of the ACM **21** (1978), no. 2, 120–126.

[Sha92] Adi Shamir, *IP=PSPACE*, Journal of the ACM, vol. 39(4), 1992, pp. 869–877.

[Sho96] Peter Shor, *Fault-tolerant quantum computation*, Proceedings of the 37th IEEE Symposium on the Foundation of Computer Science, 1996, pp. 56–65.

[Sho97] ———, *Polynomial-time algorithms for prime factorization and discrete logarithms on a quantum computer*, SIAM Journal on Computing **26** (1997), no. 5, 1484–1509.

[Sio58] Marcel Sion, *On general minmax theorems*, Journal of Mathematics (1958), 171–176.

[SR02] Robert W. Spekkens and Terry Rudolph, *Degrees of concealment and bindingness in quantum bit-commitment protocols*, Physical Review A **65** (2002), no. 1, 012310, 1–10.

[Uhl76] Armin Uhlmann, *The transition probability in the state baseof a *-algebra*, Report on Mathematical Physics **9** (1976), no. 2, 273–279.

[Uhl92] ———, *The metric of bures and the geometric phase*, Quantum Groups and Related Topics: Proceedings of the First Max Born Symposium (R. Gielerak, J. Lukierski, and Z. Popowicz, eds.), 1992, p. 267.

[VB96] Lieven Vandenberghe and Stephen Boyd, *Semidefinite programming*, SIAM Review **38** (1996), 49–95.

[vN28] John von Neumann, *Zur Theorie der Gesellschaftsspiele*, Mathematische Annalen **100** (1928), 295–320.

[vzG93] Joachim von zur Gathen, *Parallel linear algebra*, Synthesis of parallel algorithms (J.Reif, ed.), 1993.

[Wu10] Xiaodi Wu, *Equilibrium value method for the proof of QIP=PSPACE*, http://arxiv.org/pdf/201004.0264v2.pdf, 2010.

[Yao93] A. Chi-Chih Yao, *Quantum circuit complexity*, Proceedings of the 34th IEEE Symposium on the Foundation of Computer Science, 1993, pp. 352–361.