Analyzing Quantum Cryptographic Protocols Using Optimization Techniques

by

Jamie Sikora

A thesis
presented to the University of Waterloo
in fulfillment of the
thesis requirement for the degree of
Doctor of Philosophy
in
Combinatorics and Optimization

Waterloo, Ontario, Canada, 2012

I hereby declare that I am the sole author of this thesis. This is a true copy of the thesis, including any required final revisions, as accepted by my examiners.

I understand that my thesis may be made electronically available to the public.

Abstract

This thesis concerns the analysis of the unconditional security of quantum cryptographic protocols using convex optimization techniques. It is divided into the study of coin-flipping and oblivious transfer. We first examine a family of coin-flipping protocols. Almost all of the handful of explicitly described coin-flipping protocols are based on bit-commitment. To explore the possibility of finding explicit optimal or near-optimal protocols, we focus on a class which generalizes such protocols. We call these BCCF-protocols, for bit-commitment based coin-flipping. We use the semidefinite programming (SDP) formulation of cheating strategies along the lines of Kitaev to analyze the structure of the protocols.

In the first part of the thesis, we show how these semidefinite programs can be used to simplify the analysis of the protocol. In particular, we show that a particular set of cheating strategies contains an optimal strategy. This reduces the problem to optimizing a linear combination of fidelity functions over a polytope which has several benefits. First, it allows one to model cheating probabilities using a simpler class of optimization problems known as second-order cone programs (SOCPs). Second, it helps with the construction of point games due to Kitaev as described in Mochon's work. Point games were developed to give a new perspective for studying quantum protocols. In some sense, the notion of point games is dual to the notion of protocols.

There has been increased research activity in optimization concerning generalizing theory and algorithms for linear programming to much wider classes of optimization problems such as semidefinite programming. For example, semidefinite programming provides a tool for potentially improving results based on linear programming or investigating old problems that have eluded analysis by linear programming. In this sense, the history of semidefinite programming is very similar to the history of quantum computation. Quantum computing gives a generalized model of computation to tackle new and old problems, improving on and generalizing older classical techniques. Indeed, there are striking differences between linear programming and semidefinite programming as there are between classical and quantum computation. In this thesis, we strengthen this analogy by studying a family of classical coin-flipping protocols based on classical bit-commitment. Cheating strategies for these "classical BCCF-protocols" can be formulated as linear programs (LPs) which are closely related to the semidefinite programs for the quantum version. In fact, we can construct point games for the classical protocols as well using the analysis for the

quantum case. The intricate relationship between the semidefinite programming quantum protocol analysis and the linear programming classical protocol analysis is depicted in the figure below.

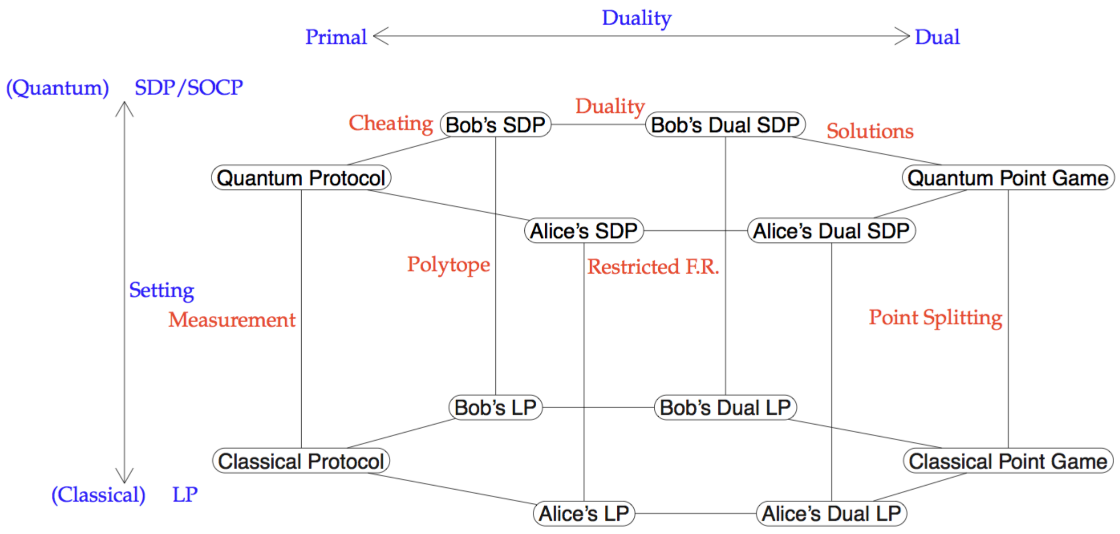


Figure 1: Relationship between classical and quantum BCCF-protocols and their point games. F.R. denotes "feasible region."

Using point games, we prove that every classical BCCF-protocol allows exactly one of the parties to entirely determine the outcome. Also, we rederive Kitaev's lower bound to show that only "classical" protocols can saturate Kitaev's analysis. Moreover, if the product of Alice and Bob's optimal cheating probabilities is 1/2, then at least one party can cheat with probability 1.

The second part concerns the design of an algorithm to search for BCCF-protocols with small bias. Most coin-flipping protocols with more than three rounds have eluded direct analysis. To better understand the properties of optimal BCCF-protocols with four or more rounds, we turn to computational experiments. We design a computational optimization approach to search for the best protocol based on the semidefinite programming formulations of cheating strategies. We create a protocol filter using cheating strategies, some of which build upon known strategies and others are based on convex optimization and linear algebra. The protocol filter efficiently eliminates candidate protocols with too

high a bias. Using this protocol filter and symmetry arguments, we perform searches in a matter of days that would have otherwise taken millions of years. Our experiments checked 10^{16} four and six round BCCF-protocols and suggest that the optimal bias is 1/4.

The third part examines the relationship between oblivious transfer, bit-commitment, and coin-flipping. We consider oblivious transfer which succeeds with probability 1 when the two parties are honest and construct a simple protocol with security provably better than any classical protocol. We also derive a lower bound by constructing a bit-commitment protocol from an oblivious transfer protocol. Known lower bounds for bit-commitment then lead to a constant lower bound on the bias of oblivious transfer. Finally, we show that it is possible to use Kitaev's semidefinite programming formulation of cheating strategies to obtain optimal lower bounds on a "forcing" variant of oblivious transfer related to coin-flipping.

Acknowledgements

First, I thank my supervisors, Ashwin Nayak and Levent Tunçel, for their advice and support over the last several years. They are both outstanding supervisors and I am very lucky to have been given the opportunity to learn from them. I also thank my thesis committee – Andrew Childs, Peter Høyer, Michele Mosca, and John Watrous – for their helpful comments and suggestions. Finally, I thank my family for their encouragement, and my wife, Caitlin, for her endless support.

Contents

Chapter 1

Introduction

This thesis is comprised of the work I have done on quantum coin-flipping and quantum oblivious transfer. My work on quantum coin-flipping, the content of Chapter 2 and Chapter 3, was done with my supervisors Ashwin Nayak and Levent Tunçel and my work on quantum oblivious transfer was done with Iordanis Kerenidis and André Chailloux.

We start with an overview of the problems and proof techniques examined in this thesis and a brief history of quantum cryptography. We then provide the necessary background on linear algebra, quantum information, semidefinite programming, formal definitions of cryptographic primitives, and quantum protocols in Section 1.4.

1.1 Overview

In this thesis, we use optimization techniques to give a detailed analysis of a family of quantum bit-commitment based coin-flipping protocols, which we call BCCF-protocols. After defining the protocols, we formulate cheating strategies for both parties as semidefinite programs (abbreviated as SDP, discussed in Subsection 1.4.4). These SDPs fully represent cheating in the protocol, that is, there is a direct connection between feasible solutions of the SDP and cheating strategies in the protocol. Moreover, the objective function is the success probability of the corresponding cheating strategy. This transforms the problem of analyzing cheating in the coin-flipping protocol into analyzing optimization problems.

Duality theory is the backbone of semidefinite programming and can often provide a new perspective. This is the case with the cheating SDPs of coin-flipping protocols as Kitaev used the duals of the cheating SDPs to derive the first constant lower bound on the bias of coin-flipping protocols. Moreover, he used these dual SDPs and other notions from convex analysis to construct "point games." We similarly use dual SDPs to construct a family of point games and show their correspondence with BCCF-protocols. This illustrates Kitaev's ideas of how protocols and "point games" are dual notions.

We then repeat the same analysis for a family of "classical" BCCF-protocols. Since these rely on classical messages and calculations, we are able to model cheating strategies as linear programs. It turns out that the linear programs for the classical cheating strategies and the semidefinite programs for the quantum strategies are very closely related. In fact, there are close connections not only between the cheating strategy formulations, but throughout the whole process of constructing the point games, as illustrated in the figure below.

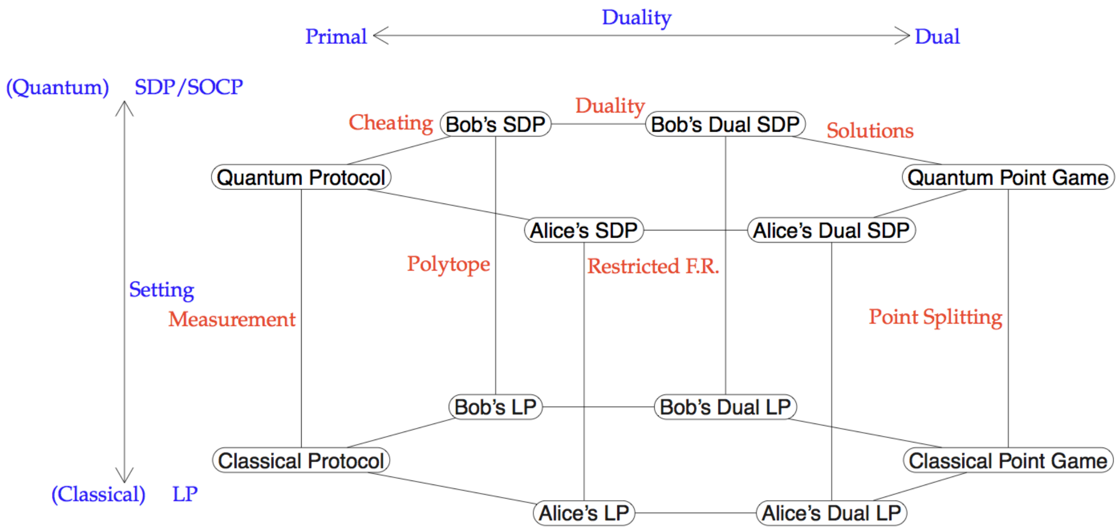


Figure 1.1: Quantum-classical connections between protocols and point games.

This figure illustrates the fact that the generalization of classical to quantum information is analogous to the generalization of linear programming to semidefinite programming.

Using point games, we prove that at least one party can cheat perfectly in every classical BCCF-protocol and that this result extends to the quantum case. Further analysis shows that exactly one party can cheat perfectly. We then rederive Kitaev's lower bound to prove that if this lower bound is saturated, then the cheating probabilities are the same as in the corresponding classical protocol. This proves that there are no BCCF-protocols with bias $\varepsilon = 1/\sqrt{2} - 1/2$.

Unfortunately, finding closed-form expressions for the optimal values of these SDPs is a very difficult problem. We therefore study the structure of the SDPs to design a search algorithm to find the best choice of protocol parameters to minimize cheating. In terms of optimization theory, we are seeking data for which these SDPs have small optimal values. By restricting the feasible region of the cheating SDPs, we reveal hidden convex structure leading to "near-optimal solutions." These near-optimal solutions are in fact near-optimal cheating strategies whose success probabilities approximate the optimal cheating probabilities. This helps to reveal some structure in these highly interactive protocols which generalizes the analysis in previous work. By exploiting these near-optimal cheating strategies, we are able to search over 10^{16} protocols to approximate the best parameters that minimize cheating. Without these strategies and other heuristics, this search would take millions of years.

Concerning the analysis of oblivious transfer, we first make connections between it and bit-commitment. Using known protocols and lower bounds for bit-commitment, we construct a protocol and derive lower bounds for oblivious transfer. We then make the connection between a "forcing" variant of oblivious transfer and coin-flipping. In particular, we show how similar cheating SDPs are involved and can be used to derive bounds on the bias.

1.2 An overview of the use of optimization techniques in quantum literature

Semidefinite programming gained popularity in the 1990s after Goemans and Williamson discovered how to use semidefinite programming to significantly improve the worst-case approximation ratio of heuristic algorithms for the MAX-CUT problem [GW95]. This further illustrated the use of semidefinite programming as a theoretical tool for analyzing important problems in combinatorics and other areas of mathematics and computer science.

We give a quick glimpse of some of the areas of quantum computation that have benefited from using SDPs and related topics from optimization theory.

Quantum information

One of the oldest uses of convex optimization techniques in quantum information is finding optimal measurements [Hol73b, YKL75], see also [Hel69, EMV03]. The first two papers predate the advent of semidefinite programming by over a decade, making this result quite remarkable. The ideas in these papers have many applications, from the hidden subgroup problem [Ip03, BCvD05, BCvD06] to quantum cryptography (see [Amb01], [KN04], [CKS10], [CK11], etc). In fact, discriminating between two quantum states is one of the central themes in this thesis.

Some related problems are unambiguous state discrimination [ESH04], distinguishing between convex sets of states [GW05], quantum channel discrimination [Sac05, PW09], and quantum strategy discrimination [Gut09, Gut10].

Another central theme in this thesis is the "closeness" of two quantum states or probability vectors as measured by the fidelity function (see Section 1.4.3). Along with studying norms useful for quantum channel discrimination, Watrous [Wat09] analyzed the fidelity function using semidefinite programming and showed that Uhlmann's and Alberti's characterizations of the fidelity function are "dual" to each other. We provide a similar analysis for the fidelity function over probability vectors in Subsection 1.4.7.

Computational complexity theory

Computational complexity theory is the study of classifying how difficult certain problems are to solve. It turns out that algorithms to approximate the optimal values of semidefinite programs have been incredibly useful in the study of quantum computational complexity theory, see [KW00] for an early example. In recent work, the matrix multiplicative weights update method for approximating the optimal value of an SDP was used by Jain, Ji, Upadhyay, and Watrous [JJUW11] to prove that the complexity class known as QIP (for quantum interactive proof systems) is equivalent to the complexity class known as PSPACE (problems that can be decided using a polynomial amount of space).

Also, using product theorems for semidefinite programming and other convex optimization problems, one can show that certain complexity classes admit perfect parallel repetition [MS07, Gut09, GSU11]. Roughly speaking, perfect parallel repetition is when two games are played simultaneously, then one cannot entangle strategies such to increase the maximum probability of winning both games. This can be used to improve the error bounds in the proof systems.

Query complexity

Query complexity is the study of determining how many queries are needed to ascertain a value of a certain black-box function f, such as the XOR of n bits. Duality theory of semidefinite programming has been used to show that many quantum adversary methods are equivalent [SS06]. The proof relies on showing that the adversary methods considered are equivalent to either the semidefinite version of minimax, or another semidefinite program that turns out to be its dual. Showing that the optimal values of these two SDPs are equal yields the result.

Recently, it has been shown by Reichardt [Rei09] that the query complexity of a Boolean function can be captured by the general adversary bound, and this bound is given by the optimal value of an SDP. This SDP can be modified to yield a norm used in the study of state conversion [LMR$^+$11]. Semidefinite programming appears in numerous papers on query complexity, see for example [BSS03, HLS07, LR11] and the references therein.

Linear optical quantum gates

A task that was thought impossible was to perform a two-qubit entangling gate in a photonic quantum computer. Indeed, a DiVincenzo criterion [DiV00] is that one needs two-qubit entangling gates to perform circuit-based quantum computation. Knill, Laflamme, and Milburn discovered a way to perform a two-qubit entangling gate using the non-linear sign shift gate (NLS gate), which in turn can be used to implement a controlled-Z gate. The NLS gate cannot be perfectly implemented using quantum mechanics, however it can be implemented in a probabilistic way using post-selection. The proposal of Knill, Laflamme, and Milburn implements this gate with probability 1/4.

The theory of convex optimization can be applied to finding the maximum success probability of implementing such gates under certain restrictions, as shown by Eisert [Eis05]. That is, by fixing several parameters, the problem becomes a convex optimization problem. What is different from many other optimization problems in quantum literature is that those in Eisert's paper are *second-order cone programs* (discussed in Subsection 1.4.4), although these were treated as SDPs. This paper is one of the few instances of a second-order cone program arising in a quantum setting.

Cryptography

We now give a brief overview of how SDPs and convex analysis have been used to tackle similar problems to those in this thesis. Firstly, Kitaev formalized a way to study the cheating strategies of quantum coin-flipping protocols using semidefinite programming. Using this, he developed point games which also use other ideas from convex analysis such as analyzing the cone of operator monotone functions and the duality of convex sets (see Subsection 1.4.2). We give an overview of Kitaev's ideas in Subsection 1.4.6.

Using SDP formulations of cheating strategies, Mochon studied a few different families of weak coin-flipping protocols [Moc04, Moc05]. He studied protocols based on public-coins, which are similar to the bit-commitment based protocols in this thesis. He was able to find optimal primal and dual solutions to the cheating SDPs, which for the first time gave a description of optimal cheating strategies for such highly interactive protocols. Then, using the convexity of optimal parameters, he was able to show a lower bound on the bias of the entire family of protocols (which could be approached as the number of messages grows large).

The goal in Chapters 2 and 3 is to analyze a family of quantum coin-flipping protocols using SDP techniques in the same manner as Mochon did for his family of protocols.

1.3 A brief history of quantum cryptography

Quantum cryptography dates back to the work of Wiesner in his seminal paper "Conjugate coding" [Wie83]. Written around 1970, the concepts were so new that it took until 1983 to get published. Among other things, this paper provides "a means for transmitting two messages either but not both of which may be received," which Wiesner called *multiplexing.* This task now goes by the name *1-out-of-2 oblivious transfer* which is one of the cryptographic primitives examined in this thesis. His idea was for the sender, call her Alice, to use a light pipe as the communication channel and have messages encoded as certain polarizations. Then the receiver, call him Bob, would measure the messages as he received them in a way that depends on the message he wants to receive. This approach is secure in practice (assuming the limitations of current technology) but, as Wiesner pointed out, is insecure in principle since an all-powerful Bob could decode both messages. This work was the beginning of quantum cryptography.

After hearing about Wiesner's work, Bennett and Brassard used similar ideas to those in Wiesner's paper to present a quantum key distribution scheme which is unconditionally secure, or information theoretically secure, against an eavesdropper [BB84], (see also [LC99, PS00, May01]). That is, a computationally unbounded eavesdropper can obtain very little information about the key. This is opposed to the security analyses based on computational assumptions such as the hardness of factoring or finding discrete logarithms. In 1984, many classical protocols for oblivious transfer and coin-flipping were based on, e.g., the hardness of factoring. However, many of these cryptosystems turn out to be vulnerable to quantum attacks using Shor's algorithm [Sho94]. This makes the unconditional security which can be promised by quantum mechanics more attractive since it does not rely on yet unproven computational conjectures.

In Bennett and Brassard's key distribution scheme, they also use messages being polarizations of light in such a way that if an eavesdropper tampered with the message, then Alice and Bob would be alerted and could abort the protocol. In the same paper, they present a coin-flipping protocol that follows in the same manner as their key distribution scheme. Roughly speaking, coin-flipping is the cryptographic task of two mistrustful parties generating a random bit over a communication channel. Much like Wiesner's multiplexing protocol, Bennett and Brassard's coin-flipping protocol is secure in practice, however insecure in principle. The work presented in these two seminal papers opened up many new problems in quantum cryptography such as authentication [BG89], the bounded storage model [DFSS08], and the problems addressed in this thesis. In particular, we are interested in finding the attainable levels of information theoretic security of coin-flipping and oblivious transfer protocols.

Quantum coin-flipping and quantum bit-commitment

Coin-flipping was introduced by Blum [Blu81] as a way for two parties to “flip a coin by telephone.” Consider the scenario where Alice and Bob are settling a divorce and need to decide who keeps the TV. They decide to settle this over the telephone, as they despise seeing each other, in the following way: Alice flips a coin and only if Bob is able to guess the value can he have the TV. The problem here is that Alice can cheat by telling Bob that his guess is wrong, even if it is not. Ideally, we would like a method such that neither Alice nor Bob can control the outcome. Unfortunately, this is impossible using classical information (assuming Alice and Bob are computationally all-powerful) since game theory tells us that at least one of them has a strategy that ensures a certain win. This motivates the use of quantum information when designing coin-flipping protocols.

Quantum coin-flipping is interesting for several reasons. Firstly, it is important in quantum cryptography being a task known as a *primitive*. Primitives are simple cryptographic tasks which are used as building blocks for larger, more elaborate protocols. For this reason, it is important to understand the attainable levels of security of such primitives, since it could affect the security of the larger protocol. Secondly, it is interesting from a general quantum computational standpoint: If we start with Alice and Bob sharing no entanglement, then how could they create an EPR (Einstein-Podolsky-Rosen) state shared between them if Alice and Bob suspect the other of cheating? Cheating can be viewed as trying to bias the amplitudes of the supposed EPR state, and we wish to design protocols such that the resulting state is as close to maximally entangled as possible.

To discuss the security of a coin-flipping protocol, we define the bias, denoted as ε, as the maximum of:

- Pr[Alice can force Bob to accept a desired outcome (without Bob aborting)] $- 1/2$,
- Pr[Bob can force Alice to accept a desired outcome (without Alice aborting)] $- 1/2$.

Aharonov, Ta-Shma, Vazirani, and Yao [ATVY00] first showed the existence of a coin-flipping protocol with bias $\varepsilon = 0.4143 < 1/2$ proving that quantum information can guarantee a level of security provably better than any classical protocol. Roughly speaking, the protocol proceeds as follows:

- Alice chooses $a \in \{0,1\}$ uniformly at random and encodes it into a quantum state. Alice sends this state to Bob.
- Bob then responds to Alice with a guess for the value of a.
- Alice then sends a and a description of the quantum state to Bob. He measures to check if the state received is in the state claimed by Alice. If so, Bob's guess being right or wrong determines the outcome of the protocol.

If Alice wants to cheat, then she must avoid detection by Bob and if Bob wants to cheat, then he must be able to learn the value of a early. By choosing the encoding quantum states in a smart way, it can be the case that neither Alice nor Bob can completely control the outcome, even if they are not computationally bounded. Ambainis [Amb01] and Spekkens and Rudolph [SR01] showed better choices of states such that the bias lowers to $\varepsilon = 1/4$, see also [NS03], [KN04]. As for lower bounds, Lo and Chau [LC97], showed that ideal coin-flipping is impossible, i.e., a bias of $\varepsilon = 0$ is impossible. In 2002, Kitaev [Kit02] used a formulation of cheating strategies as semidefinite programs to prove that the product of Alice and Bob's cheating probabilities is at least 1/2, showing that the bias satisfies $\varepsilon \geq 1/\sqrt{2} - 1/2 \approx 0.207$ for every protocol. Another proof of this lower bound was given by Gutoski and Watrous [GW07] using a different representation of quantum strategies. The problem of finding the optimal bias was resolved by Chailloux and Kerenidis [CK09] who showed the existence of protocols with bias $\varepsilon < 1/\sqrt{2} - 1/2 + \delta$ for any $\delta > 0$. The protocols they present rely only on classical messages and a related primitive known as *quantum weak coin-flipping*.

The version of coin-flipping discussed thus far is called *strong coin-flipping*. Weak coin-flipping is when Alice and Bob desire opposing outcomes and therefore we can view this primitive as having a winner and a loser. This primitive is well-suited for situations when Alice and Bob favour opposite outcomes, such as playing a game or settling a divorce. As for security, we can define the bias, denoted ε_{WCF}, in the same way, except Alice and Bob would only try to force specific outcomes. From this definition, weak coin-flipping protocols could have smaller biases than strong coin-flipping protocols. Indeed, Spekkens and Rudolph [SR02] presented a simple protocol for weak coin-flipping which has bias $\varepsilon_{\text{WCF}} = 1/\sqrt{2} - 1/2 \approx 0.207$ already matching Kitaev's lower bound for strong coin-flipping. Mochon [Moc04, Moc05] improved on the ideas in [SR02] to find protocols with biases of $\varepsilon_{\mathrm{WCF}} = 0.192$ and approaching $\varepsilon_{\mathrm{WCF}} = 1/6$, respectively. One of the reasons these protocols are able to surpass Kitaev's lower bound is that they have stronger cheat detection steps: only the "winner" gets tested for cheating. The best known lower bound for weak coin-flipping is by Ambainis [Amb01] who showed that a protocol with bias $\varepsilon_{\mathrm{WCF}}$ must use $\Omega(\log\log(1/\varepsilon_{\mathrm{WCF}}))$ rounds of communication. Then, in a breakthrough result, Mochon [Moc07] showed the existence of weak coin-flipping protocols with bias $\varepsilon_{\mathrm{WCF}} < \delta$ for any $\delta > 0$ using a development of Kitaev's called *point games.*

Point games were developed by Kitaev as a new perspective for studying coin-flipping protocols. Roughly speaking, a point game is a sequence

$$p_0 \to p_1 \to \cdots \to p_n,$$

where each p_i is a probability distribution over finitely many points in $\mathbb{R}^2_+$. Using his SDP formulation of cheating strategies in coin-flipping protocols, Kitaev found rules determining valid moves (or transitions) from p_i to p_{i+1} (more on this in Section 1.4.6). Let $P^*_{\mathrm{B},1}$ be the optimal probability Bob can force honest Alice to accept outcome 1 and let $P^*_{\mathrm{A},0}$ be the optimal probability Alice can force honest Bob to accept outcome 0. Then for any protocol and for any $\delta > 0$, there exists a point game satisfying:

- p_0 has two points $(0, 1)$ and $(1, 0)$, each having probability $1/2$,
- p_n has one point $(P^*_{\mathrm{B},1} + \delta, P^*_{\mathrm{A},0} + \delta)$,
- $p_i \to p_{i+1}$ is a "valid move," for $i \in \{0, 1, \ldots, n-1\}$.

What is surprising is that the converse is also true. If there is a point game of the form:

- p_0 has two points $(0, 1)$ and $(1, 0)$, each having probability $1/2$,
- p_n has one point $(\zeta_{\mathrm{B}}, \zeta_{\mathrm{A}})$,
- $p_i \to p_{i+1}$ is a "valid move," for $i \in \{0, 1, \ldots, n-1\}$,

then for any $\delta > 0$, there exists a weak coin-flipping protocol with $P^*_{\mathrm{B},1} \leq \zeta_{\mathrm{B}} + \delta$ and $P^*_{\mathrm{A},0} \leq \zeta_{\mathrm{A}} + \delta$. Using this new machinery, Mochon [Moc07] was able to prove the existence of point games with final point being arbitrarily close to $(1/2, 1/2)$ proving the existence of weak coin-flipping protocols with arbitrarily small bias. However, the reverse mapping from some point games to the description of a protocol is very difficult. No one has conducted this mapping for Mochon's optimal point game. The problem remains of explicitly constructing such optimal weak coin-flipping protocols and, therefore, any protocols that rely on them, such as optimal protocols for strong coin-flipping and *bit-commitment*.

Bit-commitment is a cryptographic primitive closely related to coin-flipping. Roughly speaking, bit-commitment is the task of Alice sending an encoding of a random bit a to Bob such that:

- The encoding 'hides' the value of a from Bob, and 'binds' Alice to the value a,
- Alice can reveal a and Bob can then check that the revealed bit is consistent with the encoding.

We can also define the bias of a bit-commitment protocol, denoted ε_{BC}, as the maximum of:

- Pr[Bob can infer a from the encoding without Alice aborting] $- 1/2$,
- $\displaystyle\frac{1}{2} \sum_{c \in \{0,1\}}$ Pr[Alice reveals $a = c$ without Bob aborting] $- 1/2$.

The history of quantum bit-commitment is closely related to quantum coin-flipping with the first example being the coin-flipping protocol in [BB84]. Moreover, most of the strong coin-flipping protocols previously mentioned are bit-commitment protocols with an added message from Bob who sends Alice his guess of a before Alice reveals it; the coin-flip being the correctness of Bob's guess. In this case, the bias of the bit-commitment protocol is the same as the bias of the coin-flipping protocol. This is the form of the protocols in [BB84, ATVY00, Amb01, SR01, KN04]. Mayers [May97] and Lo, Chau [LC97] proved that quantum bit-commitment protocols with $\varepsilon_{\text{BC}} = 0$ do not exist by showing that if Bob can infer no information about a before Alice reveals it, then Alice can cheat with probability 1. There has been some controversy over this *no-go* theorem, with some saying it is not entirely general. Some claim to be able to create bit-commitment protocols by circumventing some of the assumptions used. For example, one assumption is that Alice and Bob both have a complete description of the protocol before it starts. However, this is a widely accepted concept known as "Kerckhoffs's principle" [Ker83] which is also assumed in this thesis.

The result of the no-go theorem can be improved using Kitaev's lower bound for strong coin-flipping which, using the coin-flipping protocol construction above, shows that the bias can be no lower than that of strong coin-flipping. Recently, this has been improved by Chailloux and Kerenidis [CK11] who showed that $\varepsilon_{\text{BC}} \geq 0.239$ for every bit-commitment protocol. They show this bound is tight by presenting a protocol with bias $\varepsilon_{\text{BC}} < 0.239 + \delta$ for any choice of $\delta > 0$. Again, these protocols rely on optimal weak coin-flipping protocols as subroutines.

In this thesis, we construct coin-flipping protocols using bit-commitment in a way similar to the construction in [NS03]. Instead of adding the extra message from Bob before Alice reveals the value of a, we are going to blend two bit-commitment protocols in the following way. Alice chooses a bit a and encodes it in a quantum state. Bob chooses a bit b and encodes it in a quantum state. They each take turns revealing parts of their states to each other. Once the states are completely revealed, they reveal a and b to each other and they output $a \oplus b$ if cheating is not detected. Nayak and Shor showed that in every protocol of this form, one party can cheat with probability at least $9/16$. Although this is not as strong as the implicit cheating strategy from Kitaev's lower bound, it showed an explicit cheating strategy for this class of protocols.

Quantum oblivious transfer

Oblivious transfer, abbreviated OT, is the cryptographic primitive where Alice sends to Bob one of two bits but is oblivious to the bit received. Wiesner first proposed oblivious transfer as multiplexing, although the cryptographic significance was not known at the time. The first use of the term 'oblivious transfer' was by Rabin [Rab81] while discussing how to exchange secrets. This form of oblivious transfer is different than Wiesner's multiplexing and accomplished the following task:

- Alice has one bit x she wishes to transfer to Bob,
- With probability $1/2$, Bob receives the bit x, and with probability $1/2$, he receives # indicating the message was lost,

- Alice does not learn if x or # is received.

This is often referred to as “Rabin OT.” Another version of oblivious transfer was proposed by Even, Goldreich, and Lempel in [EGL85] which accomplished the following task:

- Alice has two bits x_0 and x_1 she wishes to transfer to Bob,
- With probability $1/2$, Bob receives the bit x_0, and with probability $1/2$, he receives x_1,
- Alice does not learn which bit is received.

This version is called “1-out-of-2 OT” and is an interactive version of Wiesner’s multiplexing. It was shown by Crépeau [Cré87] that these two forms of oblivious transfer are equivalent. More specifically, he showed that with a perfect 1-out-of-2 OT channel, one can implement a perfect Rabin OT channel and with access to perfect Rabin OT channels, one can implement a 1-out-of-2 OT channel (with an exponentially small probability of failure). In this thesis, we only consider 1-out-of-2 OT and refer to it as just oblivious transfer or OT (with a formal definition of OT given in Subsection 1.4.5).

The history of OT is less straightforward than quantum coin-flipping or quantum bit-commitment due to varying names, definitions, and security requirements. For example, we can have different definitions concerning the probability Bob receives the correct message or the objectives of cheating Alice or Bob. OT is a very important cryptographic task since it is universal for secure function evaluation [Kil88] (where Alice and Bob compute a function without revealing extra information about their respective inputs). This being the case, we need to worry about how one defines the security of an OT protocol when designing larger, more elaborate cryptographic systems that rely on this security. It has been proven by Lo [Lo97] that ideal oblivious transfer is impossible, even with quantum information. Since every OT protocol is imperfect, one needs to settle for some definition of security which could vary by situation.

Bennett, Brassard, Breidbard, and Wiesner [BBBW83] presented an early example of quantum oblivious transfer. They provide a way that Alice can encode her two bits x_0 and x_1 into a pure qubit state such that Bob can measure and learn either bit with probability $\cos^2(\pi/8)$. In fact, since only one qubit of information is transmitted, Bob cannot learn both bits due to information bounds [Hol73a, Nay99]. It turns out that Bob cannot obtain any information about the XOR of the two bits since the density matrices for the two cases are identical.

There has been work done on quantifying the amount of 'information' which is leaked in OT protocols, i.e., how much information Bob can get about Alice's inputs or how much information Alice can get from Bob's inputs. For example, Salvail, Schaffner, and Sotakova [SSS09] define a quantity called *information leakage* and show that oblivious transfer has constant leakage. In other work, Jain, Radhakrishnan, and Sen [JRS09] showed that for the *Set Membership Problem*, which can be thought of as 1-*out-of-n* oblivious transfer, if Bob reveals at most k bits of information about his input, then Alice must reveal at least $n/2^{O(k)}$ bits of information about her input. Oblivious transfer has also been studied in other settings such as the bounded-storage model [DFSS08] and the noisy-storage model [Sch10] which each provide a high level of security.

We now discuss the notion of security considered in this thesis. We define the bias, denoted ε_{OT}, as the maximum of the two quantities:

- Pr[Alice can guess Bob's input $b \in \{0,1\}$ without Bob aborting] $- 1/2$,
- Pr[Bob can guess Alice's bits (x_0, x_1) without Alice aborting] $- 1/2$,

where Bob's input b is the index of the bit he wants to learn. We assume that Bob gets the correct value of x_b with probability 1, when both parties are honest, and therefore we have $\varepsilon_{\text{OT}} \in (0, 1/2]$ for every protocol. Note that Bob may try to cheat to learn any function of (x_0, x_1) and then a different definition of bias would be needed. However, in this thesis, we are mostly concerned with lower bounding the bias so this definition is the most appropriate. The goal is now the same as in coin-flipping and bit-commitment which is to determine the smallest attainable bias. Note that the lower bounds discussed previously are in terms of information and do not directly translate into learning probabilities. Therefore, using this definition of bias for oblivious transfer protocols is a new measure of security we adopt for this thesis for which we give the first lower and upper bounds.

1.3.1 Contributions

Here we discuss the contributions and layout of the thesis.

Chapter 2

We define a family of coin-flipping protocols based on bit-commitment, which we call BCCF-protocols. Using SDP formulations of cheating strategies, we analyze the structure of optimal cheating strategies of Alice and Bob. In particular, we reduce the SDPs to a much simpler structure involving the optimization of fidelity functions over a polytope. Using this reduced problem, we are able to prove a bound on the dimension of messages and also to characterize optimal dual solutions to develop point games.

We then study a family of classical BCCF-protocols and show how they are related to the quantum version. Using this relationship, we also develop "classical point games" and use them to prove that every classical BCCF-protocol has bias $\varepsilon = 1/2$. In fact, we are able to conclude that exactly one party can cheat with probability 1 which also extends to the quantum case.

In Subsection 2.3, we adapt Kitaev's lower bound for BCCF-protocols and prove a theorem which roughly states that only "classical protocols" can saturate Kitaev's analysis. That is, if the product of Alice and Bob's optimal cheating probabilities equals 1/2, then one of them can cheat with probability 1. This rules out the possibility of BCCF-protocols with bias $\varepsilon = 1/\sqrt{2} - 1/2$.

Chapter 3

In this chapter, we search over the parameters defining a BCCF-protocol to seek one with the smallest bias. In Subsection 3.1, we lower bound the optimal cheating probabilities by finding feasible solutions to the cheating SDPs. By restricting the feasible regions, we reveal hidden convex structure behind some "near-optimal strategies." We use these strategies to filter out protocols with high bias. In Subsection 3.2, we examine the symmetry in the protocol parameters to reduce the number of protocols in the search. In Subsection 3.3, we develop the search algorithm by showing how the mesh is created and presenting a decent protocol which provides a cut-off point for the bias. We present our numerical findings for four round and six round searches in Subsection 3.4. We conclude with the conjecture that the minimal bias attainable by a BCCF-protocol is $\varepsilon = 1/4$.

Chapter 4

The topic of Chapter 4 is oblivious transfer. We present in Subsection 4.1 a simple protocol for oblivious transfer that has bias $\varepsilon_{\mathrm{OT}} = 1/4$ which is related to the bit-commitment protocol in [KN04]. To prove a lower bound on the bias, we construct a bit-commitment protocol using oblivious transfer and relate the corresponding cheating probabilities in Subsection 4.2. We then present (and prove in Section 4.3) a Learning-In-Sequence Lemma which provides a way for Bob to sequentially learn Alice's two input bits x_0 and x_1. Using this lemma and known lower bounds for quantum bit-commitment [CK11], we prove that the bias for every OT protocol satisfies $\varepsilon_{\mathrm{OT}} \geq 0.0852$. We conclude this chapter by studying a forcing variant of oblivious transfer in Subsection 4.4 which generalizes coin-flipping. We show how to formulate cheating strategies as SDPs which are used to derive optimal lower bounds on the bias.

1.4 Mathematical preliminaries and notation

In this section, we establish the notation and the necessary background for this thesis.

1.4.1 Linear algebra

For a finite set A, we denote by $\mathbb{R}^A$, $\mathbb{R}^A_+$, Prob^A, and $\mathbb{C}^A$ the set of real vectors, nonnegative real vectors, probability vectors, and complex vectors, respectively, each indexed by A. We use $\mathbb{R}^n$, $\mathbb{R}^n_+$, Prob^n, and $\mathbb{C}^n$ for the special case when $A = \{1, \ldots, n\}$. We denote by Σ^A and Σ^A_+ the set of Hermitian matrices and positive semidefinite matrices, respectively, each with columns and rows indexed by A. For vectors x and y, the notation $x \geq y$ denotes that $x - y$ has nonnegative entries, $x > y$ denotes that $x - y$ has positive entries, and for matrices X and Y, the notation $X \succeq Y$ denotes that $X - Y$ is positive semidefinite, and $X \succ Y$ denotes $X - Y$ is positive definite when the underlying spaces are clear from context. When we say that a matrix is positive semidefinite or positive definite, it is assumed to be Hermitian which implies that $\Sigma^A_+ \subset \Sigma^A$.

Suppose we define

$$X := \begin{bmatrix} A & B \\ B^\top & C \end{bmatrix},$$

where C is positive definite. Then $X \succeq 0$ if and only if $S := A - BC^{-1}B^\top \succeq 0$. We call S the *Schur complement.*

The Kronecker product of two matrices X and Y, denoted $X \otimes Y$, is defined as

$$X \otimes Y := \begin{bmatrix} X_{1,1}\,Y & X_{1,2}\,Y & \cdots & X_{1,n}\,Y \\ X_{2,1}\,Y & X_{2,2}\,Y & \cdots & X_{2,n}\,Y \\ \vdots & \vdots & \ddots & \vdots \\ X_{n,1}\,Y & X_{n,2}\,Y & \cdots & X_{n,n}\,Y \end{bmatrix}.$$

Note that $X \otimes Y \in \Sigma_+^{A \times B}$ when $X \in \Sigma_+^A$ and $Y \in \Sigma_+^B$ and $\mathrm{Tr}(X \otimes Y) = \mathrm{Tr}(X) \cdot \mathrm{Tr}(Y)$ when X and Y are square.

For a vector $x \in \mathbb{C}^A$, we define $\mathrm{supp}(x)$ to be the set of indices of A where x is nonzero. We define x^{-1} to be the element-wise inverse, when $x > 0$, and $\sqrt{x}$ to be the element-wise square root when $x \geq 0$. The element-wise square root of a probability vector yields a unit vector (in the Euclidean norm). This operation, in some sense, is a conversion of a probability vector to a quantum state.

The *Schatten* 1*-norm*, or *trace norm*, of a matrix X is defined as

$$\|X\|_1 := \mathrm{Tr}(\sqrt{X^* X}),$$

where X^* is the adjoint of X and $\sqrt{X}$ denotes the square root of a positive semidefinite matrix X, i.e., the positive semidefinite matrix Y such that $Y^2 = X$. Note that the 1-norm of a matrix is the sum of its singular values. We also define the 1-norm of a vector $p \in \mathbb{C}^A$ as

$$\|x\|_1 := \sum_{x \in A} |p_x|.$$

We use the notation $\bar{a}$ to denote the complement of a bit a with respect to 0 and 1 and $a \oplus b$ to denote the XOR of the bits a and b. We use $\mathbb{Z}_2^n$ to denote the set of n-bit binary strings.

For a vector $p \in \mathbb{R}^A$, we denote by $\mathrm{Diag}(p) \in \Sigma^A$ the diagonal matrix with p on the diagonal. For a matrix $X \in \Sigma^A$, we denote by $\mathrm{diag}(X) \in \mathbb{R}^A$ the vector on the diagonal of X.

For a matrix X, we denote by $\mathrm{Null}(X)$ the nullspace of X, by $\det(X)$ the determinant of X, and by $\lambda_{\max}(X)$ the largest eigenvalue of X. We denote by $\langle X, Y\rangle$ the standard inner product of matrices acting on the same space given by $\mathrm{Tr}(X^*Y)$.

1.4.2 Convex analysis

A *convex combination* of finitely many vectors $x_1, \ldots, x_n$ is any vector of the form $\sum_{i=1}^n \lambda_i x_i$, when $\lambda_1, \ldots, \lambda_n \in [0,1]$ satisfy $\sum_{i=1}^n \lambda_i = 1$. The *convex hull* of a set C is the set of convex combinations of elements of C, denoted $\mathrm{conv}(C)$. A set C is *convex* if $C = \mathrm{conv}(C)$.

A *convex function* $f : \mathbb{R}^n \to \mathbb{R} \cup \{\infty\}$ is one that satisfies

$$f(\lambda x + (1-\lambda)y) \le \lambda f(x) + (1-\lambda) f(y), \text{ for all } x, y \in \mathbb{R}^n, \lambda \in [0,1].$$

A convex function is *strictly convex* if

$$f(\lambda x + (1-\lambda)y) < \lambda f(x) + (1-\lambda) f(y), \text{ for all } x \neq y,\ x, y \in \mathbb{R}^n, \lambda \in (0,1).$$

We say that a convex function is proper if $f(x) < +\infty$ for some $x \in \mathbb{R}^n$. The *epigraph* of a function f is the set

$$\mathrm{epi}(f) := \{(x,t) : f(x) \le t\}$$

which are the points above the graph of the function. A function is convex if and only if its epigraph is a convex set.

A function $f : \mathbb{R}^n \to \mathbb{R} \cup \{-\infty\}$ is *(strictly) concave* if $-f$ is (strictly) convex, and proper when $f(x) > -\infty$ for some $x \in \mathbb{R}^n$. The *hypograph* of a function f is the set

$$\mathrm{hypo}(f) := \{(x,t) : f(x) \ge t\}$$

which are the points below the graph of the function. A function is concave if and only if its hypograph is a convex set.

Let $f_1, \ldots, f_n : \mathbb{R}^n \to \mathbb{R} \cup \{\infty\}$ be proper, convex functions. We define the *convex hull* of $\{f_1, \ldots, f_n\}$, denoted $\mathrm{conv}\{f_1, \ldots, f_n\}$, as the greatest convex function f such that $f(x) \le f_1(x), \ldots, f_n(x)$ for every $x \in \mathbb{R}^n$. Equivalently, we can define this in terms of the epigraph

$$\mathrm{conv}\{f_1, \ldots, f_n\}(x) := \inf\left\{t : (x,t) \in \mathrm{conv}(\cup_{i=1}^n \mathrm{epi}(f_i))\right\}.$$

We can similarly define the *concave hull* of $\{f_1, \ldots, f_n\}$, denoted $\text{conc}\{f_1, \ldots, f_n\}$, as

$$\text{conc}\,\{f_1, \ldots, f_n\} := -\text{conv}\,\{-f_1, \ldots, -f_n\}$$

when $f_1, \ldots, f_n : \mathbb{R}^n \to \mathbb{R} \cup \{-\infty\}$ are proper, concave functions. That is, the concave hull is the least concave function f such that $f(x) \geq f_1(x), \ldots, f_n(x)$ for every $x \in \mathbb{R}^n$, or

$$\text{conc}\{f_1, \ldots, f_n\}(x) := \sup\left\{t : (x,t) \in \text{conv}(\cup_{i=1}^n \text{hypo}(f_i))\right\}.$$

A *convex optimization problem* or *convex program* is one of the form

$$\inf_{x \in C} f(x),$$

where f is a convex function and C is a convex set. Alternatively, one could maximize a concave function over a convex set. See Subsection 1.4.4 for the types of convex programs examined in this thesis.

A set $C \subseteq \mathbb{R}^n$ is closed if it contains every limit point of sequences within C. In a complex Euclidean space, a set is compact if and only if it is closed and bounded.

We call a convex set K a cone if $\lambda K \subseteq K$, for all $\lambda > 0$. This thesis concerns the optimization of linear functions over closed, convex cones, see Subsection 1.4.4. Given a set $C \subseteq \mathbb{R}^n$, its dual cone, denoted C^*, is defined as

$$C^* := \left\{x \in \mathbb{R}^n : \langle x, y \rangle \geq 0, \text{ for all } y \in C\right\}.$$

One can check that the dual cone is always a closed, convex cone. Also, we have that $C_1 \subseteq C_2$ implies $C_1^* \supseteq C_2^*$ and the converse holds if C_1 and C_2 are closed convex cones.

A function $f : \Sigma^n \to \Sigma^m$ is said to be *operator monotone* if

$$f(X) \succeq f(Y) \quad \text{when} \quad X \succeq Y.$$

The set of operator monotone functions is a convex cone.

A function $f : \mathbb{R}^n \to \mathbb{R}$ is said to be *positively homogeneous* if

$$f(\lambda x) = \lambda f(x), \quad \text{for all } \lambda > 0.$$

A *polyhedron* is the solution set of a system of finitely many linear inequalities (or equalities). A *polytope* is a bounded polyhedron.

1.4.3 Quantum information

In this subsection, we give a brief introduction to quantum information. For a more thorough treatment of the subject, we refer the reader to [NC00].

Quantum states

Quantum states are a description of the state of a physical system, such as the spin of an electron. In the simplest case, such a state is a unit vector in a finite-dimensional Hilbert space (which is a complex Euclidean space). For example, the following vectors are quantum states in $\mathbb{C}^2$

$$|0\rangle := \begin{bmatrix} 1 \\ 0 \end{bmatrix}, \ |1\rangle := \begin{bmatrix} 0 \\ 1 \end{bmatrix}, \ |+\rangle := \frac{1}{\sqrt{2}} \begin{bmatrix} 1 \\ 1 \end{bmatrix}, \ |-\rangle := \frac{1}{\sqrt{2}} \begin{bmatrix} 1 \\ -1 \end{bmatrix}.$$

These are examples of *quantum bits* or *qubits.* The first two are standard basis vectors and can be thought of as the logical states of a standard computer. In general, a qubit can be written as

$$|\psi\rangle := \alpha_0 \, |0\rangle + \alpha_1 \, |1\rangle,$$

where $\alpha_0, \alpha_1 \in \mathbb{C}$ satisfy $|\alpha_0|^2 + |\alpha_1|^2 = 1$. This condition ensures that $|\psi\rangle$ has norm equal to 1.

Notice that $|+\rangle = \frac{1}{\sqrt{2}}|0\rangle + \frac{1}{\sqrt{2}}|1\rangle$ and $|-\rangle = \frac{1}{\sqrt{2}}|0\rangle - \frac{1}{\sqrt{2}}|1\rangle$. These states are said to be in a *superposition* of the states $|0\rangle$ and $|1\rangle$ and hold properties of being in both states at the same time. This is in part what gives quantum computers the power to efficiently tackle hard problems such as factoring [Sho94].

In general, a system may be in one of several superpositions according to some probability distribution. Suppose a quantum system is in such a state drawn from the ensemble of states $(|\psi_0\rangle, |\psi_1\rangle, \ldots, |\psi_n\rangle)$ with respective probabilities $(p_0, p_1, \ldots, p_n)$. This quantum state may be described more succinctly as a *density matrix*, defined as

$$\sum_{i=0}^{n} p_i \, |\psi_i\rangle\langle\psi_i|,$$

where $\langle\psi| := |\psi\rangle^*$ is the adjoint. (We define $\langle\psi|\phi\rangle := \langle|\phi\rangle, |\psi\rangle\rangle$ to be the inner product.) Notice that the matrix above is positive semidefinite and has unit trace. Moreover, any positive semidefinite matrix with unit trace can be written in the above form using its spectral decomposition.

Two different probability distributions over superpositions may have the same density matrix. For example, density matrices do not record "phase information," i.e., the density matrix of the state $|\psi\rangle$ is the same as that of $-|\psi\rangle$. However, two ensembles with the same density matrix behave identically under all allowed physical operations. Therefore, there is no loss in working with density matrices.

A quantum superposition given by the vector $|\psi\rangle$ can be thought of as the rank 1 density matrix $|\psi\rangle\langle\psi|$ and we call it a *pure state*. States with a density matrix of rank 2 or more are said to be *mixed*.

Quantum operations

Suppose U is a unitary operator acting on $\mathbb{C}^A$ and $|\psi\rangle \in \mathbb{C}^A$ is a quantum state. If we apply U to $|\psi\rangle$ then the resulting quantum state is $U|\psi\rangle \in \mathbb{C}^A$. Note this is a well-defined quantum state since unitary operators preserve the norm. Also, every unitary operation is a valid quantum operation.

Suppose we are given a state drawn from the ensemble $(|\psi_0\rangle, |\psi_1\rangle, \ldots, |\psi_n\rangle)$ with respective probabilities $(p_0, p_1, \ldots, p_n)$. If we apply the unitary U to the state, it is equivalent to saying the state is given to us from the ensemble $(U|\psi_0\rangle, U|\psi_1\rangle, \ldots, U|\psi_n\rangle)$ with the same probabilities. We can write the new density matrix as

$$\sum_{i=0}^{n} p_i \, U|\psi_i\rangle\langle\psi_i|U^* = U\left(\sum_{i=0}^{n} p_i \, |\psi_i\rangle\langle\psi_i|\right)U^*,$$

where U^* is the adjoint of U. Thus, if we are given a density matrix ρ and apply the unitary U, then $U\rho U^*$ is the density matrix of the new quantum state. Indeed, this matrix is still positive semidefinite with unit trace.

Quantum measurement

A quantum measurement is a means of extracting classical information from a quantum state. This is represented by a set of positive semidefinite operators $\{\Pi_1, \ldots, \Pi_n\}$ satisfying $\sum_{i=1}^{n} \Pi_i = \mathrm{I}$. This set of operators is called a *positive operator valued measure* or a POVM.

If we have a density matrix ρ and we apply the measurement $\{\Pi_1, \ldots, \Pi_n\}$, we obtain outcome "i" with probability $\langle \Pi_i, \rho \rangle$. The definitions of density matrices and measurements establish this as a well-defined probability distribution over the indices. After a measurement, the state *collapses* to one that is consistent with the outcome. For example, if each Π_i is a projection, then on outcome i, the state becomes $\Pi_i \rho \Pi_i$ (normalized). Therefore, sometimes only a limited amount of classical information may be extracted from a given quantum state. For example, if we apply the measurement $\{\Pi_0 := |0\rangle\langle 0|, \Pi_1 := |1\rangle\langle 1|\}$ to the state $|+\rangle\langle +|$, we obtain the outcome:

$$\begin{cases} \text{``0''} & \text{with probability} \quad \langle \Pi_0, |+\rangle\langle +| \rangle = 1/2, \\ \text{``1''} & \text{with probability} \quad \langle \Pi_1, |+\rangle\langle +| \rangle = 1/2. \end{cases}$$

Multiple quantum systems

Suppose we are given two quantum systems that are independently in states $|\psi_1\rangle \in \mathbb{C}^{A_1}$ and $|\psi_2\rangle \in \mathbb{C}^{A_2}$. The combined state is $|\psi_1\rangle \otimes |\psi_2\rangle \in \mathbb{C}^{A_1} \otimes \mathbb{C}^{A_2} = \mathbb{C}^{A_1 \times A_2}$. Note that the Kronecker product has the property that $\|x \otimes y\|_2 = \|x\|_2 \|y\|_2$ so the unit norm is preserved. Although we can compose two vectors to get a larger vector, it is not always possible to decompose a larger vector into two smaller vectors in this way. Consider the following state $|\Phi^+\rangle := \dfrac{1}{\sqrt{2}}[1, 0, 0, 1]^\top$. This vector cannot be expressed as $|\psi_1\rangle \otimes |\psi_2\rangle$ for any choice of $|\psi_1\rangle, |\psi_2\rangle \in \mathbb{C}^2$. A vector that cannot be expressed as a Kronecker product of two smaller vectors is said to be *entangled*.

If two disjoint quantum systems are independently in states given by density matrices $\rho_1 \in \Sigma_+^{A_1}$ and $\rho_2 \in \Sigma_+^{A_2}$, then the joint state of the combined system is given by the density matrix $\rho_1 \otimes \rho_2 \in \Sigma_+^{A_1 \times A_2}$. We make use of the properties that Kronecker products preserve positive semidefiniteness and $\mathrm{Tr}(\rho_1 \otimes \rho_2) = \mathrm{Tr}(\rho_1) \cdot \mathrm{Tr}(\rho_2)$. In this representation, it is not always possible to write a density matrix $\rho \in \Sigma_+^{A_1 \times A_2}$ as $\rho_1 \otimes \rho_2$ where $\rho_1 \in \Sigma_+^{A_1}$ and $\rho_2 \in \Sigma_+^{A_2}$. However, there is a way to describe the "part" of the state which is in A_1 using the notion of *partial trace*.

Partial trace over quantum states and marginal probability vectors

The *partial trace over* A_1 is the unique linear transformation $\mathrm{Tr}_{A_1} : \Sigma^{A_1 \times A_2} \to \Sigma^{A_2}$, which satisfies

$$\mathrm{Tr}_{A_1}(\rho_1 \otimes \rho_2) = \mathrm{Tr}(\rho_1) \cdot \rho_2,$$

for all $\rho_1 \in \Sigma^{A_1}$ and $\rho_2 \in \Sigma^{A_2}$. More precisely, given any matrix $X \in \Sigma_+^{A_1 \times A_2}$, we define Tr_{A_1} as

$$\mathrm{Tr}_{A_1}(X) := \sum_{x_1 \in A_1} \left(e_{x_1}^* \otimes \mathrm{I}_{A_2}\right) X \left(e_{x_1} \otimes \mathrm{I}_{A_2}\right),$$

where $\{e_{x_1} : x_1 \in A_1\}$ is the standard basis for $\mathbb{C}^{A_1}$. In fact, the definition is independent of the choice of basis, so long as it is orthonormal. Note that the partial trace is positive, i.e., $\mathrm{Tr}_{A_1}(X) \in \Sigma_+^{A_2}$ when $X \in \Sigma_+^{A_1 \times A_2}$, and also trace-preserving. This ensures that the new matrix is a well-defined density matrix.

Consider the scenario where two parties, Alice and Bob, hold parts of a quantum system which are jointly in some state ρ, i.e., they share a quantum state ρ over $\mathbb{C}^{A \times B}$. Then the partial trace of ρ over one space represents that quantum state over the remaining space. For example, $\mathrm{Tr}_A(\rho)$ is the density matrix representing Bob's half of the state and $\mathrm{Tr}_B(\rho)$ represents Alice's half.

Suppose we are given the density matrix $\rho \in \Sigma_+^A$. We call the pure state $|\psi\rangle \in \mathbb{C}^{A \times B}$ a *purification* of ρ if $\mathrm{Tr}_B|\psi\rangle\langle\psi| = \rho$. Note that a purification exists if $|B| \geq |A|$, and is in general not unique. An important property of purifications of the same state is that if $\mathrm{Tr}_B|\psi\rangle\langle\psi| = \mathrm{Tr}_B|\phi\rangle\langle\phi|$ for two pure states $|\psi\rangle$ and $|\phi\rangle$, then there exists a unitary U acting on $\mathbb{C}^B$ alone such that $|\psi\rangle = (\mathrm{I}_A \otimes U)\,|\phi\rangle$.

The partial trace operation is the quantum analogue of calculating marginal probability distributions. Consider a probability vector $p \in \mathrm{Prob}^{A \times B}$. We define the partial trace over A element-wise as

$$[\mathrm{Tr}_A(p)]_y = \sum_{x \in A} p_{x,y},$$

for each $y \in B$. Equivalently, we can write

$$\mathrm{Tr}_A(p) := \left(e_A^\top \otimes \mathrm{I}_B\right) p,$$

where $e_A \in \mathbb{C}^A$ is the vector of all ones. We see that this gives the marginal probability distribution of p over B. Alternatively, one may view probability vectors as diagonal positive semidefinite matrices with unit trace. Then, taking the partial trace (as defined for quantum states) corresponds exactly to the computation of marginal distributions.

Distance measures for quantum states and probability vectors

The different notions of distance between quantum states and probability vectors are very important in quantum cryptography. Here, we discuss two distance measures and show how they are related.

We define the *fidelity* of two nonnegative vectors $p, q \in \mathbb{R}_+^A$ as

$$\mathrm{F}(p,q) := \left(\sum_{x \in A} \sqrt{p_x} \sqrt{q_x} \right)^2$$

and the fidelity of two positive semidefinite matrices ρ and σ as

$$\mathrm{F}(\rho,\sigma) := \left\| \sqrt{\rho} \sqrt{\sigma} \right\|_1^2 .$$

Both definitions of the fidelity are symmetric, positively homogeneous in both arguments, and concave. Notice $\mathrm{F}(\rho,\sigma) \geq 0$ with equality if and only if $\langle \rho, \sigma \rangle = 0$ and, if ρ and σ are quantum states, $\mathrm{F}(\rho,\sigma) \leq 1$ with equality if and only if $\rho = \sigma$. An analogous statement can be made for fidelity of probability vectors.

Another distance measure is the *trace distance*. We define the trace distance between two probability vectors p and q (also called the variational distance), denoted $\Delta(p,q)$, as

$$\Delta(p,q) := \frac{1}{2} \left\| p - q \right\|_1 .$$

We similarly define the trace distance between two quantum states ρ and σ as

$$\Delta(\rho,\sigma) := \frac{1}{2} \left\| \rho - \sigma \right\|_1 .$$

Notice $\Delta(\rho,\sigma) \geq 0$ with equality if and only if $\rho = \sigma$ and $\Delta(\rho,\sigma) \leq 1$ with equality if and only if $\langle \rho, \sigma \rangle = 0$. The analogous statement can be made for the trace distance between probability vectors.

We now discuss two important notions in quantum cryptography. The first is how easily two states can be distinguished from each other. For example, if Alice gives to Bob one of two states ρ or σ chosen uniformly at random, then Bob can measure to learn whether he has been given ρ or σ with maximum probability

$$\frac{1}{2} + \frac{1}{4} \|\rho - \sigma\|_1 = \frac{1}{2} + \frac{1}{2}\Delta(\rho, \sigma),$$

proven by Helstrom [Hel69]. The second notion is *quantum steering*. Suppose Alice has given to Bob the B part of $|\phi\rangle \in \mathbb{C}^{A \times B}$. Suppose she wants to send $\mathbb{C}^A$ in such a way to convince Bob that a different state was sent, say $|\psi\rangle \in \mathbb{C}^{A \times B}$. Her most general strategy is to apply a quantum operation on $\mathbb{C}^A$ before sending it to Bob. Using Uhlmann's Theorem [Uhl76], it can be shown that if Bob measures with the POVM $\{|\psi\rangle\langle\psi|, \mathrm{I} - |\psi\rangle\langle\psi|\}$, Alice can convince him that the state was $|\psi\rangle$ with maximum probability

$$\mathrm{F}(\mathrm{Tr}_A|\psi\rangle\langle\psi|, \mathrm{Tr}_A|\phi\rangle\langle\phi|).$$

The trace distance and fidelity are closely related. The following Fuchs-van de Graaf inequalities [FvdG99] illustrate this relationship

$$1 - \sqrt{\mathrm{F}(\rho, \sigma)} \leq \Delta(\rho, \sigma) \leq \sqrt{1 - \mathrm{F}(\rho, \sigma)},$$

for all quantum states ρ and σ. The analogous statement holds for probability vectors.

1.4.4 Semidefinite programming

A natural model of optimization when studying quantum information is semidefinite programming. A *semidefinite program*, abbreviated SDP, is an optimization problem of the form

$$\begin{array}{lrrcl} \text{(P)} & \sup & \langle C, X \rangle & & \\ & \text{subject to} & \mathcal{A}(X) & = & b, \\ & & X & \in & \Sigma^n_+, \end{array}$$

where $\mathcal{A} : \Sigma^n \to \mathbb{R}^m$ is linear, $C \in \Sigma^n$, and $b \in \mathbb{R}^m$. In this subsection, we give the necessary theory and tools that are used in this thesis concerning semidefinite programming. We begin with some definitions. In (P) above, we call $\mathcal{A}(X) = b$ and $X \in \Sigma^n_+$ the *constraints*, any solution of the constraints is said to be *feasible*, and the set of feasible solutions is the *feasible region*. We call $\langle C, X \rangle$ the *objective function* and any feasible solution $\bar{X}$ has the corresponding *objective function value* $\langle C, \bar{X} \rangle$. If there exists feasible $\bar{X}$ such that

$\langle C, \bar{X}\rangle \geq \langle C, X\rangle$ for all feasible X, then $\bar{X}$ is said to be an *optimal solution* and $\langle C, \bar{X}\rangle$ to be the *optimal objective value.* Note that we need to use "sup" in the definition since SDPs with finite optimal objective values exist having no optimal solutions.

We can write the *dual* of (P) as

$$\begin{array}{lrrcl} \text{(D)} & \inf & \langle b, y\rangle & & \\ & \text{subject to} & \mathcal{A}^*(y) - S & = & C, \\ & & S & \in & \Sigma^n_+, \end{array}$$

where $\mathcal{A}^*$ is the adjoint of $\mathcal{A}$. We refer to (P) as the primal problem and to (D) as its dual. The usefulness of defining the dual in this way is apparent in the following lemmas.

Lemma 1.4.1 (Weak duality). *For every $\bar{X}$ feasible for* (P) *and* $(\bar{y}, \bar{S})$ *feasible for* (D), *we have*

$$\langle b, \bar{y}\rangle - \langle C, \bar{X}\rangle = \langle \bar{X}, \bar{S}\rangle \geq 0.$$

We call the minimum value of $\langle \bar{X}, \bar{S}\rangle$ over feasible solutions the *duality gap.*

Using weak duality, we can prove bounds on the optimal objective value of (P) and (D), i.e., the objective function value of any primal feasible solution yields a lower bound on (D) and the objective function value of any dual feasible solution yields an upper bound on (P).

We say that a primal feasible solution $\bar{X}$ is strictly feasible if $\bar{X} \succ 0$ and a dual feasible solution $(\bar{y}, \bar{S})$ is strictly feasible if $\bar{S} \succ 0$. Sometimes these are also called *Slater points.*

Under mild conditions, we have that the optimal objective values of (P) and (D) coincide. In this case, we say that the primal and dual have *zero duality gap.*

Lemma 1.4.2 (Strong duality for (P)). *If the objective function of* (P) *is bounded from above on the set of feasible solutions and there exists a strictly feasible solution, then* (D) *has an optimal solution and there is zero duality gap.*

Since the dual of the dual is the primal problem, we get the following lemma.

Lemma 1.4.3 (Strong duality for (D)). *If the objective function of* (D) *is bounded from below on the set of feasible solutions and there exists a strictly feasible solution, then* (P) *has an optimal solution and there is zero duality gap.*

Another notion in convex optimization is *complementary slackness.* In the SDP above, notice that if $\bar{X}$ is an optimal solution for (P) and $(\bar{y}, \bar{S})$ an optimal solution for (D), then $\langle \bar{X}, \bar{S} \rangle = 0$ if there is zero duality gap. This means that $\bar{X}$ is orthogonal to $\bar{S}$. Similarly, if $\bar{X}$ is a feasible solution for (P) and $(\bar{y}, \bar{S})$ a feasible solution for (D) and $\langle \bar{X}, \bar{S} \rangle = 0$, then both feasible solutions are optimal in their respective problems by weak duality. We refer to the condition $\langle X, S \rangle = 0$ as complementary slackness.

Semidefinite programming has a powerful and rich duality theory and the interested reader is referred to [WSV00, Tun10] and the references therein.

Taking the duals of SDPs with multiple variables

The SDPs encountered in this thesis are a bit more involved than the standard form above. Most of the times when dual SDPs appear in papers, the construction of the dual is omitted since the calculations are quite tedious. There are many duals that appear in this thesis, so to explain their derivations, we develop the dual of the SDP below which captures a common structure.

Consider the SDP

$$\begin{array}{llrcll}
\text{(P')} & \sup & \langle C_k, X_k \rangle & & & \\
& \text{subject to} & \mathcal{A}_1(X_1) & = & B, & \\
& & \mathcal{A}_j(X_j) & = & \mathcal{D}_j(X_{j-1}), & \text{for all } j \in \{2, \ldots, k\}, \\
& & X_i & \in & \Sigma_+^{n_i}, & \text{for all } i \in \{1, \ldots, k\},
\end{array}$$

where C_k and B are Hermitian and $\mathcal{A}_1, \ldots, \mathcal{A}_k, \mathcal{D}_2, \ldots, \mathcal{D}_k$ are Hermiticity preserving. To take the dual of (P'), we put it into the form of (P) above. Define

$$X := \sum_{i=1}^{k} |i\rangle\langle i| \otimes X_i \quad \text{and} \quad C := |k\rangle\langle k| \otimes C_k \quad \text{and} \quad b := (\text{vec}(B), 0, \ldots, 0)^\top,$$

where vec(B) is a column vector containing exactly the elements of B (we could think of vec as stacking the columns). Technically, we need b to be a real vector, but this is not an issue since we can double the dimension and keep the real and imaginary parts separate.

By defining

$$\mathcal{A}(X) := \begin{bmatrix} \mathrm{vec}(\mathcal{A}_1(X_1)) \\ \mathrm{vec}(\mathcal{A}_2(X_2)) - \mathrm{vec}(\mathcal{D}_2(X_1)) \\ \vdots \\ \mathrm{vec}(\mathcal{A}_k(X_k)) - \mathrm{vec}(\mathcal{D}_k(X_{k-1})) \end{bmatrix},$$

we see that we get an SDP in the form previously discussed. Taking its dual yields

$$\begin{array}{rrcl} \inf & \langle b, y\rangle & & \\ \text{subject to} & \mathcal{A}^*(y) - S & = & C, \\ & S & \in & \Sigma_+^{n_1+\cdots+n_k}. \end{array}$$

Note that the off-diagonal entries of C, X, and S do not factor into any constraint or either objective function. Thus, we can assume they are set to 0, but they effectively do not matter.

Let $y = (y_1, \ldots, y_k)^\top$ and let Y_i be the matrix such that $\mathrm{vec}(Y_i) = y_i$ (we can assume it is Hermitian). We can write the dual objective function as

$$\langle b, y\rangle = \langle \mathrm{vec}(B), y_1\rangle = \langle B, Y_1\rangle.$$

We can now solve for $\mathcal{A}^*(y)$ from

$$\begin{array}{rcl} \langle \mathcal{A}^*(y), X\rangle & = & \langle y, \mathcal{A}(X)\rangle \\ & = & \langle y_1, \mathrm{vec}(\mathcal{A}_1(X_1))\rangle + \langle y_2, \mathrm{vec}(\mathcal{A}_2(X_2)) - \mathrm{vec}(\mathcal{D}_2(X_1))\rangle \\ & & + \cdots + \langle y_k, \mathrm{vec}(\mathcal{A}_k(X_k)) - \mathrm{vec}(\mathcal{D}_k(X_{k-1}))\rangle \\ & = & \langle \mathcal{A}_1^*(Y_1) - \mathcal{D}_2^*(Y_2), X_1\rangle + \langle \mathcal{A}_2^*(Y_2) - \mathcal{D}_2^*(Y_3), X_2\rangle + \cdots + \langle \mathcal{A}_k^*(Y_k), X_k\rangle. \end{array}$$

We can write the dual of (P') as

$$\begin{array}{lrrcl} \text{(D')} & \inf & \langle B, Y_1\rangle & & \\ & \text{subject to} & \mathcal{A}_1^*(Y_1) & \succeq & \mathcal{D}_2^*(Y_2), \\ & & \mathcal{A}_2^*(Y_2) & \succeq & \mathcal{D}_3^*(Y_3), \\ & & & \vdots & \\ & & \mathcal{A}_k^*(Y_k) & \succeq & C_k. \end{array}$$

Complementary slackness in this case is

$$\langle X_i, \mathcal{A}_i^*(Y_i) - \mathcal{D}_{i+1}^*(Y_{i+1})\rangle = 0, \text{ for all } i \in \{1, \ldots, k-1\}, \quad \text{and} \quad \langle X_k, \mathcal{A}^*(Y_k) - C_k\rangle = 0.$$

The SDPs in this thesis are mostly of this form. Some modifications are used later, such as replacing some of the positive semidefiniteness constraints with nonnegative vector constraints. Instead of rederiving this from scratch, we can simply modify the above dual. For example, if we replace the constraint $X_i \in \Sigma_+^{n_i}$ with $x_i \in \mathbb{R}_+^{n_i}$, then all we must do is replace the "$\succeq$" with an "$\geq$" in the i'th dual constraint. The simplest way to keep track of this is to view the complementary slackness conditions. If x is a vector, then the constraint (which appears in the inner product) also needs to be a vector inequality. We review $\mathbb{R}_+^n$ constraints below in the part about linear programming.

Second-order cone programming

The second-order cone (or Lorentz cone) is defined as

$$\mathrm{SOC}^n := \left\{(x,t) \in \mathbb{R}^n : x \in \mathbb{R}^{n-1}, \, t \geq \|x\|_2\right\}.$$

A *second-order cone program*, denoted SOCP, is an optimization problem of the form

$$\begin{array}{llrcl} \text{(P)} & \sup & \langle c, (x_1, \ldots, x_k)\rangle & & \\ & \text{subject to} & A(x_1, \ldots, x_k) & = & b, \\ & & x_i & \in & \mathrm{SOC}^{n_i}, \quad \text{for all } i \in \{1, \ldots, k\}, \end{array}$$

where A is an $m \times (\sum_{i=1}^k n_k)$ matrix, $b \in \mathbb{R}^m$, and k is finite. An SOCP also has a dual which can be written as

$$\begin{array}{llrcl} \text{(D)} & \inf & \langle b, y\rangle & & \\ & \text{subject to} & A^\top y - (s_1, \ldots, s_k) & = & c, \\ & & s_i & \in & \mathrm{SOC}^{n_i}, \quad \text{for all } i \in \{1, \ldots, k\}. \end{array}$$

Note that weak duality and strong duality also hold for SOCPs for the properly modified definition of strictly feasible.

There is a related cone called the *rotated second-order cone*, defined as

$$\mathrm{RSOC}^n := \left\{(a,b,x) \in \mathbb{R}^n : a, b \in \mathbb{R}_+, \, x \in \mathbb{R}^{n-2}, \, 2ab \geq \|x\|_2^2\right\}.$$

We can optimize over the rotated second-order cone using second-order cone programming because $(x, t) \in \mathrm{SOC}^n$ if and only if $(t/2, t, x) \in \mathrm{RSOC}^{n+1}$ and $(a, b, x) \in \mathrm{RSOC}^n$ if and only if $(a+b, x, a, b) \in \mathrm{SOC}^{n+1}$. Indeed, in some optimization literature, one refers to any cone linearly isomorphic to SOC^n as a second-order cone, such as RSOC^{n+1} above. For this reason, we still call it a second-order cone program if SOC is replaced by RSOC, or any other second-order cone.

Indeed, both second-order cone constraints can be cast as positive semidefinite constraints:

$$t \geq \|x\|_2 \iff \begin{bmatrix} t & x^\top \\ x & t\,\mathrm{I} \end{bmatrix} \succeq 0 \quad \text{and} \quad a, b \geq 0,\, 2ab \geq \|x\|_2^2 \iff \begin{bmatrix} 2a & x^\top \\ x & b\,\mathrm{I} \end{bmatrix} \succeq 0.$$

This proves that second-order cone programming is a special case of semidefinite programming. However, there are some differences between semidefinite programs and second-order cone programs. One is that the algorithms for solving second-order cone programs can be more efficient and robust than those for solving semidefinite programs. We refer the interested reader to [Stu99, Stu02, Mit03, AG03] and the references therein.

Linear programming

A linear program, denoted LP, can be written as

$$\begin{array}{llrcl} \text{(P)} & \max & \langle c, x\rangle & & \\ & \text{subject to} & Ax & = & b, \\ & & x & \in & \mathbb{R}^n_+, \end{array}$$

where A is an $m \times n$ matrix, $c \in \mathbb{R}^n$ and $b \in \mathbb{R}^m$.

Linear programming is a special case of both second-order cone programming and semidefinite programming. This can be seen by casting a nonnegativity constraint $t \geq 0$ as the SOC constraint $(0, t) \in \mathrm{SOC}^2$. Associated with every linear program is its dual which is defined as

$$\begin{array}{llrcl} \text{(D)} & \min & \langle b, y\rangle & & \\ & \text{subject to} & A^\top y - s & = & c, \\ & & s & \in & \mathbb{R}^n_+. \end{array}$$

Note that in this special case, we do not require strict feasibility to guarantee strong duality. If a feasible linear program is bounded, then it and its dual attain an optimal solution and the optimal values always coincide.

1.4.5 Quantum protocols

In this thesis, we consider two-party quantum *communication protocols* in the style of Yao [Yao93]. We concentrate on a class of communication protocols relevant to coin-flipping, bit-commitment and oblivious transfer. In such protocols, two parties Alice and Bob hold some quantum state; the states with each party are initialized to a fixed pure state. The initial joint state is therefore unentangled across Alice and Bob (otherwise, Alice and Bob could perform some tasks, such as coin-flipping, trivially). The two parties then communicate in turns. Suppose it is Alice's turn. Alice performs a unitary transformation on her state and then sends part of her state to Bob. Sending part of the state does not change the overall superposition, it merely changes ownership, i.e., who has control over that part of the state. This allows Bob to apply his next unitary transformation on the state under his control. At the end of the protocol, each player performs a measurement and, if applicable, announces the outcome.

Formally, the players Alice and Bob, hold some state, which initially factors into a tensor product $\mathbb{C}^{A_0} \otimes \mathbb{C}^{B_0}$ of Hilbert spaces. The state corresponding to $\mathbb{C}^{A_0}$ is in Alice's possession, and the state corresponding to $\mathbb{C}^{B_0}$ is in Bob's possession. When the protocol starts, $\mathbb{C}^{A_0}$ is initialized to some pure state $|\psi_{A,0}\rangle$ and $\mathbb{C}^{B_0}$ is initialized to $|\psi_{B,0}\rangle$, both of which are determined by the protocol. The communication consists of $m \geq 1$ rounds of exchanging messages. Either party may start. In the i'th round, $i \geq 1$, suppose it is Alice's turn. Suppose the state space just before the round factors as $\mathbb{C}^{A_{i-1}} \otimes \mathbb{C}^{B_{i-1}}$. Alice applies a unitary operator $U_{A,i}$ to $\mathbb{C}^{A_{i-1}}$. Then, Alice "sends" part of the state to Bob. Formally, the space $\mathbb{C}^{A_{i-1}}$ factors as $\mathbb{C}^{A_i} \otimes \mathbb{C}^{M_i}$, where $\mathbb{C}^{M_i}$ denotes the state space for the i'th message. Consequently, $\mathbb{C}^{B_i} = \mathbb{C}^{M_i} \otimes \mathbb{C}^{B_{i-1}}$. In the next round, Bob may thus apply a unitary operation to the part of the state previously in Alice's control.

At the end of the m rounds, Alice and Bob measure the states in their possession according to some measurement. The outcomes of these measurements represent their outputs. We emphasize that there are no measurements until all rounds of communication are completed. A protocol with intermediate measurements may be transformed into this form by appealing to standard techniques [BV97].

We are interested in the probabilities of the different outcomes when either party "cheats." Suppose Alice and Bob have agreed upon a protocol, i.e., a set of rules for the state initialization, communication, quantum operations, and measurements. What if Alice or Bob do not follow protocol? Suppose Alice is dishonest and would like to achieve some specific goal, e.g., forcing Bob to output "0" in a coin-flipping protocol. She may use a different space for her private operations, so that her space $\mathbb{C}^{A'_i}$ may be much larger than $\mathbb{C}^{A_i}$. She may create any initial state she wants. During the communication, the only restriction is that she send a state of the correct dimension, e.g., if the protocol requires a message with three qubits in the first message, then Alice sends three qubits. Between messages, she may apply any quantum operation she wants on the state in her possession. At the end of the protocol, she may use a different measurement of her choice. For example, she may simply output "0" if this is her desired outcome (which corresponds to a trivial measurement). The rules that Alice chooses to follow instead of the protocol constitute a *cheating strategy*.

We would like to quantify the extent to which a dishonest party can cheat while the other is honest, so we focus on runs of the protocol in which at most one party is dishonest. We analyze in this thesis the maximum probability with which Alice (or Bob) can cheat in terms of a "bias," i.e., the advantage over the honest probability that a cheating party can achieve.

Formal definitions of primitives

Here we give formal definitions of the primitives considered in this thesis.

Definition 1.4.4 (**Strong coin-flipping**). *A* strong coin-flipping *protocol with bias ε is a protocol with output $c \in \{0, 1, \text{abort}\}$, satisfying:*

- *Alice and Bob start uncorrelated,*
- *if Alice and Bob are honest, then they never abort and they output the same uniformly random bit $c \in \{0, 1\}$,*

- $P^*_{\mathrm{A},c}$ *is the maximum probability dishonest Alice can force honest Bob to accept outcome* $c \in \{0, 1\}$*,*

- $P^*_{\mathrm{B},c}$ *is the maximum probability dishonest Bob can force honest Alice to accept outcome* $c \in \{0, 1\}$*,*

- $\varepsilon := \max\{P^*_{\mathrm{B},0}, P^*_{\mathrm{B},1}, P^*_{\mathrm{A},0}, P^*_{\mathrm{A},1}\} - 1/2.$

The idea is to design protocols which protect honest parties from cheating parties and there are no security guarantees when both parties are dishonest.

Definition 1.4.5 (**Weak coin-flipping**). *A* weak coin-flipping *protocol with bias* $\varepsilon_{\mathrm{WCF}}$ *is a protocol with output* $c \in \{0, 1\}$*, satisfying:*

- *Alice and Bob start uncorrelated,*

- *if Alice and Bob are honest, they output the same uniformly random bit c,*

- $P^*_{\mathrm{A},0}$ *is the maximum probability dishonest Alice can force honest Bob to accept outcome* 0*,*

- $P^*_{\mathrm{B},1}$ *is the maximum probability dishonest Bob can force honest Alice to accept outcome* 1*,*

- $\varepsilon_{\mathrm{WCF}} := \max\{P^*_{\mathrm{B},1}, P^*_{\mathrm{A},0}\} - 1/2.$

We can assume neither party aborts in a WCF protocol. If, for instance, Alice detects Bob has cheated then she may declare herself the winner, i.e., the outcome is $c = 0$. This is not the case in strong coin-flipping since there is no sense of "winning."

Definition 1.4.6 (**Bit-commitment**). *A* bit-commitment *protocol with bias* $\varepsilon_{\mathrm{BC}}$ *is a protocol with a* commit phase *and a* reveal phase*, satisfying:*

- *Alice and Bob start uncorrelated,*

- *in the commit phase, Alice interacts with Bob in order to commit to a random bit* a*,*

- *in the reveal phase, Alice interacts with Bob in order to reveal a. Bob decides to accept or reject depending on the revealed value of a and his final state. We say that Alice successfully reveals a if Bob accepts the revealed value,*

- *if Alice and Bob are both honest then Alice always successfully reveals the bit a which she has committed,*

- *Alice's maximum cheating probability is*

$$P^*_{\text{A,BC}} = \sup \frac{1}{2} \sum_{c \in \{0,1\}} \Pr[\textit{Alice successfully reveals } a = c],$$

- *Bob's maximum cheating probability is*

$$P^*_{\text{B,BC}} = \sup \Pr[\textit{Bob guesses } a \textit{ after the commit phase without Alice aborting}],$$

- $\varepsilon_{\text{BC}} := \max\{P^*_{\text{B,BC}}, P^*_{\text{A,BC}}\} - 1/2$,

where the suprema are taken over all cheating strategies of the party indicated.

We see that we can use a bit-commitment protocol to create a coin-flipping protocol. In between the commit phase and reveal phase, Bob announces his guess for Alice's bit a. Then the bias of the coin-flipping protocol equals that of the bit-commitment protocol. Note that the protocols in this thesis are not of this form.

Definition 1.4.7 (**Oblivious transfer** (**OT**))**.** *An* oblivious transfer *protocol, denoted* OT, *with bias ε_{OT} is a protocol* with inputs, *satisfying:*

- *Alice and Bob start uncorrelated,*

- *Alice inputs two uniformly random bits (x_0, x_1) and Bob inputs a uniformly random index $b \in \{0, 1\}$,*

- *when Alice and Bob are honest they never abort, Bob learns x_b perfectly, Bob gets no information about $x_{\bar{b}}$, and Alice gets no information about b,*

- $P^*_{\text{A,OT}}$ *is the maximum probability dishonest Alice can learn b without Bob aborting,*

- $P^*_{\text{B,OT}}$ *is the maximum probability dishonest Bob can learn* (x_0, x_1) *without Alice aborting,*

- $\varepsilon_{\text{OT}} = \max\{P^*_{\text{B,OT}}, P^*_{\text{A,OT}}\} - 1/2.$

When a party cheats, we only refer to the probability by which they can learn the desired values without the other party aborting. For example, when Bob cheats, we do not require that he learns either bit with probability 1.

Note that there could be some ambiguity about how a party cheats if the inputs are not chosen randomly, i.e., if Bob chooses b however he wants, then what does it mean for Alice to learn b with some probability? If Bob can freely choose his input, a suitable definition of Alice's cheating probability is

$$\max_{c \in \{0,1\}} \{\Pr[\text{Alice learns } b | \text{Bob chooses input } b = c]\} . \tag{1.1}$$

Since we are mainly concerned with lower bounds, we assume Bob and Alice choose their inputs randomly since, for example, $P^*_{\text{A,OT}}$ is a lower bound on (1.1). On the other hand, suppose we have a protocol where Bob and Alice have specific values they want to input. Then this is easily remedied by adding the extra messages at the end of the protocol specifying whether the randomly chosen inputs were the ones desired. For example, suppose $b = 0$, but Bob really wanted $b = 1$. Then he can tell Alice, "my value of b is wrong." Then Bob can switch the value of b and Alice can switch her two bits x_0 and x_1. We can do a similar thing for Alice. Thus, having the inputs chosen randomly is not an issue when designing protocols with this definition of the bias.

Definition 1.4.8 (**Forcing oblivious transfer** (Forcing-OT))**.** *A k-out-of-n forcing oblivious transfer protocol, denoted as* Forcing-OT$_{(n,k)}$*, with* forcing bias ε_{FOT}*, is a protocol satisfying:*

- *Alice and Bob start uncorrelated,*

- *Alice and Bob have* no inputs*,*

- *Alice outputs n random bits $x := (x_1, \ldots, x_n)$,*

- *Bob outputs a random k-index set b and bit string x_b consisting of x_i for $i \in b$,*

- $P^*_{\mathrm{A},(b,x_b)} := \sup\{\Pr[\textit{Alice can force Bob to output } (b, x_b)]\} = \dfrac{\varepsilon_{\mathrm{A}}}{\binom{n}{k} \cdot 2^k}$,
- $P^*_{\mathrm{B},x} := \sup\{\Pr[\textit{Bob can force Alice to output } x]\} = \dfrac{\varepsilon_{\mathrm{B}}}{2^n}$,
- $\varepsilon_{\mathrm{FOT}} = \max\{\varepsilon_{\mathrm{A}}, \varepsilon_{\mathrm{B}}\}$,

where the suprema are taken over all cheating strategies of the party indicated.

The main difference between this primitive and the standard definition of oblivious transfer is the definition of security. Here, we design protocols to protect against a dishonest party being able to *force* a desired value as the output of the other party. Standard oblivious transfer protocols are designed to protect against a dishonest party *learning* the other party's input. Notice that in coin-flipping, we can design protocols to protect against a dishonest party forcing a desired outcome, but both parties *learn* the outcome perfectly when they are honest.

This primitive is a generalization of coin-flipping since we can cast the problem of coin-flipping as a 1-out-of-1 forcing oblivious transfer protocol. Of course, in Forcing-OT$_{(1,1)}$, Alice always knows Bob's index set so the forcing bias is the only interesting notion of security in this case.

We define the bias $\varepsilon_{\mathrm{FOT}}$ as a multiplicative factor instead of additive, since the honest probabilities can be much different and in this case this definition makes more sense. To relate this bias to the one previously mentioned in coin-flipping, we have that coin-flipping protocols with bias $\varepsilon_{\mathrm{FOT}} \leq \sqrt{2} + \delta$ exist for any $\delta > 0$, see [CK09].

1.4.6 The Kitaev coin-flipping protocol formalism

Kitaev developed point games from his SDP formulation of cheating strategies for coin-flipping protocols. Here, we review this construction.

Coin-flipping protocols

We give a setting for a coin-flipping protocol which is equivalent to the general setting described earlier. This setting has a space devoted for messages and each message has the same dimension. This is done for convenience as it makes the analysis in this subsection (and at the end of Chapter 4) simpler.

A coin-flipping protocol can be described by the following parameters:

- The number of messages, denoted here as m. We can assume m is even,
- three Hilbert spaces: Alice's private space $\mathbb{C}^A$, a message space $\mathbb{C}^M$, and Bob's private space $\mathbb{C}^B$,
- a set of unitaries $\{U_{\mathrm{A},1}, U_{\mathrm{A},3}, \ldots, U_{\mathrm{A},m-1}\}$ acting on $\mathbb{C}^{A \times M}$. These correspond to Alice's messages to Bob,
- a set of unitaries $\{U_{\mathrm{B},2}, U_{\mathrm{B},4}, \ldots, U_{\mathrm{B},m}\}$ acting on $\mathbb{C}^{M \times B}$. These correspond to Bob's messages to Alice,
- a projective measurement for Alice $\{\Pi_{\mathrm{A},0}, \Pi_{\mathrm{A},1}, \Pi_{\mathrm{A,abort}}\} \subset \Sigma_+^A$ determining Alice's protocol outcome,
- a projective measurement for Bob $\{\Pi_{\mathrm{B},0}, \Pi_{\mathrm{B},1}, \Pi_{\mathrm{B,abort}}\} \subset \Sigma_+^B$ determining Bob's protocol outcome.

The protocol proceeds as follows. Alice initializes the space $\mathbb{C}^A$ to $|\psi_{A,0}\rangle$ and Bob initializes $\mathbb{C}^{M \times B}$ to $|\psi_{M,0}\rangle_M |\psi_{B,0}\rangle_B$ and sends $\mathbb{C}^M$ to Alice. Then Alice applies her first unitary $U_{\mathrm{A},1}$ and sends $\mathbb{C}^M$ to Bob. Then he applies his first unitary $U_{\mathrm{B},2}$ and returns $\mathbb{C}^M$ to Alice. They repeat this until Bob applies his last unitary $U_{\mathrm{B},n}$. Then they both measure their private spaces to get the outcome of the protocol. This process is depicted in Figure 1.2.

The protocol parameters must satisfy the requirements that

1. Alice and Bob do not abort when both are honest.
2. They output the same bit when they are honest, and that bit is randomly generated.

If we let $|\psi\rangle \in \mathbb{C}^{A \times M \times B}$ be the state at the end of the protocol when Alice and Bob are honest, both requirements are satisfied when

$$\langle \Pi_{\mathrm{A},0} \otimes \mathrm{I}_M \otimes \Pi_{\mathrm{B},0}, |\psi\rangle\langle\psi| \rangle = \langle \Pi_{\mathrm{A},1} \otimes \mathrm{I}_M \otimes \Pi_{\mathrm{B},1}, |\psi\rangle\langle\psi| \rangle = \frac{1}{2}. \tag{1.2}$$

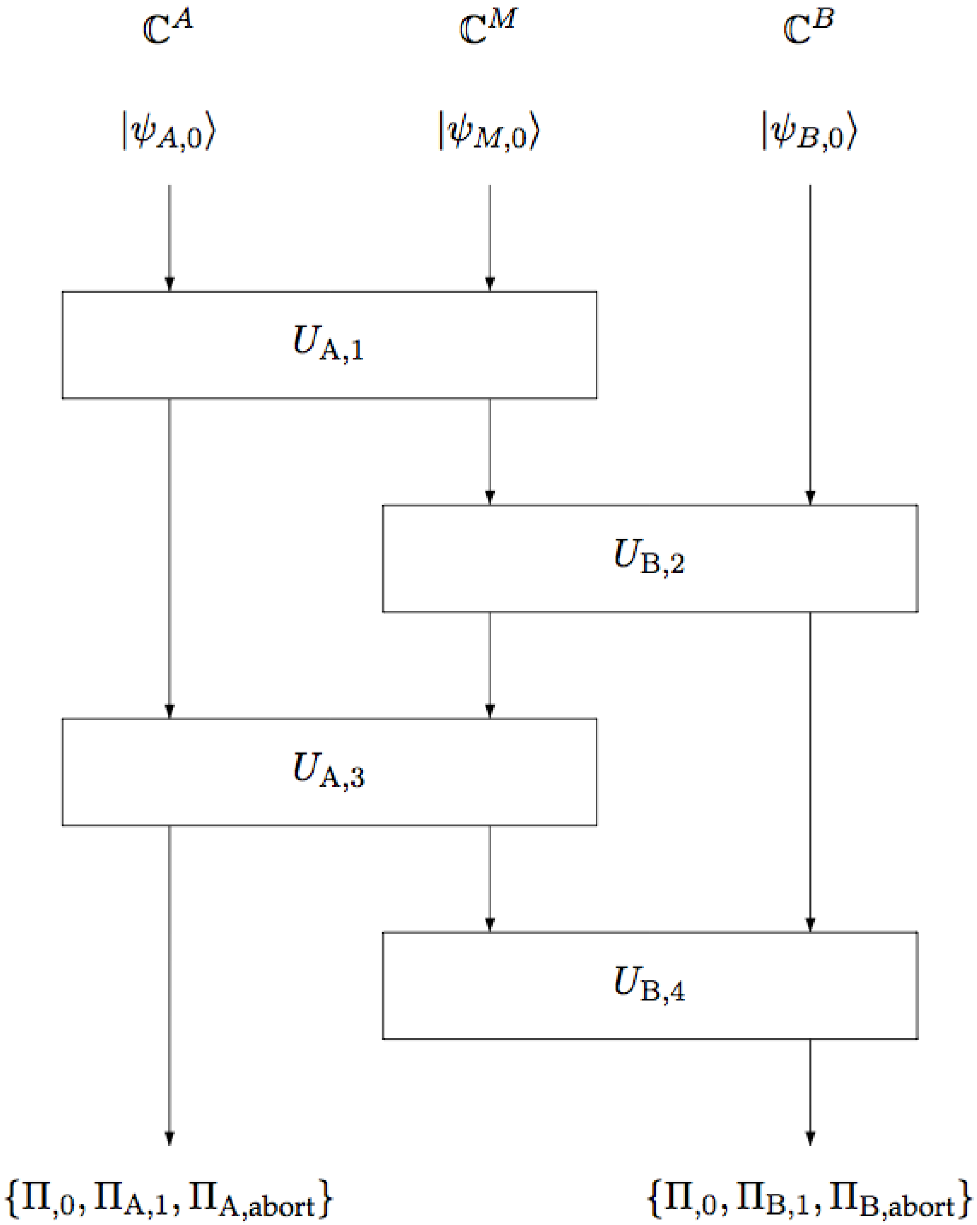


Figure 1.2: Four-message coin-flipping protocol setting.

Cheating SDPs

We can calculate the extent cheating Bob can force honest Alice to output a fixed desired outcome, say $c \in \{0, 1\}$, by solving the following SDP:

$$\begin{array}{rrcll}
P^*_{\mathrm{B},c} = \max & \langle \Pi_{\mathrm{A},c}, \rho_{A,n} \rangle & & & \\
\text{subject to} & \rho_{A,0} & = & |\psi_{A,0}\rangle\langle\psi_{A,0}|, & \\
& \rho_{A,i} & = & \rho_{A,i-1}, & \text{for all } i \text{ even}, \\
& \mathrm{Tr}_M \tilde{\rho}_{A,i} & = & \rho_{A,i}, & \text{for all } i \text{ even}, \\
& \rho_{A,i} & = & \mathrm{Tr}_M \left[U_{\mathrm{A},i} \tilde{\rho}_{A,i-1} U^*_{\mathrm{A},i} \right], & \text{for all } i \text{ odd}, \\
& \rho_{A,i} & \in & \Sigma^A_+, & \text{for all } i, \\
& \tilde{\rho}_{A,i} & \in & \Sigma^{A \times M}_+, & \text{for all } i \text{ even}.
\end{array}$$

The variables describe the parts of the quantum state under Alice's control during different times in the protocol as depicted in Figure 1.3. The constraints model how much cheating Bob can change the current state of the protocol in each message and the objective function is the probability Alice accepts outcome $c \in \{0, 1\}$ by measuring the state she has at the end of the protocol.

We get a very similar SDP for cheating Alice by switching the projections and interchanging the "odd" constraints with the "even" ones:

$$\begin{array}{rrcll}
P^*_{\mathrm{A},c} = \max & \langle \Pi_{\mathrm{B},c}, \rho_{B,n} \rangle & & & \\
\text{subject to} & \rho_{B,0} & = & |\psi_{B,0}\rangle\langle\psi_{B,0}|, & \\
& \rho_{B,i} & = & \rho_{B,i-1}, & \text{for all } i \text{ odd}, \\
& \mathrm{Tr}_M \tilde{\rho}_{B,i} & = & \rho_{B,i}, & \text{for all } i \text{ odd}, \\
& \rho_{B,i} & = & \mathrm{Tr}_M \left[U_{\mathrm{B},i} \tilde{\rho}_{B,i-1} U^*_{\mathrm{B},i} \right], & \text{for all } i \text{ even}, \\
& \rho_{B,i} & \in & \Sigma^B_+, & \text{for all } i, \\
& \tilde{\rho}_{B,i} & \in & \Sigma^{M \times B}_+, & \text{for all } i \text{ odd}.
\end{array}$$

The variables for a cheating Alice are also depicted in Figure 1.3. These SDPs are referred to as Alice and Bob's cheating SDPs.

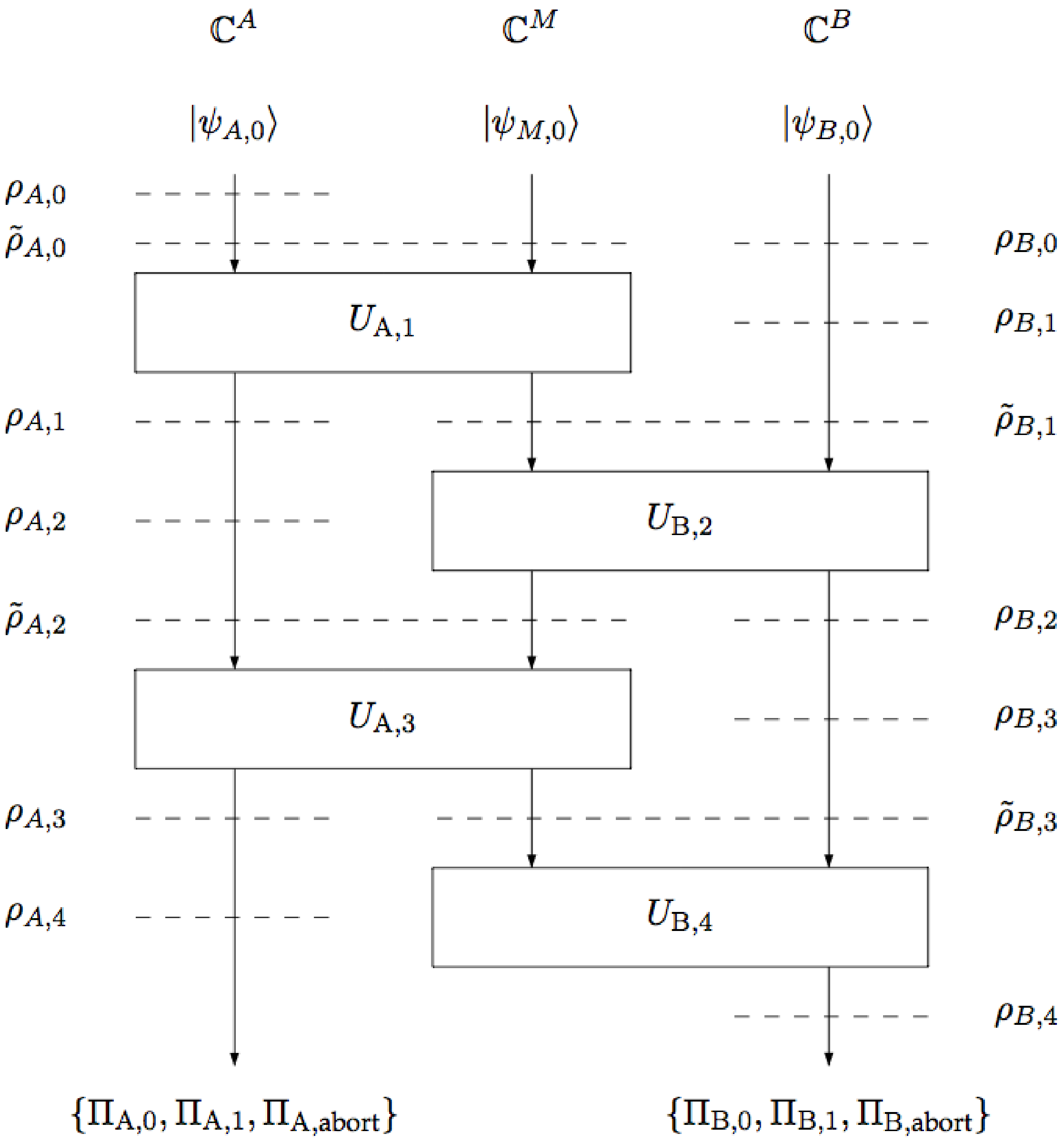


Figure 1.3: Context of primal variables in a four-message coin-flipping protocol.

The duals of the above SDPs are as follows:

$$\begin{array}{rrcll} \inf & \multicolumn{4}{l}{\langle Z_{A,0}, |\psi_{A,0}\rangle\langle\psi_{A,0}|\rangle} \\ \text{subject to} & Z_{A,i-1} \otimes \mathrm{I}_M & \succeq & U^*_{\mathrm{A},i}(Z_{A,i} \otimes \mathrm{I}_M)U_{\mathrm{A},i}, & \text{for all } i \text{ odd}, \\ & Z_{A,i-1} & = & Z_{A,i}, & \text{for all } i \text{ even}, \\ & Z_{A,n} & = & \Pi_{\mathrm{A},c}, & \end{array}$$

and

$$\begin{array}{rrcll} \inf & \multicolumn{4}{l}{\langle Z_{B,0}, |\psi_{B,0}\rangle\langle\psi_{B,0}|\rangle} \\ \text{subject to} & Z_{B,i-1} \otimes \mathrm{I}_M & \succeq & U^*_{\mathrm{B},i}(Z_{B,i} \otimes \mathrm{I}_M)U_{\mathrm{B},i}, & \text{for all } i \text{ even}, \\ & Z_{B,i-1} & = & Z_{B,i}, & \text{for all } i \text{ odd}, \\ & Z_{B,n} & = & \Pi_{\mathrm{B},c}. & \end{array}$$

Kitaev's lower bound for strong coin-flipping

We can derive a lower bound on the bias of any strong coin-flipping protocol by examining feasible dual solutions. Since the dual SDPs have strictly feasible solutions and the objective function is bounded on the feasible region, there is zero duality gap. Therefore, for any $\delta > 0$, we can find feasible dual solutions $(Z_{B,0}, \ldots, Z_{B,n})$ and $(Z_{A,0}, \ldots, Z_{A,n})$, such that

$$P^*_{\mathrm{A},0} + \delta > \langle Z_{B,0}, |\psi_{B,0}\rangle\langle\psi_{B,0}|\rangle \quad \text{and} \quad P^*_{\mathrm{B},0} + \delta > \langle Z_{A,0}, |\psi_{A,0}\rangle\langle\psi_{A,0}|\rangle.$$

Therefore, we have

$$\begin{array}{rcl} \left(P^*_{\mathrm{B},0} + \delta\right)\left(P^*_{\mathrm{A},0} + \delta\right) & > & \langle Z_{B,0}, |\psi_{B,0}\rangle\langle\psi_{B,0}|\rangle\langle Z_{A,0}, |\psi_{A,0}\rangle\langle\psi_{A,0}|\rangle \\ & = & \langle Z_{A,0} \otimes \mathrm{I}_M \otimes Z_{B,0}, |\psi_{B,0}\rangle\langle\psi_{B,0}| \otimes |\psi_{M,0}\rangle\langle\psi_{M,0}| \otimes |\psi_{A,0}\rangle\langle\psi_{A,0}|\rangle \\ & = & \langle Z_{A,0} \otimes \mathrm{I}_M \otimes Z_{B,0}, |\psi_0\rangle\langle\psi_0|\rangle, \end{array}$$

where we define $|\psi_i\rangle$ to be the state after Bob applies $U_{\mathrm{B},i}$ in an honest run of the protocol, for $i \in \{1, \ldots, n\}$. From the dual constraints, we have

$$\begin{array}{rcl} \langle Z_{A,i} \otimes \mathrm{I}_M \otimes Z_{B,i}, |\psi_i\rangle\langle\psi_i|\rangle & \geq & \langle U^*_{\mathrm{A},i+1}(Z_{A,i+1} \otimes \mathrm{I}_M)U_{\mathrm{A},i+1} \otimes Z_{B,i}, |\psi_i\rangle\langle\psi_i|\rangle \\ & = & \langle Z_{A,i+1} \otimes \mathrm{I}_M \otimes Z_{B,i}, U_{\mathrm{A},i+1}|\psi_i\rangle\langle\psi_i|U^*_{\mathrm{A},i+1}\rangle \\ & \geq & \langle Z_{A,i+1} \otimes U^*_{\mathrm{B},i+1}(\mathrm{I}_M \otimes Z_{B,i+1})U_{\mathrm{B},i+1}, U_{\mathrm{A},i+1}|\psi_i\rangle\langle\psi_i|U^*_{\mathrm{A},i+1}\rangle \\ & = & \langle Z_{A,i+1} \otimes \mathrm{I}_M \otimes Z_{B,i+1}, |\psi_{i+1}\rangle\langle\psi_{i+1}|\rangle, \end{array}$$

for all $i \in \{1, \ldots, n\}$. We can compute

$$\langle Z_{A,n} \otimes \mathrm{I}_M \otimes Z_{B,n}, |\psi_n\rangle\langle\psi_n| \rangle = \langle \Pi_{\mathrm{A},0} \otimes \mathrm{I}_M \otimes \Pi_{\mathrm{B},0}, |\psi_n\rangle\langle\psi_n| \rangle = 1/2,$$

from condition (1.2). Taking the limit as $\delta \to 0$, we get

$$P^*_{\mathrm{B},0} P^*_{\mathrm{A},0} \geq \frac{1}{2} \implies \max\left\{P^*_{\mathrm{B},0}, P^*_{\mathrm{A},0}\right\} \geq \frac{1}{\sqrt{2}} \implies \varepsilon \geq \frac{1}{\sqrt{2}} - \frac{1}{2}.$$

This lower bound was later reproven by Gutoski and Watrous [GW07] using a different representation of quantum strategies.

Notice that we can reproduce the proof above using dual feasible solutions for Bob cheating towards 1 and Alice cheating towards 0. In this case, we get the final condition

$$\langle Z_{A,n} \otimes \mathrm{I}_M \otimes Z_{B,n}, |\psi_n\rangle\langle\psi_n| \rangle = \langle \Pi_{\mathrm{A},0} \otimes \mathrm{I}_M \otimes \Pi_{\mathrm{B},1}, |\psi_n\rangle\langle\psi_n| \rangle = 0.$$

This gives a trivial bound on the product of the cheating probabilities. However, Kitaev used this to create point games. We refer the reader to [Moc07] for the full details of the construction of general point games as the details are not needed for this thesis. However, we discuss the construction of point games for specific protocols (which we later apply to the family of protocols in Chapter 2).

Point games

Let $\mathrm{eig}(Z)$ denote the set of eigenvalues for an operator Z and let $\Pi_Z^{[\lambda]}$ denote the projection onto the eigenspace of Z corresponding to eigenvalue $\lambda \in \mathrm{eig}(Z)$. For a quantum state $\sigma \in \Sigma^n_+$, and $X, Y \in \Sigma^n_+$, denote by $\mathrm{Prob}(X, Y, \sigma) : \mathbb{R}^2 \to \mathbb{R}_+$ the function

$$\mathrm{Prob}(X, Y, \sigma) := \sum_{\lambda \in \mathrm{eig}(X)} \sum_{\mu \in \mathrm{eig}(Y)} \langle \Pi_X^{[\lambda]} \otimes \Pi_Y^{[\mu]}, \sigma \rangle \; [\lambda, \mu],$$

where we use the notation $[\lambda, \mu] : \mathbb{R}^2 \to \mathbb{R}$ to denote the function that takes value 1 on input (λ, μ) and 0 otherwise. Note this function has finite support. Using this definition, we can create a point game from feasible dual variables as follows

$$p_{n-i} := \mathrm{Prob}(Z_{B,i}, Z_{A,i}, \mathrm{Tr}_M |\psi_i\rangle\langle\psi_i|),$$

where $|\psi_i\rangle \in \mathbb{C}^{A \times M \times B}$ is the state after Bob applies $U_{\mathrm{B},i}$ in an honest run of the protocol. Consider the dual SDPs for weak coin-flipping, i.e., Bob trying to force outcome 1 and Alice trying to force outcome 0. Then any dual feasible solution yields a point game. We can calculate $p_0 = \frac{1}{2}[0,1] + \frac{1}{2}[1,0]$, which acts as the starting point of the point game. Notice for any $\delta > 0$, there exists a large constant Λ such that

$$Z_{A,0}(\delta) := (\langle \psi_{A,0}|Z_{A,0}|\psi_{A,0}\rangle + \delta)\,|\psi_{A,0}\rangle\langle\psi_{A,0}| + \Lambda\,(\mathrm{I} - |\psi_{A,0}\rangle\langle\psi_{A,0}|) \succeq Z_{A,0},$$

which can be proved using the Schur complement after writing $Z_{A,0}$ in a basis containing $|\psi_{A,0}\rangle$. Notice $(Z_{A,0}(\delta), Z_{A,1}, \ldots, Z_{A,n})$ is feasible if $(Z_{A,0}, Z_{A,1}, \ldots, Z_{A,n})$ is feasible and has the same objective function value as $\delta \to 0$. If we replace $Z_{A,0}$ with $Z_{A,0}(\delta)$, and replace $Z_{B,0}$ with the properly modified definition of $Z_{B,0}(\delta)$, we get that the final point is

$$p_n = 1\,[Z_{A,0} + \delta, Z_{B,0} + \delta]\,.$$

By strong duality, we see that we can choose the dual feasible solutions such that this final point gets arbitrarily close to $\left[P^*_{\mathrm{B},0}, P^*_{\mathrm{A},0}\right]$.

A point game $p_0 \to p_1 \to \cdots \to p_n$ with final point $(\zeta_{\mathrm{B}}, \zeta_{\mathrm{A}})$ can be defined independent of protocols. Define $[x] : \mathbb{R} \to \mathbb{R}$ to be the function that takes value 1 on input x and equals 0, otherwise. Then $p_0 \to p_1 \to \cdots \to p_n$ is a point game if each p_i is a function with finite support, $p_0 = \frac{1}{2}[0,1] + \frac{1}{2}[1,0]$, $p_n = 1\,[\zeta_{\mathrm{B}}, \zeta_{\mathrm{A}}]$, and the moves (or transitions) $p_i \to p_{i+1}$ have one of the following forms

- $p_i = \sum_{a \in A} p_{i,a}\,[x_a, y] \to p_{i+1} = \sum_{b \in B} p_{i+1,b}\,[z_b, y]$ (called a horizontal move),
- $p_i = \sum_{a \in A} p_{i,a}\,[y, x_a] \to p_{i+1} = \sum_{b \in B} p_{i+1,b}\,[y, z_b]$ (called a vertical move),

where $\sum_{a \in A} p_{i,a} = \sum_{b \in B} p_{i+1,b}$ (conservation of probability) and

$$\sum_{b \in B} p_{i+1,b}[z_b] - \sum_{a \in A} p_{i,a}[x_a] \in \mathrm{OMF}^*,$$

where OMF is the cone of operator monotone functions. The purpose of the second condition is beyond the scope of this thesis, but it is used to prove that if there is a point game with final point $(\zeta_{\rm B}, \zeta_{\rm A})$, then there exists a coin-flipping protocol with $P^*_{\rm B,1} \leq \zeta_{\rm B} + \delta$ and $P^*_{\rm A,0} \leq \zeta_{\rm A} + \delta$, for any $\delta > 0$ [Moc07]. Mochon proved that there exists a point game with final point $(1/2 + \delta, 1/2 + \delta)$, for any $\delta > 0$, proving the existence of weak coin-flipping protocols with arbitrarily small bias.

1.4.7 Technical lemmas

In this part, we present a few lemmas which are helpful during the analysis in this thesis.

Subspace Lemma

This first lemma is useful when developing the point games for the family of protocols discussed in the next chapter. It is used to simplify dual constraints.

Lemma 1.4.9 (Subspace Lemma). *For a vector $|\psi\rangle \in \mathbb{C}^n$, a set $S \subseteq \Sigma^n$, and a continuous, monotonically nondecreasing function F, we have*

$$\inf_{X,Y \in \Sigma^n} \{F(\langle\psi|X|\psi\rangle) : X \succeq Y,\, Y \in S\} = \inf_{X,Y \in \Sigma^n} \{F(\langle\psi|X|\psi\rangle) : \langle\psi|X|\psi\rangle \geq \langle\psi|Y|\psi\rangle,\, Y \in S\}.$$

Proof. The proof follows by noticing that for any $X \in \Sigma^A$, and any $\delta > 0$, there is a positive constant Λ such that

$$X(\delta) := \left(\langle\psi|X|\psi\rangle + \delta\right)|\psi\rangle\langle\psi| + \Lambda\left(\mathrm{I} - |\psi\rangle\langle\psi|\right) \succeq X,$$

as noted in Subsection 1.4.6. Thus, we can assume X has $|\psi\rangle$ as an eigenvector and we are only concerned with minimizing X on the subspace spanned by $|\psi\rangle\langle\psi|$. Suppose $X \succeq Y$, we see this implies

$$\langle\psi|X|\psi\rangle \geq \langle\psi|Y|\psi\rangle. \tag{1.3}$$

This is also a sufficient condition since any X that satisfies (1.3), we have for all $\delta > 0$, there is a Λ, such that $X(\delta) \succeq Y$ (by the same argument showing $X(\delta) \succeq X$). □

This lemma can be generalized. We can use this lemma whenever the constraint on X is satisfied by replacing it with $X(\delta)$ for $\delta > 0$. The most complicated constraints that arise later in this thesis are of the form

$$\sum_{x \in A} W_{x,y} \otimes |x\rangle\langle x| \otimes \mathrm{I}_B \succeq C,$$

where $W_{x,y}$ are the variables and the objective function is continuous and nondecreasing on $\langle \phi | W_{x,y} | \phi \rangle$. We see that a necessary condition is

$$\sum_{x \in A} \langle \phi | W_{x,y} | \phi \rangle \cdot |x\rangle\langle x| \otimes \mathrm{I}_B \succeq (\langle \phi | \otimes \mathrm{I}_A \otimes \mathrm{I}_B)\, C\, (|\phi\rangle \otimes \mathrm{I}_A \otimes \mathrm{I}_B).$$

By using a properly modified definition for $W_{x,y}(\delta)$, we have that this condition is also sufficient. The idea is to increase the eigenvalues on subspaces that do not affect the objective function.

SDP characterization of the fidelity of probability vectors

We characterize the fidelity of probability vectors using semidefinite programming. Note that this is very closely related to the analysis of the fidelity function of quantum states in [Wat09]. However, the primal SDP is slightly different and the details of the proofs are useful later in this thesis.

Lemma 1.4.10. *For any $p, q \in \mathbb{R}^A_+$, we have*

$$\mathrm{F}(p,q) = \max\{\langle X, \sqrt{p}\sqrt{p}^\top \rangle : \mathrm{diag}(X) = q,\ X \in \Sigma^A_+\}.$$

Proof. Notice that $\bar{X} := \sqrt{q}\sqrt{q}^\top$ is a feasible solution to the SDP with objective function value $\mathrm{F}(p,q)$. All that remains to show is that it is an optimal solution. If $p = 0$, then we are done, so assume $p \neq 0$. The dual can be written as

$$\inf\{\langle y, q\rangle : \mathrm{Diag}(y) \succeq \sqrt{p}\sqrt{p}^\top, y \in \mathbb{R}^A\}.$$

Define y, as a function of $\varepsilon > 0$, entry-wise as

$$y_a(\varepsilon) := \begin{cases} (\sqrt{\mathrm{F}(p,q)} + \varepsilon)\frac{\sqrt{p_a}}{\sqrt{q_a}} & \text{if} \quad p_a, q_a > 0, \\ \frac{(\sqrt{\mathrm{F}(p,q)}+\varepsilon)\|p\|_1}{\varepsilon} & \text{if} \quad q_a = 0, \\ \varepsilon & \text{if} \quad p_a = 0, q_a > 0. \end{cases}$$

We can check that $\langle y(\varepsilon), q\rangle \to \mathrm{F}(p,q)$ as $\varepsilon \to 0$, so it suffices to show that $y(\varepsilon)$ is dual feasible for all $\varepsilon > 0$. To show this, we use the following trick. For any $y > 0$,

$$\begin{aligned} \mathrm{Diag}(y) \succeq \sqrt{p}\sqrt{p}^\top \quad &\iff \quad \mathrm{I}_A \succeq \mathrm{Diag}(y)^{-1/2}\sqrt{p}\sqrt{p}^\top\mathrm{Diag}(y)^{-1/2} \\ &\iff \quad 1 \geq \sqrt{p}^\top \mathrm{Diag}(y)^{-1}\sqrt{p} \\ &\iff \quad 1 \geq \sum_{a \in A} \frac{p_a}{y_a}, \end{aligned}$$

noting $\mathrm{Diag}(y)^{-1/2}\sqrt{p}\sqrt{p}^\top\mathrm{Diag}(y)^{-1/2}$ is rank 1 so the largest eigenvalue is equal to its trace. From this, we can check that $y(\varepsilon)$ is feasible for all sufficiently small $\varepsilon > 0$. □

The proof above shows that

$$\begin{aligned}\inf\{\langle y, q\rangle : \mathrm{Diag}(y) \succeq \sqrt{p}\sqrt{p}^\top, y \in \mathbb{R}^A\} &= \inf\{\langle y, q\rangle : \langle y^{-1}, p\rangle \leq 1, y > 0\} \\ &= \inf_{y>0}\{\langle y, q\rangle\langle y^{-1}, p\rangle\},\end{aligned}$$

which is the classical version of Alberti's Theorem [Alb83], which states that

$$\mathrm{F}(\rho, \sigma) = \inf_{X\succ 0} \langle X, \rho\rangle\langle X^{-1}, \sigma\rangle,$$

for any quantum states ρ and σ.

From the proof of Lemma 1.4.10, we see that we can apply the same trick to the inequality

$$\mathrm{Diag}(y) \otimes \mathrm{I}_A \succeq |\psi\rangle\langle\psi| \quad (y > 0)$$

to get the equivalent condition

$$1 \geq \langle\psi|\mathrm{Diag}(y)^{-1} \otimes \mathrm{I}_A|\psi\rangle,$$

which works for any $|\psi\rangle \in \mathbb{C}^{A\times A}$. In particular, we have the following lemma.

Lemma 1.4.11. *For any $p \in \mathbb{R}^A_+$, we have that*

$$\{y > 0 : \mathrm{Diag}(y) \succeq \sqrt{p}\sqrt{p}^\top\} = \{y > 0 : \mathrm{Diag}(y) \otimes \mathrm{I}_A \succeq |\psi\rangle\langle\psi|\},$$

where $|\psi\rangle := \sum_{a\in A} \sqrt{p_a}\,|a\rangle|a\rangle$.

Using Lemma 1.4.10, we can prove fidelity is concave.

Lemma 1.4.12. *The classical fidelity function is concave on the nonnegative orthant.*

Proof. Let p_1, p_2 be two nonnegative vectors and $\lambda \in [0, 1]$. We know from Lemma 1.4.10 that $X_i = \sqrt{p_i}\sqrt{p_i}^\top$ is an optimal solution to

$$\max\{\langle X_i, \sqrt{q}\sqrt{q}^\top\rangle : \mathrm{diag}(X_i) = p_i,\, X_i \succeq 0\},$$

for $i \in \{1, 2\}$. Define $\bar{X} := \lambda X_1 + (1-\lambda) X_2$. Then we have

$$\begin{aligned} \mathrm{F}(\lambda p_1 + (1-\lambda) p_2, q) &= \max\{\langle X, \sqrt{q}\sqrt{q}^\top\rangle : \mathrm{diag}(X) = \lambda p_1 + (1-\lambda) p_2,\ X \succeq 0\} \\ &\geq \langle \bar{X}, \sqrt{q}\sqrt{q}^\top\rangle \quad \text{since } \bar{X} \text{ is feasible} \\ &= \lambda\,\mathrm{F}(p_1, q) + (1-\lambda)\,\mathrm{F}(p_2, q), \end{aligned}$$

proving it is concave. □

Note, we can also prove it is concave by showing the Hessian is negative semidefinite on the interior of the nonnegative orthant. Since it is not negative definite, we know fidelity is not strictly concave. For example, choose

$$\lambda := \frac{1}{2}, \quad p_1 := \begin{bmatrix} 1 \\ 0 \\ 0 \end{bmatrix}, \quad p_2 := \begin{bmatrix} 0 \\ 1 \\ 0 \end{bmatrix}, \quad \text{and} \quad q := \begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix}.$$

Then

$$0 = \mathrm{F}(\lambda p_1 + (1-\lambda) p_2, q) \not> \lambda\,\mathrm{F}(p_1, q) + (1-\lambda)\,\mathrm{F}(p_2, q) = 0.$$

The concavity of the fidelity function of quantum states can be proved using Uhlmann's Theorem [Uhl76], which states that

$$\mathrm{F}(\rho, \sigma) = \max_{A, |\psi\rangle, |\phi\rangle} |\langle\psi|\phi\rangle|^2,$$

where $\mathrm{Tr}_A |\psi\rangle\langle\psi| = \rho$ and $\mathrm{Tr}_A |\phi\rangle\langle\phi| = \sigma$.

Largest eigenvalue and trace distance characterizations

Lemma 1.4.13. *For $\eta, \tau \in \mathbb{R}$ and $p, q \in \mathrm{Prob}^n$, we have*

$$\lambda_{\max}\left(\eta\sqrt{p}\sqrt{p}^\top + \tau\sqrt{q}\sqrt{q}^\top\right) = \frac{1}{2}\left(\eta + \tau + \sqrt{(\eta-\tau)^2 + 4\eta\tau\,\mathrm{F}(p, q)}\right).$$

Proof. Since we can write $\mathrm{F}(p, q) = \left(\sqrt{p}^\top\sqrt{q}\right)^2$, we can apply a unitary to both $\sqrt{p}$ and $\sqrt{q}$ and both sides of the equality we want to prove are unaffected. Choose a unitary U such that

$$U\sqrt{p} = [1, 0, 0, \ldots, 0]^\top \quad \text{and} \quad U\sqrt{q} = [\sin\theta, \cos\theta, 0, \ldots, 0]^\top,$$

for some $\theta \in [0, 2\pi)$. Then we can write $\mathrm{F}(p, q) = \sin^2\theta$. Let λ_2 be the second largest eigenvalue of $\eta\sqrt{p}\sqrt{p}^\top + \tau\sqrt{q}\sqrt{q}^\top$, or equivalently, of $\eta U\sqrt{p}\sqrt{p}^\top U^* + \tau U\sqrt{q}\sqrt{q}^\top U^*$. Then

$$\lambda_{\max} + \lambda_2 = \mathrm{Tr}(\eta\sqrt{p}\sqrt{p}^\top + \tau\sqrt{q}\sqrt{q}^\top) = \eta + \tau$$

and, by taking the determinant of the only nonzero block, we get

$$\lambda_{\max} \cdot \lambda_2 = \eta\tau\cos^2\theta = \eta\tau(1 - \mathrm{F}(p, q))$$

implying $\lambda_{\max} = \frac{1}{2}\left(\eta + \tau + \sqrt{(\eta-\tau)^2 + 4\eta\tau\mathrm{F}(p,q)}\right)$, as desired. □

Note that Lemma 1.4.13 shows that switching the roles of η and τ does not affect the largest eigenvalue.

Lemma 1.4.14. *For $\beta_0, \beta_1 \in \mathrm{Prob}^B$, we have*

$$\sum_{y\in B} \max_{a\in\{0,1\}} \{\beta_{a,y}\} = 1 + \Delta(\beta_0, \beta_1).$$

Proof. Notice that

$$\sum_{y\in B} \max_{a\in\{0,1\}} \{\beta_{a,y}\} + \sum_{y\in B} \min_{a\in\{0,1\}} \{\beta_{a,y}\} = 2$$

and

$$\sum_{y\in B} \max_{a\in\{0,1\}} \{\beta_{a,y}\} - \sum_{y\in B} \min_{a\in\{0,1\}} \{\beta_{a,y}\} = 2\Delta(\beta_0, \beta_1).$$

Putting it together, we have

$$2 + 2\Delta(\beta_0, \beta_1) = 2\sum_{y\in B} \max_{a\in\{0,1\}} \{\beta_{a,y}\},$$

as desired. □

Chapter 2

Coin-flipping protocols based on bit-commitment

The content of this chapter is based on my work with Ashwin Nayak and Levent Tunçel.

We now turn our attention to analyzing a family of quantum coin-flipping protocols which are based on bit-commitment. This type of protocol was first considered in [NS03] as a way to blend two bit-commitment protocols together to create a strong coin-flipping protocol. This style of protocol is also similar to the weak coin-flipping protocols in [Moc05] which are analyzed using semidefinite programming. Like the analysis in [Moc05], this chapter is concerned with gaining a better understanding of a family of coin-flipping protocols by studying their cheating SDPs.

The structure of this chapter is as follows:

- Section 2.1: We describe a family of coin-flipping protocols based on bit-commitment. We call these BCCF-protocols.

- Section 2.1.1: We formulate the cheating strategies as SDPs.

- Section 2.1.2: We prove that we can simplify the SDPs by considering specific strategies for each party. This results in the cheating probabilities being a maximum of a linear combination of fidelity functions over probability vectors constrained to be in a certain polytope. We call these problems the *reduced* SDPs or *reduced problems.*

- Section 2.1.3: We show an immediate consequence of examining the reduced SDPs, that we can assume each message consists of one qubit.

- Section 2.1.4: We show the simple nature of the reduced SDPs by modelling them as SOCPs. That is, we can characterize the bias using a simpler class of optimization problems.

- Section 2.1.5: Using the reduced SDPs and their duals, we develop the point games corresponding to BCCF-protocols. We give a protocol-independent definition of "BCCF-point games" and use it to define "BCCF-point game pairs" which is an analog for strong coin-flipping. This leads to a relationship between the cheating probabilities in BCCF-protocols and the "final points" of BCCF-point game pairs.

- Section 2.2: This section describes a family of "classical BCCF-protocols." Using LP formulations of cheating strategies, we develop their point games. Throughout the process, we see how closely connected these protocols and point games are to their quantum counterparts, and how the analysis is very similar. We use the point games to (re)prove these classical protocols have bias $\varepsilon = 1/2$.

- Section 2.3: We present modified versions of Kitaev's lower bounds for BCCF-protocols that arise from the reduced cheating SDPs. We then show that if the lower bounds are saturated, then the protocol is "classical" in a sense described later. This rules out the possibility of quantum BCCF-protocols attaining a bias of $\varepsilon = 1/\sqrt{2} - 1/2$ and shows a deeper connection to the classical version.

2.1 A family of quantum coin-flipping protocols

We now describe a family of protocols where Alice and Bob each choose a random bit and commit to their respective bits by exchanging quantum states. Then they reveal their bits, and send the remaining part of their quantum states for cheat detection purposes. Each party checks the received state against the state claimed and, if both parties accept the revealed values, the outcome of the protocol is the XOR of the committed bits.

The difficulty in designing a good protocol is in deciding how Alice and Bob should commit their bits to each other. If Alice or Bob leak too much information too early, then the other party has more information to form a cheating strategy. Thus, we try to maintain a balance between the two parties so as to minimize the bias they can achieve by cheating.

Consider the following Cartesian product of finite sets $A = A_1 \times \cdots \times A_n$ corresponding to Alice's first n messages, and $\alpha_0, \alpha_1 \in \text{Prob}^A$. Define the following two quantum states

$$|\psi_a\rangle := \sum_{x \in A} \sqrt{\alpha_{a,x}}\, |xx\rangle \in \mathbb{C}^{A \times A'} \quad \text{for} \quad a \in \{0,1\},$$

where A' is a copy of A. The reason we define the state over $\mathbb{C}^A$ and a copy is because in the protocol, Alice sends states in $\mathbb{C}^A$ while retaining copies in $\mathbb{C}^{A'}$ until the end of the protocol. Since Alice chooses $a \in \{0,1\}$ uniformly at random, we can represent her initial state as

$$|\psi\rangle := \sum_{a \in \{0,1\}} \frac{1}{\sqrt{2}} |aa\rangle \otimes |\psi_a\rangle \in \mathbb{C}^{A_0 \times A_0' \times A \times A'},$$

where $A_0 = A_0' = \{0,1\}$. This yields the space $\mathbb{C}^{A_0 \times A_0'}$ to hold two copies of Alice's bit $a \in \{0,1\}$.

We now describe the setting for Bob's messages. Consider the following Cartesian product of finite sets $B = B_1 \times \cdots \times B_n$ corresponding to Bob's first n messages to Alice, and $\beta_0, \beta_1 \in \text{Prob}^B$. Define the following two quantum states

$$|\phi_b\rangle := \sum_{y \in B} \sqrt{\beta_{b,y}}\, |yy\rangle \in \mathbb{C}^{B \times B'} \quad \text{for} \quad b \in \{0,1\},$$

where B' is a copy of B. Since Bob chooses $b \in \{0,1\}$ uniformly at random, we can represent his initial state as

$$|\phi\rangle := \sum_{b \in \{0,1\}} \frac{1}{\sqrt{2}} |bb\rangle \otimes |\phi_b\rangle \in \mathbb{C}^{B_0 \times B_0' \times B \times B'},$$

where $B_0 = B_0' = \{0,1\}$. The space $\mathbb{C}^{B_0 \times B_0'}$ holds two copies of Bob's bit $b \in \{0,1\}$. We now describe the steps of the protocol.

Protocol 2.1.1 (BCCF-protocol)**.**

(i) *Alice prepares the state $|\psi\rangle$ and Bob prepares the state $|\phi\rangle$, as described above.*

(ii) *For i from 1 to n: Alice sends $\mathbb{C}^{A_i}$ to Bob who replies with $\mathbb{C}^{B_i}$.*

(iii) *Alice fully reveals her bit by sending* $\mathbb{C}^{A'_0}$. *She also sends* $\mathbb{C}^{A'}$ *which Bob uses to check if she was honest. Bob then reveals his bit by sending* $\mathbb{C}^{B'_0}$. *He also sends* $\mathbb{C}^{B'}$ *which Alice uses to check if he was honest.*

(iv) *Alice applies the measurement* $\{\Pi_{\mathrm{A},0}, \Pi_{\mathrm{A},1}, \Pi_{\mathrm{A,abort}}\} \subset \Sigma_+^{A_0 \times B'_0 \times B \times B'}$, *where*

$$\begin{aligned}
\Pi_{\mathrm{A},0} &:= \sum_{b \in \{0,1\}} |b\rangle\langle b| \otimes |b\rangle\langle b| \otimes |\phi_b\rangle\langle\phi_b|, \\
\Pi_{\mathrm{A},1} &:= \sum_{b \in \{0,1\}} |\bar{b}\rangle\langle\bar{b}| \otimes |b\rangle\langle b| \otimes |\phi_b\rangle\langle\phi_b|, \qquad \textit{and} \\
\Pi_{\mathrm{A,abort}} &:= \mathrm{I} - \Pi_{\mathrm{A},0} - \Pi_{\mathrm{A},1}.
\end{aligned}$$

Bob applies the measurement $\{\Pi_{\mathrm{B},0}, \Pi_{\mathrm{B},1}, \Pi_{\mathrm{B,abort}}\} \subset \Sigma_+^{B_0 \times A'_0 \times A \times A'}$, *where*

$$\begin{aligned}
\Pi_{\mathrm{B},0} &:= \sum_{a \in \{0,1\}} |a\rangle\langle a| \otimes |a\rangle\langle a| \otimes |\psi_a\rangle\langle\psi_a|, \\
\Pi_{\mathrm{B},1} &:= \sum_{a \in \{0,1\}} |\bar{a}\rangle\langle\bar{a}| \otimes |a\rangle\langle a| \otimes |\psi_a\rangle\langle\psi_a|, \qquad \textit{and} \\
\Pi_{\mathrm{B,abort}} &:= \mathrm{I} - \Pi_{\mathrm{B},0} - \Pi_{\mathrm{B},1}.
\end{aligned}$$

A six-round BCCF-protocol is depicted in Figure 2.1. Note that the measurements check two things. First, it checks whether the outcome, $a \oplus b$, is 0 or 1. The first two terms determine this, i.e., whether $a = b$ or if $a \neq b$. Second, it checks whether the other party was honest. For example, if Alice's measurement projects onto a space where $b = 0$ and Bob's messages are not equal to $|\phi_0\rangle$, then Alice could detect Bob has cheated and abort.

BCCF-protocols are parameterized by A, B, and four probability vectors α_0, α_1, β_0, β_1. It is a difficult problem to solve for the best choices of these parameters. Indeed, we do not even have an upper bound on the sizes of A or B in an optimal protocol. However, we can solve for the bias of a protocol once these parameters are fixed using the optimization techniques in the next subsection.

We note here how this protocol fits the form discussed in Subsection 1.4.6. One difference is that we "send" the space, rather than swap the state with the message space (to have it swapped into the other parties' private space). Another difference is that the messages in BCCF-protocols could have varying dimension. These types of messages can easily be cast in the form in Subsection 1.4.6, although the extra information clutters the underlying problem.

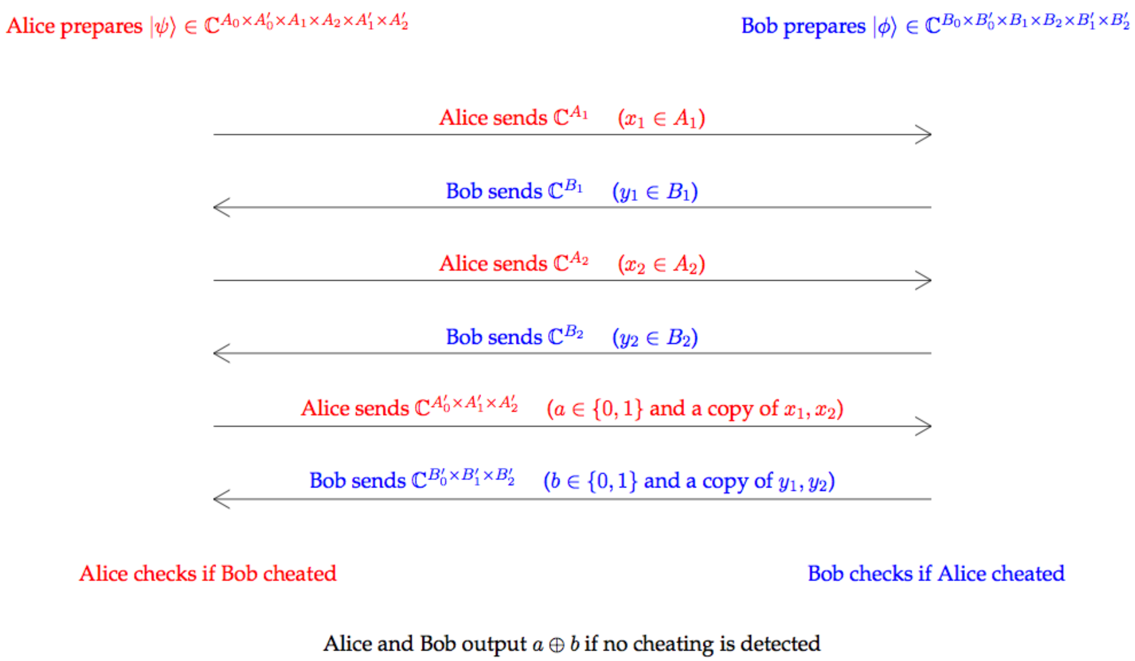

Alice prepares $|\psi\rangle \in \mathbb{C}^{A_0 \times A_0' \times A_1 \times A_2 \times A_1' \times A_2'}$
Bob prepares $|\phi\rangle \in \mathbb{C}^{B_0 \times B_0' \times B_1 \times B_2 \times B_1' \times B_2'}$
Alice sends $\mathbb{C}^{A_1}$ $(x_1 \in A_1)$
Bob sends $\mathbb{C}^{B_1}$ $(y_1 \in B_1)$
Alice sends $\mathbb{C}^{A_2}$ $(x_2 \in A_2)$
Bob sends $\mathbb{C}^{B_2}$ $(y_2 \in B_2)$
Alice sends $\mathbb{C}^{A_0' \times A_1' \times A_2'}$ $(a \in \{0,1\}$ and a copy of $x_1, x_2)$
Bob sends $\mathbb{C}^{B_0' \times B_1' \times B_2'}$ $(b \in \{0,1\}$ and a copy of $y_1, y_2)$
Alice checks if Bob cheated
Bob checks if Alice cheated
Alice and Bob output $a \oplus b$ if no cheating is detected

Figure 2.1: Six-round BCCF-protocol. Alice's actions in red, Bob's actions in blue.

2.1.1 Quantum cheating strategy formulations

We can formulate strategies for cheating Bob and cheating Alice as semidefinite programs in the same manner as Kitaev [Kit02]. The extent to which Bob can cheat is captured by the following lemma.

Lemma 2.1.2. *Bob's optimal cheating probability for forcing honest Alice to accept the outcome $c \in \{0,1\}$ is given by the optimal objective value of the following semidefinite program:*

$$\begin{array}{rrrll}
P^*_{\mathrm{B},c} = \sup & \langle \rho_F, \Pi_{\mathrm{A},c} \rangle & & & \\
\text{subject to} & \mathrm{Tr}_{B_1}(\rho_1) & = & \mathrm{Tr}_{A_1} |\psi\rangle\langle\psi|, & \\
& \mathrm{Tr}_{B_j}(\rho_j) & = & \mathrm{Tr}_{A_j}(\rho_{j-1}), & \forall j \in \{2, \ldots, n\}, \\
& \mathrm{Tr}_{B' \times B_0'}(\rho_F) & = & \mathrm{Tr}_{A' \times A_0'}(\rho_n), & \\
& \rho_j & \in & \Sigma_+^{A_0 \times A_0' \times B_1 \times \cdots \times B_j \times A_{j+1} \times \cdots \times A_n \times A'}, & \forall j \in \{1, \ldots, n\}, \\
& \rho_F & \in & \Sigma_+^{A_0 \times B_0' \times B \times B'}. &
\end{array}$$

The actions of a cheating Bob and the variables in the SDP above are depicted in Figure 2.2 below. Note that the data defining this SDP is real and thus we can restrict ourselves to real matrix variables without loss of generality. This is because the real part of any complex feasible solution is also feasible and has the same objective function value.

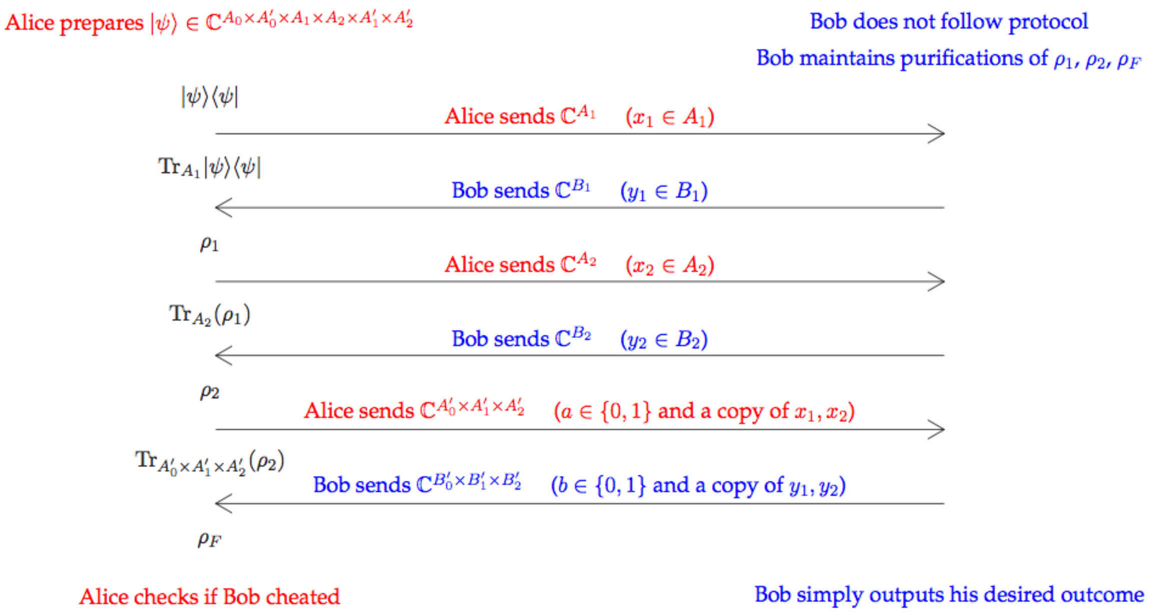

Alice prepares $|\psi\rangle \in \mathbb{C}^{A_0 \times A_0' \times A_1 \times A_2 \times A_1' \times A_2'}$
Bob does not follow protocol
Bob maintains purifications of ρ_1, ρ_2, ρ_F
$|\psi\rangle\langle\psi|$
Alice sends $\mathbb{C}^{A_1}$ $(x_1 \in A_1)$
$\mathrm{Tr}_{A_1}|\psi\rangle\langle\psi|$
Bob sends $\mathbb{C}^{B_1}$ $(y_1 \in B_1)$
ρ_1
Alice sends $\mathbb{C}^{A_2}$ $(x_2 \in A_2)$
$\mathrm{Tr}_{A_2}(\rho_1)$
Bob sends $\mathbb{C}^{B_2}$ $(y_2 \in B_2)$
ρ_2
Alice sends $\mathbb{C}^{A_0' \times A_1' \times A_2'}$ ($a \in \{0,1\}$ and a copy of x_1, x_2)
$\mathrm{Tr}_{A_0' \times A_1' \times A_2'}(\rho_2)$
Bob sends $\mathbb{C}^{B_0' \times B_1' \times B_2'}$ ($b \in \{0,1\}$ and a copy of y_1, y_2)
ρ_F
Alice checks if Bob cheated
Bob simply outputs his desired outcome

Figure 2.2: Bob cheating in a six-round BCCF-protocol. Alice's actions in red, Bob's actions in blue.

Proof. The variables are the density matrices under Alice's control after each of Bob's messages. The partial trace is trace-preserving, so any feasible solution satisfies

$$\mathrm{Tr}(\rho_F) = \mathrm{Tr}(\rho_n) = \cdots = \mathrm{Tr}(\rho_1) = \mathrm{Tr}|\psi\rangle\langle\psi| = 1.$$

Since $\rho_1, \ldots, \rho_n, \rho_F$ are constrained to be positive semidefinite, they are valid quantum states.

Bob sends $\mathbb{C}^{B_1}$ to Alice replacing $\mathbb{C}^{A_1}$ which was sent to Bob. Thus, ρ_1, the density matrix Alice has after Bob's first message, satisfies

$$\mathrm{Tr}_{B_1}(\rho_1) = \mathrm{Tr}_{A_1}|\psi\rangle\langle\psi|.$$

Similarly, for each $j \in \{2, \ldots, n\}$, we have the constraint

$$\mathrm{Tr}_{B_j}(\rho_j) = \mathrm{Tr}_{A_j}(\rho_{j-1}),$$

for each ρ_j after Bob's j'th message, and

$$\mathrm{Tr}_{B' \times B'_0}(\rho_F) = \mathrm{Tr}_{A' \times A'_0}(\rho_n),$$

for ρ_F being the state Alice has at the end of the protocol. She then measures ρ_F and accepts c with probability $\langle \rho_F, \Pi_{A,c} \rangle$.

These constraints are necessary conditions on the states under Alice's control. We now show that every feasible solution to the above problem yields a valid cheating strategy for Bob with success probability equal to the objective function value of the feasible solution. He can find such a strategy by maintaining a purification of each density matrix under Alice's control. For example, suppose the protocol starts in the state $|\psi\rangle \otimes |\phi'\rangle$, where Bob creates $|\phi'\rangle \in \mathbb{C}^K := \mathbb{C}^{B_0 \times B'_0 \times B \times B' \times K'}$ and $\mathbb{C}^{K'}$ is extra space Bob uses to cheat. Consider $|\tau\rangle \in \mathbb{C}^{A_0 \times A'_0 \times A \times A' \times K}$ a purification of ρ_1 and $|\eta\rangle := |\psi\rangle \otimes |\phi'\rangle$, a purification of $|\psi\rangle$. Since $\mathrm{Tr}_{B_1}(\rho_1) = \mathrm{Tr}_{A_1}|\psi\rangle\langle\psi|$, we have

$$\mathrm{Tr}_{A_1 \times K}|\tau\rangle\langle\tau| = \mathrm{Tr}_{B_1}(\rho_1) = \mathrm{Tr}_{A_1}|\psi\rangle\langle\psi| = \mathrm{Tr}_{A_1 \times K}|\eta\rangle\langle\eta|.$$

Thus, there exists a unitary U which acts on $\mathbb{C}^{A_1 \times K}$ which maps $|\tau\rangle$ to $|\eta\rangle$. If Bob applies this unitary after Alice's first message and sends $\mathbb{C}^{B_1}$ back then he creates ρ_1 under Alice's control. The same argument can be applied to the remaining constraints.

The states corresponding to an honest Bob yield a feasible solution. Attainment of an optimal solution follows from continuity of the objective function and from the feasible region being nonempty and compact. An optimal solution yields an optimal cheating strategy. □

In a similar fashion, we formulate cheating strategies for Alice in the lemma below.

Lemma 2.1.3. *Alice's optimal cheating probability for forcing honest Bob to accept the outcome* $c \in \{0, 1\}$ *is given by the optimal objective value of the following semidefinite* *program:*

$$\begin{array}{rrcll}
P^*_{\mathrm{A},c} = \sup & \langle \sigma_F, \Pi_{\mathrm{B},c} \otimes \mathrm{I}_{B'_0 \times B'} \rangle & & & \\
\text{subject to} & \mathrm{Tr}_{A_1}(\sigma_1) & = & |\phi\rangle\langle\phi|, & \\
& \mathrm{Tr}_{A_j}(\sigma_j) & = & \mathrm{Tr}_{B_{j-1}}(\sigma_{j-1}), & \forall j \in \{2, \ldots, n\}, \\
& \mathrm{Tr}_{A' \times A'_0}(\sigma_F) & = & \mathrm{Tr}_{B_n}(\sigma_n), & \\
& \sigma_j & \in & \Sigma_+^{B_0 \times B'_0 \times A_1 \times \cdots \times A_j \times B_j \times \cdots \times B_n \times B'}, & \forall j \in \{1, \ldots, n\}, \\
& \sigma_F & \in & \Sigma_+^{B_0 \times B'_0 \times A'_0 \times A \times A' \times B'}. &
\end{array}$$

The actions of a cheating Alice and the variables in the SDP above are depicted in Figure 2.3. Again, we can restrict ourselves to real matrices without loss of generality.

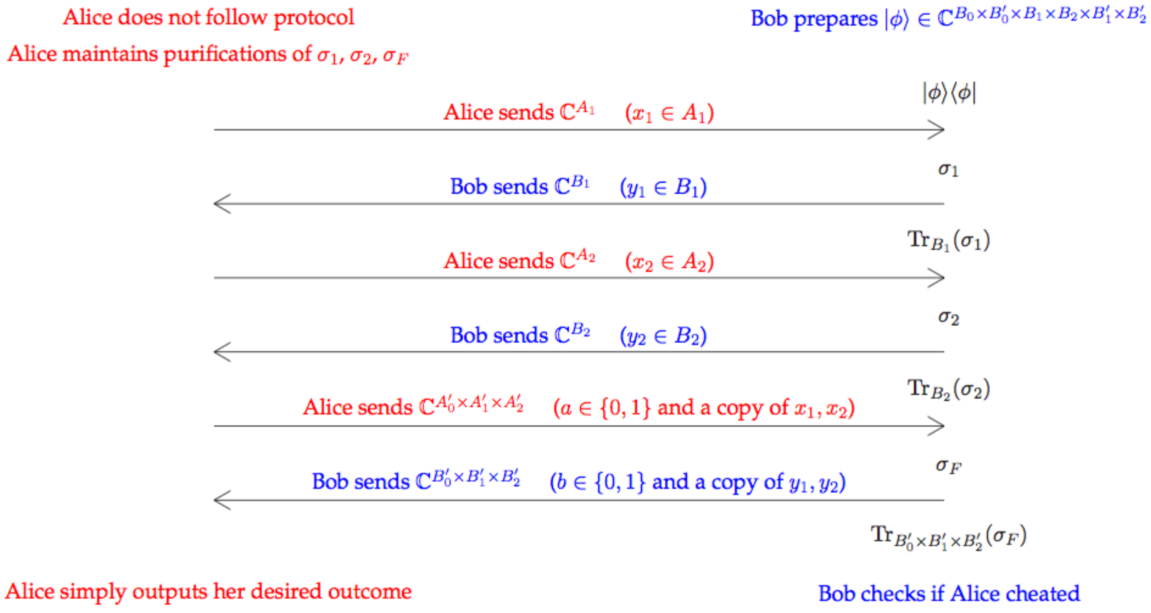

Alice does not follow protocol
Alice maintains purifications of $\sigma_1, \sigma_2, \sigma_F$
Bob prepares $|\phi\rangle \in \mathbb{C}^{B_0 \times B'_0 \times B_1 \times B_2 \times B'_1 \times B'_2}$
$|\phi\rangle\langle\phi|$
Alice sends $\mathbb{C}^{A_1}$ $(x_1 \in A_1)$
σ_1
Bob sends $\mathbb{C}^{B_1}$ $(y_1 \in B_1)$
$\mathrm{Tr}_{B_1}(\sigma_1)$
Alice sends $\mathbb{C}^{A_2}$ $(x_2 \in A_2)$
σ_2
Bob sends $\mathbb{C}^{B_2}$ $(y_2 \in B_2)$
$\mathrm{Tr}_{B_2}(\sigma_2)$
Alice sends $\mathbb{C}^{A'_0 \times A'_1 \times A'_2}$ $(a \in \{0,1\}$ and a copy of $x_1, x_2)$
σ_F
Bob sends $\mathbb{C}^{B'_0 \times B'_1 \times B'_2}$ $(b \in \{0,1\}$ and a copy of $y_1, y_2)$
$\mathrm{Tr}_{B'_0 \times B'_1 \times B'_2}(\sigma_F)$
Alice simply outputs her desired outcome
Bob checks if Alice cheated

Figure 2.3: Alice cheating in a six-round BCCF-protocol. Alice's actions in red, Bob's actions in blue.

Proof. The proof for this is almost exactly the same as the case for cheating Bob. We let the variables be the states in Bob's control after each of Alice's messages. There are two key differences. One is that Alice sends the first message and Bob sends the last, explaining the slightly different constraints. Secondly, Bob measures only the $\mathbb{C}^{B_0 \times A'_0 \times A \times A'}$ part of his state after Alice's last message, i.e., he measures the state described by $\mathrm{Tr}_{B_0' \times B'}(\sigma_F)$. Note that the adjoint of the partial trace can be written as $\mathrm{Tr}_A^*(Y) = Y \otimes \mathrm{I}_A$, which is evident from the definition. Therefore, we can write

$$\langle \mathrm{Tr}_{B_0' \times B'}(\sigma_F), \Pi_{\mathrm{B},c} \rangle = \langle \sigma_F, \Pi_{\mathrm{B},c} \otimes \mathrm{I}_{B_0' \times B'} \rangle,$$

which explains the objective function. □

We refer to these SDPs as Bob's and Alice's cheating SDPs, respectively. Analyzing and solving these SDPs numerically get increasingly difficult and time consuming as n increases since the dimensions of the positive semidefinite variables increase exponentially with the number of qubits in the communication. We make use of the reductions described in the next subsection.

2.1.2 Reduced formulations of quantum strategies

We now prove that the cheating SDPs can have a certain, restricted form while retaining the same optimal objective function value. That is, we cut down the feasible region to something that is much cleaner and illustrates the simple communication of the protocol. The main technique used in proving that we do not cut off any optimal solutions comes from duality theory of semidefinite programming. We generalize the following idea. If we wish to prove that a certain feasible solution is optimal for the primal problem, it suffices to exhibit a dual solution with the same objective function value. Here, we claim that a restricted feasible region contains an optimal solution. Let p_1^* be the optimal value of the original SDP, p_2^* be the optimal value of the restricted SDP, and let d_1^* and d_2^* be the optimal values of the respective dual problems and assume all of them are finite. We want to show that $p_1^* = p_2^*$. Suppose the restricted problem and its dual have zero duality gap (which can be guaranteed if restricting the feasible region does not cut off all the strictly feasible points). Then if we can prove that $d_1^* \leq d_2^*$, we have

$$p_1^* \leq d_1^* \leq d_2^* = p_2^* \leq p_1^*,$$

proving $p_1^* = p_2^*$ as desired. To show $d_1^* \leq d_2^*$, it suffices to find a restriction of the dual of the original SDP to get to a problem equivalent to the dual of the restricted SDP. This is depicted in Figure 2.4.

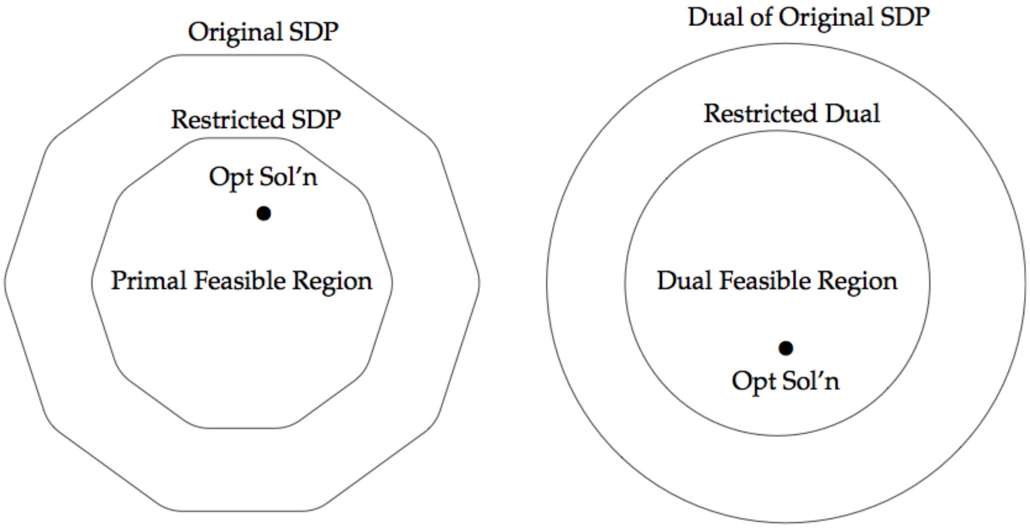


Figure 2.4: There exist optimal solutions in certain restrictions of the feasible regions of the cheating SDPs.

Cheating Bob

Let x_i denote an element of A_i and let y_i denote an element of B_i, for each $i \in \{1, \ldots, n\}$. We now restrict the feasible region of Bob's cheating SDPs by defining the following primal feasible solution. Intuitively, Bob creates a state similar to his honest state, but the amplitudes are being changed during each message according to all of the previously revealed messages. The solution is given below

$$\bar{\rho}_j := \sum_{x_1} \cdots \sum_{x_j} |x_1, \ldots, x_j\rangle\langle x_1, \ldots, x_j| \otimes |\psi_{x_1, \ldots, x_j}\rangle\langle\psi_{x_1, \ldots, x_j}| \otimes \mathrm{Diag}(p_j),$$

for $j \in \{1, \ldots, n\}$, and

$$\bar{\rho}_F := \sum_{a \in \{0,1\}} |aa\rangle\langle aa| \otimes |\psi'_a\rangle\langle\psi'_a|,$$

where $p_j \in \mathbb{R}_+^{A_1 \times B_1 \times \cdots \times A_j \times B_j}$ is a variable,

$$|\psi_{x_1, \ldots, x_j}\rangle := \frac{1}{\sqrt{2}} \sum_{x_{j+1}} \cdots \sum_{x_n} \sum_{a \in \{0,1\}} \sqrt{\alpha_{a,x}}\, |aa\rangle |x_{j+1}, \ldots, x_n\rangle |x_{j+1}, \ldots, x_n\rangle,$$

and

$$|\psi'_a\rangle := \sum_{y \in B} \sqrt{\frac{1}{2} \sum_{x \in A} \alpha_{a,x} [p_n]_{x,y}}\, |yy\rangle, \quad \forall a \in \{0,1\}.$$

The new objective function for forcing outcome 0 becomes

$$\langle \bar{\rho}_F, \Pi_{\mathrm{A},0} \rangle = \frac{1}{2} \sum_{a \in \{0,1\}} \mathrm{F}\left((\alpha_a \otimes \mathrm{I}_A)^\top p_n,\, \beta_a \right)$$

and the variables $(p_1, \ldots, p_n)$ belong to a polytope which we refer to as *Bob's cheating polytope*, defined below.

Definition 2.1.4. *We define* Bob's cheating polytope*, denoted* $\mathcal{P}_\mathrm{B}$*, as the set of vectors* $(p_1, p_2, \ldots, p_n)$ *satisfying*

$$\begin{aligned}
\mathrm{Tr}_{B_1}(p_1) &= e_{A_1}, \\
\mathrm{Tr}_{B_2}(p_2) &= p_1 \otimes e_{A_2}, \\
&\vdots \\
\mathrm{Tr}_{B_n}(p_n) &= p_{n-1} \otimes e_{A_n}, \\
p_j &\in \mathbb{R}_+^{A_1 \times B_1 \times \cdots \times A_j \times B_j}, \text{ for all } j \in \{1, \ldots, n\},
\end{aligned}$$

where e_{A_j} *denotes the vector of all ones in the corresponding space* $\mathbb{C}^{A_j}$*.*

Since we have restricted the feasible region of a maximization SDP, we have proved that

$$P^*_{\mathrm{B},0} \geq \max \left\{ \frac{1}{2} \sum_{a \in \{0,1\}} \mathrm{F}\left((\alpha_a \otimes \mathrm{I}_B)^\top p_n,\, \beta_a \right) : (p_1, \ldots, p_n) \in \mathcal{P}_\mathrm{B} \right\}.$$

By changing the value of $\bar{\rho}_F \in \Sigma_+^{A_0 \times B'_0 \times B \times B'}$ above to

$$\bar{\rho}_F = \sum_{a \in \{0,1\}} |a\bar{a}\rangle\langle a\bar{a}| \otimes |\psi'_a\rangle\langle\psi'_a|,$$

we get

$$P^*_{\mathrm{B},1} \geq \max \left\{ \frac{1}{2} \sum_{a \in \{0,1\}} \mathrm{F}\left((\alpha_a \otimes \mathrm{I}_B)^\top p_n,\, \beta_{\bar{a}} \right) : (p_1, \ldots, p_n) \in \mathcal{P}_\mathrm{B} \right\}.$$

This swaps Bob's choice of commitment reveal in the last message.

We now argue that the above inequalities hold with equality.

Theorem 2.1.5 (Bob's reduced problems)**.** *For the* BCCF*-protocol defined by the parameters* $\alpha_0, \alpha_1 \in \text{Prob}^A$ *and* $\beta_0, \beta_1 \in \text{Prob}^B$*, we have*

$$P^*_{\text{B},0} = \max \left\{ \frac{1}{2} \sum_{a \in \{0,1\}} \text{F}\left((\alpha_a \otimes \text{I}_B)^\top p_n,\, \beta_a \right) : (p_1, \ldots, p_n) \in \mathcal{P}_\text{B} \right\}$$

and

$$P^*_{\text{B},1} = \max \left\{ \frac{1}{2} \sum_{a \in \{0,1\}} \text{F}\left((\alpha_a \otimes \text{I}_B)^\top p_n,\, \beta_{\bar{a}} \right) : (p_1, \ldots, p_n) \in \mathcal{P}_\text{B} \right\}.$$

Note that we sometimes refer to this reduced problem as Bob's *reduced* SDP. This implies we have replaced the fidelity with its SDP characterization from Subsection 1.4.7. We do the same for cheating Alice, to come.

Proof. We begin by proving this for the case of $P^*_{\text{B},0}$. Consider the dual of Bob's cheating SDP below:

$$\begin{array}{rrcl}
P^*_{\text{B},0} \;=\; \inf & \langle W_1, \text{Tr}_{A_1} |\psi\rangle\langle\psi| \rangle & & \\
\text{subject to} & W_j \otimes \text{I}_{B_j} & \succeq & W_{j+1} \otimes \text{I}_{A_{j+1}}, \\
& & & \text{for all } \; j \in \{1, \ldots, n-1\}, \\
& W_n \otimes \text{I}_{B_n} & \succeq & W_{n+1} \otimes \text{I}_{A'} \otimes \text{I}_{A'_0}, \\
& W_{n+1} \otimes \text{I}_{B'} \otimes \text{I}_{B'_0} & \succeq & \Pi_{\text{A},0}, \\
& W_j & \in & \Sigma^{A_0 \times A'_0 \times B_1 \times \cdots \times B_{j-1} \times A_{j+1} \times \cdots \times A_n \times A'}, \\
& & & \text{for all } j \in \{1, \ldots, n\}, \\
& W_{n+1} & \in & \Sigma^{A_0 \times B}.
\end{array}$$

We now define a restriction of the following form:

$$W_j := \sum_{x_1} \sum_{y_1} \cdots \sum_{y_{j-1}} \sum_{x_j} |x_1, y_1, \ldots, y_{j-1}, x_j\rangle\langle x_1, y_1, \ldots, y_{j-1}, x_j| \otimes W_{j, x_1, y_1, \ldots, y_{j-1}, x_j},$$

for $j \in \{1, \ldots, n\}$, and

$$W_{n+1} := \sum_{a \in \{0,1\}} |a\rangle\langle a| \otimes \text{Diag}(v_a).$$

Under this restriction, we have the following problem:

$$\begin{array}{rrcl}
d_2^* \; = \; \inf & \displaystyle\sum_{x_1} \langle W_{1,x_1}, |\psi_{x_1}\rangle\langle\psi_{x_1}| \rangle & & \\
\text{subject to} & W_{j,x_1,y_1,\ldots,y_{j-1},x_j} & \succeq & \displaystyle\sum_{x_{j+1}} |x_{j+1}\rangle\langle x_{j+1}| \otimes \mathrm{I}_{A_{j+1}} \otimes W_{j+1,x_1,y_1,\ldots,y_j,x_{j+1}}, \\
& & & \text{for all } \; j \in \{1,\ldots,n-1\}, \\
& & & (x_1,\ldots,x_j) \in A_1 \times \cdots \times A_j, \\
& & & (y_1,\ldots,y_j) \in B_1 \times \cdots \times B_j, \\
& W_{n,x_1,y_1,\ldots,y_{n-1},x_n} & \succeq & \displaystyle\sum_{a \in \{0,1\}} v_{a,y} \, |a\rangle\langle a| \otimes \mathrm{I}_{A'_0}, \\
& \mathrm{Diag}(v_a) & \succeq & \sqrt{\beta_a}\sqrt{\beta_a}^{\top}, \text{for all } a \in \{0,1\},
\end{array}$$

where the last constraint was obtained using Lemma 1.4.11. The last constraint changes to $\mathrm{Diag}(v_a) \succeq \sqrt{\beta_{\bar{a}}}\sqrt{\beta_{\bar{a}}}^{\top}$, for all $a \in \{0,1\}$, if Bob is cheating towards 1 and the rest of the proof follows similarly in this case. Note that this shows $d_2^* \geq P_{\mathrm{B},0}^*$.

Since the objective function only depends on W_{1,x_1} in the subspace $|\psi_{x_1}\rangle\langle\psi_{x_1}|$, we apply the Subspace Lemma (Lemma 1.4.9) to the first constraint and replace it with

$$\begin{array}{rcl}
\langle\psi_{x_1}|W_{1,x_1}|\psi_{x_1}\rangle & \geq & \langle\psi_{x_1}| \displaystyle\sum_{x_2} |x_2\rangle\langle x_2| \otimes \mathrm{I}_{A_2} \otimes W_{2,x_1,y_1,x_2} |\psi_{x_1}\rangle \\
& = & \displaystyle\sum_{x_2} \langle\psi_{x_1,x_2}|W_{2,x_1,y_1,x_2}|\psi_{x_1,x_2}\rangle.
\end{array}$$

Examining the next constraint, we need to choose W_{2,x_1,y_1,x_2} to satisfy

$$W_{2,x_1,y_1,x_2} \succeq \sum_{x_3} |x_3\rangle\langle x_3| \otimes \mathrm{I}_{A_3} \otimes W_{3,x_1,y_2,x_2,y_2,x_3}.$$

Since the objective function value only depends on $\langle\psi_{x_1,x_2}|W_{2,x_1,y_1,x_2}|\psi_{x_1,x_2}\rangle$, we can repeat the same argument and replace the constraint by

$$\langle\psi_{x_1,x_2}|W_{2,x_1,y_1,x_2}|\psi_{x_1,x_2}\rangle \geq \sum_{x_3} \langle\psi_{x_1,x_2,x_3}|W_{3,x_1,y_2,x_2,y_2,x_3}|\psi_{x_1,x_2,x_3}\rangle.$$

Continuing in this fashion, we can replace each constraint to get the following problem with the same optimal objective value:

$$\begin{aligned}
\text{inf} \quad & \sum_{x_1} \langle W_{1,x_1}, |\psi_{x_1}\rangle\langle\psi_{x_1}|\rangle \\
\text{s.t.} \quad \langle\psi_{x_1,\dots,x_j}|W_{j,x_1,y_1,\dots,y_{j-1},x_j}|\psi_{x_1,\dots,x_j}\rangle \;\; &\geq \;\; \sum_{x_{j+1}} \langle\psi_{x_1,\dots,x_{j+1}}|W_{j+1,x_1,y_1,\dots,y_j,x_{j+1}}|\psi_{x_1,\dots,x_{j+1}}\rangle \\
& \qquad \text{for all } \; j \in \{1,\dots,n-1\}, \\
& \qquad (x_1,\dots,x_{j+1}) \in A_1 \times \cdots \times A_{j+1}, \\
& \qquad (y_1,\dots,y_j) \in B_1 \times \cdots \times B_j, \\
\langle\psi_x|W_{n,x_1,y_1,\dots,y_{n-1},x_n}|\psi_x\rangle \;\; &\geq \;\; \sum_{a\in\{0,1\}} \alpha_{a,x}\, v_{a,y}, \text{for all } x \in A, y \in B, \\
\operatorname{Diag}(v_a) \;\; &\succeq \;\; \sqrt{\beta_a}\sqrt{\beta_a}^{\top}, \text{for all } a \in \{0,1\}.
\end{aligned}$$

Define

$$w_{j,x_1,y_1,\dots,y_{j-1},x_j} := \langle\psi_{x_1,\dots,x_j}|W_{j,x_1,y_1,\dots,y_{j-1},x_j}|\psi_{x_1,\dots,x_j}\rangle,$$

for all $j \in \{1,\dots,n-1\}$, $(x_1,\dots,x_{j+1}) \in A_1 \times \cdots \times A_{j+1}$, $(y_1,\dots,y_j) \in B_1 \times \cdots \times B_j$, to get the problem

$$\begin{aligned}
d_2^* \; = \; \text{inf} \quad & \sum_{x_1} w_{1,x_1} \\
\text{subject to} \quad w_{j,x_1,y_1,\dots,y_{j-1},x_j} \;\; &\geq \;\; \sum_{x_{j+1}} w_{j+1,x_1,y_1,\dots,y_j,x_{j+1}}, \\
& \qquad \text{for all } \; j \in \{1,\dots,n-1\}, \\
& \qquad (x_1,\dots,x_{j+1}) \in A_1 \times \cdots \times A_{j+1}, \\
& \qquad (y_1,\dots,y_j) \in B_1 \times \cdots \times B_j, \\
w_{n,x_1,y_1,\dots,y_{n-1},x_n} \;\; &\geq \;\; \sum_{a\in\{0,1\}} \frac{1}{2}\alpha_{a,x}\, v_{a,y}, \text{for all } x \in A, y \in B, a \in \{0,1\}, \\
\operatorname{Diag}(v_a) \;\; &\succeq \;\; \sqrt{\beta_a}\sqrt{\beta_a}^{\top}, \forall a \in \{0,1\}.
\end{aligned}$$

This problem has a strictly feasible solution and the objective function is bounded from below on the feasible region, thus strong duality holds and there is zero duality gap. The dual of this problem is

$$d_2^* = \max_{\substack{(p_1,\dots,p_n)\in\mathcal{P}_{\mathrm{B}} \\ \rho_0,\rho_1\in\Sigma_+^B}} \left\{ \sum_{a\in\{0,1\}} \frac{1}{2}\langle\rho_a, \sqrt{\beta_a}\sqrt{\beta_a}^{\top}\rangle : \operatorname{diag}(\rho_a) = (\alpha_a \otimes \mathrm{I}_B)^{\top} p_n,\; \forall a \in \{0,1\} \right\},$$

which is equivalent to the reduced problem by Lemma 1.4.10. Therefore, we have $P_{\mathrm{B},0}^* = d_2^*$, as desired. □

Cheating Alice

We now restrict the feasible region of Alice's cheating SDPs by defining the following primal feasible solution. Intuitively, this strategy is similar to that of cheating Bob. The solution is given below

$$\bar{\sigma}_j := \sum_{y_1} \cdots \sum_{y_{j-1}} |y_1, \ldots, y_{j-1}\rangle\langle y_1, \ldots, y_{j-1}| \otimes |\phi_{y_1,\ldots,y_{j-1}}\rangle\langle \phi_{y_1,\ldots,y_{j-1}}| \otimes \mathrm{Diag}(s_j),$$

for $j \in \{2, \ldots, n\}$, and

$$\bar{\sigma}_F := \sum_{a \in A_0'} \sum_{y \in B} |a\rangle\langle a| \otimes |y\rangle\langle y| \otimes |\phi_y\rangle\langle \phi_y| \otimes |\phi'_{a,y}\rangle\langle \phi'_{a,y}|,$$

where $s_j \in \mathbb{R}_+^{A_1 \times B_1 \times \cdots \times B_{j-1} \times A_j}$ and $s \in \mathbb{R}_+^{A_0' \times A \times B}$ are variables,

$$|\phi_{y_1,\ldots,y_{j-1}}\rangle := \frac{1}{\sqrt{2}} \sum_{y_j} \cdots \sum_{y_n} \sum_{b \in \{0,1\}} \sqrt{\beta_{b,y}}\, |bb\rangle |y_j, \ldots, y_n\rangle |y_j, \ldots, y_n\rangle,$$

and

$$|\phi'_{a,y}\rangle := \sum_{x \in A} \sqrt{s_{a,y,x}}\, |xx\rangle, \quad \forall y \in B, a \in \{0,1\}.$$

With this restriction, we can write the new objective function for forcing outcome 0 as

$$\langle \bar{\sigma}_F, \Pi_{\mathrm{B},0} \otimes \mathrm{I}_{B_0' \times B'} \rangle = \frac{1}{2} \sum_{a \in A_0'} \sum_{y \in B'} \beta_{a,y}\, \mathrm{F}(s^{(a,y)}, \alpha_a),$$

where $s^{(a,y)} \in \mathbb{C}^A$ is defined as the restriction of s with a and y fixed. We can define it element-wise as $[s^{(a,y)}]_x := s_{a,y,x}$. The new objective function for forcing outcome 1 is

$$\langle \bar{\sigma}_F, \Pi_{\mathrm{B},1} \otimes \mathrm{I}_{B_0' \times B'} \rangle = \frac{1}{2} \sum_{a \in A_0'} \sum_{y \in B'} \beta_{\bar{a},y}\, \mathrm{F}(s^{(a,y)}, \alpha_a).$$

The variables $(s_1, \ldots, s_n, s)$ belong to a polytope which we refer to as *Alice's cheating polytope*, defined below.

Definition 2.1.6. *We define* Alice's cheating polytope*, denoted* $\mathcal{P}_{\mathrm{A}}$*, as the set of vectors* $(s_1, s_2, \ldots, s_n, s)$ *satisfying*

$$\begin{array}{rcl}
\mathrm{Tr}_{A_1}(s_1) & = & 1, \\
\mathrm{Tr}_{A_2}(s_2) & = & s_1 \otimes e_{B_1}, \\
& \vdots & \\
\mathrm{Tr}_{A_n}(s_n) & = & s_{n-1} \otimes e_{B_{n-1}}, \\
\mathrm{Tr}_{A_0'}(s) & = & s_n \otimes e_{B_n}, \\
s_1 & \in & \mathbb{R}_+^{A_1}, \\
s_j & \in & \mathbb{R}_+^{A_1 \times B_1 \times \cdots \times B_{j-1} \times A_j}, \textit{ for all } j \in \{2, \ldots, n\}, \\
s & \in & \mathbb{R}_+^{A \times B \times A_0'},
\end{array}$$

where e_{B_j} *is the vector of all ones in the corresponding space* $\mathbb{C}^{B_j}$*.*

We have proved

$$P^*_{\mathrm{A},0} \geq \max \left\{ \frac{1}{2} \sum_{a \in \{0,1\}} \sum_{y \in B} \beta_{a,y} \, \mathrm{F}(s^{(a,y)}, \alpha_a) \; : \; (s_1, \ldots, s_n, s) \in \mathcal{P}_{\mathrm{A}} \right\}$$

and

$$P^*_{\mathrm{A},1} \geq \max \left\{ \frac{1}{2} \sum_{a \in \{0,1\}} \sum_{y \in B} \beta_{\bar{a},y} \, \mathrm{F}(s^{(a,y)}, \alpha_a) \; : \; (s_1, \ldots, s_n, s) \in \mathcal{P}_{\mathrm{A}} \right\}.$$

The following theorem shows that the above inequalities hold with equality.

Theorem 2.1.7 (Alice's reduced problems)**.** *For the* BCCF*-protocol defined by the parameters* $\alpha_0, \alpha_1 \in \mathrm{Prob}^A$ *and* $\beta_0, \beta_1 \in \mathrm{Prob}^B$*, we have*

$$P^*_{\mathrm{A},0} = \max \left\{ \frac{1}{2} \sum_{a \in \{0,1\}} \sum_{y \in B} \beta_{a,y} \, \mathrm{F}(s^{(a,y)}, \alpha_a) \; : \; (s_1, \ldots, s_n, s) \in \mathcal{P}_{\mathrm{A}} \right\}$$

and

$$P^*_{\mathrm{A},1} = \max \left\{ \frac{1}{2} \sum_{a \in \{0,1\}} \sum_{y \in B} \beta_{\bar{a},y} \, \mathrm{F}(s^{(a,y)}, \alpha_a) \; : \; (s_1, \ldots, s_n, s) \in \mathcal{P}_{\mathrm{A}} \right\}.$$

Proof. Consider the dual to Alice's cheating SDP for forcing outcome 0, below:

$$\begin{array}{rrcl}
P^*_{\mathrm{A},0} \; = \; \inf & \langle Z_1, |\phi\rangle\langle\phi| \rangle & & \\
\text{subject to} & Z_j \otimes \mathrm{I}_{A_j} & \succeq & Z_{j+1} \otimes \mathrm{I}_{B_j}, \\
 & & & \text{for all } j \in \{1, \ldots, n\}, \\
 & Z_{n+1} \otimes \mathrm{I}_{A'} \otimes \mathrm{I}_{A'_0} & \succeq & \Pi_{\mathrm{B},0} \otimes \mathrm{I}_{B'_0} \otimes \mathrm{I}_{B'}, \\
 & Z_j & \in & \Sigma^{B_0 \times B'_0 \times A_1 \times \cdots \times A_{j-1} \times B_j \times \cdots \times B_n \times B'}, \\
 & & & \text{for all } j \in \{1, \ldots, n, n+1\}.
\end{array}$$

Consider the following restriction:

$$Z_{j+1} := \sum_{x_1} \sum_{y_1} \cdots \sum_{x_j} \sum_{y_j} |x_1, y_1, \ldots, x_j, y_j\rangle\langle x_1, y_1, \ldots, x_j, y_j| \otimes Z_{j+1, x_1, y_1, \ldots, x_j, y_j},$$

for $j \in \{1, \ldots, n\}$. Substituting this into the constraints, we get the following new problem

$$\begin{array}{rrcl}
d^*_2 \; = \; \inf & \langle Z_1, |\phi\rangle\langle\phi| \rangle & & \\
\text{subject to} & Z_1 & \succeq & \displaystyle\sum_{y_1} |y_1\rangle\langle y_1| \otimes \mathrm{I}_{B_1} \otimes Z_{2, x_1, y_1}, \\
 & Z_{j, x_1, y_1, \ldots, x_{j-1}, y_{j-1}} & \succeq & \displaystyle\sum_{y_j} |y_j\rangle\langle y_j| \otimes \mathrm{I}_{B_j} \otimes Z_{j+1, x_1, y_1, \ldots, x_j, y_j}, \\
 & & & \text{for all } j \in \{2, \ldots, n\}, \\
 & & & (x_1, \ldots, x_j) \in A_1 \times \cdots \times A_j, \\
 & & & (y_1, \ldots, y_j) \in B_1 \times \cdots \times B_j, \\
 & \displaystyle\sum_{x \in A} Z_{n+1, x, y} \otimes |x\rangle\langle x| \otimes \mathrm{I}_{A'} & \succeq & |a\rangle\langle a| \otimes \mathrm{I}_{B'_0} \otimes |\psi_a\rangle\langle\psi_a|, \; \forall a \in \{0, 1\}, y \in B.
\end{array}$$

This shows that $d^*_2 \geq P^*_{\mathrm{A},0}$. Applying the Subspace Lemma (Lemma 1.4.9) recursively, as in the case for cheating Bob, we get the following problem with the same optimal objective value

$$\begin{array}{rrcl}
\inf & \langle Z_1, |\phi\rangle\langle\phi| \rangle & & \\
\text{s.t.} & \langle \phi_{y_1, \ldots, y_{j-1}} | Z_{j, x_1, y_1, \ldots, x_{j-1}, y_{j-1}} | \phi_{y_1, \ldots, y_{j-1}} \rangle & \geq & \displaystyle\sum_{y_j} \langle \phi_{y_1, \ldots, y_j} | Z_{j+1, x_1, y_1, \ldots, x_j, y_j} | \phi_{y_1, \ldots, y_j} \rangle, \\
 & & & \text{for all } j \in \{1, \ldots, n\}, \\
 & & & (x_1, \ldots, x_j) \in A_1 \times \cdots \times A_j, \\
 & & & (y_1, \ldots, y_j) \in B_1 \times \cdots \times B_j, \\
 & \displaystyle\sum_{x \in A} \langle \phi_y | Z_{n+1, x, y} | \phi_y \rangle \, |x\rangle\langle x| \otimes \mathrm{I}_{A'} & \succeq & \tfrac{1}{2} \beta_{a, y} \, |\psi_a\rangle\langle\psi_a|, \text{ for all } a \in \{0, 1\}, y \in B.
\end{array}$$

Defining

$$z_{j+1,x_1,y_1,\dots,x_j,y_j} := \langle \phi_{y_1,\dots,y_j} | Z_{j+1,x_1,y_1,\dots,x_j,y_j} | \phi_{y_1,\dots,y_j} \rangle,$$

for $j \in \{1,\dots,n\}$, $(x_1,\dots,x_j) \in A_1 \times \cdots \times A_j$, and $(y_1,\dots,y_j) \in B_1 \times \cdots \times B_j$, and

$$\mathrm{Diag}(z_{n+1}^{(y)}) := \sum_{x \in A} \langle \phi_y | Z_{n+1,x,y} | \phi_y \rangle \, |x\rangle\langle x|,$$

for $y \in B$, we get the following problem:

$$\begin{array}{rrcl} d_2^* \; = \; \inf & & z_1 & \\ \text{subject to} & z_{j,x_1,y_1,\dots,x_{j-1},y_{j-1}} & \geq & \displaystyle\sum_{y_j \in B_j} z_{j+1,x_1,y_1,\dots,x_j,y_j}, \\ & & & \text{for all } j \in \{1,\dots,n\}, \\ & & & (x_1,\dots,x_j) \in A_1 \times \cdots \times A_j, \\ & & & (y_1,\dots,y_j) \in B_1 \times \cdots \times B_j, \\ & \mathrm{Diag}(z_{n+1}^{(y)}) & \succeq & \frac{1}{2}\beta_{a,y} \sqrt{\alpha_a}\sqrt{\alpha_a}^\top, \text{ for all } y \in B, a \in \{0,1\}. \end{array}$$

This problem has a strictly feasible solution and the objective function is bounded from below on the feasible region, thus it and its dual have zero duality gap. The dual of this problem is

$$\max_{\substack{(s_1,\dots,s_n,s) \in \mathcal{P}_\mathrm{A} \\ \sigma_{a,y} \in \Sigma_+^A}} \left\{ \frac{1}{2} \sum_{a \in \{0,1\}} \sum_{y \in B} \beta_{a,y} \langle \sigma_{a,y}, \sqrt{\alpha_a}\sqrt{\alpha_a}^\top \rangle : \mathrm{diag}(\sigma_{a,y}) = s^{(a,y)}, \, \forall a \in \{0,1\}, y \in B \right\}$$

which is equivalent to Alice's reduced problem for forcing outcome 0 by Lemma 1.4.10 and has optimal objective value d_2^*. Therefore, we have $P_{\mathrm{A},0}^* = d_2^*$ as desired.

The case for forcing outcome 1 is almost the exact same, except every occurrence of α_a is replaced with $\alpha_{\bar{a}}$. The dual thus becomes

$$\max_{\substack{(s_1,\dots,s_n,s) \in \mathcal{P}_\mathrm{A} \\ \sigma_{a,y} \in \Sigma_+^A}} \left\{ \frac{1}{2} \sum_{a \in \{0,1\}} \sum_{y \in B} \beta_{a,y} \langle \sigma_{a,y}, \sqrt{\alpha_{\bar{a}}}\sqrt{\alpha_{\bar{a}}}^\top \rangle : \mathrm{diag}(\sigma_{a,y}) = s^{(a,y)}, \, \forall a \in \{0,1\}, y \in B \right\}.$$

Since the last constraint is symmetric in a, we can replace $s^{(a,y)}$ with $s^{(\bar{a},y)}$ (and $\sigma_{a,y}$ with $\sigma_{\bar{a},y}$) and the optimal objective value does not change. We can write it as

$$\max_{\substack{(s_1,\dots,s_n,s) \in \mathcal{P}_\mathrm{A} \\ \sigma_{a,y} \in \Sigma_+^A}} \left\{ \frac{1}{2} \sum_{a \in \{0,1\}} \sum_{y \in B} \beta_{\bar{a},y} \langle \sigma_{a,y}, \sqrt{\alpha_a}\sqrt{\alpha_a}^\top \rangle : \mathrm{diag}(\sigma_{a,y}) = s^{(a,y)}, \, \forall a \in \{0,1\}, y \in B \right\},$$

as desired. □

We note here that we can get similar reductions if Alice chooses a with a non-uniform probability distribution and similarly for Bob. It only changes the $1/2$ multiplicative factor in the reduced problems to something that depends on a and the proofs are nearly identical.

We can also prove these two theorems using the primal SDPs alone. These proofs rely on the fact that for any $x \in \mathbb{C}^A$, the set

$$\{\lambda \, xx^* : \lambda > 0\}$$

is an extreme ray of the cone of positive semidefinite matrices. That is, if $X_1, X_2 \in \Sigma_+^A$ satisfy $X_1 + X_2 = \lambda \, xx^*$, for some $\lambda > 0$, then $X_1 = \lambda_1 \, xx^*$ and $X_2 = \lambda_2 \, xx^*$ for some $\lambda_1, \lambda_2 \geq 0$, satisfying $\lambda_1 + \lambda_2 = \lambda$.

The structure of certain optimal solutions shown in this subsection was an observation after numerically solving some of the cheating SDPs from the last subsection. We note that there are some similarities between the reduced problems above and the optimal solutions of the cheating SDPs for the weak coin-flipping protocols in [Moc05]. The protocols Mochon considers in [Moc05] also give rise to "reduced problems" being the maximization of fidelity functions over a polytope. However, the analysis is much cleaner in Mochon's work since the objective function only involves a single fidelity function as opposed to the linear combination of fidelity functions that arise in BCCF-protocols. This difference is due to the fact that weak coin-flipping protocols often allow a stronger cheat detection step than those for strong coin-flipping. Having a single fidelity function allowed Mochon to construct an optimal solution using a dynamic programming approach. The structure of the objective functions in the reduced problems above for BCCF-protocols does not reveal an obvious way to solve it using dynamic programming, making this family of protocols harder to analyze.

A succinct way to write the duals of the reduced cheating problems

In this part, we present a simple form for the duals of the reduced cheating SDPs. We show that we only need to consider the variables in the positive semidefiniteness constraints, since the linear inequalities reveal how to optimally assign the rest of the variables.

Sometimes it is easier to work with the succinct form developed in this section because handling many dual variables can overcomplicate simple ideas.

Consider Bob's reduced cheating SDP, below, using the SDP characterization for fidelity

$$P^*_{\mathrm{B},0} = \max_{\substack{(p_1,\ldots,p_n)\in\mathcal{P}_\mathrm{B} \\ \rho_0,\rho_1\in\Sigma_+^B}} \left\{ \sum_{a\in\{0,1\}} \frac{1}{2}\langle \rho_a, \sqrt{\beta_a}\sqrt{\beta_a}^\top \rangle : \mathrm{diag}(\rho_a) = (\alpha_a \otimes \mathrm{I}_B)^\top p_n, \, \forall a \in \{0,1\} \right\}.$$

The dual is

$$\begin{array}{rrcll}
\inf & \mathrm{Tr}_{A_1}(w_1) & & & \\
\text{subject to} & w_1 \otimes e_{B_1} & \geq & \mathrm{Tr}_{A_2}(w_2), & \\
 & w_2 \otimes e_{B_2} & \geq & \mathrm{Tr}_{A_3}(w_3), & \\
 & & \vdots & & \\
 & w_n \otimes e_{B_n} & \geq & \frac{1}{2}\sum_{a\in\{0,1\}} \alpha_a \otimes v_a, & \\
 & \mathrm{Diag}(v_a) & \succeq & \sqrt{\beta_a}\sqrt{\beta_a}^\top, & \text{for all } a \in \{0,1\}.
\end{array}$$

Let us examine the first constraint $w_1 \otimes e_{B_1} \geq \mathrm{Tr}_{A_2}(w_2)$. This is equivalent to saying $w_{1,x_1} \geq \sum_{x_2} w_{2,x_1,y_1,x_2}$ for all $x_1 \in A_1$, $y_1 \in B_1$. Once we fix a value for w_2, an optimal choice of w_1 is $w_{1,x_1} = \max_{y_1} \sum_{x_2} w_{2,x_1,y_1,x_2}$. Using this idea, we can rewrite Bob's dual as

$$\inf_{\mathrm{Diag}(v_a)\succeq\sqrt{\beta_a}\sqrt{\beta_a}^\top} \left\{ \sum_{x_1} \max_{y_1} \sum_{x_2} \max_{y_2} \cdots \sum_{x_n} \max_{y_n} \sum_{a} \frac{1}{2} \alpha_{a,x} v_{a,y} \right\}$$

and Alice's as

$$\inf_{\mathrm{Diag}(z^{(y)}_{n+1})\succeq\frac{1}{2}\beta_{a,y}\sqrt{\alpha_a}\sqrt{\alpha_a}^\top} \left\{ \max_{x_1} \sum_{y_1} \cdots \max_{x_n} \sum_{y_n} z_{n+1,x,y} \right\},$$

each for forcing outcome 0. We can switch β_0 with β_1 to get the succinct forms for forcing outcome 1.

Sometimes these forms for the duals are convenient, as seen in the next subsection.

2.1.3 An SDP proof for why qubit messages are sufficient

In this subsection, we show how the succinct representation of the duals helps us prove a novel result, that we can bound the dimension of the messages in a BCCF-protocol.

However, the analysis in the rest of the thesis is general and does not assume this result unless otherwise stated.

We use the reduced cheating SDPs to prove that we can assume $A_i = B_i = \{0, 1\}$, that is, each message is a single qubit. More specifically, we show that for any BCCF-protocol, there exists another BCCF-protocol with qubit messages where the bias is no larger. We prove it for Alice's messages as the proof for Bob's messages is nearly identical.

Suppose we have a protocol defined by

$$A = A_1 \times \cdots A_n, \ B = B_1 \times \cdots \times B_n, \ \alpha_0, \alpha_1 \in \mathrm{Prob}^A, \ \beta_0, \beta_1 \in \mathrm{Prob}^B.$$

Suppose Alice's i'th message has large dimension, that is, $|A_i| > 2$. We define a new protocol by replacing A_i with $A'_i \times A''_i$, where $A_i \subseteq A'_i \times A''_i$. Notice that α_0 and α_1 can be viewed as probability distributions over $A_1 \times \cdots A_{i-1} \times A'_i \times A''_i \times A_{i+1} \times \cdots \times A_n$ in the obvious way. We also add a "dummy" message from Bob by adding B_d in between B_i and B_{i+1}. This dummy message needs to be independent of the protocol, so we can suppose Bob sends $|0\rangle$. This effectively replaces β_b with $\beta'_b := \beta_b \otimes [1, 0]_d^\top$, for each $b \in \{0, 1\}$. If Alice and Bob cannot cheat more in this new protocol, then we can repeat these arguments to show that all of Alice's messages are qubit messages by inductively breaking up the $\mathbb{C}^{A_i}$ spaces.

Bob's cheating probabilities do not increase

We now show that Bob cannot use the extra message to cheat more in the new protocol. We show this by constructing a dual feasible solution.

In the original protocol, cheating Bob can force outcome 0 with maximum probability given by the optimal objective value of the following problem

$$\inf_{\mathrm{Diag}(v_a) \succeq \sqrt{\beta_a}\sqrt{\beta_a}^\top} \left\{ \sum_{x_1} \max_{y_1} \sum_{x_2} \max_{y_2} \cdots \sum_{x_n} \max_{y_n} \sum_{a} \frac{1}{2} \alpha_{a,x} v_{a,y} \right\}.$$

In the new protocol, cheating Bob can force outcome 0 with maximum probability given by the optimal objective value of the following problem

$$\inf_{\mathrm{Diag}(\tilde{v}_a) \succeq \sqrt{\beta'_a}\sqrt{\beta'_a}^\top} \left\{ \sum_{x_1} \max_{y_1} \sum_{x_2} \max_{y_2} \cdots \sum_{x'_i \in A'_i} \max_{y_d \in B_d} \sum_{x''_i \in A''_i} \cdots \sum_{x_n} \max_{y_n} \sum_{a} \frac{1}{2} \alpha_{a,x} \tilde{v}_{a,y'} \right\}.$$

For any (v_0, v_1) feasible in the first problem, we can define a solution feasible in the second problem $(\tilde{v}_0, \tilde{v}_1) := \left(v_0 \otimes [1,0]_d^\top, v_1 \otimes [1,0]_d^\top\right)$ with the same objective function value. Notice the same argument holds if we switch β_0 with β_1 and β_0' with β_1', i.e., if Bob wants outcome 1. Since these are minimization problems, Bob can cheat no more in the new protocol.

Alice's cheating probabilities do not increase

We now show that Alice cannot use her extra message to cheat more in the new protocol. To show this, we repeat the same argument as in the case for cheating Bob.

In the original protocol, cheating Alice can force outcome 0 with maximum probability given by the optimal objective value of the following problem

$$\inf_{\mathrm{Diag}(z_{n+1}^{(y)}) \succeq \frac{1}{2}\beta_{a,y}\sqrt{\alpha_a}\sqrt{\alpha_a}^\top} \left\{ \max_{x_1} \sum_{y_1} \cdots \max_{x_n} \sum_{y_n} z_{n+1,x,y} \right\}.$$

In the new protocol, cheating Alice can force outcome 0 with maximum probability given by the optimal objective value of the following problem

$$\inf_{\mathrm{Diag}(\tilde{z}_{n+1}^{(y)}) \succeq \frac{1}{2}\beta'_{a,y}\sqrt{\alpha_a}\sqrt{\alpha_a}^\top} \left\{ \max_{x_1} \sum_{y_1} \cdots \max_{x_i' \in A_i'} \sum_{y_d \in B_d} \max_{x_i'' \in A_i''} \cdots \max_{x_n} \sum_{y_n} \tilde{z}_{n+1,x,y} \right\}.$$

For any z_{n+1} feasible in the first problem, we can define a solution feasible in the second problem $\tilde{z}_{n+1} := z_{n+1} \otimes [1,0]_d^\top$ with the same objective function value. Notice the same argument holds if we switch β_0 with β_1 and β_0' with β_1', i.e., if Alice wants outcome 1. Since these are minimization problems, Alice can cheat no more in the new protocol.

2.1.4 SOCP formulations of quantum strategies

We show that the reduced SDPs can be modelled using a simpler class of optimization problems. We give details in this subsection and explain the significance to solving these problems numerically.

We start by first explaining how to model fidelity as an SOCP. Suppose we are given the problem

$$\max\{\sqrt{\mathrm{F}(p,q)} : q \in \mathbb{R}^n_+ \cap S\} = \max\left\{ \sum_{i=1}^{n} \sqrt{p_i}\, t_i : t_i^2 \leq q_i,\, \forall i \in \{1,\ldots,n\},\, q \in \mathbb{R}^n_+ \cap S \right\},$$

where $p \in \mathbb{R}^n_+$ and $S \subseteq \mathbb{R}^n$. We can replace $t_i^2 \leq q_i$ with the equivalent constraint $(1/2, q_i, t_i) \in \mathrm{RSOC}^3$, for all $i \in \{1,\ldots,n\}$. Therefore, we can maximize the fidelity using n rotated second-order cone constraints.

For the same reason, we can use second-order cone programming to solve a problem of the form

$$\max\left\{ \sum_{j=1}^{m} a_j \sqrt{\mathrm{F}(p_j,q_j)} : (q_1,\ldots,q_m) \in \mathbb{R}^{mn}_+ \cap S' \right\},$$

where $a \in \mathbb{R}^m_+$ and $S' \subseteq \mathbb{R}^{mn}$. However, this does not apply directly to the reduced problems in Subsection 2.1.2 since we need to optimize over a linear combination of fidelities and $f(x) = x^2$ is not a concave function. For example, Alice's reduced problem is of the form

$$\max\left\{ \sum_{j=1}^{m} a_j\, \mathrm{F}(p_j,q_j) : (q_1,\ldots,q_m) \in \mathbb{R}^{mn}_+ \cap S' \right\}.$$

The root of this problem arises from the fact that the fidelity function, which is concave, is a composition of a concave function with a convex function, thus we cannot break it into these two steps. Even though the above analysis does not work to capture the reduced cheating SDPs as SOCPs, it does have a desirable property that it only uses $O(n)$ second-order cone constraints and perhaps this formulation will be useful for future applications.

We now explain how to model the reduced problems as SOCPs directly.

Lemma 2.1.8. *For $p, q \in \mathbb{R}^n_+$, we have*

$$\mathrm{F}(p,q) = \max\left\{ \frac{1}{\sqrt{2}} \sum_{i,j=1}^{n} \sqrt{p_i p_j}\, t_{i,j} : (q_i, q_j, t_{i,j}) \in \mathrm{RSOC}^3,\ \textit{for all } i,j \in \{1,\ldots,n\} \right\}.$$

Proof. For every $i, j \in \{1,,\ldots,n\}$, we have $(q_i, q_j, t_{i,j}) \in \mathrm{RSOC}^3$ if and only if $q_i, q_j \geq 0$, and $2q_iq_j \geq t_{i,j}^2$. Thus, $t_{i,j} = \sqrt{2q_iq_j}$ is optimal with objective function value

$$\frac{1}{\sqrt{2}} \sum_{i,j=1}^{n} \sqrt{2p_ip_jq_iq_j} = \mathrm{F}(p,q),$$

as desired. □

This lemma provides an SOCP representation for the hypograph of the fidelity function. Recall that the hypograph of a concave function is a convex set. Also, the dimension of the hypograph of $\mathrm{F}(\cdot, q) : \mathbb{R}^n_+ \to \mathbb{R}$ is equal to n (assuming $q > 0$). Since the hypograph is $O(n)$-dimensional and convex, there exists a *self-concordant barrier function* for the set with complexity parameter $O(n)$, shown by Nesterov and Nemirovski [NN94]. This allows the derivation of interior-point methods for the underlying convex optimization problem which use $O(\sqrt{n}\log(1/\epsilon))$ iterations, where ϵ is an accuracy parameter. The above lemma uses $\Omega(n^2)$ second-order cone constraints and the usual treatment of these "cone constraints" with optimal self-concordant barrier functions lead to interior-point methods with an iteration complexity bound of $O(n\log(1/\epsilon))$. It is conceivable that there exist better convex representations of the hypograph of the fidelity function than the one we provided in Lemma 2.1.8.

Finding efficient SOCP formulations of the fidelity

We use the analysis of the fidelity function in Subsection 1.4.7 to develop an SOCP formulation of the fidelity function using $O(n)$ SOC constraints as opposed to the $\Omega(n^2)$ above. Note that this SOCP characterization is through a dual problem and does not characterize the hypograph of the fidelity function.

Lemma 2.1.9. *For $p, q \in \mathbb{R}^n_+$, we have*

$$\mathrm{F}(p,q) = \inf\{\langle v, p\rangle : \langle \xi, q\rangle \leq 1,\ (v_i, \xi_i, \sqrt{2}) \in \mathrm{RSOC}^3,\ \textit{for all } i \in \{1,\ldots,n\}\}.$$

Proof. Using Lemma 1.4.10, we have

$$\begin{aligned} \mathrm{F}(p,q) &= \max\{\langle \rho, \sqrt{q}\sqrt{q}^\top\rangle : \mathrm{diag}(\rho) = p,\ \rho \succeq 0\} \\ &= \inf\{\langle v, p\rangle : \mathrm{Diag}(v) \succeq \sqrt{q}\sqrt{q}^\top\} \\ &= \inf\{\langle v, p\rangle : \langle v^{-1}, q\rangle \leq 1,\ v > 0\}. \end{aligned}$$

Notice that

$$(v_i, \xi_i, \sqrt{2}) \in \mathrm{RSOC}^3 \iff \xi_i, v_i > 0,\ 2\xi_i v_i \geq 2 \iff \xi_i, v_i > 0,\ \xi_i \geq \frac{1}{v_i}.$$

Since we can relax $\xi_i = \dfrac{1}{v_i}$ to $\xi_i \geq \dfrac{1}{v_i}$, the result follows. □

Note, we can write the RSOC constraint above as

$$(v, \xi, \sqrt{2}) \in \mathrm{RSOC}^3 \iff \begin{bmatrix} \xi & 1 \\ 1 & v \end{bmatrix} \succeq 0.$$

We can similarly formulate the fidelity of the quantum states ρ and σ using the following SDP version of Alberti's Theorem [Alb83]

$$\mathrm{F}(\rho, \sigma) = \inf \left\{ \langle X, \rho \rangle : \langle R, \sigma \rangle \leq 1, \begin{bmatrix} X & \mathrm{I} \\ \mathrm{I} & R \end{bmatrix} \succeq 0 \right\}.$$

Note that the identity matrices on the off-diagonal blocks force X and R to be invertible.

We can now calculate the bias of a BCCF-protocol using second-order cone programming as shown in the following corollary.

Corollary 2.1.10. *For the* BCCF*-protocol defined by the parameters* $\alpha_0, \alpha_1 \in \mathrm{Prob}^A$ *and* $\beta_0, \beta_1 \in \mathrm{Prob}^B$*, we have*

$$P^*_{\mathrm{B},0} = \inf_{\substack{\langle \xi_a, \beta_a \rangle \leq 1 \\ (\xi_{a,y}, v_{a,y}, \sqrt{2}) \in \mathrm{RSOC}^3}} \left\{ \sum_{x_1} \max_{y_1} \sum_{x_2} \max_{y_2} \cdots \sum_{x_n} \max_{y_n} \sum_{a} \frac{1}{2} \alpha_{a,x} v_{a,y} \right\},$$

$$P^*_{\mathrm{B},1} = \inf_{\substack{\langle \xi_a, \beta_{\bar{a}} \rangle \leq 1 \\ (\xi_{a,y}, v_{a,y}, \sqrt{2}) \in \mathrm{RSOC}^3}} \left\{ \sum_{x_1} \max_{y_1} \sum_{x_2} \max_{y_2} \cdots \sum_{x_n} \max_{y_n} \sum_{a} \frac{1}{2} \alpha_{a,x} v_{a,y} \right\},$$

$$P^*_{\mathrm{A},0} = \inf_{\substack{\frac{\beta_{a,y}}{2} \langle \mu^{(y)}, \alpha_a \rangle \leq 1 \\ (\mu_{x,y}, z_{n+1,x,y}, \sqrt{2}) \in \mathrm{RSOC}^3}} \left\{ \max_{x_1} \sum_{y_1} \cdots \max_{x_n} \sum_{y_n} z_{n+1,x,y} \right\},$$

$$P^*_{\mathrm{A},1} = \inf_{\substack{\frac{\beta_{\bar{a},y}}{2} \langle \mu^{(y)}, \alpha_a \rangle \leq 1 \\ (\mu_{x,y}, z_{n+1,x,y}, \sqrt{2}) \in \mathrm{RSOC}^3}} \left\{ \max_{x_1} \sum_{y_1} \cdots \max_{x_n} \sum_{y_n} z_{n+1,x,y} \right\}.$$

2.1.5 Point games for BCCF-protocols

In this subsection, we develop the point games corresponding to BCCF-protocols providing us with a new perspective for studying them. In theory, one can use them to prove protocols with a certain bias exist. In Section 2.2, we use them to lower bound the bias of "classical BCCF-protocols."

We start by examining Kitaev's lower bound involving the quantities $P^*_{\text{B},1}$ and $P^*_{\text{A},0}$. Since we are concerned with strong coin-flipping, the choice of Bob desiring outcome 1 and Alice desiring outcome 0 for this part is somewhat arbitrary. However, this way we can compare them to point games for other classes of weak coin-flipping protocols (see [Moc07]). We later show that we lose no generality in choosing these two values, as we consider all four values simultaneously by viewing the point games in pairs (more on this later).

For any $\delta > 0$, we can choose $(W_1, \ldots, W_{n+1})$ feasible in the dual of Bob's cheating SDP and $(Z_1, \ldots, Z_{n+1})$ feasible in the dual of Alice's cheating SDP such that

$$\begin{aligned}
\left(P^*_{\text{B},1} + \delta\right)\left(P^*_{\text{A},0} + \delta\right) &> \langle W_1 \otimes Z_1, \text{Tr}_{A_1} |\xi_1\rangle\langle\xi_1|\rangle \\
&\geq \langle W_1 \otimes Z_2, \text{Tr}_{B_1} |\xi'_1\rangle\langle\xi'_1|\rangle \\
&\geq \langle W_2 \otimes Z_2, \text{Tr}_{A_2} |\xi_2\rangle\langle\xi_2|\rangle \\
&\geq \langle W_2 \otimes Z_3, \text{Tr}_{B_2} |\xi'_2\rangle\langle\xi'_2|\rangle \\
&\vdots \\
&\geq \langle W_{n+1} \otimes Z_{n+1}, \text{Tr}_{A'_0 \times A'} |\xi_{n+1}\rangle\langle\xi_{n+1}|\rangle \\
&\geq \langle W_{n+1} \otimes \Pi_{\text{B},0}, \text{Tr}_{B'_0 \times B'} |\xi'_{n+1}\rangle\langle\xi'_{n+1}|\rangle \\
&\geq \langle \Pi_{\text{A},1} \otimes \Pi_{\text{B},0}, |\xi_{n+2}\rangle\langle\xi_{n+2}|\rangle \\
&= 0,
\end{aligned}$$

where $|\xi_j\rangle$ and $|\xi'_j\rangle$ are equal to $|\psi\rangle|\phi\rangle$ with the spaces permuted accordingly. Note that these are dual variables from the original cheating SDPs, not the reduced version. The dual variables for the reduced version are scaled eigenvalues of the corresponding dual variables above. However, we do reconstruct Kitaev's proof above using the reduced SDPs in Section 2.3.

Recall for a quantum state σ, we denote by $\text{Prob}(X, Y, \sigma)$ the function

$$\text{Prob}(X, Y, \sigma) := \sum_{\lambda \in \text{eig}(X)} \sum_{\mu \in \text{eig}(Y)} \langle \Pi_X^{[\lambda]} \otimes \Pi_Y^{[\mu]}, \sigma \rangle \; [\lambda, \mu] \,,$$

where $[\lambda, \mu] : \mathbb{R}^2 \to \mathbb{R}$ denotes the function that takes value 1 on input (λ, μ) and 0 otherwise. We create a point game as follows:

$$\begin{array}{rcll}
p_0 & := & \text{Prob}(\Pi_{\text{A},1} \otimes \Pi_{\text{B},0}, |\xi_{n+2}\rangle\langle\xi_{n+2}|), & \\
p'_1 & := & \text{Prob}(W_{n+1} \otimes \Pi_{\text{B},0}, \text{Tr}_{B'_0 \times B'} |\xi'_{n+1}\rangle\langle\xi'_{n+1}|), & \\
p_1 & := & \text{Prob}(W_{n+1} \otimes Z_{n+1}, \text{Tr}_{A'_0 \times A'} |\xi'_{n+1}\rangle\langle\xi'_{n+1}|), & \\
p'_{(n+2)-j} & := & \text{Prob}(W_j \otimes Z_{j+1}, \text{Tr}_{B_j} |\xi'_j\rangle\langle\xi'_j|), & \text{for all } j \in \{1, \ldots, n\}, \\
p_{(n+2)-j} & := & \text{Prob}(W_j \otimes Z_j, \text{Tr}_{A_j} |\xi_j\rangle\langle\xi_j|), & \text{for all } j \in \{1, \ldots, n\},
\end{array}$$

noting that the i'th point corresponds to the i'th last message in the protocol. The reason we define point games this way is so that they always have the same starting state and it is shown later that the final point captures the two objective function values of the two dual feasible solutions. The reverse time order ensures that we always start with the same p_0 and aim to get a desirable last point, instead of the other way around.

This gives rise to the point game moves (or transitions):

$$p_0 \to p'_1 \to p_1 \to \cdots \to p'_j \to p_j \to \cdots \to p'_{n+1} \to p_{n+1},$$

which we give context to in the next subsection.

First, we calculate $\text{Prob}(W_j, Z_j, \text{Tr}_{A_j} |\xi_j\rangle\langle\xi_j|)$, for $j \in \{1, \ldots, n\}$.

Definition 2.1.11. *For a string $z \in \{0,1\}^*$, we define $p(z)$ as the probability of string z being revealed during an honest run of a fixed* BCCF*-protocol.*

Note we have $p(x_1, \ldots, x_j) = \langle \psi_{x_1,\ldots,x_j} | \psi_{x_1,\ldots,x_j} \rangle$, for all $(x_1, \ldots, x_j) \in A_1 \times \cdots \times A_j$, and $p(y_1, \ldots, y_j) = \langle \psi_{y_1,\ldots,y_j} | \psi_{y_1,\ldots,y_j} \rangle$, for all $(y_1, \ldots, y_j) \in B_1 \times \cdots \times B_j$, for $j \in \{1, \ldots, n\}$.

From the proof of the reduced problems, we can assume an optimal choice of W_j has eigenvalues $\frac{w_{j,x_1,y_1,\ldots,y_{j-1},x_j}}{p(x_1,\ldots,x_j)}$, where w_j is the corresponding variable in the dual of Bob's reduced cheating SDP. Note that $p(x_1, \ldots, x_j) = 0$ implies $w_{j,x_1,y_1,\ldots,y_{j-1},x_j} = 0$, so we do not need to worry about this case (nor the division by 0). The same argument holds in the following cases whenever there is an issue of dividing by 0. The positive eigenvalues have respective eigenspaces

$$\Pi_{W_j}^{[x_1,y_1,\ldots,y_{j-1},x_j]} := |x_1,y_1,\ldots,y_{j-1},x_j\rangle\langle x_1,y_1,\ldots,y_{j-1},x_j| \otimes |\tilde{\psi}_{x_1,\ldots,x_j}\rangle\langle\tilde{\psi}_{x_1,\ldots,x_j}|,$$

where $|\tilde{\psi}_{x_1,\ldots,x_j}\rangle$ is $|\psi_{x_1,\ldots,x_j}\rangle$ normalized. The other eigenvalues do not contribute to the points (this can be verified since these eigenvalues already contribute to probabilities adding to 1). Similarly, an optimal choice of Z_j has eigenvalues $\frac{z_{j,x_1,y_1,\ldots,x_{j-1},y_{j-1}}}{p(y_1,\ldots,y_{j-1})}$, where z_j is the corresponding variable in the dual of Alice's reduced cheating SDP, with respective eigenspaces

$$\Pi_{Z_j}^{[x_1,y_1,\ldots,x_{j-1},y_{j-1}]} := |x_1,y_1,\ldots,x_{j-1},y_{j-1}\rangle\langle x_1,y_1,\ldots,x_{j-1},y_{j-1}| \otimes |\tilde{\phi}_{y_1,\ldots,y_{j-1}}\rangle\langle\tilde{\phi}_{y_1,\ldots,y_{j-1}}|,$$

where, again, $|\tilde{\phi}_{y_1,\ldots,y_{j-1}}\rangle$ is $|\phi_{y_1,\ldots,y_{j-1}}\rangle$ normalized.

From these eigenspaces, we can compute

$$\begin{aligned}
&\langle \Pi_{W_j}^{[x_1',y_1',\ldots,y_{j-1}',x_j']} \otimes \Pi_{Z_j}^{[x_1,y_1,\ldots,x_{j-1},y_{j-1}]}, \mathrm{Tr}_{A_j}|\xi_j\rangle\langle\xi_j|\rangle \\
= \;& \delta_{x_1,x_1'}\cdots\delta_{x_{j-1},x_{j-1}'}\delta_{y_1,y_1'}\cdots\delta_{y_{j-1},y_{j-1}'}\, p(x_1,y_1,\ldots,y_{j-1},x_j).
\end{aligned}$$

Thus, we have the point

$$\begin{aligned}
p_{(n+2)-j} &:= \mathrm{Prob}(W_j \otimes Z_j, \mathrm{Tr}_{A_j}|\xi_j\rangle\langle\xi_j|) \\
&= \sum_{x_1}\sum_{y_1}\cdots\sum_{y_{j-1}}\sum_{x_j} p(x_1,y_1,\ldots,y_{j-1},x_j)\left[\frac{w_{j,x_1,y_1,\ldots,y_{j-1},x_j}}{p(x_1,\ldots,x_j)}, \frac{z_{j,x_1,y_1,\ldots,x_{j-1},y_{j-1}}}{p(y_1,\ldots,y_{j-1})}\right].
\end{aligned}$$

We can similarly calculate

$$\begin{aligned}
p'_{(n+2)-j} &:= \mathrm{Prob}(W_j, Z_{j+1}, \mathrm{Tr}_{B_j}|\xi_j'\rangle\langle\xi_j'|) \\
&= \sum_{x_1}\sum_{y_1}\cdots\sum_{y_j}\sum_{x_j} p(x_1,y_1,\ldots,x_j,y_j)\left[\frac{w_{j,x_1,y_1,\ldots,y_{j-1},x_j}}{p(x_1,\ldots,x_j)}, \frac{z_{j+1,x_1,y_1,\ldots,x_j,y_j}}{p(y_1,\ldots,y_j)}\right].
\end{aligned}$$

The first three points are different from above, they concern the last few messages in the protocol. Nonetheless, the process is the same and we can calculate them to be

$$\begin{aligned}
p_1 &= \sum_{a\in\{0,1\}}\sum_{x\in A}\sum_{y\in B} p(x,y,a)\left[v_{a,y}, \frac{z_{n+1,x,y}}{p(y)}\right], \\
p_1' &= \sum_{b\in\{0,1\}}\sum_{y\in B}\frac{1}{2}p(y,\bar{b})\,[v_{b,y},0] + \sum_{b,y}\frac{1}{2}p(y,b)\,[v_{b,y},1], \\
p_0 &= \frac{1}{2}\,[1,0] + \frac{1}{2}\,[0,1],
\end{aligned}$$

noting $z_{n+1,x,y} > 0$ when $p(y) > 0$.

We call any point game for a BCCF-protocol a BCCF*-point game*. In the next subsection, we describe rules for moving from one point to the next in any BCCF-point game yielding a protocol independent definition.

2.1.6 Describing BCCF-point games using basic moves

Below are some basic point moves (or transitions) as Mochon describes them in [Moc07].

Definition 2.1.12 (Basic moves)**.**

- *Point raising*
$$q\,[w, z] \to q\,[w, z'], \quad (z \leq z'),$$
- *Point merging*
$$q_1\,[w, z_1] + q_2\,[w, z_2] \to (q_1 + q_2)\left[w, \frac{q_1 z_1 + q_2 z_2}{q_1 + q_2}\right],$$
- *Point splitting*
$$(q_1 + q_2)\left[w, \frac{q_1 + q_2}{\left(\frac{q_1}{z_1}\right) + \left(\frac{q_2}{z_2}\right)}\right] \to q_1\,[w, z_1] + q_2\,[w, z_2], \quad (z_1, z_2 \neq 0).$$

An example of point splitting and point raising can be seen in Figure 2.5 and examples of point mergings can be seen in Figures 2.6 and 2.7. Using a slight abuse of the definition of point splitting, if we perform a point split then raise the points, we still refer to this as a point split (for reasons that will be clear later). Also, we can merge or split on more than two points by repeating the process two points at a time.

These are moves in the second coordinate (keeping the first coordinate fixed) called *vertical moves*, and we similarly define *horizontal moves* acting on the first coordinate (keeping the second coordinate fixed).

Mochon gives a rough interpretation of these moves in [Moc07]. We can think of point raising as receiving a message, point merging as generating a message, and point splitting as checking a message via quantum measurement. These interpretations apply to the family of weak coin-flipping protocols in [Moc05], and we show they also apply to BCCF-protocols.

Below are some special cases of these moves which are useful when describing BCCF-point games.

- *Probability splitting*
$$(q_1 + q_2)[z, w] \to q_1[z, w] + q_2[z, w],$$
- *Probability merging*
$$q_1[z, w] + q_2[z, w] \to (q_1 + q_2)[z, w],$$
- *Aligning*
$$q_1[z_1, w] + q_2[z_2, w] \to q_1[\max\{z_1, z_2\}, w] + q_2[\max\{z_1, z_2\}, w].$$

Probability splitting is the special case of point splitting where the resulting points have the same value and probability merging is the special case of point merging where all the points have the same value. Aligning is just raising two points to the maximum of the two (usually so a merge can be performed on the other coordinate).

We now show that each move in a BCCF-point game can be described using basic moves. Consider the first transition:
$$\frac{1}{2}[1, 0] + \frac{1}{2}[0, 1] \to \sum_{b \in \{0,1\}} \sum_{y \in B} \frac{1}{2} p(y, \bar{b})\, [v_{b,y}, 0] + \sum_{b \in \{0,1\}} \sum_{y \in B} \frac{1}{2} p(y, b)\, [v_{b,y}, 1],$$
which can be described in two steps. First,
$$\frac{1}{2}[0, 1] \to \sum_{b \in \{0,1\}} \sum_{y \in B} \frac{1}{2} p(y, b)\, [v_{b,y}, 1],$$
is just probability splitting followed by point raising (in the first coordinate). The step
$$\frac{1}{2}[1, 0] \to \sum_{b \in \{0,1\}} \sum_{y \in B} \frac{1}{2} p(y, \bar{b})\, [v_{b,y}, 0],$$
is a point splitting. To see this, we have the dual constraint
$$\mathrm{Diag}(v_a) \succeq \sqrt{\beta_{\bar{a}}}\sqrt{\beta_{\bar{a}}}^{\top}, \text{for all } a \in \{0, 1\}.$$

We have seen that this is equivalent to the condition

$$\sum_{y\in B} \frac{\beta_{\bar{a},y}}{v_{a,y}} \leq 1,$$

when $v_a > 0$, which is the condition for a point split. Technically, a point split would have this inequality satisfied with equality, but we can always raise the points such that we get an inequality. As explained earlier, we just call this a point split.

We can interpret the point raise as Alice accepting Bob's last message b, and the point split as Alice checking Bob's state at the end of the protocol using her measurement. Note that these are the last two actions of a BCCF-protocol.

We can do something similar for the second transition below

$$\sum_{b\in\{0,1\}} \sum_{y\in B} \frac{1}{2} p(y,\bar{b})\,[v_{b,y},0] + \sum_{b\in\{0,1\}} \sum_{y\in B} \frac{1}{2} p(y,b)\,[v_{b,y},1]$$
$$\to \quad \sum_{a\in\{0,1\}} \sum_{x\in A} \sum_{y\in B} p(x,y,a) \left[v_{a,y}, \frac{z_{n+1,x,y}}{p(y)}\right].$$

To get this, for every $b \in \{0,1\}$, $y \in \operatorname{supp}(\beta_b)$, we point split

$$[v_{b,y},1] \to \sum_{x\in A} \alpha_{b,x} \left[v_{b,y}, \frac{2z_{n+1,x,y}}{\beta_{b,y}}\right].$$

This is a valid point split since we have the dual constraint

$$\operatorname{Diag}\left(\frac{2z^{(y)}_{n+1}}{\beta_{b,y}}\right) \succeq \sqrt{\alpha_b}\sqrt{\alpha_b}^{\top}, \text{for all } b \in \{0,1\}, y \in \operatorname{supp}(\beta_b).$$

The points at this stage can be seen in Figure 2.5 for the special case of a four-round BCCF-protocol with $|A| = |B| = 2$ (noting that $p(y,b) = \frac{1}{2}\beta_{b,y}$).

For the other points, we perform the probability splitting:

$$\sum_{b\in\{0,1\}} \sum_{y\in B} \frac{1}{2}\, p(y,\bar{b})\,[v_{b,y},0] \to \sum_{b\in\{0,1\}} \sum_{y\in B} \sum_{x\in A} \frac{1}{2}\, p(y,\bar{b})\,\alpha_{b,x}\,[v_{b,y},0]\,,$$

yielding the current state

$$\sum_{b\in\{0,1\}} \sum_{y\in B} \sum_{x\in A} \frac{1}{2}\,\alpha_{b,x} \left(p(y,\bar{b})\,[v_{b,y},0] + p(y,b)\left[v_{b,y}, \frac{2z_{n+1,x,y}}{\beta_{b,y}}\right]\right).$$

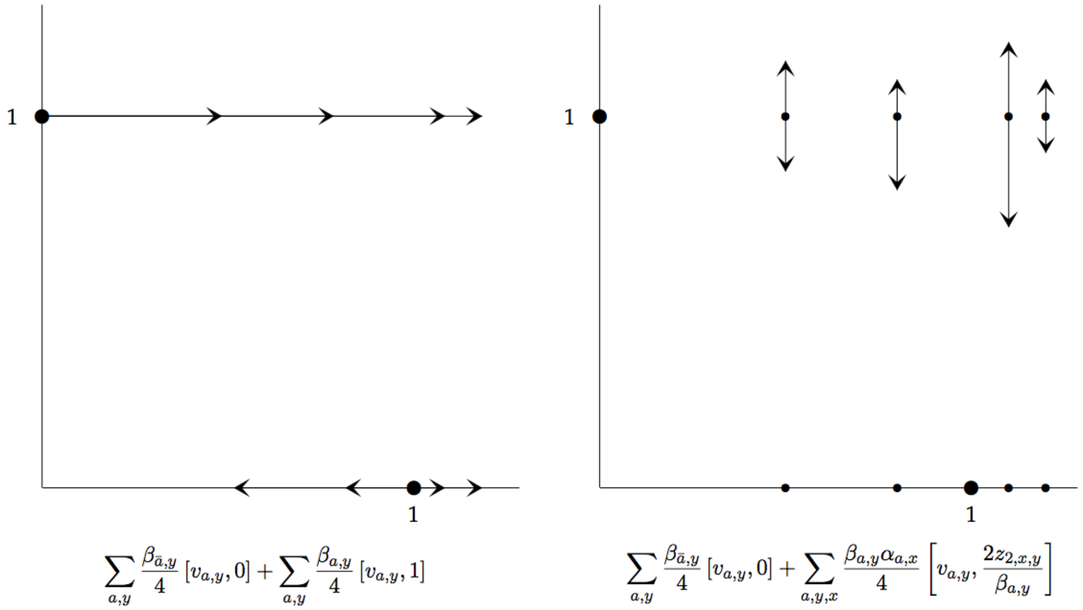

1
1
1
1
$\sum_{a,y} \frac{\beta_{\bar{a},y}}{4} [v_{a,y}, 0] + \sum_{a,y} \frac{\beta_{a,y}}{4} [v_{a,y}, 1]$
$\sum_{a,y} \frac{\beta_{\bar{a},y}}{4} [v_{a,y}, 0] + \sum_{a,y,x} \frac{\beta_{a,y}\alpha_{a,x}}{4} \left[v_{a,y}, \frac{2z_{2,x,y}}{\beta_{a,y}}\right]$

Figure 2.5: Left: Point splitting $[1, 0]$ and point raising $[0, 1]$. Right: Four point splittings. (Point labels omitted for clarity.)

Merging the part in the brackets yields

$$\sum_{b\in\{0,1\}} \sum_{y\in B} \sum_{x\in A} \frac{1}{2}\alpha_{b,x} p(y) \left[v_{b,y}, \frac{z_{n+1,x,y}}{p(y)}\right] = \sum_{a\in\{0,1\}} \sum_{y\in B} \sum_{x\in A} p(x,y,a) \left[v_{a,y}, \frac{z_{n+1,x,y}}{p(y)}\right],$$

where the quantity on the right just relabelled b as a. The transitions here were point splitting, point merging, and point raising (from the dual constraint on $z_{x,y}$, we can think of it as being a maximum over a, corresponding to a raise). These correspond to Bob checking Alice, Bob generating b, and Bob receiving a, respectively.

Fortunately, the rest of the transitions are straightforward. To explain the transition

$$\begin{aligned} &\sum_{a\in\{0,1\}} \sum_{y\in B} \sum_{x\in A} p(x,y,a) \left[v_{a,y}, \frac{z_{n+1,x,y}}{p(y)}\right] \\ \to &\sum_{y\in B} \sum_{x\in A} p(x,y) \left[\frac{w_{n,x_1,y_1,\ldots,y_{n-1},x_n}}{p(x)}, \frac{z_{n+1,x,y}}{p(y)}\right], \end{aligned}$$

all we do is merge a, then align $y_n \in B_n$ in the first coordinate. To see why this is valid, we have the dual constraint

$$w_{n,x_1,y_1,\ldots,y_{n-1},x_n} \geq \sum_{a\in\{0,1\}} \frac{1}{2}\alpha_{a,x}\, v_{a,y} = \sum_{a\in\{0,1\}} p(x,a)\, v_{a,y}.$$

This corresponds to Alice generating a and receiving Bob's message $y_n \in B_n$. This is depicted in Figure 2.6, below.

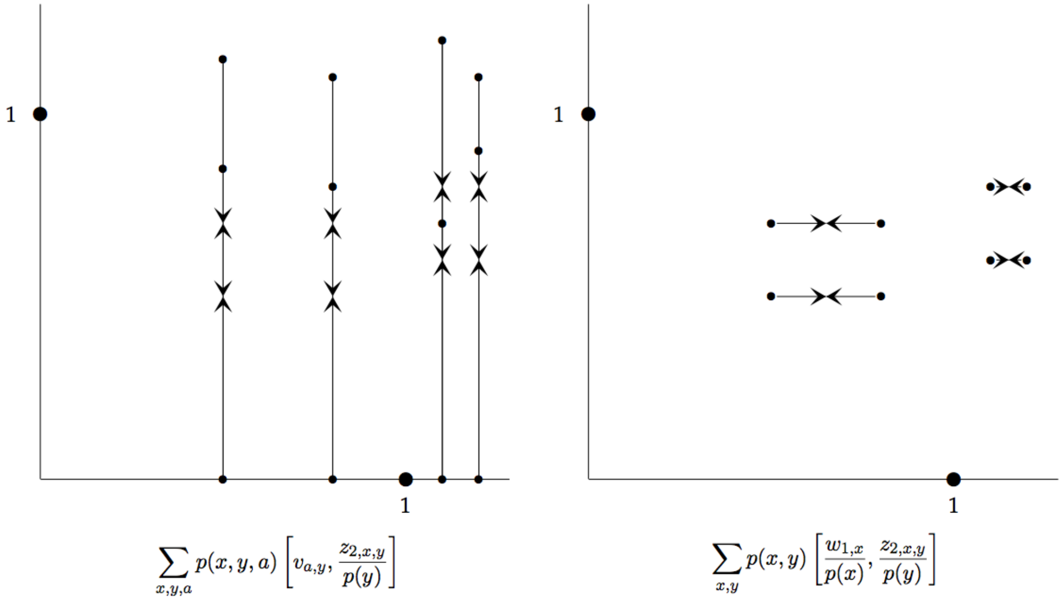

1
1
$\sum_{x,y,a} p(x,y,a)\left[v_{a,y}, \frac{z_{2,x,y}}{p(y)}\right]$
1
1
$\sum_{x,y} p(x,y)\left[\frac{w_{1,x}}{p(x)}, \frac{z_{2,x,y}}{p(y)}\right]$

Figure 2.6: Left: Eight point merges (vertical). Right: Four point merges (horizontal). (Point labels omitted for clarity.)

We show one more transition and the rest follow similarly. To show the transition

$$\sum_{y\in B}\sum_{x\in A} p(x,y)\left[\frac{w_{n,x_1,y_1,\ldots,y_{n-1},x_n}}{p(x)}, \frac{z_{n+1,x,y}}{p(y)}\right]$$

$$\to \sum_{y_1}\cdots\sum_{y_{n-1}}\sum_{x\in A} p(x_1,y_1,\ldots,y_{n-1},x_n)\left[\frac{w_{n,x_1,y_1,\ldots,y_{n-1},x_n}}{p(x)}, \frac{z_{n,x_1,y_1,\ldots,x_{n-1},y_{n-1}}}{p(y_1,\ldots,y_{n-1})}\right],$$

we merge on $y_n \in B_n$ then align $x_n \in A_n$ in the second coordinate. The dual constraint corresponding to this is

$$z_{n,x_1,y_1,\ldots,x_{n-1},y_{n-1}} \geq \sum_{y_n} z_{n+1,x,y}.$$

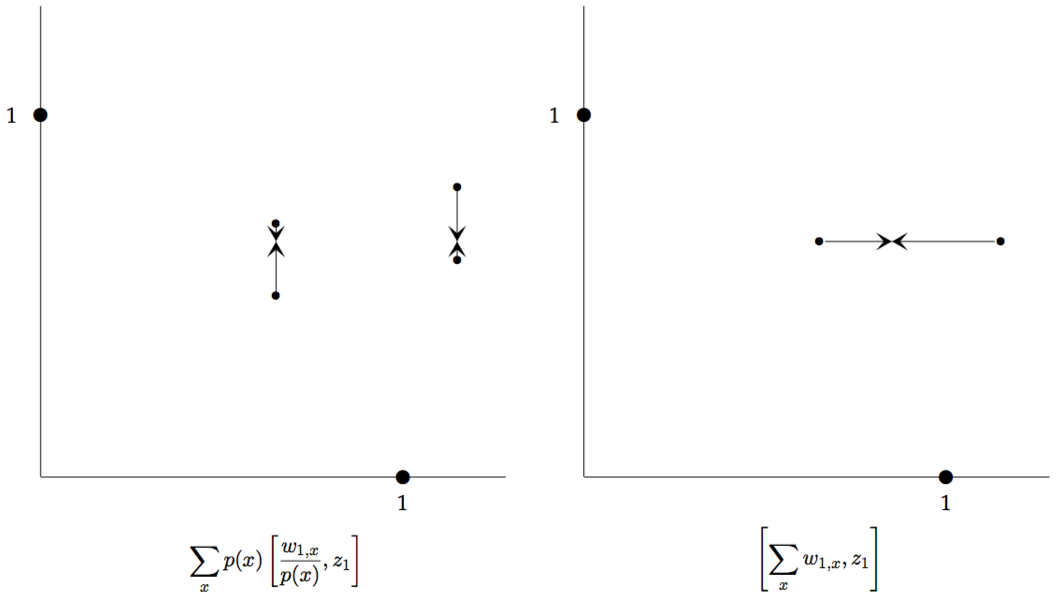
We can continue in this fashion until we get to the last point

$$\sum_{x_1} p(x_1) \left[\frac{w_{1,x_1}}{p(x_1)}, z_1 \right],$$

where z_1 is Alice's dual objective function value. If we merge on x_1, we get Bob's dual objective function value in the first coordinate

$$\left[\sum_{x_1} w_{1,x_1}, z_1 \right].$$

Therefore, if $(w_1, \ldots, w_n, v_0, v_1)$ is feasible for the dual of Bob's reduced cheating SDP and $(z_1, \ldots, z_n, z_{n+1})$ is feasible for the dual of Alice's reduced cheating SDP, then the final point of the point game is comprised of the two dual objective function values, as seen in Figure 2.7, below.



1
1
$\sum_x p(x) \left[\frac{w_{1,x}}{p(x)}, z_1 \right]$
1
1
$\left[\sum_x w_{1,x}, z_1 \right]$

Figure 2.7: Left: Two point merges (vertical). Right: Last point merge (horizontal). (Point labels omitted for clarity.)

We summarize this entire process as a list of basic moves on the following page.

Point Game 2.1.13 (BCCF-point game with final point $[\zeta_{\mathrm{B}}, \zeta_{\mathrm{A}}]$ from basic moves).

$$
\begin{aligned}
& \frac{1}{2}\,[1,0] + \frac{1}{2}\,[0,1] && \\
\to\ & \sum_{a\in\{0,1\}} \frac{1}{4}\ [1,0] + \sum_{a\in\{0,1\}} \sum_{y\in B} \frac{1}{4}\beta_{a,y}\ [0,1] && \text{prob. splitting} \\
\to\ & \sum_{a\in\{0,1\}} \sum_{y\in B} \frac{1}{4}\beta_{\bar{a},y}\ [v_{a,y},0] + \sum_{a\in\{0,1\}} \sum_{y\in B} \frac{1}{4}\beta_{a,y}\ [0,1] && \text{point splitting} \\
\to\ & \sum_{a\in\{0,1\}} \sum_{y\in B} \frac{1}{4}\beta_{\bar{a},y}\ [v_{a,y},0] + \sum_{a\in\{0,1\}} \sum_{y\in B} \frac{1}{4}\beta_{a,y}\ [v_{a,y},1] && \text{point raises} \\
\to\ & \sum_{a\in\{0,1\}} \sum_{y\in B} \frac{1}{4}\beta_{\bar{a},y}\ [v_{a,y},0] + \sum_{a\in\{0,1\}} \sum_{y\in B} \sum_{x\in A} \frac{1}{4}\beta_{a,y}\alpha_{a,x}\ \left[v_{a,y}, \frac{2z_{n+1,x,y}}{\beta_{a,y}}\right] && \text{point splitting} \\
\to\ & \sum_{a\in\{0,1\}} \sum_{y\in B} \sum_{x\in A} \left(\frac{1}{4}\beta_{\bar{a},y}\alpha_{a,x}\ [v_{a,y},0] + \frac{1}{4}\beta_{a,y}\alpha_{a,x}\ \left[v_{a,y}, \frac{2z_{n+1,x,y}}{\beta_{a,y}}\right]\right) && \text{prob. splitting} \\
=\ & \sum_{a\in\{0,1\}} \sum_{y\in B} \sum_{x\in A} \frac{1}{4}\alpha_{a,x} \left(\beta_{\bar{a},y}\ [v_{a,y},0] + \beta_{a,y}\ \left[v_{a,y}, \frac{2z_{n+1,x,y}}{\beta_{a,y}}\right]\right) && \\
\to\ & \sum_{a\in\{0,1\}} \sum_{y\in B} \sum_{x\in A} \frac{1}{4}\alpha_{a,x} \left(\sum_{b\in\{0,1\}} \beta_{b,y}\right) \left[v_{a,y}, \frac{z_{n+1,x,y}}{p(y)}\right] && \text{merges} \\
=\ & \sum_{a\in\{0,1\}} \sum_{y\in B} \sum_{x\in A} p(x,y,a) \left[v_{a,y}, \frac{z_{n+1,x,y}}{p(y)}\right] && \\
\to\ & \sum_{y\in B} \sum_{x\in A} p(x,y) \left[\frac{w_{n,x_1,y_1,\ldots,y_{n-1},x_n}}{p(x)}, \frac{z_{n+1,x,y}}{p(y)}\right] && \text{merge } a, \\
& && \text{then align } y_n \\
\to\ & \sum_{y_1,\ldots,y_{n-1}} \sum_{x\in A} p(x,y_1,\ldots,y_{n-1}) \left[\frac{w_{n,x_1,y_1,\ldots,y_{n-1},x_n}}{p(x)}, \frac{z_{n,x_1,y_1,\ldots,x_{n-1},y_{n-1}}}{p(y_1,\ldots,y_{n-1})}\right] && \text{merge } y_n, \\
& && \text{then align } x_n \\
& \vdots && \\
\to\ & \sum_{x_1} p(x_1) \left[\frac{w_{1,x_1}}{p(x_1)}, \zeta_{\mathrm{A}}\right] && \text{merge } y_1, \\
& && \text{then align } x_1 \\
\to\ & 1\,[\zeta_{\mathrm{B}}, \zeta_{\mathrm{A}}] && \text{merge } x_1.
\end{aligned}
$$

Therefore, an optimal assignment of variables in the duals of the reduced cheating SDPs corresponds to a minimal choice of ζ_{B} and ζ_{A}. We now argue that these duals attain an optimal solution. Since the optimal objective values are bounded above by 1, we can upper bound the values on all of the variables in the duals accordingly (it can be shown that $v_{a,y} \leq 2|A|$, for all $a \in \{0,1\}, y \in B$ and the rest of the variables in the four duals are bounded above by 1). Also, they are bounded below by 0 from the positive semidefiniteness constraints. Since we are optimizing a continuous function over a compact set, we have that an optimal solution exists.

From the point game above, we see that the only freedom is in how we choose the point splits, the rest of the points are determined from the merges and aligns. We have seen this idea before, in the succinct form of the duals of the reduced SDPs in Subsection 2.1.2. In each of the succinct forms of these duals, the only freedom is in how we choose to satisfy the last constraints. Once these variables were fixed, there was an obvious way to choose an optimal assignment of the rest of the variables. Coincidentally, the last constraints in each dual correspond to the point splits in the point game.

This brings us to the following protocol independent definition of BCCF-point games.

Definition 2.1.14 (BCCF-point game (protocol independent definition))**.** *A* BCCF*-point game defined on the parameters $\alpha_0, \alpha_1 \in \mathrm{Prob}^A$ and $\beta_0, \beta_1 \in \mathrm{Prob}^B$, with final point $[\zeta_{\mathrm{B}}, \zeta_{\mathrm{A}}]$, is any point game of the form*

$$p_0 := \frac{1}{2}\,[1,0] + \frac{1}{2}\,[0,1] \to p_1 \to p_2 \to \cdots \to p_m := [\zeta_{\mathrm{B}}, \zeta_{\mathrm{A}}]\,,$$

where the transitions are exactly the basic moves as described in Point Game 2.1.13.

As mentioned above, one only has the freedom to choose how the points are split at the beginning, the rest of the points are determined. Thus, every choice of point splitting yields a potentially different point game (keeping $\alpha_0, \alpha_1 \in \mathrm{Prob}^A$ and $\beta_0, \beta_1 \in \mathrm{Prob}^B$ fixed). A BCCF-point game is defined on the parameters $\alpha_0, \alpha_1 \in \mathrm{Prob}^A$ and $\beta_0, \beta_1 \in \mathrm{Prob}^B$ which are the same parameters that uniquely define a BCCF-protocol. However, there could be many point games corresponding to these same parameters. The analogous concept for BCCF-protocols is that there could be many cheating strategies for the same protocol. Of course, there is an optimal cheating strategy just as there is an optimal BCCF-point game.

The above definition is protocol independent since we have defined starting points, an ending point, and a description of how to move the points around. Indeed, the "rules" for the point moves correspond exactly to dual feasible solutions with objective function values being the two coordinates of the final point. This yields the following lemma which is the application of weak and strong duality in the language of protocols and point games.

Lemma 2.1.15. *Suppose* $[\zeta_{\mathrm{B}}, \zeta_{\mathrm{A}}]$ *is the final point of a* BCCF*-point game defined on the parameters* $\alpha_0, \alpha_1 \in \mathrm{Prob}^A$ *and* $\beta_0, \beta_1 \in \mathrm{Prob}^B$*. Then*

$$P^*_{\mathrm{B},1} \leq \zeta_{\mathrm{B}} \quad \text{and} \quad P^*_{\mathrm{A},0} \leq \zeta_{\mathrm{A}},$$

where $P^*_{\mathrm{B},1}$ *and* $P^*_{\mathrm{A},0}$ *are the optimal cheating probabilities for Bob forcing* 1 *and Alice forcing* 0*, respectively, in the corresponding* BCCF*-protocol. Moreover, there exists a* BCCF*-point game with final point* $\left[P^*_{\mathrm{B},1}, P^*_{\mathrm{A},0}\right]$*.*

In this thesis, we are concerned with bounding the bias of strong coin-flipping protocols, and therefore would like to bound all four cheating probabilities. Recall that Alice and Bob's two cheating probabilities are swapped when β_0 and β_1 are swapped. This motivates the following definition.

Definition 2.1.16 (BCCF-point game pair)**.** *Suppose we have a* BCCF*-point game defined on the parameters* $\alpha_0, \alpha_1 \in \mathrm{Prob}^A$ *and* $\beta_0, \beta_1 \in \mathrm{Prob}^B$ *with final point* $[\zeta_{\mathrm{B},0}, \zeta_{\mathrm{A},1}]$*. Also, suppose we have another* BCCF*-point game defined by the parameters* $\alpha_0, \alpha_1 \in \mathrm{Prob}^A$ *and* $\beta'_0 = \beta_1, \beta'_1 = \beta_0 \in \mathrm{Prob}^B$ *with final point* $[\zeta_{\mathrm{B},1}, \zeta_{\mathrm{A},0}]$*. We call the two point games a* BCCF*-point game pair, defined by the parameters* $\alpha_0, \alpha_1 \in \mathrm{Prob}^A$ *and* $\beta_0, \beta_1 \in \mathrm{Prob}^B$*, with final point* $[\zeta_{\mathrm{B},0}, \zeta_{\mathrm{B},1}, \zeta_{\mathrm{A},0}, \zeta_{\mathrm{A},1}]$*.*

It is worth commenting that BCCF-point game pairs are defined over certain parameters even though one of the point games in the pair is defined over swapped parameters.

Using Lemma 2.1.15, we have the following theorem.

Theorem 2.1.17. *Suppose* $[\zeta_{\mathrm{B},0}, \zeta_{\mathrm{B},1}, \zeta_{\mathrm{A},0}, \zeta_{\mathrm{A},1}]$ *is the final point of a* BCCF*-point game pair defined on the parameters* $\alpha_0, \alpha_1 \in \mathrm{Prob}^A$ *and* $\beta_0, \beta_1 \in \mathrm{Prob}^B$*. Then*

$$P^*_{\mathrm{B},0} \leq \zeta_{\mathrm{B},0}, \quad P^*_{\mathrm{B},1} \leq \zeta_{\mathrm{B},1}, \quad P^*_{\mathrm{A},0} \leq \zeta_{\mathrm{A},0}, \;\; \text{and} \;\; P^*_{\mathrm{A},1} \leq \zeta_{\mathrm{A},1},$$

where $P^*_{\mathrm{B},0}$*,* $P^*_{\mathrm{B},1}$*,* $P^*_{\mathrm{A},0}$*,* $P^*_{\mathrm{A},1}$ *are the optimal cheating probabilities for the corresponding* BCCF*-protocol. Moreover, there exists a* BCCF*-point game pair with final point*

$$[P^*_{\mathrm{B},0}, P^*_{\mathrm{B},1}, P^*_{\mathrm{A},0}, P^*_{\mathrm{A},1}].$$

2.2 A related family of classical coin-flipping protocols

In this section, we describe a family of classical protocols which is the classical counterpart to quantum BCCF-protocols. That is, we choose messages according to the underlying probability distributions (instead of in a superposition) and we have a modified cheat detection step at the end of the protocol.

Consider the following Cartesian product of finite sets $A = A_1 \times \cdots \times A_n$ which are used for Alice's first n messages to Bob. Suppose we are given two probability vectors $\alpha_0, \alpha_1 \in \mathrm{Prob}^A$. Define $A_0 := \{0, 1\}$ for Alice's committed bit and A'_0 to be a copy.

Consider the following Cartesian product of finite sets $B = B_1 \times \cdots \times B_n$ which are used for Bob's first n messages. Suppose we are given two probability vectors $\beta_0, \beta_1 \in \mathrm{Prob}^B$. Define $B_0 := \{0, 1\}$ for Bob's committed bit and B'_0 to be a copy.

We now describe the communication of the protocol.

Protocol 2.2.1 (Classical BCCF-protocol)**.**

- *Alice chooses* $a \in A_0$ *uniformly at random and samples* $x \in A$ *with probability* $\alpha_{a,x}$*.*
- *Bob chooses* $b \in B_0$ *uniformly at random and samples* $y \in B$ *with probability* $\beta_{a,y}$*.*
- *For* i *from* 1 *to* n*: Alice sends* $x_i \in A_i$ *to Bob who replies with* $y_i \in B_i$*.*
- *Alice fully reveals her bit by sending* $a \in A_0$ *to Bob. If* $x \notin \mathrm{supp}(\alpha_a)$*, Bob aborts.*
- *Bob fully reveals his bit by sending* $b \in B_0$ *to Alice. If* $y \notin \mathrm{supp}(\beta_b)$*, Alice aborts.*
- *The outcome of the protocol is* $a \oplus b$*, if no one aborts.*

The rest of this section illustrates the connections between this classical protocol and the quantum version.

2.2.1 Classical cheating strategy formulations

We can similarly formulate optimal cheating strategies as optimization problems. In this case, we use linear programming. Recall Bob's cheating polytope, denoted $\mathcal{P}_{\mathrm{B}}$, is the set of vectors $(p_1, p_2, \ldots, p_n)$ satisfying

$$\begin{array}{rcl} \mathrm{Tr}_{B_1}(p_1) & = & e_{A_1}, \\ \mathrm{Tr}_{B_2}(p_2) & = & p_1 \otimes e_{A_2}, \\ & \vdots & \\ \mathrm{Tr}_{B_n}(p_n) & = & p_{n-1} \otimes e_{A_n}, \\ p_j & \in & \mathbb{R}_+^{A_1 \times B_1 \times \cdots \times A_j \times B_j}, \text{ for all } j \in \{1, \ldots, n\}, \end{array}$$

where e_{A_j} denotes the vector of all ones in the corresponding space $\mathbb{C}^{A_j}$. Alice's cheating polytope, denoted $\mathcal{P}_{\mathrm{A}}$, is the set of vectors $(s_1, s_2, \ldots, s_n, s)$ satisfying

$$\begin{array}{rcl} \mathrm{Tr}_{A_1}(s_1) & = & 1, \\ \mathrm{Tr}_{A_2}(s_2) & = & s_1 \otimes e_{B_1}, \\ & \vdots & \\ \mathrm{Tr}_{A_n}(s_n) & = & s_{n-1} \otimes e_{B_{n-1}}, \\ \mathrm{Tr}_{A'_0}(s) & = & s_n \otimes e_{B_n}, \\ s_1 & \in & \mathbb{R}_+^{A_1}, \\ s_j & \in & \mathbb{R}_+^{A_1 \times B_1 \times \cdots \times B_{j-1} \times A_j}, \text{ for all } j \in \{2, \ldots, n\}, \\ s & \in & \mathbb{R}_+^{A \times B \times A'_0}, \end{array}$$

where e_{B_j} is the vector of all ones in the corresponding space $\mathbb{C}^{B_j}$.

We use these to characterize classical cheating strategies in the lemma below.

Lemma 2.2.2. *The four optimal cheating probabilities in the classical protocol are given by the optimal objective values of the following linear programs:*

$$P^*_{\mathrm{B},0} = \max \left\{ \frac{1}{2} \sum_{a \in A'_0} \sum_{y \in \mathrm{supp}(\beta_a)} \sum_{x \in A} \alpha_{a,x} \, p_{n,x,y} : (p_1, \ldots, p_n) \in \mathcal{P}_{\mathrm{B}} \right\},$$

$$P^*_{\mathrm{B},1} = \max \left\{ \frac{1}{2} \sum_{a \in A'_0} \sum_{y \in \mathrm{supp}(\beta_{\bar{a}})} \sum_{x \in A} \alpha_{a,x} \, p_{n,x,y} : (p_1, \ldots, p_n) \in \mathcal{P}_{\mathrm{B}} \right\},$$

$$P^*_{\mathrm{A},0} = \max \left\{ \frac{1}{2} \sum_{a \in A'_0} \sum_{y \in B} \sum_{x \in \mathrm{supp}(\alpha_a)} \beta_{a,y} s_{a,x,y} : (s_1, \ldots, s_n, s) \in \mathcal{P}_\mathrm{A} \right\},$$

$$P^*_{\mathrm{A},1} = \max \left\{ \frac{1}{2} \sum_{a \in A'_0} \sum_{y \in B} \sum_{x \in \mathrm{supp}(\alpha_{\bar{a}})} \beta_{a,y} s_{a,x,y} : (s_1, \ldots, s_n, s) \in \mathcal{P}_\mathrm{A} \right\}.$$

Proof. We shall prove this for the case of cheating Bob as the case for cheating Alice is almost identical. By examining Alice's cheat detection, we see that if we switch the roles of β_0 and β_1 then we also switch $P^*_{\mathrm{B},0}$ and $P^*_{\mathrm{B},1}$, so we only need to prove the $P^*_{\mathrm{B},0}$ case.

After receiving the first message from Alice, Bob must choose a message to send. He can do this probabilistically by choosing $y_1 \in B_1$ with probability p_{1,x_1,y_1}, yielding the first constraint in Bob's cheating polytope. Notice that his message can depend on Alice's first message. We can similarly argue that the probabilities with which he chooses the rest of his messages are captured by the rest of the constraints in the cheating polytope with the exception of the last message. For the last message, we assume that Bob replies with $b = a$, where $a \in A_0$ was Alice's last message, if he desires outcome 0 and $b = \bar{a}$ otherwise. Therefore, this decision is deterministic and is not represented by the cheating polytope.

All that remains is to explain the objective function. Since Bob chooses his last message deterministically, the quantity $\frac{1}{2}\alpha_{a,x}\, p_{n,x,y}$ is the probability that Alice reveals (x, a) and Bob reveals (y, a). If he reveals y when $\beta_{a,y} = 0$, he gets caught cheating, otherwise, his choice of b is accepted. Therefore the objective function captures the total probability Alice accepts an outcome of 0. □

These are very similar to the quantum cheating probabilities except for the nonlinearity in the objective functions. For example, in the quantum setting, cheating Alice's objective function is

$$\frac{1}{2} \sum_{a \in \{0,1\}} \sum_{y \in B} \beta_{a,y} \, \mathrm{F}(s^{(a,y)}, \alpha_a)$$

and for the classical setting, it is

$$\frac{1}{2} \sum_{a \in \{0,1\}} \sum_{y \in B} \beta_{a,y} \, \langle s^{(a,y)}, e_{\mathrm{supp}(\alpha_a)} \rangle,$$

where $e_{\mathrm{supp}(\alpha_a)}$ is the $0, 1$ vector taking value 1 only on the support of α_a. We have a similar observation for Bob. What is surprising is that we can capture the communication for both settings with the same respective cheating polytopes.

To better understand this connection, we can write the objective function of Alice's reduced cheating SDP as

$$\frac{1}{2} \sum_{a \in \{0,1\}} \sum_{y \in B} \beta_{a,y} \, \langle \sqrt{s^{(a,y)}} \sqrt{s^{(a,y)}}^\top, \sqrt{\alpha_a} \sqrt{\alpha_a}^\top \rangle.$$

Then the objective function for Alice's LP can be recovered if we replace $\sqrt{\alpha_a}\sqrt{\alpha_a}^\top$ with $\mathrm{Diag}(e_{\mathrm{supp}(\alpha_a)})$. Suppose we define a new projection

$$\Pi_{\mathrm{B},0} := \sum_{a \in \{0,1\}} |a\rangle\langle a| \otimes |a\rangle\langle a| \otimes \mathrm{Diag}(e_{\mathrm{supp}(\alpha_a)}) \otimes \mathrm{I}_{B'}.$$

A quick check shows that we can repeat the entire proof of the reduced cheating problems with this new projection if we also replace each occurrence of $\sqrt{\alpha_a}\sqrt{\alpha_a}^\top$ with $\mathrm{Diag}(e_{\mathrm{supp}(\alpha_a)})$. Similar statements can be made if we redefine the other projections as

$$\Pi_{\mathrm{B},1} := \sum_{a \in \{0,1\}} |\bar{a}\rangle\langle \bar{a}| \otimes |a\rangle\langle a| \otimes \mathrm{Diag}(e_{\mathrm{supp}(\alpha_a)}) \otimes \mathrm{I}_{B'},$$

$$\Pi_{\mathrm{A},0} := \sum_{a \in \{0,1\}} |a\rangle\langle a| \otimes |a\rangle\langle a| \otimes \mathrm{Diag}(e_{\mathrm{supp}(\beta_a)}) \otimes \mathrm{I}_{A'},$$

$$\Pi_{\mathrm{A},1} := \sum_{a \in \{0,1\}} |\bar{a}\rangle\langle \bar{a}| \otimes |a\rangle\langle a| \otimes \mathrm{Diag}(e_{\mathrm{supp}(\beta_a)}) \otimes \mathrm{I}_{A'}.$$

This proves two things. First, it proves that if we weaken the quantum cheat detection, we recover the optimal cheating probabilities for the corresponding classical protocol. Second, it gives us a recipe for developing the point games. Notice that the eigenvalues of the dual variables are the same as in the quantum case, it is just that we have the stronger constraints:

$$\begin{array}{rcr} \mathrm{Diag}(v_a) \succeq \mathrm{Diag}(e_{\mathrm{supp}(\beta_{\bar{a}})}) & \text{compared to} & \mathrm{Diag}(v_a) \succeq \sqrt{\beta_{\bar{a}}}\sqrt{\beta_{\bar{a}}}^\top, \\ \mathrm{Diag}(z_{n+1}^{(y)}) \succeq \frac{1}{2}\beta_{a,y}\mathrm{Diag}(e_{\mathrm{supp}(\alpha_a)}) & \text{compared to} & \mathrm{Diag}(z_{n+1}^{(y)}) \succeq \frac{1}{2}\beta_{a,y}\sqrt{\alpha_a}\sqrt{\alpha_a}^\top. \end{array}$$

Any solution of the constraints on the left satisfies the respective constraint on the right since

$$\mathrm{Diag}(e_{\mathrm{supp}(\beta_{\bar{a}})}) \succeq \sqrt{\beta_{\bar{a}}}\sqrt{\beta_{\bar{a}}}^\top \quad \text{and} \quad \frac{1}{2}\beta_{a,y}\mathrm{Diag}(e_{\mathrm{supp}(\alpha_a)}) \succeq \frac{1}{2}\beta_{a,y}\sqrt{\alpha_a}\sqrt{\alpha_a}^\top.$$

Since the feasible region is smaller in the classical case, we get that the optimal objective value cannot be less than the quantum case. This makes sense since the classical protocol has a weaker cheat detection step and we could have larger cheating probabilities. We can think of the classical case having more general strategies since the cheat detection step in the quantum version rules out certain strategies from being optimal. In this sense, the classical primal feasible regions are larger and the classical dual feasible regions are smaller. This is similar to the relationship between the duality of convex sets. We have that $C_1 \subseteq C_2$ implies $C_1^* \supseteq C_2^*$ and the converse holds if C_1 and C_2 are closed convex cones. This relationship is depicted in Figure 2.8, below.

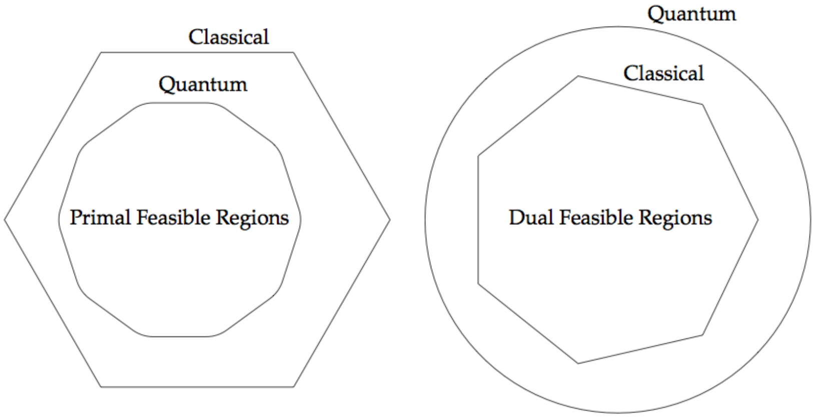


Figure 2.8: Relationship between primal and dual feasible regions. Roughly speaking, duality reverses containment.

2.2.2 Point games for classical BCCF-protocols and security analysis

In this subsection, we develop the classical analog to the quantum BCCF-point games. Using these "classical point games," we prove that at least one party can cheat with probability 1 in any classical BCCF-protocol. A closer analysis shows that both cannot cheat with probability 1, which holds true for quantum BCCF-protocols as well.

Since point games are defined in terms of dual SDPs, we use the above embedding of the classical cheating LPs into SDPs to construct classical BCCF-point games. Due to the similarities, very little about the quantum BCCF-point games needs to be changed to attain classical BCCF-point games; we only need to change the definitions of Alice and Bob's projections. Of course, the dual solutions may be different due to the stronger constraints for the classical version. The only differences are in the first few points (corresponding to the last few steps in Kitaev's proof that involve the projections). A quick calculation shows that these points are the same as well. The reason for this is because, in Bob's projections, we replace $|\psi_a\rangle\langle\psi_a|$ with $\mathrm{Diag}(e_{\mathrm{supp}(\alpha_a)}) \otimes \mathrm{I}_{A'}$, but they have the same inner product with the honest state of the protocol

$$\langle |\psi_a\rangle\langle\psi_a|, |\psi_a\rangle\langle\psi_a| \rangle = \langle |\psi_a\rangle\langle\psi_a|, \mathrm{Diag}(e_{\mathrm{supp}(\alpha_a)}) \otimes \mathrm{I}_{A'} \rangle = 1.$$

A similar argument holds for Alice's projections as well.

Thus, the only difference between the classical point games are the values of the points, which are derived from slightly different dual constraints. Let us examine the point splits. In the quantum case, these are derived from the constraints

$$\mathrm{Diag}(v_a) \succeq \sqrt{\beta_{\bar{a}}}\sqrt{\beta_{\bar{a}}}^\top \quad \text{and} \quad \mathrm{Diag}(z_{n+1}^{(y)}) \succeq \frac{1}{2}\beta_{a,y}\sqrt{\alpha_a}\sqrt{\alpha_a}^\top, \ \forall a \in \{0,1\}, y \in B.$$

In the classical case, the corresponding constraints are

$$\mathrm{Diag}(\tilde{v}_a) \succeq \mathrm{Diag}(e_{\mathrm{supp}(\beta_{\bar{a}})}) \quad \text{and} \quad \mathrm{Diag}(\tilde{z}_{n+1}^{(y)}) \succeq \frac{1}{2}\beta_{a,y}\mathrm{Diag}(e_{\mathrm{supp}(\alpha_a)}), \ \forall a \in \{0,1\}, y \in B.$$

It is easy to see that $\tilde{v}_a = e_{\mathrm{supp}(\beta_{\bar{a}})}$ and

$$\tilde{z}_{n+1,x,y} = \begin{cases} \frac{1}{2}\beta_{0,y} & \text{if} \quad x \in \mathrm{supp}(\alpha_0) \setminus \mathrm{supp}(\alpha_1), \\ \frac{1}{2}\beta_{1,y} & \text{if} \quad x \in \mathrm{supp}(\alpha_1) \setminus \mathrm{supp}(\alpha_0), \\ \frac{1}{2}\max_{a\in\{0,1\}}\{\beta_{a,y}\} & \text{if} \quad x \in \mathrm{supp}(\alpha_0) \cap \mathrm{supp}(\alpha_1), \\ 0 & \text{otherwise}, \end{cases}$$

are optimal assignments of these variables. Recall the two point splittings:

$$\frac{1}{2}[1,0] \to \sum_{b\in\{0,1\}}\sum_{y\in B}\frac{1}{4}\beta_{\bar{a},y}\,[\tilde{v}_{a,y}, 0] \quad \text{and} \quad [v_{a,y}, 1] \to \sum_{x\in A}\alpha_{a,x}\left[v_{a,y}, \frac{2\tilde{z}_{n+1,x,y}}{\beta_{a,y}}\right].$$

We see that these are probability splittings in this case (with possibly a point raise in the case of $x \in \text{supp}(\alpha_0) \cap \text{supp}(\alpha_1)$). These probability splittings are in contrast to the point splittings in the quantum case. The rest of the constraints are the same as in the quantum case and correspond to point merging, probability merging, and aligning. Therefore, the only difference between quantum BCCF-point games and the classical version is that non-trivial point splittings are allowed in the quantum version. Therefore, we get the following definition.

Definition 2.2.3 (Classical BCCF-point game (protocol independent definition))**.** *A classical* BCCF*-point game defined on the parameters $\alpha_0, \alpha_1 \in \text{Prob}^A$ and $\beta_0, \beta_1 \in \text{Prob}^B$, with final point $[\zeta_{\text{B}}, \zeta_{\text{A}}]$, is a quantum* BCCF*-point game defined by the same parameters and having the same final point but the point splittings are trivial (i.e., they are probability splittings).*

Using this definition, we define *classical* BCCF*-point game pairs* analogously to the quantum version.

To complete the picture, we now present the classical version of Theorem 2.1.17.

Theorem 2.2.4. *Suppose $[\zeta_{\text{B},0}, \zeta_{\text{B},1}, \zeta_{\text{A},0}, \zeta_{\text{A},1}]$ is the final point of a classical* BCCF*-point game pair defined on the parameters $\alpha_0, \alpha_1 \in \text{Prob}^A$ and $\beta_0, \beta_1 \in \text{Prob}^B$. Then*

$$P^*_{\text{B},0} \leq \zeta_{\text{B},0}, \quad P^*_{\text{B},1} \leq \zeta_{\text{B},1}, \quad P^*_{\text{A},0} \leq \zeta_{\text{A},0}, \;\text{ and }\; P^*_{\text{A},1} \leq \zeta_{\text{A},1},$$

where $P^*_{\text{B},0}$, $P^*_{\text{B},1}$, $P^*_{\text{A},0}$, $P^*_{\text{A},1}$ are the optimal cheating probabilities for the corresponding classical BCCF*-protocol. Moreover, there exists a classical* BCCF*-point game pair with final point*

$$[P^*_{\text{B},0}, P^*_{\text{B},1}, P^*_{\text{A},0}, P^*_{\text{A},1}].$$

Figure 2.9 (on the next page) depicts the intricate connections between quantum and classical BCCF-protocols and their point games.

Security analysis of classical BCCF-protocols

We start by giving an alternative proof that these classical protocols have bias $\varepsilon = 1/2$ using the language of point games.

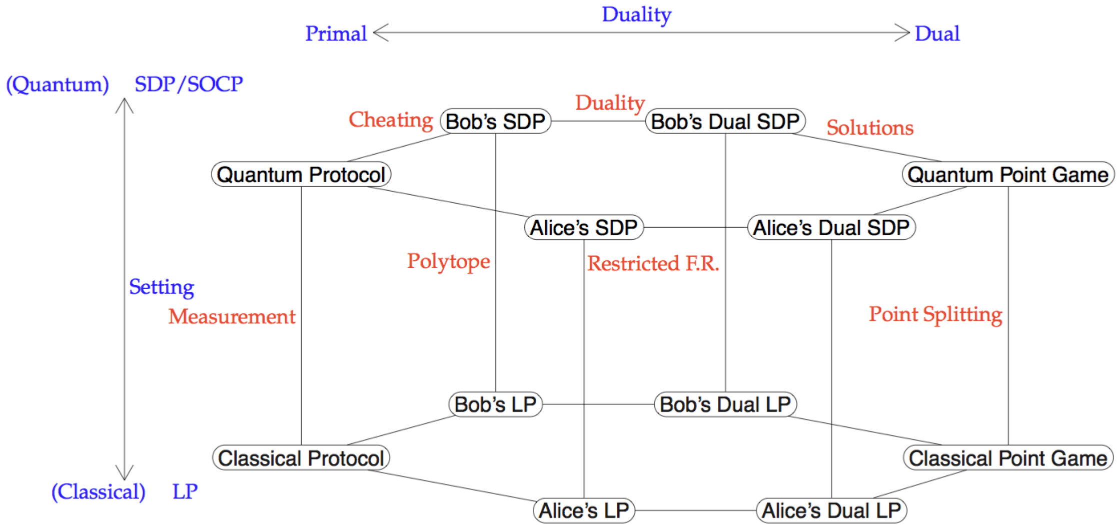


Figure 2.9: Quantum and classical BCCF-protocol crystal structure.

Lemma 2.2.5. *Suppose we have the following point game*

$$p_0 := \frac{1}{2}[0,1] + \frac{1}{2}[1,0] \to p_1 \to \cdots \to p_{m-1} \to p_m := [\zeta_{\mathrm{B}}, \zeta_{\mathrm{A}}],$$

where each move is either point raising, point merging, probability merging, or probability splitting. Then $\zeta_{\mathrm{B}} \geq 1$ *or* $\zeta_{\mathrm{A}} \geq 1$.

Proof. Suppose for a contradiction that $\zeta_{\mathrm{B}}, \zeta_{\mathrm{A}} < 1$ and let $i \in \{1, \ldots, m\}$ be the smallest index such that p_i has a point of the form $[\zeta_{\mathrm{B},i}, \zeta_{\mathrm{A},i}]$ with $\zeta_{\mathrm{B},i}, \zeta_{\mathrm{A},i} < 1$. Now, since p_{i-1} has no such points, $[\zeta_{\mathrm{B},i}, \zeta_{\mathrm{A},i}]$ could not have been generated from a point raise, a probability merge, nor a probability split. Thus, $p_{i-1} \to p_i$ must be a point merge and suppose without loss of generality it acted on the first coordinate. Then p_{i-1} has two points $q_1 [\zeta_1, \zeta_{\mathrm{A},i}]$ and $q_2 [\zeta_2, \zeta_{\mathrm{A},i}]$ with

$$\frac{q_1 \zeta_1 + q_2 \zeta_2}{q_1 + q_2} = \zeta_{\mathrm{B},i} < 1 \implies \zeta_1 < 1 \text{ or } \zeta_2 < 1,$$

a contradiction to the minimality of i. □

Using the above lemma and Theorem 2.2.4, we have the following corollary.

Corollary 2.2.6. *Every classical* BCCF*-protocol has bias* $\varepsilon = 1/2$.

There are two special cases of classical protocols we consider in greater detail. Recall the points in the point game (before merging on a in the first coordinate)

$$\sum_{a\in\{0,1\}}\sum_{y\in B}\sum_{x\in A} p(x,y,a)\left[v_{a,y}, \frac{z_{n+1,x,y}}{p(y)}\right]. \tag{2.1}$$

The first case we consider is when $\alpha_0, \alpha_1, \beta_0, \beta_1 > 0$. Then we can set $v_{a,y} = 1$ for all $a \in \{0,1\}, y \in B$ and $z_{n+1,x,y} = \frac{1}{2}\max_{a\in\{0,1\}} \beta_{a,y}$ for all $a \in \{0,1\}, x \in A, y \in B$. After the merges and aligns, we have the final point being

$$\left[1, \sum_{y\in B} \max_{a\in\{0,1\}} \frac{1}{2}\beta_{a,y}\right] = \left[1, \frac{1}{2} + \frac{1}{2}\Delta(\beta_0,\beta_1)\right],$$

using Lemma 1.4.14. We can see that this is a BCCF-point game with an optimal assignment of dual variables. Thus, Bob can cheat towards 1 perfectly and Alice can force 0 with probability $\frac{1}{2} + \frac{1}{2}\Delta(\beta_0,\beta_1)$ as seen in Figure 2.10, below.

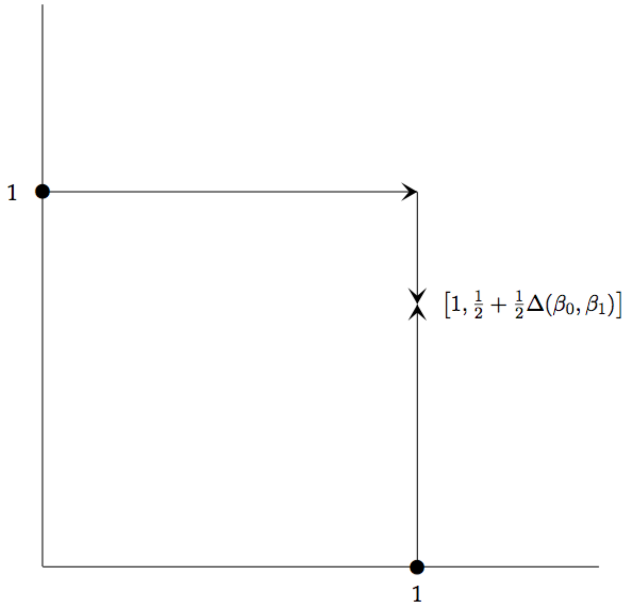


Figure 2.10: Classical BCCF-point game corresponding to a BCCF-protocol favouring cheating Bob.

These two quantities are invariant under switching β_0 and β_1, thus $P^*_{\text{B},0} = P^*_{\text{B},1} = 1$ and $P^*_{\text{A},0} = P^*_{\text{A},1} = \frac{1}{2} + \frac{1}{2}\Delta(\beta_0, \beta_1)$. The corresponding optimal cheating strategies in the classical BCCF-protocol are obvious by noticing the cheat detection step does nothing when the vectors have full support. Bob can send anything during the first n messages and then return $b = a$. Alice can send a corresponding to her best guess of b from her information about $y \in B$, i.e., she can cheat with the probability she can infer b from $y \in B$. An interesting observation is that since an optimal solution for the (classical) dual LPs is feasible in the (quantum) dual SDPs, we have that $P^*_{\text{A},0}, P^*_{\text{A},1} \leq \frac{1}{2} + \frac{1}{2}\Delta(\beta_0, \beta_1)$ for every quantum BCCF-protocol as well. This can be interpreted as follows. Suppose we change the order of the messages in the BCCF-protocol in Alice's favour, so that Bob's first n messages are sent first, followed by *all* of Alice's messages, then finally Bob's last message. Then Alice's new cheating probability would be $\frac{1}{2} + \frac{1}{2}\Delta(\beta_0, \beta_1)$ and would be an obvious upper bound on the amount she can cheat in the original protocol (since she gets information about b sooner than intended). This argument works for the classical and quantum versions.

It may seem that classical protocols favour a cheating Bob, but this is not always the case. Consider the case when $\beta_0 \perp \beta_1$ and $\alpha_0, \alpha_1 > 0$. Then we have $\frac{z_{n+1,x,y}}{p(y)} = 1$ for all $y \in \text{supp}(\beta_0) \cup \text{supp}(\beta_1)$, thus the second coordinate equals 1 for all points in (2.1), and remains that way until the end of the point game. This proves Alice can cheat with probability 1, which is obvious since Bob's first message fully reveals b and she can always pass the cheat detection step. The extent to which Bob can cheat depends on the choice of α_0 and α_1 and can be calculated as

$$\sum_{x_1} \max_a \sum_{x_2} \cdots \sum_{x_n} \frac{1}{2}\alpha_{a,x} = \frac{1}{2} + \frac{1}{2}\Delta\left(\text{Tr}_{A_2 \times \cdots \times A_n}(\alpha_0), \text{Tr}_{A_2 \times \cdots \times A_n}(\alpha_1)\right),$$

using Lemma 1.4.14. This is a distance measure between the two marginal distributions over Alice's first message x_1. This point game is depicted in Figure 2.11, on the next page.

Bob can cheat with this probability since he can choose b equal to his best guess for a from his information about x_1. Once his first message is sent, he must keep his choice of b or he will be caught cheating with certainty. These cheating probabilities do not depend

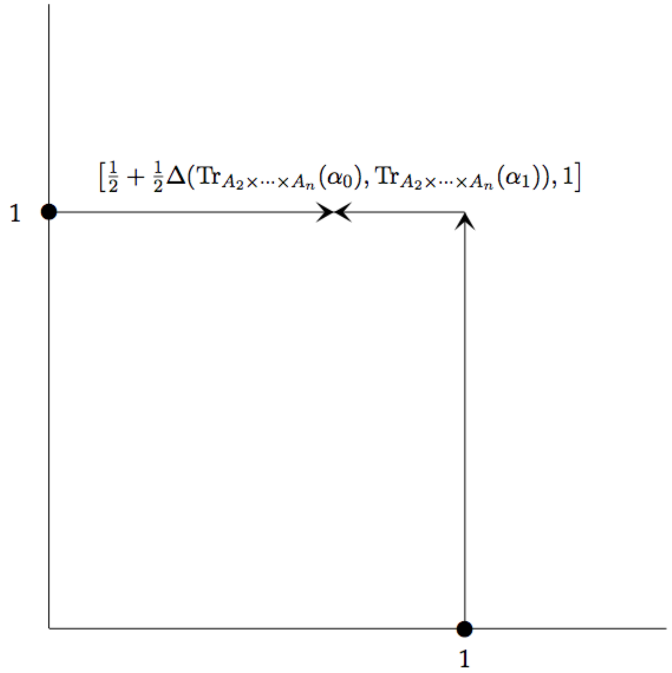

$[\frac{1}{2}+\frac{1}{2}\Delta(\mathrm{Tr}_{A_2\times\cdots\times A_n}(\alpha_0),\mathrm{Tr}_{A_2\times\cdots\times A_n}(\alpha_1)),1]$
1
1

Figure 2.11: Classical BCCF-point game corresponding to a BCCF-protocol favouring cheating Alice.

on β_0 or β_1, so we have $P^*_{\mathrm{A},0} = P^*_{\mathrm{A},1} = 1$ and

$$P^*_{\mathrm{B},0} = P^*_{\mathrm{B},1} = \frac{1}{2} + \frac{1}{2}\Delta\left(\mathrm{Tr}_{A_2\times\cdots\times A_n}(\alpha_0), \mathrm{Tr}_{A_2\times\cdots\times A_n}(\alpha_1)\right).$$

Therefore, a classical BCCF-protocol could favour either party. This raises the question: Can we find a BCCF-protocol such that both parties can perfectly control the outcome? We now argue that no such classical, and hence no such quantum, BCCF-protocol exists. Assume for a contradiction that this is the case. Then we must have

$$1 = P^*_{\mathrm{A},0} \leq \frac{1}{2} + \frac{1}{2}\Delta(\beta_0, \beta_1) \leq 1$$

which implies $\beta_0 \perp \beta_1$. The only way for Bob to cheat with probability 1 is to have complete information about a after Alice's first message, implying $\mathrm{Tr}_{A_2\times\cdots\times A_n}(\alpha_0) \perp \mathrm{Tr}_{A_2\times\cdots\times A_n}(\alpha_1)$. This can only be the case when $\alpha_0 \perp \alpha_1$ and in this case, we have argued before that Alice must stick to her choice of a after her first message. Since she has no information about b before the start of the protocol, she can only cheat with probability $1/2$, a contradiction. We have proved the following theorem.

Theorem 2.2.7. *In any quantum* BCCF*-protocol, at most one party can cheat with probability* 1*. In any classical* BCCF*-protocol, exactly one party can cheat with probability* 1*.*

2.2.3 Extreme points of the cheating polytopes

This subsection examines the extreme points of Alice and Bob's cheating polytopes which appear in both the quantum and classical cheating strategy formulations. We show that deterministic strategies correspond to the extreme points of the cheating polytopes. One can argue this directly from the properties of the protocol. However, we give a strictly algebraic proof based on the properties of the cheating polytopes.

Definition 2.2.8. *An extreme point of a convex set C is a point $x \in C$ such that if $x = \lambda y + (1-\lambda)z$, for $\lambda \in (0,1)$, $y \neq z$, then $y \notin C$ or $z \notin C$.*

We start with a well-known fact.

Fact 2.2.9. *Suppose $\tilde{x} \in \{x \geq 0 : \Gamma x = b\}$. Then $\tilde{x}$ is an extreme point of $\{x \geq 0 : \Gamma x = b\}$* if and only if *there does not exist nonzero $u \in \mathrm{Null}(\Gamma)$ with $\mathrm{supp}(u) \subseteq \mathrm{supp}(\tilde{x})$.*

Lemma 2.2.10. *Suppose $(p_1, \ldots, p_n) \in \mathcal{P}_{\mathrm{B}}$ and $(s_1, \ldots, s_n, s) \in \mathcal{P}_{\mathrm{A}}$. Then the vectors are extreme points of the respective polytopes if and only if they are Boolean, i.e., all of their entries are* 0 *or* 1*.*

Proof. We prove it for Bob's cheating polytope as the proof for Alice's is nearly identical. Suppose $(p_1, \ldots, p_n) \in \mathcal{P}_{\mathrm{B}}$ is Boolean, we show it is an extreme point. Let Bob's polytope $\mathcal{P}_{\mathrm{B}}$ be represented by the linear system $\Gamma(p_1, \ldots, p_n) = b$, $(p_1, \ldots, p_n) \geq 0$. Let $(u_1, \ldots, u_n) \in \mathrm{Null}(\Gamma)$ satisfy $\mathrm{supp}(u_1, \ldots, u_n) \subseteq \mathrm{supp}(p_1, \ldots, p_n)$. We argue that $(u_1, \ldots, u_n)$ must be the zero vector. The constraint on p_1 is $\sum_{y_1} p_{1,x_1,y_1} = 1$ for all $x_1 \in A_1$. Therefore, since p_1 is Boolean, there is exactly one value of y_1 for every x_1 such that $p_{1,x_1,y_1} = 1$. These are the only entries of u_1 that can be nonzero, but since

$(u_1, \ldots, u_n) \in \text{Null}(\Gamma)$ we must have that entry equal to 0. We can repeat this argument to get $u_i = 0$ for all $i \in \{1, \ldots, n\}$. Therefore, $(p_1, \ldots, p_n)$ is an extreme point.

Conversely, suppose $(p_1, \ldots, p_n) \in \mathcal{P}_\text{B}$ is not Boolean. Let i be the smallest index where p_i is not Boolean. If $i > 1$, define $u_j := 0$ for $j \in \{1, \ldots, i-1\}$. Let $(\hat{x}_1, \hat{y}_1, \ldots, \hat{x}_i, \hat{y}_i)$ be an index such that $p_{i,\hat{x}_1,\hat{y}_1,\ldots,\hat{x}_i,\hat{y}_i} \in (0,1)$. From the constraints, we must have another $\hat{y}_i'$ such that $p_{i,\hat{x}_1,\hat{y}_1,\ldots,\hat{x}_i,\hat{y}_i'} \in (0,1)$ as well (since they must add to 1). Now define $u_{i,\hat{x}_1,\hat{y}_1,\ldots,\hat{x}_i,\hat{y}_i} := t$, for some $t \neq 0$, and $u_{i,\hat{x}_1,\hat{y}_1,\ldots,\hat{x}_i,\hat{y}_i'} := -t$, and the rest of the entries of u_i to be 0. We define u_{i+1} to be equal to p_{i+1}, but we scale each entry such that

$$\text{Tr}_{B_{i+1}}(u_{i+1}) = u_i \otimes e_{A_{i+1}}.$$

We inductively define u_j in this way for all $j \in \{i+1, \ldots, n\}$. Therefore, since we scaled $(p_1, \ldots, p_n)$ to get $(u_1, \ldots, u_n)$, we have $\text{supp}(u_1, \ldots, u_n) \subseteq \text{supp}(p_1, \ldots, p_n)$ and $(u_1, \ldots, u_n) \in \text{Null}(\Gamma)$ implying $(p_1, \ldots, p_n)$ cannot be an extreme point. □

We see that extreme points of the cheating polytopes correspond to the strategies where Alice and Bob choose their next bit deterministically depending on the bits revealed.

Corollary 2.2.11. *In a classical* BCCF*-protocol, Alice and Bob each have an optimal cheating strategy which is deterministic.*

Proof. In a linear program whose feasible region does not contain lines, if there exists an optimal solution then there exists an optimal solution which is an extreme point of the feasible region. The result follows since the feasible region is nonempty and compact implying the existence of an optimal solution. □

2.3 Using classical protocols to lower bound the quantum bias

In this section, we prove that no quantum BCCF-protocol can have bias $\varepsilon = 1/\sqrt{2} - 1/2$. More specifically, we prove that only protocols that share optimal cheating probabilities with their classical counterpart can saturate Kitaev's lower bound on the product of the cheating probabilities. This shows yet another connection between quantum and classical BCCF-protocols.

We start with rederiving Kitaev's lower bound using the reduced SDPs. The dual of Bob's reduced SDP can be written as

$$\begin{array}{rrcll} \inf & \mathrm{Tr}_{A_1}(w_1) & & & \\ \text{subject to} & w_1 \otimes e_{B_1} & \geq & \mathrm{Tr}_{A_2}(w_2), & \\ & w_2 \otimes e_{B_2} & \geq & \mathrm{Tr}_{A_3}(w_3), & \\ & & \vdots & & \\ & w_n \otimes e_{B_n} & \geq & \frac{1}{2}\sum_{a \in \{0,1\}} \alpha_a \otimes v_a, & \\ & \mathrm{Diag}(v_a) & \succeq & \sqrt{\beta_a}\sqrt{\beta_a}^\top, & \text{for all } a \in \{0,1\}, \end{array}$$

for cheating towards 0, and the dual of Alice's reduced SDP can be written as

$$\begin{array}{rrcll} \inf & z_1 & & & \\ \text{subject to} & z_1 \cdot e_{A_1} & \geq & \mathrm{Tr}_{B_1}(z_2), & \\ & z_2 \otimes e_{A_2} & \geq & \mathrm{Tr}_{B_2}(z_3), & \\ & & \vdots & & \\ & z_n \otimes e_{A_n} & \geq & \mathrm{Tr}_{B_n}(z_{n+1}), & \\ & \mathrm{Diag}(z_{n+1}^{(y)}) & \succeq & \frac{1}{2}\beta_{a,y}, \sqrt{\alpha_a}\sqrt{\alpha_a}^\top, & \text{for all } a \in \{0,1\}, y \in B, \end{array}$$

for cheating towards 0. We rederive Kitaev's lower bound using these duals as follows. We start with

$$\begin{array}{rcl} P^*_{\mathrm{B},0}P^*_{\mathrm{A},0} & = & \mathrm{Tr}_{A_1}(w_1)\, z_1 \\ & = & \langle \mathrm{Tr}_{A_1}(w_1), z_1 \rangle \\ & = & \langle w_1, z_1 \otimes e_{A_1} \rangle \\ & \geq & \langle w_1, \mathrm{Tr}_{B_1}(z_2) \rangle \\ & = & \langle w_1 \otimes e_{B_1}, z_2 \rangle \\ & \geq & \langle \mathrm{Tr}_{A_2}(w_2), z_2 \rangle \\ & \vdots & \\ & \geq & \langle w_n \otimes e_{B_n}, z_{n+1} \rangle. \end{array}$$

We look at the quantity $\langle w_n \otimes e_{B_n}, z_{n+1} \rangle$ separately since the following analysis is slightly different. We use the inequality

$$w_n \otimes e_{B_n} \geq \frac{1}{2} \sum_{a \in \{0,1\}} \alpha_a \otimes v_a$$

and decompose

$$z_{n+1} = \sum_{y \in B} z_{n+1}^{(y)} \otimes e_y$$

to get

$$\begin{aligned}
\langle w_n \otimes e_{B_n}, z_{n+1} \rangle &\geq \frac{1}{2} \sum_{a \in \{0,1\}} \langle \alpha_a \otimes v_a, z_{n+1}^{(y)} \otimes e_y \rangle \\
&= \frac{1}{2} \sum_{a \in \{0,1\}} \sum_{y \in B} \langle \alpha_a, z_{n+1}^{(y)} \rangle \langle v_a, e_y \rangle \\
&= \frac{1}{2} \sum_{a \in \{0,1\}} \sum_{y \in B} \langle \sqrt{\alpha_a} \sqrt{\alpha_a}^\top, \mathrm{Diag}(z_{n+1}^{(y)}) \rangle \langle \mathrm{Diag}(v_a), e_y e_y^\top \rangle \\
&\geq \frac{1}{2} \sum_{a \in \{0,1\}} \sum_{y \in B} \langle \sqrt{\alpha_a} \sqrt{\alpha_a}^\top, \frac{1}{2} \beta_{a,y} \sqrt{\alpha_a} \sqrt{\alpha_a}^\top \rangle \langle \mathrm{Diag}(v_a), e_y e_y^\top \rangle \\
&= \frac{1}{4} \sum_{a \in \{0,1\}} \langle \mathrm{Diag}(v_a), \mathrm{Diag}(\beta_a) \rangle \\
&= \frac{1}{4} \sum_{a \in \{0,1\}} \langle \mathrm{Diag}(v_a), \sqrt{\beta_a} \sqrt{\beta_a}^\top \rangle \\
&\geq \frac{1}{4} \sum_{a \in \{0,1\}} \langle \sqrt{\beta_a} \sqrt{\beta_a}^\top, \sqrt{\beta_a} \sqrt{\beta_a}^\top \rangle \\
&= \frac{1}{2}.
\end{aligned}$$

Therefore, we get Kitaev's lower bound $P_{\mathrm{A},0}^* P_{\mathrm{B},0}^* \geq 1/2$ implying that $P_{\mathrm{A},0}^* \geq 1/\sqrt{2}$ or $P_{\mathrm{B},0}^* \geq 1/\sqrt{2}$. Note this was later reproven by Gutoski and Watrous (for general coin-flipping protocols) using a different representation of cheating strategies in [GW07].

We get the inequality $P_{\mathrm{A},1}^* P_{\mathrm{B},1}^* \geq 1/2$ by switching β_0 with β_1 in the proof above (and the dual variables accordingly). Using these two lower bounds, we show that it is impossible to have a quantum BCCF-protocol with bias $\varepsilon = 1/\sqrt{2} - 1/2$ by proving Kitaev's bounds can only be saturated with protocols where one party can cheat perfectly. More specifically, we show that if there exists four dual solutions that saturate both of Kitaev's bounds

$$P_{\mathrm{A},0}^* P_{\mathrm{B},0}^* \geq 1/2 \quad \text{and} \quad P_{\mathrm{A},1}^* P_{\mathrm{B},1}^* \geq 1/2,$$

then all four of the dual solutions must be in the restricted part of the feasible regions defined by the classical dual LPs, depicted in Figure 2.12.

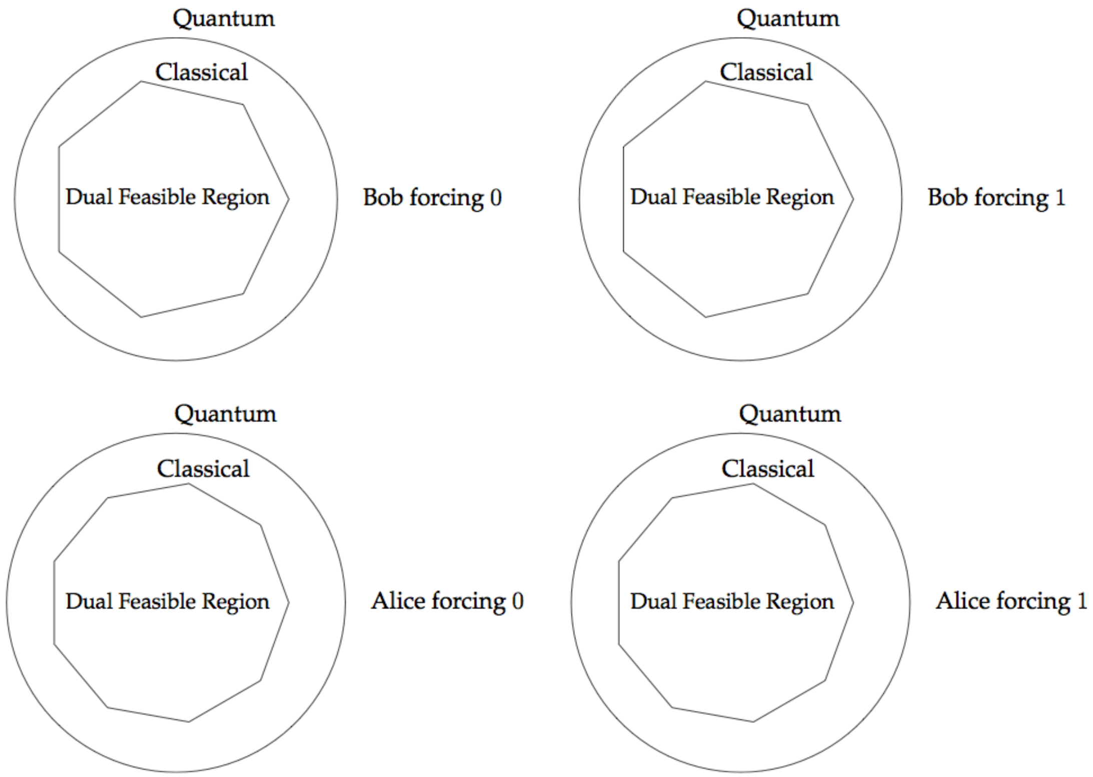


Figure 2.12: The four "quantum" dual feasible regions and the four "classical" dual feasible regions. If Kitaev's lower bounds are saturated by a quantum BCCF-protocol, then the four optimal dual solutions are in the smaller "classical" feasible regions.

Theorem 2.3.1. *Suppose a quantum* BCCF*-protocol satisfies*

$$P^*_{\mathrm{A},0}\,P^*_{\mathrm{B},0} = \frac{1}{2} \quad \text{and} \quad P^*_{\mathrm{A},1}\,P^*_{\mathrm{B},1} = \frac{1}{2}.$$

Then the cheating probabilities are the same as in the corresponding classical protocol.

Proof. We look at Kitaev's proof above, and note that if it were saturated, then every inequality must hold with equality. Therefore, we get that

$$\mathrm{Diag}(v_a) \succeq \sqrt{\beta_a}\sqrt{\beta_a}^{\top} \tag{2.2}$$

has no slack on the subspace spanned by $\sqrt{\beta_a}\sqrt{\beta_a}^{\top}$, i.e.,

$$\langle \mathrm{Diag}(v_a) - \sqrt{\beta_a}\sqrt{\beta_a}^{\top}, \sqrt{\beta_a}\sqrt{\beta_a}^{\top} \rangle = 0 \iff \langle \mathrm{Diag}(v_a), \sqrt{\beta_a}\sqrt{\beta_a}^{\top} \rangle = 1, \tag{2.3}$$

for both $a \in \{0,1\}$. Consider $v_a = e_{\mathrm{supp}(\beta_a)}$, this satisfies the constraint (2.2) and satisfies the condition (2.3). We show this choice is unique (on $\mathrm{supp}(\beta_a)$). Consider the optimization problems

$$\begin{aligned} & \inf\left\{\langle \mathrm{Diag}(v_a), \sqrt{\beta_a}\sqrt{\beta_a}^\top\rangle : \mathrm{Diag}(v_a) \succeq \sqrt{\beta_a}\sqrt{\beta_a}^\top\right\} \\ = \;& \inf\left\{\sum_{y \in \mathrm{supp}(\beta_a)} v_{a,y}\beta_{a,y} : \sum_{y \in \mathrm{supp}(\beta_a)} \frac{\beta_{a,y}}{v_{a,y}} \leq 1, \; v_{a,y} > 0\right\}. \end{aligned}$$

Obviously $v_a = e_{\mathrm{supp}(\beta_a)}$ is an optimal solution since 1 is a lower bound on the optimal objective value. Suppose there are two optimal solutions v' and v''. Then $\frac{1}{2}v' + \frac{1}{2}v''$ has the same objective value, but satisfies the constraint

$$\sum_{y \in \mathrm{supp}(\beta_a)} \frac{\beta_{a,y}}{v_{a,y}} \leq 1$$

with strict inequality since the function $\sum_{y \in \mathrm{supp}(\beta_a)} \frac{\beta_{a,y}}{v_{a,y}}$ is strictly convex. Thus, we can scale $\frac{1}{2}v' + \frac{1}{2}v''$ to get a better objective function value, a contradiction. Therefore, if Kitaev's bound is saturated, we must have $v_{a,y} = 1$ for all $a \in \{0,1\}$, $y \in \mathrm{supp}(\beta_a)$.

We argue the same about Alice's dual variables $z_{n+1}^{(y)}$. If the Kitaev inequalities are saturated, we have

$$\langle \sqrt{\alpha_a}\sqrt{\alpha_a}^\top, \mathrm{Diag}(z_{n+1}^{(y)}) - \frac{1}{2}\beta_{a,y}\sqrt{\alpha_a}\sqrt{\alpha_a}^\top\rangle = 0 \iff \langle \sqrt{\alpha_a}\sqrt{\alpha_a}^\top, \mathrm{Diag}(z_{n+1}^{(y)})\rangle = \frac{1}{2}\beta_{a,y},$$

for all a, y such that $v_{a,y} > 0$, i.e., for all $y \in \mathrm{supp}(\beta_a)$.

Similar to the arguments above, we need $[z_{n+1}^{(y)}]_x = \frac{1}{2}\beta_{a,y}$ for $a \in \{0,1\}$, $x \in \mathrm{supp}(\alpha_a)$, and $y \in \mathrm{supp}(\beta_a)$.

To summarize, if we have Kitaev's bounds saturated, then the optimal dual solutions satisfy

$$\mathrm{Diag}(v_a) \succeq \mathrm{Diag}(e_{\mathrm{supp}(\beta_a)}) \quad \text{and} \quad \mathrm{Diag}(z_{n+1}^{(y)}) \succeq \frac{1}{2}\beta_{a,y}\,\mathrm{Diag}(e_{\mathrm{supp}(\alpha_a)}), \forall a \in \{0,1\}, y \in B,$$

which are exactly the constraints in the dual LPs for the classical version. Therefore, the protocol must have the property that relaxing the cheat detection steps in $\Pi_{\mathrm{A},0}$ and $\Pi_{\mathrm{B},0}$ (obtaining the classical cheat detection) preserves the two cheating probabilities. We can repeat the same argument with Alice and Bob cheating towards 1 and get the two corresponding classical cheating probabilities. Therefore, we have all four cheating probabilities are equal to those of the corresponding classical protocol, as desired. □

Since every classical protocol allows one party to cheat perfectly, we have the following result.

Corollary 2.3.2. $\varepsilon = 1/\sqrt{2} - 1/2$ *is impossible for any* BCCF*-protocol.*

The proof of Theorem 2.3.1 gives necessary conditions on classical protocols that saturate Kitaev's bound. Note from the condition on $z_{n+1}^{(y)}$, we have

$$[z_{n+1}]_{x,y} = \frac{1}{2}\beta_{a,y} \text{ when } \beta_{a,y}, \alpha_{a,x} > 0.$$

In the case when $\alpha_0, \alpha_1, \beta_0, \beta_1 > 0$, then β_0 must equal β_1. This makes sense since Bob can easily cheat with probability 1, but if $\beta_0 \neq \beta_1$, then Alice could cheat with probability greater than 1/2. In the case when $\alpha_0 \perp \alpha_1$, the condition above tells us nothing, but it is easy to see that Alice fully reveals a in the first message, thus she can cheat with probability 1/2 and Bob can cheat with probability 1.

2.3.1 A better lower bound on a special case

We prove that there are no BCCF-protocols with bias less than $\varepsilon = 1/4$ for the special case when $A = B = \{0, 1\}$. This is the simplest non-trivial BCCF-protocol where Alice and Bob each have two messages; the first message is a qubit, and the second message reveals their committed bit.

We make use of the strengthened Fuchs-van de Graaf inequality

$$1 - \mathrm{F}(\rho_0, \rho_1) \leq \Delta(\rho_0, \rho_1),$$

when ρ_0, ρ_1 are qubit states (see [SR01]).

Let

$$\beta_0 = [r, 1-r]^\top \quad \text{and} \quad \beta_1 = [1-t, t]^\top,$$

where $r \geq 1 - t$ (we can switch β_0 and β_1 if this is not the case and the bias remains unchanged).

We present a fact here which is proved as part of two more general theorems (Theorem 3.1.2 and Theorem 3.1.9) in Chapter 3. This fact presents lower bounds on the extent Alice and Bob can cheat. Note that the result of this subsection is independent of Chaper 3.

Fact 2.3.3. *For any four-round* BCCF*-protocol, we have*

- $P^*_{\mathrm{B},0} \geq \frac{1}{2} + \frac{1}{2}\Delta(\alpha_0, \alpha_1)$,
- $P^*_{\mathrm{B},0} \geq \frac{1}{2} + \frac{1}{2}\sqrt{\mathrm{F}(\beta_0, \beta_1)}$,
- $P^*_{\mathrm{A},0} \geq \frac{1}{2}\lambda_{\max}\left(r\sqrt{\alpha_0}\sqrt{\alpha_0}^\top + t\sqrt{\alpha_1}\sqrt{\alpha_1}^\top\right) = \frac{1}{4}\left(r + t + \sqrt{(r-t)^2 + 4rt\mathrm{F}(\alpha_0, \alpha_1)}\right)$,

where r and t are as above.

Suppose Bob can cheat with probability at most 3/4. Then

$$\frac{1}{2} + \frac{1}{2}\sqrt{\mathrm{F}(\beta_0, \beta_1)} \leq \frac{3}{4} \implies 2\Delta(\beta_0, \beta_1) \geq 1 \implies r + t \geq \frac{3}{2} \implies r, t \geq \frac{1}{2}.$$

Another consequence is

$$\frac{1}{2} + \frac{1}{2}\sqrt{\mathrm{F}(\beta_0, \beta_1)} \leq 3/4 \implies \sqrt{r(1-t)} + \sqrt{t(1-r)} \leq \frac{1}{2}$$

which can be used to bound t as

$$t \geq f(r) := \frac{r}{2} + \frac{1}{4} + \sqrt{3r(1-r)}.$$

Since Bob can cheat with probability at most 3/4, we also have that

$$\frac{1}{2} + \frac{1}{2}\Delta(\alpha_0, \alpha_1) \leq \frac{3}{4} \implies \mathrm{F}(\alpha_0, \alpha_1) \geq \frac{1}{2}.$$

Using these inequalities, Alice can cheat with probability

$$\begin{aligned}
P^*_{\mathrm{A},0} &\geq \frac{1}{2}\lambda_{\max}\left(r\sqrt{\alpha_0}\sqrt{\alpha_0}^\top + t\sqrt{\alpha_1}\sqrt{\alpha_1}^\top\right) \\
&= \frac{1}{4}\left(r + t + \sqrt{(r-t)^2 + 4rt\,\mathrm{F}(\alpha_0, \alpha_1)}\right) \\
&\geq \frac{1}{4}\left(r + t + \sqrt{r^2 + t^2}\right) \qquad \text{(since } \mathrm{F}(\alpha_0, \alpha_1) \geq 1/2\text{)} \\
&\geq \frac{1}{4}\left(r + f(r) + \sqrt{r^2 + f(r)^2}\right) \qquad \text{(since } t \geq f(r)\text{)} \\
&\geq 3/4 \qquad \text{(minimized at } r = 1 \text{ on the interval } [1/2, 1]\text{)}.
\end{aligned}$$

Therefore, one of Alice or Bob can cheat with probability at least $3/4$ in this case.

Note that all three of the lower bounds in Fact 2.3.3 could be less than Kitaev's bound $1/\sqrt{2} \approx 0.707$ when $|A|$ and $|B|$ are larger than 2. Therefore, we cannot get an interesting lower bound using Fact 2.3.3 even if we could strengthen the Fuchs-van de Graaf inequalities for higher dimensional states.

Chapter 3

A computational search for BCCF-protocols with small bias

The content of this chapter is based on my work with Ashwin Nayak and Levent Tunçel.

In the previous chapter, we defined a family of coin-flipping protocols and proved a lower bound on the bias. In this chapter, we are concerned with finding the best upper bound. Upper bounds are achieved by exhibiting a protocol with a proof of its bias. With highly interactive protocols, it is a difficult task to simply point out a protocol that performs well. Instead, we have designed an algorithm to search for protocols with small bias over the parameters $\alpha_0, \alpha_1 \in \text{Prob}^A$ and $\beta_0, \beta_1 \in \text{Prob}^B$.

We start with a simple idea: We create a finite mesh over the parameters and, for each point in the mesh, calculate the four cheating probabilities then the bias. However, it is not hard to be convinced that this task grows very expensive as the mesh becomes finer. We therefore introduce heuristics to decrease the number of protocols needed to be tested and also to decrease the time it takes to test each protocol. This process is broken into the following steps.

- Section 3.1: We examine feasible cheating strategies for Alice and Bob by studying the reduced cheating SDPs. By restricting the feasible region in certain ways, we reveal certain cheating strategies. The set of strategies we obtain creates a *protocol filter*. The protocol filter is used to quickly eliminate protocols with high bias from the search.

- Section 3.2: We study the symmetry of the parameters defining BCCF-protocols and show how certain permutations of the indices preserve the bias. This drastically reduces the number of protocols needed to be checked.

- Section 3.3: We describe how the mesh is generated and exhibit a protocol with bias $\varepsilon = 1/4$. We give two proofs of the bias of this protocol, one quantum and one using the cheating SDPs. This gives a bias cut-off point. We then describe the algorithm using these ideas and those developed in the previous two sections.

- Section 3.4: We present the numerical results of several searches. We present the results in tables indicating how well each strategy in the filter performed and how many protocols made it all the way through the filter. We also test the algorithm by randomly offsetting the mesh. We conclude with the conjecture that $\varepsilon = 1/4$ is the smallest attainable bias for BCCF-protocols.

3.1 The protocol filter

In this section, we describe ways to approximate the optimal cheating probabilities by finding feasible solutions to Alice and Bob's reduced cheating SDPs. Why do we care about feasible solutions that may not be optimal? The algorithm tests many protocols so the idea is to have simple checks to see whether a protocol is a good candidate for being optimal. For example, suppose we have the success probability of a certain cheating strategy for, say, Bob. Then for a given set of parameters, if this strategy succeeds with high probability, then we can rule out these parameters as being a good candidate and save the time it would have taken to solve the SDPs (or SOCPs).

We illustrate this idea using Kitaev's lower bound below.

Theorem 3.1.1 ([Kit02, GW07]). *For* any *coin-flipping protocol, we have*

$$P^*_{\mathrm{A},0} P^*_{\mathrm{B},0} \geq \frac{1}{2} \quad \textit{and} \quad P^*_{\mathrm{A},1} P^*_{\mathrm{B},1} \geq \frac{1}{2}.$$

Suppose that we compute and find $P^*_{\mathrm{A},0} \approx 1/2$, that is, the protocol is very secure concerning Alice cheating towards 0. Then, from Kitaev's bound, we know that $P^*_{\mathrm{B},0} \approx 1$ and the protocol is very insecure concerning cheating Bob. Therefore, we do not need to solve for $P^*_{\mathrm{B},0}$ since we know this protocol has large bias.

The remainder of this section is divided into cheating Alice and cheating Bob. We discuss cheating strategies for each and for the special cases of four-round and six-round protocols.

3.1.1 Cheating Alice

We now present a theorem which captures some of Alice's cheating strategies.

Theorem 3.1.2. *For any* BCCF*-protocol, we can bound Alice's optimal cheating probability as follows:*

$$\begin{aligned} P^*_{\mathrm{A},0} &\geq \frac{1}{2}\sum_{y\in B} \mathrm{conc}_{a\in\{0,1\}} \left\{\beta_{a,y} \mathrm{F}(\cdot, \alpha_a)\right\}(v) && (3.1) \\ &\geq \frac{1}{2}\lambda_{\max}\left(\eta\sqrt{\alpha_0}\sqrt{\alpha_0}^\top + \tau\sqrt{\alpha_1}\sqrt{\alpha_1}^\top\right) && (3.2) \\ &\geq \left(\frac{1}{2} + \frac{1}{2}\sqrt{\mathrm{F}(\alpha_0,\alpha_1)}\right)\left(\frac{1}{2} + \frac{1}{2}\Delta(\beta_0,\beta_1)\right), && (3.3) \end{aligned}$$

where

$$\eta := \sum_{\substack{y\in B:\\ \beta_{0,y}\geq\beta_{1,y}}} \beta_{0,y} \quad \text{and} \quad \tau := \sum_{\substack{y\in B:\\ \beta_{0,y}<\beta_{1,y}}} \beta_{1,y},$$

and $\sqrt{v}$ is the normalized, nonnegative principal eigenvector of $\eta\sqrt{\alpha_0}\sqrt{\alpha_0}^\top + \tau\sqrt{\alpha_1}\sqrt{\alpha_1}^\top$. We call (3.1) *Alice's improved eigenstrategy,* (3.2) *her eigenstrategy, and* (3.3) *her three-round strategy.*

Furthermore, in a six-round protocol, we have

$$\begin{aligned} P^*_{\mathrm{A},0} &\geq \frac{1}{2}\lambda_{\max}\left(\eta'\sqrt{\mathrm{Tr}_{A_2}(\alpha_0)}\sqrt{\mathrm{Tr}_{A_2}(\alpha_0)}^\top + \tau'\sqrt{\mathrm{Tr}_{A_2}(\alpha_1)}\sqrt{\mathrm{Tr}_{A_2}(\alpha_1)}^\top\right) && (3.4) \\ &\geq \left(\frac{1}{2} + \frac{1}{2}\sqrt{\mathrm{F}(\mathrm{Tr}_{A_2}(\alpha_0), \mathrm{Tr}_{A_2}(\alpha_1))}\right)\left(\frac{1}{2} + \frac{1}{2}\Delta(\mathrm{Tr}_{B_2}(\beta_0), \mathrm{Tr}_{B_2}(\beta_1))\right), && (3.5) \end{aligned}$$

where

$$\eta' := \sum_{\substack{y_1 \in B_1 : \\ [\mathrm{Tr}_{B_2}(\beta_0)]_{y_1} \geq [\mathrm{Tr}_{B_2}(\beta_1)]_{y_1}}} [\mathrm{Tr}_{B_2}(\beta_0)]_{y_1} \quad \text{and} \quad \tau' := \sum_{\substack{y_1 \in B_1 : \\ [\mathrm{Tr}_{B_2}(\beta_0)]_{y_1} < [\mathrm{Tr}_{B_2}(\beta_1)]_{y_1}}} [\mathrm{Tr}_{B_2}(\beta_1)]_{y_1}.$$

We call (3.4) *Alice's six-round eigenstrategy and* (3.5) *her measuring strategy. We get bounds for* $P^*_{\mathrm{A},1}$ *when we switch the roles of* β_0 *and* β_1 *above.*

Note that only the improved eigenstrategy is affected by switching β_0 and β_1 (as long as we are willing to accept a slight modification to how we break ties in the definitions of η, η', τ, and τ').

We now briefly describe the strategies that yield the corresponding cheating probabilities in Theorem 3.1.2.

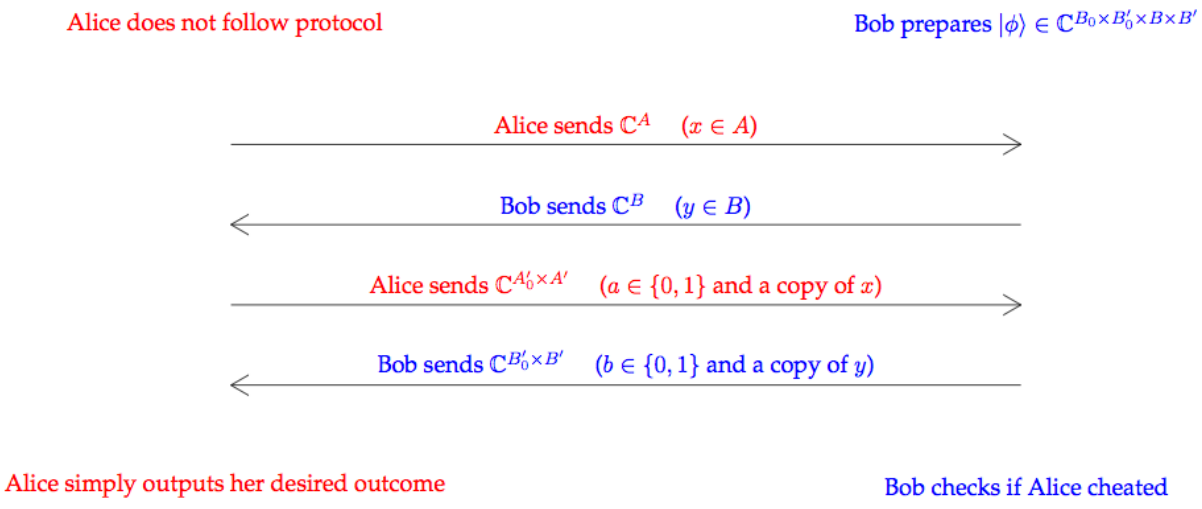

Alice does not follow protocol
Bob prepares $|\phi\rangle \in \mathbb{C}^{B_0 \times B_0' \times B \times B'}$
Alice sends $\mathbb{C}^A$ $(x \in A)$
Bob sends $\mathbb{C}^B$ $(y \in B)$
Alice sends $\mathbb{C}^{A_0' \times A'}$ $(a \in \{0,1\}$ and a copy of $x)$
Bob sends $\mathbb{C}^{B_0' \times B'}$ $(b \in \{0,1\}$ and a copy of $y)$
Alice simply outputs her desired outcome
Bob checks if Alice cheated

Figure 3.1: Alice cheating in a four-round BCCF-protocol.

Her three-round strategy is to send σ_1 as if it were a standard three-round protocol, measure Bob's message to try to learn b, and reply with a deterministically, depending on her desired outcome. Her eigenstrategy is the same as her three-round strategy, except the first message is optimized. The improved eigenstrategy has the same first message as her eigenstrategy, but the last message is optimized. These strategies work for the general case as well, were Alice treats her first n messages like the first one in a four-round protocol.

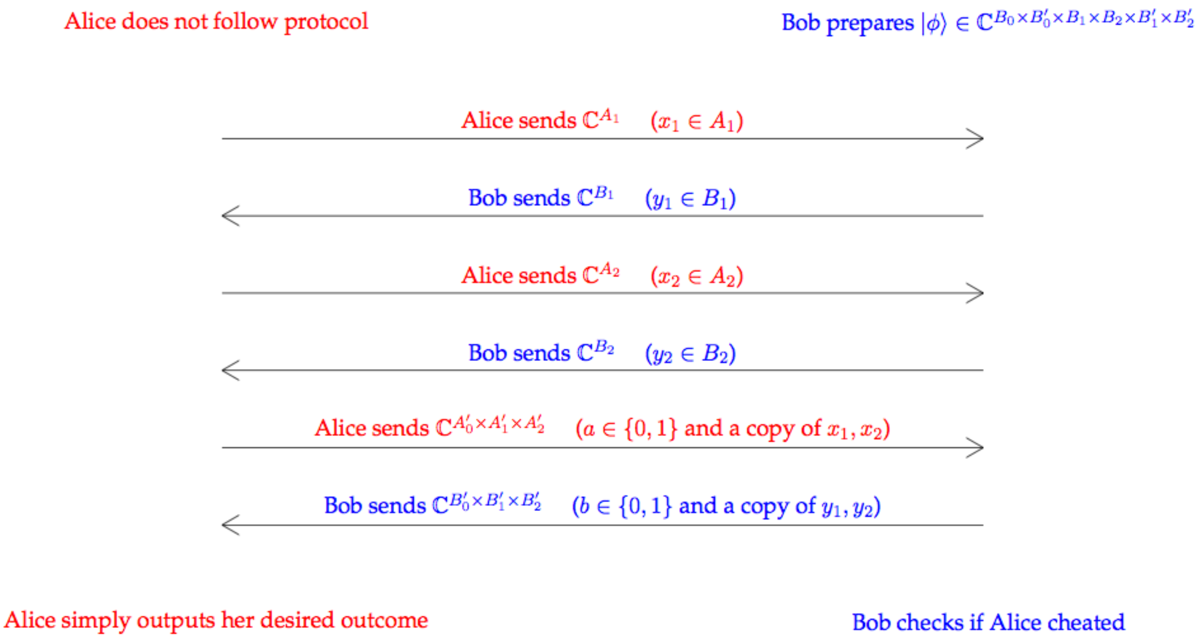

Alice does not follow protocol
Bob prepares $|\phi\rangle \in \mathbb{C}^{B_0 \times B_0' \times B_1 \times B_2 \times B_1' \times B_2'}$
Alice sends $\mathbb{C}^{A_1}$ $(x_1 \in A_1)$
Bob sends $\mathbb{C}^{B_1}$ $(y_1 \in B_1)$
Alice sends $\mathbb{C}^{A_2}$ $(x_2 \in A_2)$
Bob sends $\mathbb{C}^{B_2}$ $(y_2 \in B_2)$
Alice sends $\mathbb{C}^{A_0' \times A_1' \times A_2'}$ $(a \in \{0,1\}$ and a copy of $x_1, x_2)$
Bob sends $\mathbb{C}^{B_0' \times B_1' \times B_2'}$ $(b \in \{0,1\}$ and a copy of $y_1, y_2)$
Alice simply outputs her desired outcome
Bob checks if Alice cheated

Figure 3.2: Alice cheating in a six-round BCCF-protocol.

The six-round version is depicted above. Her six-round eigenstrategy is the same as the four-round version, except she tries to learn b from the first message, and only her first message is optimized. Her measuring strategy is the same as her three-round strategy in the four-round version, she sends the first message as in a three-round protocol then measures Bob's first message to try to learn b.

Developing Alice's strategies

Recall Alice's reduced SDP

$$P^*_{\mathrm{A},0} = \max \left\{ \frac{1}{2} \sum_{a \in \{0,1\}} \sum_{y \in B} \beta_{a,y} \, \mathrm{F}(s^{(a,y)}, \alpha_a) \ : \ (s_1, \ldots, s_n, s) \in \mathcal{P}_{\mathrm{A}} \right\},$$

where $\mathcal{P}_{\mathrm{A}}$ is the set of vectors $(s_1, s_2, \ldots, s_n, s)$ satisfying

$$\begin{array}{rcl}
\mathrm{Tr}_{A_1}(s_1) & = & 1, \\
\mathrm{Tr}_{A_2}(s_2) & = & s_1 \otimes e_{B_1}, \\
 & \vdots & \\
\mathrm{Tr}_{A_n}(s_n) & = & s_{n-1} \otimes e_{B_{n-1}}, \\
\mathrm{Tr}_{A'_0}(s) & = & s_n \otimes e_{B_n}, \\
s_1 & \in & \mathbb{R}_+^{A_1}, \\
s_j & \in & \mathbb{R}_+^{A_1 \times B_1 \times \cdots \times B_{j-1} \times A_j}, \text{ for all } j \in \{2, \ldots, n\}, \\
s & \in & \mathbb{R}_+^{A \times B \times A'_0}.
\end{array}$$

To get a feasible solution, suppose Alice guesses b before she reveals a in the following way. If Bob reveals $y \in B$, then Alice guesses $b = 0$ if $\beta_{0,y} \geq \beta_{1,y}$ and $b = 1$ if $\beta_{0,y} < \beta_{1,y}$. Let Alice's guess be denoted by $f(y)$, so

$$f(y) = \arg\max_{a} \{\beta_{a,y}\} \in \{0, 1\},$$

and we set $f(y) = 0$ in the case of a tie. We have chosen a way to satisfy the last constraint in Alice's cheating polytope, but we can choose how Alice sends her first n messages $s_1, \ldots, s_n$. We make one more restriction, we set $s_n = d \otimes e_B$ and optimize over $d \in \mathbb{R}_+^A$. We can easily satisfy the rest of the constraints given any d by choosing each variable as the corresponding marginal probability distribution.

Under these restrictions, we have that Alice's reduced problem can be written as

$$\max_{d \in \mathrm{Prob}^A} \left\{ \frac{1}{2} \sum_{y \in B} \beta_{f(y),y} \mathrm{F}(d, \alpha_{f(y)}) \right\} = \max_{d \in \mathrm{Prob}^A} \left\{ \frac{\eta}{2} \, \mathrm{F}(d, \alpha_0) + \frac{\tau}{2} \, \mathrm{F}(d, \alpha_1) \right\}.$$

We can simplify this using the following lemma.

Lemma 3.1.3. *For nonnegative vectors $\{z_1, \ldots, z_m\} \subset \mathbb{R}_+^n$, we have that*

$$\max \left\{ \sum_{i=1}^{m} \mathrm{F}(p, z_i) : p \in \mathrm{Prob}^n \right\} = \lambda_{\max} \left(\sum_{i=1}^{m} \sqrt{z_i} \sqrt{z_i}^\top \right),$$

where $\lambda_{\max}$ denotes the largest eigenvalue. Furthermore, an optimal solution is the entry-wise square of the normalized, nonnegative principal eigenvector.

Proof. Since $\sum_{i=1}^m \mathrm{F}(p, z_i) = \sum_{i=1}^m \langle \sqrt{p}\sqrt{p}^\top, \sqrt{z_i}\sqrt{z_i}^\top \rangle = \sqrt{p}^\top \left(\sum_{i=1}^m \sqrt{z_i}\sqrt{z_i}^\top \right) \sqrt{p}$, where $\sqrt{\cdot}$ is the entry-wise square root, the maximization problem reduces to

$$\max \left\{ \sqrt{p}^\top \left(\sum_{i=1}^m \sqrt{z_i}\sqrt{z_i}^\top \right) \sqrt{p} : p \in \mathrm{Prob}^n \right\}.$$

If the nonnegativity constraint were not present, the optimal value would be attained by setting $\sqrt{p}$ to be a normalized principal eigenvector of $\sum_{i=1}^m \sqrt{z_i}\sqrt{z_i}^\top$. Because $\sum_{i=1}^m \sqrt{z_i}\sqrt{z_i}^\top$ has nonnegative entries, we know there exists a nonnegative principal eigenvector by the Perron-Frobenius Theorem. Since this does not violate the nonnegativity constraint in the problem, this value for p is an optimal solution yielding an optimal objective value of $\lambda_{\max}\left(\sum_{i=1}^m \sqrt{z_i}\sqrt{z_i}^\top \right)$. □

Using this lemma, Alice can cheat with probability

$$\frac{1}{2}\lambda_{\max}\left(\eta\sqrt{\alpha_0}\sqrt{\alpha_0}^\top + \tau\sqrt{\alpha_1}\sqrt{\alpha_1}^\top \right),$$

which we call Alice's *eigenstrategy*.

Since $\frac{1}{2}(\eta + \tau) = \sum_y \max_a \left\{ \frac{1}{2}\beta_{a,y} \right\}$, we have

$$\begin{aligned} \frac{1}{2}\lambda_{\max}\left(\eta\sqrt{\alpha_0}\sqrt{\alpha_0}^\top + \tau\sqrt{\alpha_1}\sqrt{\alpha_1}^\top \right) &= \frac{1}{4}\left(\eta + \tau + \sqrt{(\eta-\tau)^2 + 4\eta\tau\,\mathrm{F}(\alpha_0,\alpha_1)} \right) \\ &\geq \frac{1}{4}\left(\eta + \tau + \sqrt{(\eta-\tau)^2\mathrm{F}(\alpha_0,\alpha_1) + 4\eta\tau\,\mathrm{F}(\alpha_0,\alpha_1)} \right) \\ &= \frac{1}{4}\left(\left(1 + \sqrt{\mathrm{F}(\alpha_0,\alpha_1)} \right)(\eta+\tau) \right) \\ &= \left(\frac{1}{2} + \frac{1}{2}\sqrt{\mathrm{F}(\alpha_0,\alpha_1)} \right)\left(\frac{1}{2} + \frac{1}{2}\Delta(\beta_0,\beta_1) \right), \end{aligned}$$

using Lemma 1.4.14 and Lemma 1.4.13.

This lower bound has a natural interpretation. This is the strategy where Alice ignores all of Bob's messages until $\mathbb{C}^{B_n}$ is sent. Then she measures the state in $\mathbb{C}^B$ to learn b with probability $\frac{1}{2} + \frac{1}{2}\Delta(\beta_0,\beta_1)$. She then tries to get past Bob's cheat detection and can do so with probability $\frac{1}{2} + \frac{1}{2}\sqrt{\mathrm{F}(\alpha_0,\alpha_1)}$. We call this Alice's *three-round strategy* since it combines optimal strategies for three-round protocols [Amb01, NS03, KN04]. It makes sense that this is a lower bound on the success probability of Alice's eigenstrategy since her eigenstrategy is optimized after the same restriction present in three-round strategy (choosing b from the most likely choice from knowledge of y).

We can also examine how Alice can choose her last message optimally supposing she has already sent her first n messages. I.e., suppose we fix some $s_1, \ldots, s_n$ satisfying the first n constraints of her cheating polytope and we want to optimize over s satisfying $\mathrm{Tr}_{A'_0}(s) = c$, where $c := s_n \otimes e_{B_n}$ is now constant. In this case, the only constraint is $\mathrm{Tr}_{A'_0}(s) = c$ which can be written as $\sum_{a \in \{0,1\}} s^{(a,y)} = c$, for each $y \in B$, where again, $s^{(a,y)}$ is the restriction of s with a and y fixed. Now we get the following optimization problem

$$\begin{array}{rrcll} \max & \frac{1}{2} \sum_{a \in \{0,1\}} \sum_{y \in B} \beta_{a,y} \mathrm{F}(s^{(a,y)}, \alpha_a) & & & \\ \text{subject to} & \sum_{a \in \{0,1\}} s^{(a,y)} & = & c, & \text{for all } y \in B, \\ & s^{(a,y)} & \in & \mathbb{R}^A_+. & \end{array}$$

If we rewrite this as

$$\begin{array}{rrcll} \max & \frac{1}{2} \sum_{y \in B} \sum_{a \in \{0,1\}} \mathrm{F}(s^{(a,y)}, \beta_{a,y}\alpha_a) & & & \\ \text{subject to} & \sum_{a \in \{0,1\}} s^{(a,y)} & = & c, & \text{for all } y \in B, \\ & s^{(a,y)} & \in & \mathbb{R}^A_+, & \end{array}$$

we see that this is a separable problem over $y \in B$. That is, for each fixed $\tilde{y} \in B$, we need to solve the optimization problem

$$G_{\tilde{y}}(c) := \max \left\{ \frac{1}{2} \sum_{a \in \{0,1\}} \mathrm{F}(s^{(a,\tilde{y})}, \beta_{a,\tilde{y}}\alpha_a) : \sum_{a \in \{0,1\}} s^{(a,\tilde{y})} = c, \; s^{(a,\tilde{y})} \in \mathbb{R}^A_+, \, \forall a \in \{0,1\} \right\}.$$

This optimization problem has a special structure.

Definition 3.1.4. *The* infimal convolution *of the convex functions $f_1, f_2, \ldots, f_n$, where $f_1, \ldots, f_n : \mathbb{R}^n \to \mathbb{R} \cup \{\infty\}$, is*

$$(f_1 \,\Box\, f_2 \,\Box\, \cdots \,\Box\, f_n)(d) := \inf_{x \in \mathbb{R}^n} \left\{ \sum_{i=1}^n f_i(x_i) : \sum_{i=1}^n x_i = d \right\}.$$

We do not need to worry about the nonnegativity constraints on the variables since we can define the convex function $-\mathrm{F}(p,q) = +\infty$ if p or q is not nonnegative. Note for every $p \in \mathbb{R}^n_+$, that $-\mathrm{F}(p, \cdot)$ is a *proper, convex function*, i.e., it is convex and $-\mathrm{F}(p,q) < +\infty$ for some $q \in \mathbb{R}^n_+$. Using these properties and the fact that $-\mathrm{F}(p,\cdot)$ is positively homogeneous, we show a way to express $G_{\tilde{y}}$.

Recall that for proper, convex functions $f_1, \ldots, f_n : \mathbb{R}^n \to \mathbb{R} \cup \{\infty\}$ we define the convex hull of $\{f_1, \ldots, f_n\}$ as the greatest convex function f such that $f(x) \leq f_1(x), \ldots, f_n(x)$ for every $x \in \mathbb{R}^n$. To write down explicitly what the convex hull is, we use the following definition.

Definition 3.1.5. *We define the* right scalar multiplication *of a function f as*

$$(f\lambda)(x) = \begin{cases} \lambda f(\lambda^{-1}x) & \text{for} \quad \lambda > 0, \\ 0 & \text{for} \quad \lambda = 0, x = 0, \\ +\infty & \text{for} \quad \lambda = 0, x \neq 0. \end{cases}$$

Thus, $(f\lambda) = f$ for all $\lambda > 0$ if and only if f is positively homogeneous.

Theorem 3.1.6 (Rockafellar, [Roc70, page 38])**.** *Let $f_1, \ldots, f_n : \mathbb{R}^n \to \mathbb{R} \cup \{\infty\}$ be proper, convex functions. Then we have*

$$\operatorname{conv}\{f_1, \ldots, f_n\}(d) = \inf_{\lambda \in \mathrm{Prob}^n} \left\{ ((f_1\lambda_1) \,\Box\, (f_2\lambda_2) \,\Box\, \cdots \,\Box\, (f_n\lambda_n))(d) \right\}.$$

We now present a theorem which helps capture Alice's cheating probability.

Theorem 3.1.7. *Suppose $f_1, \ldots, f_n : \mathbb{R}^n \to \mathbb{R} \cup \{\infty\}$ are positively homogenous, proper, convex functions and $f_i(0) = 0$, for all $i \in \{1, \ldots, n\}$. Then*

$$\operatorname{conv}\{f_1, \ldots, f_n\} = f_1 \,\Box\, f_2 \,\Box\, \cdots \,\Box\, f_n.$$

Proof. From the results above, it suffices to prove that we can assume $\lambda > 0$ in Theorem 3.1.6. Now suppose $\lambda_i = 0$ for i in index set I, positive otherwise. Since we are minimizing over x, we need $x_i = 0$ for $i \in I$, otherwise $(f_i 0)(x_i) = +\infty$. Then we have

$$\begin{aligned} ((f_1\lambda_1) \,\Box\, \cdots \,\Box\, (f_n\lambda_n))(d) &= \inf_{x \in \mathbb{R}^n} \left\{ \sum_{i \in I} (f_i 0)(x_i) + \sum_{i \notin I} f_i(x_i) : \sum_{i \in I} x_i + \sum_{i \notin I} x_i = d \right\} \\ &= \inf_{x \in \mathbb{R}^n} \left\{ \sum_{i \notin I} f_i(x_i) : \sum_{i \notin I} x_i = d \right\} \\ &\geq \inf_{x \in \mathbb{R}^n} \left\{ \sum_{i=1}^n f_i(x_i) : \sum_{i=1}^n x_i = d \right\} \\ &= (f_1 \,\Box\, f_2 \,\Box\, \cdots \,\Box\, f_n)(d), \end{aligned}$$

which can be attained by choosing $\lambda > 0$, since each f_i is positively homogenous. □

We have

$$\begin{aligned} G_{\tilde{y}}(c) &= \max \left\{ \frac{1}{2} \sum_{a \in \{0,1\}} \mathrm{F}(s^{(a,\tilde{y})}, \beta_{a,\tilde{y}} \alpha_a) : \sum_{a \in \{0,1\}} s^{(a,\tilde{y})} = c, \, s^{(a,\tilde{y})} \in \mathbb{R}_+^A, \, \forall a \in \{0,1\} \right\} \\ &= - \min \left\{ \frac{-1}{2} \sum_{a \in \{0,1\}} \mathrm{F}(s^{(a,\tilde{y})}, \beta_{a,\tilde{y}} \alpha_a) : \sum_{a \in \{0,1\}} s^{(a,\tilde{y})} = c, \, s^{(a,\tilde{y})} \in \mathbb{R}_+^A, \, \forall a \in \{0,1\} \right\} \\ &= - \left(\frac{-1}{2} \mathrm{F}(\cdot, \beta_{0,\tilde{y}} \alpha_0) \right) \, \square \, \left(\frac{-1}{2} \mathrm{F}(\cdot, \beta_{1,\tilde{y}} \alpha_1) \right) (c) \\ &= -\mathrm{conv} \left\{ \frac{-1}{2} \beta_{0,\tilde{y}} \mathrm{F}(\cdot, \alpha_0), \frac{-1}{2} \beta_{1,\tilde{y}} \mathrm{F}(\cdot, \alpha_1) \right\} (c) \\ &= \mathrm{conc} \left\{ \frac{1}{2} \beta_{0,\tilde{y}} \mathrm{F}(\cdot, \alpha_0), \frac{1}{2} \beta_{1,\tilde{y}} \mathrm{F}(\cdot, \alpha_1) \right\} (c). \end{aligned}$$

Thus, we can write Alice's optimization problem as

$$P^*_{\mathrm{A},0} = \max_c \sum_{y \in B} \mathrm{conc} \left\{ \frac{1}{2} \beta_{0,y} \mathrm{F}(\cdot, \alpha_0), \frac{1}{2} \beta_{1,y} \mathrm{F}(\cdot, \alpha_1) \right\} (c),$$

where c ranges over all feasible choices of the first n messages. Note the special case of $n = 1$, then this is simply $c \in \mathrm{Prob}^{A_1} = \mathrm{Prob}^A$. Alice's *improved eigenstrategy* is when Alice chooses c according to her eigenstrategy, yet reveals a optimally as above.

This may seem like an optimal solution since we are optimizing over the first n messages, then optimizing over the last message, but we are not optimizing over them simultaneously. However, this is a very good approximation of the optimal solution as seen in the numerical tests in Subsection 3.4.

Alice cheating in the six-round version

We want to maximize the objective function $\frac{1}{2} \sum_{a \in \{0,1\}} \sum_{y_1 \in B_1} \sum_{y_2 \in B_2} \beta_{a,y_1 y_2} \mathrm{F}(s^{(a,y_1 y_2)}, \alpha_a)$

over (s_1, s_2, s) satisfying:

$$\begin{array}{rcl} \mathrm{Tr}_{A_1}(s_1) & = & 1, \\ \mathrm{Tr}_{A_2}(s_2) & = & s_1 \otimes e_{B_1}, \\ \mathrm{Tr}_{A_0'}(s) & = & s_2 \otimes e_{B_2}, \\ s_1 & \in & \mathbb{R}_+^{A_1}, \\ s_2 & \in & \mathbb{R}_+^{A_1 \times B_1 \times A_2}, \\ s & \in & \mathbb{R}_+^{A_1 \times A_2 \times B_1 \times B_2 \times A_0'}. \end{array}$$

We suppose that Alice chooses her commitment a based on the most likely choice of b after seeing $y_1 \in B_1$ from Bob's first message. Let $f'(y_1) = \arg\max_{a \in A_0'} \{[\mathrm{Tr}_{B_2}(\beta_a)]_{y_1}\}$ and 0 in the case of a tie. The last constraint can be written as $\sum_{a \in A_0'} s^{(a, y_1 y_2)} = s_2^{(y_1)}$, for all $y_1 \in B_1$, where $s_2^{(y_1)}$ is the projection of s_2 with the index y_1 fixed. Alice's choice of commitment is equivalent to setting $s^{(a, y_1, y_2)} = s_2^{(y_1)}$, if $a = f'(y_1)$, and 0 otherwise. Now we set $s_2^{(y_1)} = s_2^0$, if $0 = f'(y_1)$, and $s_2^{(y_1)} = s_2^1$, if $1 = f'(y_1)$, where we optimize $s_2^0, s_2^1 \in \mathbb{R}_+^{A_1 \times A_2}$. s_2^0 and s_2^1 are the second messages which only depend on her first message and the inferred value of b from y_1.

The new objective function can be written as

$$\begin{array}{rcl} \frac{1}{2} \sum_{a \in A_0'} \sum_{y_1 \in B_1, y_2 \in B_2} \beta_{a, y_1 y_2} \mathrm{F}(s^{(a, y_1 y_2)}, \alpha_a) & = & \frac{1}{2} \sum_{y_1 \in B_1} \left[\sum_{y_2 \in B_2} \beta_{f'(y_1), y_1 y_2} \right] \mathrm{F}(s_2^{f'(y_1)}, \alpha_{f'(y_1)}) \\ & = & \frac{\eta'}{2} \mathrm{F}(s_2^0, \alpha_0) + \frac{\tau'}{2} \mathrm{F}(s_2^1, \alpha_1). \end{array}$$

Since the only constraints remaining are $\mathrm{Tr}_{A_2}(s_2^0) = s_1 = \mathrm{Tr}_{A_2}(s_2^1)$, we now optimize over each choice of s_2^0 and s_2^1 separately using the following lemma.

Lemma 3.1.8. *For $\alpha \in \mathbb{R}_+^{A_1 \times A_2}$ and $c \in \mathbb{R}_+^{A_1}$, we have*

$$\max \left\{ \mathrm{F}(p, \alpha) : \mathrm{Tr}_{A_2}(p) = c, \ p \in \mathbb{R}_+^{A_1 \times A_2} \right\} \geq \mathrm{F}(c, \mathrm{Tr}_{A_2}(\alpha)).$$

The inequality can be shown to hold with equality by Uhlmann's theorem. However, we prove the inequality by exhibiting a feasible solution which is also useful for the analysis of cheating Bob.

Proof. For each $x_1 \in A_1$, $x_2 \in A_2$, define p_{x_1,x_2} as

$$p_{x_1,x_2} := \begin{cases} c_{x_1} \dfrac{\alpha_{x_1,x_2}}{[\mathrm{Tr}_{A_2}(\alpha)]_{x_1}} & \text{if } [\mathrm{Tr}_{A_2}(\alpha)]_{x_1} > 0, \\ \\ c_{x_1} \dfrac{1}{|A_2|} & \text{if } [\mathrm{Tr}_{A_2}(\alpha)]_{x_1} = 0. \end{cases}$$

Then we have $p \geq 0$ is feasible since $[\mathrm{Tr}_{A_2}(p)]_{x_1} = c_{x_1}$ and it has objective function value

$$\begin{aligned} \mathrm{F}(p, \alpha) &= \left(\sum_{\substack{x_1,x_2: \\ \alpha_{x_1,x_2} > 0}} \sqrt{p_{x_1,x_2}} \sqrt{\alpha_{x_1,x_2}} \right)^2 \\ &= \left(\sum_{\substack{x_1,x_2: \\ \alpha_{x_1,x_2} > 0}} \sqrt{c_{x_1} \frac{\alpha_{x_1,x_2}}{[\mathrm{Tr}_{A_2}(\alpha)]_{x_1}}} \sqrt{\alpha_{x_1,x_2}} \right)^2 \\ &= \left(\sum_{x_1} \sqrt{c_{x_1}} \sqrt{[\mathrm{Tr}_{A_2}(\alpha)]_{x_1}} \right)^2, \\ &= \mathrm{F}(c, \mathrm{Tr}_{A_2}(\alpha)), \end{aligned}$$

as desired. □

Using Lemma 3.1.8, we can write Alice's problem as

$$\max_{c \in \mathrm{Prob}^{A_1}} \frac{\eta'}{2} \mathrm{F}(c, \mathrm{Tr}_{A_2}(\alpha_0)) + \frac{\tau'}{2} \mathrm{F}(c, \mathrm{Tr}_{A_2}(\alpha_1))$$

which has optimal objective value

$$\frac{1}{2} \lambda_{\max} \left(\eta' \sqrt{\mathrm{Tr}_{A_2}(\alpha_0)} \sqrt{\mathrm{Tr}_{A_2}(\alpha_0)}^\top + \tau' \sqrt{\mathrm{Tr}_{A_2}(\alpha_1)} \sqrt{\mathrm{Tr}_{A_2}(\alpha_1)}^\top \right).$$

This can be lower bounded by

$$\left(\frac{1}{2} + \frac{1}{2} \sqrt{\mathrm{F}(\mathrm{Tr}_{A_2}(\alpha_0), \mathrm{Tr}_{A_2}(\alpha_1))} \right) \left(\frac{1}{2} + \frac{1}{2} \Delta(\mathrm{Tr}_{B_2}(\beta_0), \mathrm{Tr}_{B_2}(\beta_1)) \right)$$

which is the success probability of the strategy where Alice measures the first message to learn b early and then tries to change the value of a. She can learn b with probability

$\frac{1}{2} + \frac{1}{2}\Delta(\mathrm{Tr}_{B_2}(\beta_0), \mathrm{Tr}_{B_2}(\beta_1))$. She can successfully change the value of a with probability $\frac{1}{2} + \frac{1}{2}\sqrt{\mathrm{F}(\mathrm{Tr}_{A_2}(\alpha_0), \mathrm{Tr}_{A_2}(\alpha_1))}$. Thus, she can cheat with probability at least

$$\left(\frac{1}{2} + \frac{1}{2}\sqrt{\mathrm{F}(\mathrm{Tr}_{A_2}(\alpha_0), \mathrm{Tr}_{A_2}(\alpha_1))}\right)\left(\frac{1}{2} + \frac{1}{2}\Delta(\mathrm{Tr}_{B_2}(\beta_0), \mathrm{Tr}_{B_2}(\beta_1))\right).$$

3.1.2 Cheating Bob

We now present a theorem capturing some of Bob's cheating strategies.

Theorem 3.1.9. *For any* BCCF*-protocol, we can bound Bob's optimal cheating probabilities as follows:*

$$P^*_{\mathrm{B},0} \geq \frac{1}{2} + \frac{1}{2}\sqrt{\mathrm{F}(\beta_0, \beta_1)} \tag{3.1}$$

and

$$P^*_{\mathrm{B},0} \geq \frac{1}{2} + \frac{1}{2}\Delta(\mathrm{Tr}_{A_2 \times \cdots \times A_n}(\alpha_0), \mathrm{Tr}_{A_2 \times \cdots \times A_n}(\alpha_1)). \tag{3.2}$$

We call (3.1) *Bob's ignoring strategy and* (3.2) *his measuring strategy.*

In a four-round protocol, we have

$$\begin{aligned} P^*_{\mathrm{B},0} &\geq \frac{1}{2}\sum_{a \in \{0,1\}} \mathrm{F}\left(\sum_{x \in A} \alpha_{a,x}(v_x), \beta_a\right) && (3.3) \\ &\geq \frac{1}{2}\sum_{x \in A} \lambda_{\max}\left(\sum_{a \in \{0,1\}} \alpha_{a,x}\sqrt{\beta_a}\sqrt{\beta_a}^\top\right) && (3.4) \\ &\geq \frac{1}{2} + \frac{1}{2}\Delta(\alpha_0, \alpha_1), \quad \frac{1}{2} + \frac{1}{2}\sqrt{\mathrm{F}(\beta_0, \beta_1)}, \end{aligned}$$

where v_x is the entry-wise square of the normalized, nonnegative principal eigenvector of

$$\sum_{a \in \{0,1\}} \alpha_{a,x}\sqrt{\beta_a}\sqrt{\beta_a}^\top.$$

We call (3.3) *Bob's eigenstrategy and* (3.4) *his eigenstrategy lower bound.*

In a six-round protocol, we have

$$
\begin{aligned}
P^*_{\mathrm{B},0} &\geq \frac{1}{2} \sum_{a \in A'_0} \mathrm{F}\left(\sum_{x \in A} \alpha_{a,x} \tilde{p_2}^{(x)}, \beta_a \right) && (3.5) \\
&\geq \frac{1}{2} \lambda_{\max} \left(\kappa \sqrt{\mathrm{Tr}_{B_2}(\beta_0)} \sqrt{\mathrm{Tr}_{B_2}(\beta_0)}^\top + \omega \sqrt{\mathrm{Tr}_{B_2}(\beta_1)} \sqrt{\mathrm{Tr}_{B_2}(\beta_1)}^\top \right) && (3.6) \\
&\geq \left(\frac{1}{2} + \frac{1}{2} \sqrt{\mathrm{F}(\mathrm{Tr}_{B_2}(\beta_0), \mathrm{Tr}_{B_2}(\beta_1))} \right) \left(\frac{1}{2} + \frac{1}{2} \Delta(\alpha_0, \alpha_1) \right), && (3.7)
\end{aligned}
$$

where

$$
\kappa = \sum_{\substack{x \in A: \\ \alpha_{0,x} \geq \alpha_{1,x}}} \alpha_{0,x}, \qquad \omega = \sum_{\substack{x \in A: \\ \alpha_{0,x} < \alpha_{1,x}}} \alpha_{1,x}, \qquad g(x) = \arg\max_a \{\alpha_{a,x}\},
$$

$$
[\tilde{p_2}^{(x)}] y_1, y_2 := \begin{cases} c_{y_1} \dfrac{\beta_{g(x),y_1,y_2}}{[\mathrm{Tr}_{B_2}(\beta_{g(x)})]_{y_1}} & \textit{if} \quad [\mathrm{Tr}_{B_2}(\beta_{g(x)})]_{y_1} > 0, \\ \\ c_{y_1} \dfrac{1}{|B_2|} & \textit{if} \quad [\mathrm{Tr}_{B_2}(\beta_{g(x)})]_{y_1} = 0, \end{cases}
$$

and c is the entry-wise square of the normalized, nonnegative principal eigenvector of

$$
\kappa \sqrt{\mathrm{Tr}_{B_2}(\beta_0)} \sqrt{\mathrm{Tr}_{B_2}(\beta_0)}^\top + \omega \sqrt{\mathrm{Tr}_{B_2}(\beta_1)} \sqrt{\mathrm{Tr}_{B_2}(\beta_1)}^\top .
$$

We call (3.5) *Bob's six-round eigenstrategy,* (3.6) *his six-round eigenstrategy lower bound, and* (3.7) *his three-round strategy. Furthermore, if $|A_i| = |B_i|$, for all $i \in \{1, \ldots, n\}$, then*

$$
P^*_{\mathrm{B},0} \geq \frac{1}{2} \sum_{a \in \{0,1\}} \mathrm{F}(\alpha_a, \beta_a), \tag{3.8}
$$

which we call Bob's returning strategy. We can lower bound $P^*_{\mathrm{B},1}$ as well if we switch the roles of β_0 and β_1 above.

Note that the only strategies that are affected by switching β_0 and β_1 are the eigenstrategies and the returning strategy.

We now briefly describe the strategies that yield the corresponding cheating probabilities in Theorem 3.1.9.

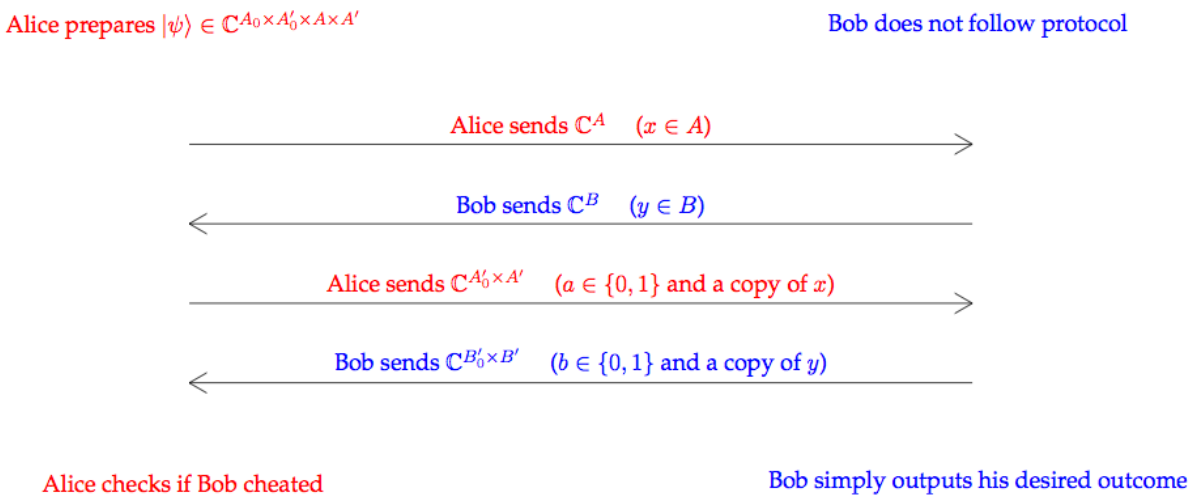

Alice prepares $|\psi\rangle \in \mathbb{C}^{A_0 \times A_0' \times A \times A'}$
Bob does not follow protocol
Alice sends $\mathbb{C}^A$ $(x \in A)$
Bob sends $\mathbb{C}^B$ $(y \in B)$
Alice sends $\mathbb{C}^{A_0' \times A'}$ ($a \in \{0,1\}$ and a copy of x)
Bob sends $\mathbb{C}^{B_0' \times B'}$ ($b \in \{0,1\}$ and a copy of y)
Alice checks if Bob cheated
Bob simply outputs his desired outcome

Figure 3.3: Bob cheating in a four-round BCCF-protocol.

His ignoring strategy is to ignore all of Alice's messages until a is revealed, then try to change his value of b. His measuring strategy is to measure Alice's first message and choose b according to his best guess for a. His returning strategy is to send Alice's messages right back to her. For the four-round eigenstrategy, Bob's first message is a principal eigenvector depending on Alice's first message.

For the six-round version (figure on the next page), his three-round strategy is to send his first message as in a three-round protocol, measure Alice's second message to obtain a guess for a, then try to change the value of b. His six-round eigenstrategy is the same as his three-round strategy, except the first message is optimized in a way described later in this section.

Developing Bob's strategies

Bob's returning strategy is to send Alice's messages right back to her (if the dimensions agree). If $\alpha_0 = \beta_0$ and $\alpha_1 = \beta_1$, then this strategy allows Bob to win perfectly if he wants outcome 0 since if $a = 0$ and Alice sends $|\psi_0\rangle$, then Bob replies with $b = a$ and the state to be checked is $|\psi_0\rangle = |\phi_0\rangle$. To evaluate the success probability of this strategy for any choice of parameters, it is easier to use the original cheating SDP as opposed to the reduced

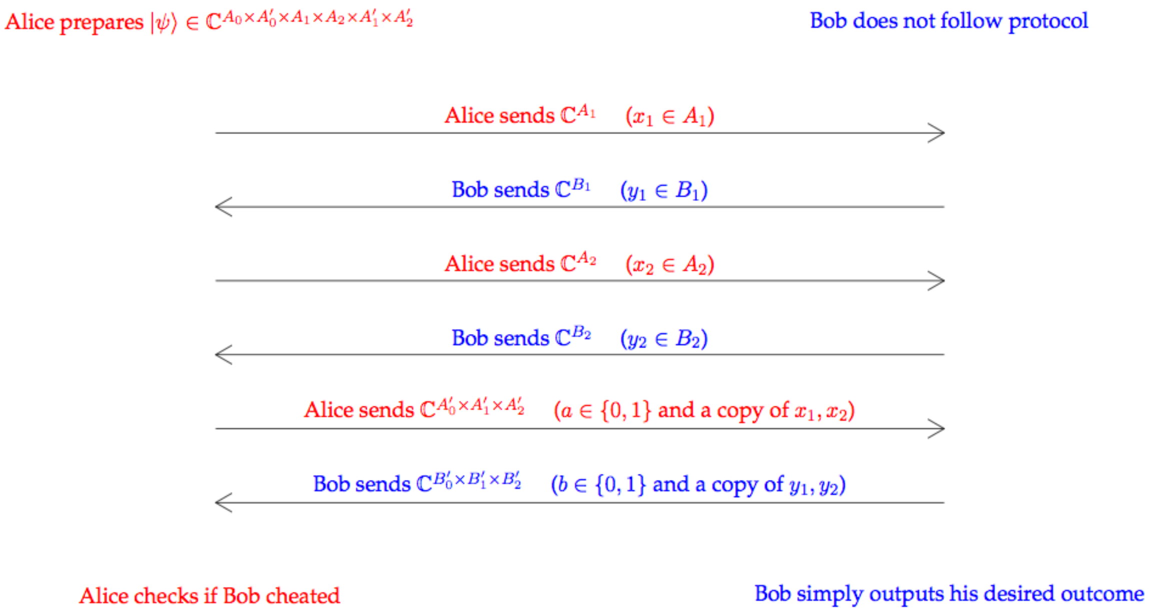

Alice prepares $|\psi\rangle \in \mathbb{C}^{A_0\times A_0'\times A_1\times A_2\times A_1'\times A_2'}$
Bob does not follow protocol
Alice sends $\mathbb{C}^{A_1}$ $(x_1 \in A_1)$
Bob sends $\mathbb{C}^{B_1}$ $(y_1 \in B_1)$
Alice sends $\mathbb{C}^{A_2}$ $(x_2 \in A_2)$
Bob sends $\mathbb{C}^{B_2}$ $(y_2 \in B_2)$
Alice sends $\mathbb{C}^{A_0'\times A_1'\times A_2'}$ $(a \in \{0,1\}$ and a copy of $x_1, x_2)$
Bob sends $\mathbb{C}^{B_0'\times B_1'\times B_2'}$ $(b \in \{0,1\}$ and a copy of $y_1, y_2)$
Alice checks if Bob cheated
Bob simply outputs his desired outcome

Figure 3.4: Bob cheating in a six-round BCCF-protocol.

cheating SDP. This cheating strategy corresponds to the feasible solution

$$\bar{\rho}_1 = \bar{\rho}_2 = \cdots = \bar{\rho}_n = \bar{\rho}_F = |\psi\rangle\langle\psi|$$

which has success probability

$$\langle \bar{\rho}_F, \Pi_{\mathrm{A},0} \rangle = \langle |\psi\rangle\langle\psi|, \Pi_{\mathrm{A},0} \rangle = \frac{1}{2} \sum_{a \in \{0,1\}} \mathrm{F}(\alpha_a, \beta_a).$$

This is clearly optimal when $\alpha_0 = \beta_0$ and $\alpha_1 = \beta_1$.

Recall Bob's reduced problem below

$$P^*_{\mathrm{B},0} = \max \left\{ \frac{1}{2} \sum_{a \in \{0,1\}} \mathrm{F}\left((\alpha_a \otimes \mathrm{I}_B)^\top p_n,\, \beta_a \right) : (p_1, \ldots, p_n) \in \mathcal{P}_\mathrm{B} \right\},$$

where $\mathcal{P}_{\mathrm{B}}$ is the set of vectors $(p_1, p_2, \ldots, p_n)$ satisfying

$$\begin{array}{rcl}
\mathrm{Tr}_{B_1}(p_1) & = & e_{A_1}, \\
\mathrm{Tr}_{B_2}(p_2) & = & p_1 \otimes e_{A_2}, \\
 & \vdots & \\
\mathrm{Tr}_{B_n}(p_n) & = & p_{n-1} \otimes e_{A_n}, \\
p_j & \in & \mathbb{R}_+^{A_1 \times B_1 \times \cdots \times A_j \times B_j}, \text{ for all } j \in \{1, \ldots, n\}.
\end{array}$$

There is a strategy for Bob that works for any n and is very important in the search algorithm. This is the strategy where Bob ignores all of Alice's messages and tries to choose b after learning a from Alice. By ignoring Alice's messages, he is effectively setting $p_n = e_A \otimes d$, for some $d \in \mathrm{Prob}^B$, which we optimize. Under this restriction, he can cheat with probability

$$\begin{array}{rcl}
\max\limits_{d \in \mathrm{Prob}^B} \dfrac{1}{2} \sum\limits_{a \in \{0,1\}} \mathrm{F}\left((\alpha_a \otimes \mathrm{I}_B)^\top (e_A \otimes d),\, \beta_a\right) & = & \max\limits_{d \in \mathrm{Prob}^B} \dfrac{1}{2} \sum\limits_{a \in \{0,1\}} \mathrm{F}\left(d, \beta_a\right) \\
 & = & \dfrac{1}{2} \lambda_{\max}\left(\sqrt{\beta_0}\sqrt{\beta_0}^\top + \sqrt{\beta_1}\sqrt{\beta_1}^\top\right) \\
 & = & \dfrac{1}{2} + \dfrac{1}{2}\sqrt{\mathrm{F}(\beta_0, \beta_1)},
\end{array}$$

using Lemma 3.1.3 and Lemma 1.4.13. The reason this strategy is important is that it is relatively cheap to compute, only depends on half of the parameters, and performs quite well under these circumstances. We call this Bob's *ignoring strategy.*

Another strategy for Bob is to measure Alice's first message, choose b accordingly, then follow the rest of the protocol honestly. This is called Bob's *measuring strategy* and succeeds with probability

$$\frac{1}{2} + \frac{1}{2}\Delta(\mathrm{Tr}_{A_2 \times \cdots \times A_n}(\alpha_0), \mathrm{Tr}_{A_2 \times \cdots \times A_n}(\alpha_1)),$$

when $n \geq 2$. We discuss the case of $n = 1$, i.e. four-round protocols, separately.

Bob cheating in the four-round version

In four-round protocols, some cheating strategies exist that do not work for protocols with more rounds. One reason is because Bob has all of Alice's $\mathbb{C}^A$ space before he must send any messages. We show that Bob can use this to his advantage. One example is Bob's measuring strategy can simply be written as

$$\frac{1}{2} + \frac{1}{2}\Delta(\alpha_0, \alpha_1).$$

Similar to cheating Alice, we can develop an eigenstrategy for Bob. For the special case of four-round protocols, notice that Bob's cheating polytope contains only the constraints $\mathrm{Tr}_B(p_1) = e_A$ and $p_1 \in \mathbb{R}_+^{A \times B}$. This can be rewritten as $p_1^{(x)} \in \mathrm{Prob}^B$ for all $x \in A$. Also, $\mathrm{F}\left((\alpha_a \otimes \mathrm{I}_B)^\top p_1,\, \beta_a\right)$ can be written as $\mathrm{F}\left(\sum_{x \in A} \alpha_{a,x} p_1^{(x)}, \beta_a\right)$. Thus, we can rewrite Bob's reduced problem as

$$P^*_{\mathrm{B},0} = \max \left\{ \frac{1}{2} \sum_{a \in \{0,1\}} \mathrm{F}\left(\sum_{x \in A} \alpha_{a,x} p_1^{(x)}, \beta_a \right) : p_1^{(x)} \in \mathrm{Prob}^B, \text{ for all } x \in A \right\}.$$

Since fidelity is concave, we have that $\mathrm{F}\left(\sum_{x \in A} \alpha_{a,x} p_1^{(x)}, \beta_a\right) \geq \sum_{x \in A} \alpha_{a,x} \mathrm{F}(p_1^{(x)}, \beta_a)$. Therefore, Bob's optimal cheating probability is bounded below by

$$\max \left\{ \frac{1}{2} \sum_{x \in A} \sum_{a \in \{0,1\}} \alpha_{a,x} \mathrm{F}(p_1^{(x)}, \beta_a) : p_1^{(x)} \in \mathrm{Prob}^B, \text{ for all } x \in A \right\}$$

which separates over $x \in A$. That is, we can choose each $p_1^{(x)} \in \mathrm{Prob}^B$ separately to maximize

$$\sum_{a \in \{0,1\}} \alpha_{a,x} \mathrm{F}(p_1^{(x)}, \beta_a).$$

Lemma 3.1.3 shows that each subproblem has an optimal objective value of

$$\frac{1}{2} \lambda_{\max} \left(\sum_{a \in \{0,1\}} \alpha_{a,x} \sqrt{\beta_a} \sqrt{\beta_a}^\top \right).$$

Thus, we know that

$$P^*_{\mathrm{B},0} \geq \frac{1}{2} \sum_{x \in A} \lambda_{\max} \left(\sum_{a \in \{0,1\}} \alpha_{a,x} \sqrt{\beta_a} \sqrt{\beta_a}^\top \right).$$

Since we are using the concavity of the objective function to bound it, we are getting a loose bound. Notice that solving the smaller separated problems yields a solution which is feasible for the original problem. Therefore, we can substitute this into the original objective function to get a better lower bound on Bob's optimal cheating probability. We call this Bob's *eigenstrategy.*

Since eigenvalues are expensive to compute, we can bound this quantity by

$$\begin{aligned}\frac{1}{2}\sum_{x \in A} \lambda_{\max}\left(\sum_{a \in \{0,1\}} \alpha_{a,x}\sqrt{\beta_a}\sqrt{\beta_a}^\top\right) &\geq \min_{\beta_0,\beta_1 \in \text{Prob}^B} \frac{1}{2}\sum_{x \in A} \lambda_{\max}\left(\sum_{a \in \{0,1\}} \alpha_{a,x}\sqrt{\beta_a}\sqrt{\beta_a}^\top\right) \\ &= \frac{1}{2}\sum_{x \in A} \max_{a \in \{0,1\}} \{\alpha_{a,x}\} \\ &= \frac{1}{2} + \frac{1}{2}\Delta(\alpha_0, \alpha_1),\end{aligned}$$

where the last equality follows from Lemma 1.4.14.

Since $f(X) = \lambda_{\max}(X)$ is subadditive, i.e., $\lambda_{\max}(X_1) + \lambda_{\max}(X_2) \geq \lambda_{\max}(X_1 + X_2)$, we have that

$$\begin{aligned}\frac{1}{2}\sum_{x \in A} \lambda_{\max}\left(\sum_{a \in \{0,1\}} \alpha_{a,x}\sqrt{\beta_a}\sqrt{\beta_a}^\top\right) &\geq \frac{1}{2}\lambda_{\max}\left(\sum_{x \in A}\sum_{a \in \{0,1\}} \alpha_{a,x}\sqrt{\beta_a}\sqrt{\beta_a}^\top\right) \\ &= \frac{1}{2}\lambda_{\max}\left(\sum_{a \in \{0,1\}} \sqrt{\beta_a}\sqrt{\beta_a}^\top\right) \\ &= \frac{1}{2} + \frac{1}{2}\sqrt{\mathrm{F}(\beta_0, \beta_1)},\end{aligned}$$

using Lemma 1.4.13. Therefore, Bob's eigenstrategy lower bound performs better than both his measuring strategy and ignoring strategy.

Bob cheating in the six-round version

We want to maximize the objective function $\frac{1}{2}\sum_{a \in \{0,1\}} \mathrm{F}((\alpha_a \otimes \mathrm{I}_{B_1 \times B_2})^\top p_2, \beta_a)$ over (p_1, p_2)

satisfying:

$$\begin{array}{rcl} \mathrm{Tr}_{B_1}(p_1) & = & e_{A_1}, \\ \mathrm{Tr}_{B_2}(p_2) & = & p_1 \otimes e_{A_2}, \\ p_1 & \in & \mathbb{R}_+^{A_1 \times B_1}, \\ p_2 & \in & \mathbb{R}_+^{A_1 \times B_1 \times A_2 \times B_2}. \end{array}$$

Like the four-round version, we can lower bound the objective function as

$$\frac{1}{2} \sum_{a \in A_0'} \mathrm{F}\left(\sum_{x \in A} \alpha_{a,x} p_2^{(x)}, \beta_a \right) \geq \frac{1}{2} \sum_{x \in A} \sum_{a \in A_0'} \mathrm{F}(p_2^{(x)}, \alpha_{a,x}\beta_a).$$

We make the restriction $p_1^{(x_1)} = c$, for all $x_1 \in A_1$, for a fixed value of c which we optimize later. We now analyze the subproblem

$$\max \left\{ \sum_{a \in A_0'} \mathrm{F}(p_2^{(x)}, \alpha_{a,x}\beta_a) : \mathrm{Tr}_{B_2}(p_2^{(x)}) = c,\, p_2^{(x)} \in \mathbb{R}_+^{B} \right\}$$

using the following lemma.

Lemma 3.1.10. *For $\beta_0, \beta_1 \in \mathbb{R}_+^{B_1 \times B_2}$ and $c \in \mathbb{R}_+^{B_1}$, we have*

$$\max \left\{ \sum_{a \in \{0,1\}} \mathrm{F}(p, \beta_a) : \mathrm{Tr}_{B_2}(p) = c,\, p \in \mathbb{R}_+^{B_1 \times B_2} \right\} \geq \mathrm{F}(c, \mathrm{Tr}_{B_2}(\beta_{\tilde{a}})),$$

for any $\tilde{a} \in \{0, 1\}$.

Proof. Just fix a value for a then choose $p \in \arg\max \left\{ \mathrm{F}(p, \beta_a) : \mathrm{Tr}_{B_2}(p) = c,\, p \in \mathbb{R}_+^{B_1 \times B_2} \right\}$. The result follows since $\mathrm{F}(p, \beta_{\bar{a}}) \geq 0$ and by Lemma 3.1.8. □

We now apply Lemma 3.1.10 to lower bound Alice's problem as

$$\sum_{a \in A_0'} \mathrm{F}(p_2^{(x)}, \alpha_{a,x}\beta_a) \geq \alpha_{g(x),x}\, \mathrm{F}(c, \mathrm{Tr}_{B_2}(\beta_{g(x)})),$$

where $g(x) := \arg\max_{a \in A_0'} \{\alpha_{a,x}\}$, and 0 in the case of a tie.

Substituting this into the subproblem, we have

$$\begin{aligned}
P^*_{\mathrm{B},0} &\geq \max_{c \in \mathrm{Prob}^{B_1}} \frac{1}{2} \sum_{x \in A} \alpha_{g(x),x} \, \mathrm{F}(c, \mathrm{Tr}_{B_2}(\beta_{g(x)})) \\
&= \max_{c \in \mathrm{Prob}^{B_1}} \frac{\kappa}{2} \, \mathrm{F}(c, \mathrm{Tr}_{B_2}(\beta_0)) + \frac{\omega}{2} \, \mathrm{F}(c, \mathrm{Tr}_{B_2}(\beta_1)) \\
&= \frac{1}{2} \lambda_{\max} \left(\kappa \sqrt{\mathrm{Tr}_{B_2}(\beta_0)} \sqrt{\mathrm{Tr}_{B_2}(\beta_0)}^\top + \omega \sqrt{\mathrm{Tr}_{B_2}(\beta_1)} \sqrt{\mathrm{Tr}_{B_2}(\beta_1)}^\top \right) \qquad (3.6) \\
&\geq \left(\frac{1}{2} + \frac{1}{2} \Delta(\alpha_0, \alpha_1) \right) \left(\frac{1}{2} + \frac{1}{2} \sqrt{\mathrm{F}(\mathrm{Tr}_{B_2}(\beta_0), \mathrm{Tr}_{B_2}(\beta_1))} \right), \qquad (3.7)
\end{aligned}$$

by the same arguments used for cheating Alice. We call (3.6) Bob's six-round eigenstrategy lower bound. The quantity (3.7) corresponds to the strategy where Bob measures Alice's second message to try to learn a early, then tries to change the value of b. He can learn a after Alice's second message with probability $\frac{1}{2} + \frac{1}{2}\Delta(\alpha_0, \alpha_1)$. He can change the value of b with probability $\frac{1}{2} + \frac{1}{2}\sqrt{\mathrm{F}(\mathrm{Tr}_{B_2}(\beta_0), \mathrm{Tr}_{B_2}(\beta_1))}$. Thus, he can cheat with probability at least

$$\left(\frac{1}{2} + \frac{1}{2} \sqrt{\mathrm{F}(\mathrm{Tr}_{B_2}(\beta_0), \mathrm{Tr}_{B_2}(\beta_1))} \right) \left(\frac{1}{2} + \frac{1}{2} \Delta(\alpha_0, \alpha_1) \right).$$

We call this Bob's *three-round strategy.*

Although we used many bounds in developing the quantity (3.6), such as the lower bound in Lemma 3.1.10, we can recover some of the losses by generating the corresponding feasible solution and computing its objective function value for the original objective function. For example, we can calculate c as the entry-wise square of the normalized, nonnegative principal eigenvector of

$$\kappa \sqrt{\mathrm{Tr}_{B_2}(\beta_0)} \sqrt{\mathrm{Tr}_{B_2}(\beta_0)}^\top + \omega \sqrt{\mathrm{Tr}_{B_2}(\beta_1)} \sqrt{\mathrm{Tr}_{B_2}(\beta_1)}^\top,$$

then calculate $p_2^{(x)}$ for each value of x from the construction of the feasible solution in the proof of Lemma 3.1.8. We call this Bob's *six-round eigenstrategy.*

3.2 Protocol symmetry

In this section, we discuss symmetry in BCCF-protocols. Symmetry is very important in computational optimization since it can greatly reduce the time for computations. For BCCF-protocols, we show that there is much symmetry in the parameters.

3.2.1 Symmetry in local permutations

We show that if we permute the elements of A_i, or B_i, for some fixed $i \in \{1, \ldots, n\}$, then this does not change the bias of the protocol. We first show that cheating Bob is unaffected.

Cheating Bob: Bob's reduced problems are to maximize $\frac{1}{2} \sum_{a \in A_0'} \mathrm{F}\left((\alpha_a \otimes \mathrm{I}_B)^\top p_n,\, \beta_a\right)$, for forcing outcome 0, and $\frac{1}{2} \sum_{a \in A_0'} \mathrm{F}\left((\alpha_a \otimes \mathrm{I}_B)^\top p_n,\, \beta_{\bar{a}}\right)$, for forcing outcome 1, over the polytope $\mathcal{P}_{\mathrm{B}}$ defined as the set of all vectors $(p_1, p_2, \ldots, p_n)$ satisfying

$$\begin{array}{rcl}
\mathrm{Tr}_{B_1}(p_1) & = & e_{A_1}, \\
\mathrm{Tr}_{B_2}(p_2) & = & p_1 \otimes e_{A_2}, \\
& \vdots & \\
\mathrm{Tr}_{B_n}(p_n) & = & p_{n-1} \otimes e_{A_n}, \\
p_j & \in & \mathbb{R}_+^{A_1 \times B_1 \times \cdots \times A_j \times B_j}, \text{ for all } j \in \{1, \ldots, n\}.
\end{array}$$

Suppose we are given a new protocol where the elements of A_i have been permuted, for some $i \in \{1, \ldots, n\}$ (and likewise the entries of α_a for $a \in \{0, 1\}$). We can write the entries of $(\alpha_a \otimes \mathrm{I}_B)^\top p_n$ as

$$[(\alpha_a \otimes \mathrm{I}_B)^\top p_n]_y = \sum_{x \in A} \alpha_{a,x} p_{n,x,y},$$

for each $y \in B$. If we permute the entries in p_n corresponding to A_i (and likewise for every variable in the polytope) we get the same objective function value. Thus, dishonest Bob cannot cheat more or less than the original protocol.

Now suppose we are given a new protocol where the elements of B_i have been permuted for some $i \in \{1, \ldots, n\}$. We can write

$$\mathrm{F}\left((\alpha_a \otimes \mathrm{I}_B)^\top p_n,\, \beta_a\right) = \left(\sqrt{(\alpha_a \otimes \mathrm{I}_B)^\top p_n}^\top \sqrt{\beta_a}\right)^2.$$

If we permute the entries in p_n corresponding to B_i (and likewise for every variable in the polytope) we get the same objective function value. A similar argument holds for $P^*_{\mathrm{B},1}$. In both cases, Bob's two cheating probabilities are unaffected.

Cheating Alice: To show the bias remains unchanged, we still need to check that cheating Alice is unaffected by these permutations. Alice's reduced problems are to maximize the objective function $\frac{1}{2}\sum_{a \in A_0'} \sum_{y \in B} \beta_{a,y} \ \mathrm{F}(s^{(a,y)}, \alpha_a)$ for outcome 0, and to maximize $\frac{1}{2}\sum_{a \in A_0'} \sum_{y \in B} \beta_{\bar{a},y} \ \mathrm{F}(s^{(a,y)}, \alpha_a)$ for outcome 1, over the set of all vectors $(s_1, s_2, \ldots, s_n, s)$ satisfying

$$\begin{array}{rcl}
\mathrm{Tr}_{A_1}(s_1) & = & 1, \\
\mathrm{Tr}_{A_2}(s_2) & = & s_1 \otimes e_{B_1}, \\
 & \vdots & \\
\mathrm{Tr}_{A_n}(s_n) & = & s_{n-1} \otimes e_{B_{n-1}}, \\
\mathrm{Tr}_{A_0'}(s) & = & s_n \otimes e_{B_n}, \\
s_j & \in & \mathbb{R}_+^{A_1 \times B_1 \times \cdots \times B_{j-1} \times A_j}, \text{ for all } j \in \{1, \ldots, n\}, \\
s & \in & \mathbb{R}_+^{A \times B \times A_0'}.
\end{array}$$

By examining the above problem, we see that the same arguments that apply to cheating Bob also apply to cheating Alice. We can simply permute any feasible solution to account for any permutation in A_i or B_i.

Note that these arguments only hold for "local" permutations, i.e., if we permute the elements between Cartesian products in $A_1 \times \cdots \times A_n$ the bias may change.

3.2.2 Symmetry between probability vectors

We now show a different kind of symmetry in the protocols. Recall the four objective functions below

$$P^*_{\mathrm{B},0} = \frac{1}{2}\sum_{a \in A_0'} \mathrm{F}\left((\alpha_a \otimes \mathrm{I}_B)^\top p_n, \, \beta_a\right) \quad \text{and} \quad P^*_{\mathrm{B},1} = \frac{1}{2}\sum_{a \in A_0'} \mathrm{F}\left((\alpha_a \otimes \mathrm{I}_B)^\top p_n, \, \beta_{\bar{a}}\right)$$

for Bob and

$$P^*_{\mathrm{A},0} = \frac{1}{2}\sum_{y \in B} \sum_{a \in \{0,1\}} \beta_{a,y} \mathrm{F}(s^{(a,y)}, \alpha_a) \quad \text{and} \quad P^*_{\mathrm{A},1} = \frac{1}{2}\sum_{y \in B} \sum_{a \in \{0,1\}} \beta_{\bar{a},y} \mathrm{F}(s^{(a,y)}, \alpha_a)$$

for Alice. We argue that the four quantities above are not effected if we switch β_0 and β_1 and simultaneously switch α_0 and α_1. This is obvious for cheating Bob, but it requires explanation for cheating Alice. The only constraints involving $s^{(a,y)}$ can be written as

$$\sum_{a \in A_0'} s^{(a,y)} = s_n^{(y_1, \ldots, y_{n-1})},$$

for all $y = (y_1, \ldots, y_{n-1}, y_n) \in B$. Since this constraint is symmetric about a, the result follows.

We have shown that switching the roles of β_0 and β_1 switches $P^*_{\text{A},0}$ and $P^*_{\text{A},1}$ and also switches $P^*_{\text{B},0}$ and $P^*_{\text{B},1}$. With these symmetries, we can effectively switch the roles of α_0 and α_1 and the roles of β_0 and β_1 independently and the bias is unaffected.

How we apply the symmetry in the four and six-round versions

Since we are able to switch the roles of α_0 and α_1, we can assume α_0 has the largest entry out of α_0 and α_1 and similarly that β_0 has the largest entry out of β_0 and β_1.

In the four-round version, since we can permute the elements of $A = A_1$, we can also assume α_0 has entries that are non-decreasing. This allows us to upper bound all the entries of α_0 and α_1 by the last entry in α_0. We can do this simultaneously for β_0 and β_1.

In the six-round version, we need to be careful when applying the symmetry arguments, we cannot permute all of the entries in α_0. The symmetry only applies to local permutations so we can only partially order them. We can order A_2 such that the entries $\alpha_{0,\tilde{x}_1 x_2}$ do not decrease for *one particular index* $\tilde{x}_1 \in A_1$. It is convenient to choose the index corresponding to the largest entry. Then we can order the last block of entries in α_0 such that they do not decrease. Note that the last entry in α_0 is now the largest among all the entries in α_0 and α_1. We can do this simultaneously for β_0 and β_1.

3.3 The search algorithm

In this section, we describe an algorithm for finding BCCF-protocols with small bias. We start with the following search algorithm prototype.

Pseudo-algorithm for finding BCCF-protocols with small bias

For each choice of α_0, α_1, β_0, and β_1 (modulo the symmetry):

- Use the protocol filter to test if the protocol has large bias.
- Solve the necessary SDPs (or SOCPs) characterizing the bias.
- If the protocol has small bias: stop and output the protocol parameters.

We now discuss a few key steps to speed up the search and how to iterate over the parameters.

3.3.1 A protocol with bias $1/4$

We now present a protocol and give two proofs of its bias. This acts as a cut-off point for a "small bias." We can extend the three-round protocol in [KN04] to a four-round BCCF-protocol by defining

$$A = \{0,1,2\}\,,\ B = \{0,1\}\,,\quad \alpha_0 = \frac{1}{2}\,[1,0,1]^\top\,,\ \alpha_1 = \frac{1}{2}\,[0,1,1]^\top\,,\ \beta_0 = [1,0]^\top\,,\ \beta_1 = [0,1]^\top\,.$$

Notice that $\beta_0 \perp \beta_1$, so b is completely revealed in Bob's first message. We now show that the bias of this protocol is $\varepsilon = 1/4$.

A quantum proof

We examine the most general cheating strategies of Alice and Bob.

Cheating Alice: We see that Bob's first message completely reveals his committed bit. Therefore, Alice must choose her first message, call it $\sigma \in \Sigma_+^A$, such that when she learns b after Bob's message, she can fool him into thinking that she chose $a = b$ at the beginning of the protocol (if she wants outcome 0). If $\sigma \in \Sigma_+^A$ was sent in the first message, and b is revealed, then the success probability is

$$\mathrm{F}(\sigma, \mathrm{Tr}_B|\psi_b\rangle\langle\psi_b|) = \mathrm{F}(\sigma, \mathrm{Diag}(\alpha_b)),$$

where $|\psi_b\rangle := \frac{1}{\sqrt{2}}|bb\rangle + \frac{1}{\sqrt{2}}|22\rangle \in \mathbb{C}^3 \otimes \mathbb{C}^3$, as discussed in Subsection 1.4.3. Therefore, the optimal success probability is

$$\max_{\sigma} \sum_{b \in \{0,1\}} \frac{1}{2} \, \mathrm{F}(\sigma, \mathrm{Diag}(\alpha_b)),$$

where $\sigma \in \Sigma_+^A$ is a density matrix. Define the following quantum channel

$$\Xi(X) := \mathrm{Diag}(\mathrm{diag}(X)).$$

Since the fidelity does not decrease after applying a quantum channel to both inputs, we have

$$\begin{aligned}
\max_{\sigma} \sum_{b \in \{0,1\}} \frac{1}{2} \, \mathrm{F}(\sigma, \mathrm{Diag}(\alpha_b)) &\leq \max_{\sigma} \sum_{b \in \{0,1\}} \frac{1}{2} \, \mathrm{F}(\Xi(\sigma), \Xi(\mathrm{Diag}(\alpha_b))) \\
&= \max_{c \in \mathrm{Prob}^A} \sum_{b \in \{0,1\}} \frac{1}{2} \, \mathrm{F}(c, \alpha_b) \\
&= \frac{1}{2} \lambda_{\max} \left(\sqrt{\alpha_0} \sqrt{\alpha_0}^\top + \sqrt{\alpha_1} \sqrt{\alpha_1}^\top \right) \\
&= \frac{1}{2} + \frac{1}{2} \sqrt{\mathrm{F}(\alpha_0, \alpha_1)} \\
&= \frac{3}{4},
\end{aligned}$$

using Lemmas 3.1.3 and 1.4.13. This upper bound can be achieved by choosing σ to be diagonal. This is independent of whether she wants outcome 0 or 1, thus $P^*_{\mathrm{A},0} = P^*_{\mathrm{A},1} = 3/4$.

Cheating Bob: If Bob wants to cheat, he must infer the value of a from Alice's first message since $\beta_0 \perp \beta_1$ which binds him to a value of b from his first message. Thus, he can cheat with probability given by

$$\frac{1}{2} + \frac{1}{2} \Delta(\alpha_0, \alpha_1) = 3/4,$$

as discussed in Subsection 1.4.3. Therefore, $P^*_{\mathrm{B},0} = P^*_{\mathrm{B},1} = 3/4$ as well, proving $\varepsilon = 1/4$ for this protocol.

An SDP proof

Numerically solving for the cheating probabilities for this protocol shows that

$$P^*_{\mathrm{A},0} = P^*_{\mathrm{A},1} = P^*_{\mathrm{B},0} = P^*_{\mathrm{B},1} = 3/4.$$

However, when it comes to proving the bias of a protocol, one should take numerical findings such as these as supporting evidence only. Thus, we prove these are the optimal values of the four cheating SDPs by presenting feasible primal and dual solutions with matching objective function values. Note, sometimes it suffices to just prove an upper bound. In these cases, one could just find feasible solutions for the dual SDPs.

Cheating Alice: We can adapt the quantum proof (or the output from numerical tests) to find a feasible primal solution to Alice's reduced cheating SDP for forcing outcome 0. Define

$$(s_1, s^{(0,0)}, s^{(0,1)}, s^{(1,0)}, s^{(1,1)}) := (c, c, 0, 0, c), \quad \text{where } c := \begin{bmatrix} 1/6 \\ 1/6 \\ 2/3 \end{bmatrix} \in \mathrm{Prob}^A.$$

We have $(s_1, s) \in \mathcal{P}_\mathrm{A}$ with objective function value

$$\frac{1}{2} \sum_{a \in \{0,1\}} \sum_{y \in B} \beta_{a,y} \, \mathrm{F}(s^{(a,y)}, \alpha_a) = \frac{1}{2} \, \mathrm{F}(c, \alpha_0) + \frac{1}{2} \, \mathrm{F}(c, \alpha_1) = 3/4.$$

Therefore, $P^*_{\mathrm{A},0} \geq 3/4$. Alice's dual SDP for forcing outcome 0 is

$$\begin{array}{rrcl} \inf & \mathrm{Tr}_A(w) & & \\ \text{subject to} & w \otimes e_B & \geq & \dfrac{1}{2} \begin{bmatrix} 1/2 \\ 0 \\ 1/2 \end{bmatrix} \otimes v_0 + \dfrac{1}{2} \begin{bmatrix} 0 \\ 1/2 \\ 1/2 \end{bmatrix} \otimes v_1, \\ & \mathrm{Diag}(v_0) & \succeq & \begin{bmatrix} 1 & 0 \\ 0 & 0 \end{bmatrix}, \\ & \mathrm{Diag}(v_1) & \succeq & \begin{bmatrix} 0 & 0 \\ 0 & 1 \end{bmatrix}, \end{array}$$

which has the feasible solution

$$(v_0, v_1, w) = \left(\begin{bmatrix} 1 \\ 0 \end{bmatrix}, \begin{bmatrix} 0 \\ 1 \end{bmatrix}, \begin{bmatrix} 1/4 \\ 1/4 \\ 1/4 \end{bmatrix} \right)$$

with corresponding objective function value 3/4. Thus, $P^*_{\mathrm{A},0} \leq 3/4$ proving $P^*_{\mathrm{A},0} = 3/4$. By switching

$$v_0 \longleftrightarrow v_1, \quad s^{(0,0)} \longleftrightarrow s^{(0,1)}, \quad \text{and} \quad s^{(1,0)} \longleftrightarrow s^{(1,1)},$$

we get primal and dual feasible solutions for Alice forcing outcome 1, both having objective function value 3/4. Therefore, we have $P^*_{\mathrm{A},1} = 3/4$, as well.

Cheating Bob: We now repeat the process for cheating Bob. Define $p \in \mathbb{R}^{A \times B}_+$ as

$$p := [1, 0,\ 0, 1,\ 1/2, 1/2]^\top \in \mathcal{P}_\mathrm{B},$$

which has objective function value

$$\frac{1}{2} \sum_{a \in \{0,1\}} \mathrm{F}\left((\alpha_a \otimes \mathrm{I}_A)^\top p,\ \beta_a \right) = \frac{1}{2} [(\alpha_0 \otimes \mathrm{I}_A)^\top p]_0 + [(\alpha_1 \otimes \mathrm{I}_A)^\top p]_1 = 3/4$$

for forcing outcome 0. By defining

$$p := [0, 1,\ 1, 0,\ 1/2, 1/2]^\top \in \mathcal{P}_\mathrm{B},$$

we get

$$\frac{1}{2} \sum_{a \in \{0,1\}} \mathrm{F}\left((\alpha_a \otimes \mathrm{I}_A)^\top p,\ \beta_{\bar{a}} \right) = \frac{1}{2} [(\alpha_0 \otimes \mathrm{I}_A)^\top p]_1 + [(\alpha_1 \otimes \mathrm{I}_A)^\top p]_0 = 3/4$$

for forcing outcome 1. This proves $P^*_{\mathrm{B},0}, P^*_{\mathrm{B},1} \geq 3/4$. The dual for forcing outcome 0 is

$$\begin{array}{rrcl} \inf & z_1 & & \\ \text{subject to} & z_1 \cdot e_A & \geq & z_2^{(0)} + z_2^{(1)} \\ & \mathrm{Diag}(z_2^{(0)}) & \succeq & \frac{1}{2} \begin{bmatrix} 1/2 & 0 & 1/2 \\ 0 & 0 & 0 \\ 1/2 & 0 & 1/2 \end{bmatrix}, \\ & \mathrm{Diag}(z_2^{(1)}) & \succeq & \frac{1}{2} \begin{bmatrix} 0 & 0 & 0 \\ 0 & 1/2 & 1/2 \\ 0 & 1/2 & 1/2 \end{bmatrix}. \end{array}$$

We see that this has the feasible solution

$$(z_1, z_2^{(0)}, z_2^{(1)}) := \left(\frac{3}{4}, \begin{bmatrix} 3/4 \\ 0 \\ 3/8 \end{bmatrix}, \begin{bmatrix} 0 \\ 3/4 \\ 3/8 \end{bmatrix} \right)$$

with objective function value 3/4. By switching $z_2^{(0)} \longleftrightarrow z_2^{(1)}$, we get a feasible dual solution for Bob forcing outcome 1, with objective function value 3/4. Thus, $P^*_{\mathrm{B},0}, P^*_{\mathrm{B},1} \leq 3/4$, proving $P^*_{\mathrm{B},0}, P^*_{\mathrm{B},1} = 3/4$ as desired.

Note that the SDP proof is longer in this case (in part because we reproduced the SDPs and their duals). However, it has the advantage of working for protocols with many rounds. Proving upper bounds on cheating is a very difficult task and this is where the duality of semidefinite programming becomes a very useful tool.

3.3.2 Making a finite mesh

We can only search a finite number of protocols so we create a finite approximation of an uncountably infinite set using a mesh. We fix a dimension d and a precision $p \in (0, 1)$ and range over entries in α_0, α_1, β_0, and β_1 in increments of p keeping them as probability vectors. For example, in the four-round version, the values of $\alpha_{0,0}$ range over

$$\{0, p, 2p, \ldots, 1-p, 1\}.$$

Note we only consider $p = 1/p'$ for some positive integer p' so that we use the endpoints of the intervals. For values of $\alpha_{0,1}$, we range over

$$\{0, p, 2p, \ldots, 1-\alpha_{0,0}\},$$

and so forth. Note that by creating the mesh in this way, it is very easy to exploit the symmetry discussed in Section 3.2. We show numerically that this symmetry helps by dramatically reducing the number of protocols needed to be tested. This is important since there are $\binom{d+p'-1}{p'}^4$ protocols to test (before symmetry).

3.3.3 The refined algorithm

Using these ideas, we can further refine the algorithm.

Search algorithm for finding BCCF-protocols with small bias

For each protocol in the finite mesh (modulo the symmetry):

- Use the protocol filter to test if the protocol has bias $\varepsilon \geq 1/4$.
  If so, move on to the next protocol.
- Solve the necessary SDPs (or SOCPs).
  If any have optimal value 3/4 or larger (or 2/3 or less) move on to the next protocol.
  Else, output the protocol parameters and bias $\varepsilon < 1/4$.

Suppose for example that $P^*_{\mathrm{A},0} \leq 2/3$. Then $P^*_{\mathrm{B},0} \geq 3/4$ from Kitaev's bound so we move on to the next protocol.

We test the search algorithm for the cases of four and six-round protocols and for certain dimensions and mesh precisions in the following section.

3.4 Numerical results

Computational Platform. Programs were ran on Matlab, Version 7.12.0.635, on an SGI XE C1103 with 2x 3.2 GHz 4-core Intel X5672 x86 CPUs processor, and 10 GB memory, running Linux.

Semidefinite programs were solved using SeDuMi 1.3, a program for solving semidefinite programs in Matlab [Stu99, Stu02].

3.4.1 Four-round search

We list the filter cheating strategies in the tables below and give an estimate of how long the success probability for each strategy takes to compute (averaging over 1000 randomly generated $\alpha_0, \alpha_1, \beta_0, \beta_1$). We then give tables detailing how well the filter performs for four-round protocols for dimension $d \in \{2, \ldots, 9\}$ and for various values of precision p.

Note that for the cheating probability cut-off, we use 0.7499 instead of 0.75. This is because many cheating strategies in the filter succeed with probability exactly 0.75 and the filter would let these boundary cases through which slows down the search.

Notice the two strategies F1 and F2 are special because they only involve two of the four probability distributions. Preliminary tests show that first generating β_0 and β_1 and checking with F1 is much faster than first generating α_0 and α_1 and checking with F2, even though F2 is much faster to compute. If we were to calculate F2 first, we would have to calculate F1 for every choice of $(\alpha_0, \alpha_0, \beta_0, \beta_1)$ that F2 did not filter out, roughly squaring the number of F1 computations. For this reason, it is better to have the cheaper strategy second.

We can also justify the placement of $P^*_{\mathrm{A},0}$ before $P^*_{\mathrm{B},0}$ or $P^*_{\mathrm{B},1}$. The strategies F8 and F9 performed very well in preliminary tests and the cheating probabilities are very close to $P^*_{\mathrm{B},0}$ and $P^*_{\mathrm{B},1}$. Thus, if a protocol gets through the F8 and F9 filter strategies, then it is likely that $P^*_{\mathrm{B},0}$ and $P^*_{\mathrm{B},1}$ will also be less than 0.7499. This is why we place $P^*_{\mathrm{A},0}$ first. Note that we solve for $P^*_{\mathrm{B},0}$, $P^*_{\mathrm{B},1}$, $P^*_{\mathrm{A},0}$, and $P^*_{\mathrm{A},1}$ using the SDP formulation of the reduced problems. Preliminary tests show that SeDuMi ran into numerical issues while solving the SOCPs. Fortunately, the filter works so well that few SDPs are needed to be solved.

Table 3.1: Running times for filter strategies for a four-round protocol when $d = 5$ (1 of 2)

Success Probability	Comp. Time (s)	Code
$\frac{1}{2} + \frac{1}{2}\sqrt{\mathrm{F}(\beta_0, \beta_1)}$	0.000034429	F1
$\frac{1}{2} + \frac{1}{2}\Delta(\alpha_0, \alpha_1)$	0.000004640	F2
$\left(\frac{1}{2} + \frac{1}{2}\sqrt{\mathrm{F}(\alpha_0, \alpha_1)}\right)\left(\frac{1}{2} + \frac{1}{2}\Delta(\beta_0, \beta_1)\right)$	0.000025980	F3
$\frac{1}{2}\sum_{a\in\{0,1\}} \mathrm{F}(\alpha_a, \beta_a)$	0.000023767	F4
$\frac{1}{2}\sum_{a\in\{0,1\}} \mathrm{F}(\alpha_a, \beta_{\bar{a}})$	0.000018019	F5
$\frac{1}{2}\lambda_{\max}\left(\left(\sum_{y:\beta_{0,y}\geq\beta_{1,y}} \beta_{0,y}\right)\sqrt{\alpha_0}\sqrt{\alpha_0}^\top + \left(\sum_{y:\beta_{0,y}<\beta_{1,y}} \beta_{1,y}\right)\sqrt{\alpha_1}\sqrt{\alpha_1}^\top\right)$ ($\sqrt{v}$ is the principal eigenvector)	0.000036613	F6
$\frac{1}{2}\sum_{x\in A} \lambda_{\max}\left(\sum_{a\in\{0,1\}} \alpha_{a,x}\sqrt{\beta_a}\sqrt{\beta_a}^\top\right)$ ($\sqrt{v_x}$ is the principal eigenvector for each $x \in A$)	0.000073010	F7
$\frac{1}{2}\sum_{a\in\{0,1\}} \mathrm{F}\left(\sum_{x\in A} \alpha_{a,x}(v_x), \beta_a\right)$	0.000697611	F8
$\frac{1}{2}\sum_{a\in\{0,1\}} \mathrm{F}\left(\sum_{x\in A} \alpha_{a,x}(v_x), \beta_{\bar{a}}\right)$	0.000532954	F9

Table 3.2: Running times for filter strategies for a four-round protocol when $d = 5$ (2 of 2)

Success Probability	Comp. Time (s)	Code
$\sum_{y \in B} \operatorname{conc}\left\{\frac{1}{2}\beta_{0,y}\mathrm{F}(\cdot, \alpha_0), \frac{1}{2}\beta_{1,y}\mathrm{F}(\cdot, \alpha_1)\right\}(v)$	0.122971205	F10
$\sum_{y \in B} \operatorname{conc}\left\{\frac{1}{2}\beta_{1,y}\mathrm{F}(\cdot, \alpha_0), \frac{1}{2}\beta_{0,y}\mathrm{F}(\cdot, \alpha_1)\right\}(v)$	0.123375678	F11
$P^*_{\mathrm{A},0}$	0.149814373	SDPA0
$\dfrac{1}{2P^*_{\mathrm{A},0}}$	0.000000947	F12
$P^*_{\mathrm{B},0}$	0.070846378	SDPB0
$P^*_{\mathrm{A},1}$	0.149176117	SDPA1
$\dfrac{1}{2P^*_{\mathrm{A},1}}$	0.000000760	F13
$P^*_{\mathrm{B},1}$	0.070479449	SDPB1

Table 3.3: The number of four-round protocols that get through each strategy in the filter for $d = 2$

d = 2	p = 1/500	p = 1/1000	p = 1/1250	p = 1/1500	p = 1/2000
Protocols	6.30 e+10	1.00 e+12	2.44 e+12	5.07 e+12	1.60 e+13
Symmetry	3,969,126,001	63,001,502,001	153,566,799,376	318,097,128,001	1,004,006,004,001
F1	96,706,535	1,499,479,974	3,636,609,280	7,506,289,309	23,607,143,560
F2	72,336,875	1,123,112,000	2,724,552,320	5,624,716,125	17,693,560,000
F3	5	27	50	67	124
F4	0	0	0	0	0
F5	0	0	0	0	0
F6	0	0	0	0	0
F7	0	0	0	0	0
F8	0	0	0	0	0

Table 3.4: The number of four-round protocols that get through each strategy in the filter for $d = 3$

d = 3	p = 1/5	p = 1/10	p = 1/20	p = 1/30	p = 1/50
Protocols	1.94e+05	1.89e+07	2.84e+09	6.05e+10	3.09e+12
Symmetry	4,356	272,484	29,430,625	55,436,7025	25,475,990,544
F1	1,254	37,584	2,175,425	30,985,220	1,020,080,292
F2	665	19,656	1,300,042	19,366,256	662,158,728
F3	49	470	22,282	225,098	4,414,994
F4	29	261	11,667	110,931	2,028,518
F5	28	258	11,495	109,515	2,009,141
F6	28	241	10,405	96,464	1,765,114
F7	0	3	54	148	1,158
F8	0	0	0	0	0

Table 3.5: The number of four-round protocols that get through each strategy in the filter for $d = 4$

d = 4	p = 1/10	p = 1/12	p = 1/16	p = 1/20	1/24	1/30
Protocols	6.69 e+09	4.28e+10	8.81e+11	9.83 e+12	7.31 e+13	8.86 e+14
Symmetry	13,498,276	74,166,544	1,154,640,400	10,334,552,281	69,927,455,844	736,486,643,344
F1	2,432,188	12,616,580	146,114,000	934,856,164	5,916,006,936	49,798,933,264
F2	1,036,030	5,616,810	71,246,700	489,282,376	3,170,626,956	27,760,130,976
F3	66,623	302,547	3,185,895	19,670,642	101,703,667	738,284,522
F4	46,734	209,747	2,061,868	12,000,187	59,503,895	406,963,112
F5	46,531	208,961	2,054,891	11,962,104	59,353,374	406,099,637
F6	42,591	198,192	1,886,782	11,004,125	54,702,075	367,847,304
F7	329	756	3,439	17,144	55,929	190,699
F8	0	0	0	0	0	0

Table 3.6: The number of four-round protocols that get through each strategy in the filter for $d = 5$

d = 5	p = 1/5	p = 1/8	p = 1/10	p = 1/12
Protocols	2.52 e+08	6.00 e+10	1.00 e+12	1.09 e+13
Symmetry	240,100	29,539,225	284,529,424	2,485,919,881
F1	105,840	9,467,770	66,257,504	567,544,997
F2	37,584	2,687,906	22,774,544	203,983,360
F3	8,561	241,420	2,440,765	17,794,655
F4	7,423	201,569	1,937,298	13,682,059
F5	7,417	200,965	1,933,833	13,665,087
F6	7,417	189,144	1,790,144	13,117,165
F7	0	1415	10,790	43,459
F8	0	0	0	0

Table 3.7: The number of four-round protocols that get through each strategy in the filter for $d = 6$

d = 6	p = 1/7	p = 1/8	p = 1/9	p=1/10	p = 1/11	p = 1/12
Protocols	3.93e+11	2.74e+12	1.60e+13	8.13e+13	3.64e+14	1.46e+15
Symmetry	53,144,100	265,950,864	1,021,825,156	3,534,302,500	12,577,398,201	46,107,255,076
F1	25,070,310	107,583,876	387,459,886	1,034,786,700	3,605,814,648	13,370,558,568
F2	7,276,924	23,294,007	123,246,328	287,251,218	1,330,224,696	3,841,063,848
F3	1,744,038	2,811,374	25,114,451	42,503,208	258,455,916	468,218,324
F4	1,551,522	2,526,900	21,682,087	36,628,517	214,823,642	390,846,158
F5	1,550,617	2,524,052	21,666,437	36,594,682	214,698,072	390,649,931
F6	1,451,038	2,419,474	20,598,749	34,117,986	203,605,433	377,899,946
F7	9,169	13,976	57,720	174,118	526,077	1,153,864
F8	0	0	0	0	0	0

Table 3.8: The number of four-round protocols that get through each strategy in the filter for $d = 7$

d = 7	p = 1/5	p=1/6	p = 1/7	p = 1/8	p = 1/9	p = 1/10
Protocols	4.55e+10	7.28e+11	8.67e+12	8.13e+13	6.27e+14	4.11e+15
Symmetry	3,709,476	46,963,609	289,374,121	1,730,643,201	7,402,021,225	30,490,398,225
F1	2,270,754	26,952,849	161,111,181	841,297,023	3,456,456,125	10,915,707,495
F2	495,180	3,154,266	36,330,756	136,788,372	851,509,125	2,419,940,743
F3	149,806	369,434	10,277,699	20,469,535	216,148,269	449,464,967
F4	142,255	351,290	9,583,747	19,200,670	197,250,330	409,366,494
F5	142,241	351,219	9,582,215	19,194,692	197,214,454	409,185,885
F6	142,241	351,219	9,034,728	18,734,072	187,977,589	383,402,064
F7	0	0	60,155	91,787	512,171	1,804,382
F8	0	0	0	0	0	0

Table 3.9: The number of four-round protocols that get through each strategy in the filter for $d = 8$

d = 8	p = 1/4	p = 1/5	p = 1/6	p = 1/7	p = 1/8	p = 1/9
Protocols	1.18e+10	3.93e+11	8.67e+12	1.38e+14	1.71e+15	1.71e+16
Symmetry	1,572,516	11,532,816	179,345,664	1,293,697,024	9,018,161,296	42,352,405,209
F1	1,054,614	7,797,216	115,131,024	814,855,040	5,050,850,268	23,061,817,617
F2	60,552	1,356,936	9,766,192	142,862,430	606,597,735	4,417,668,742
F3	0	431,956	1,254,420	44,457,239	106,851,420	1,276,499,496
F4	0	417,759	1,213,728	42,541,702	102,719,851	1,204,238,273
F5	0	417,741	1,213,629	42,539,430	102,710,139	1,204,173,244
F6	0	417,741	1,213,629	40,425,272	101,061,706	1,151,097,965
F7	0	0	0	277,225	452,792	3,194,346
F8	0	0	0	0	0	0

Table 3.10: The number of four-round protocols that get through each strategy in the filter for $d = 9$

d = 9	p = 1/3	p = 1/4	p = 1/5	p = 1/6	p = 1/7	p = 1/8
Protocols	7.41e+08	6.00e+10	2.74e+12	8.13e+13	1.71e+15	2.74e+16
Symmetry	164,025	3,744,225	32,069,569	594,433,161	4,957,145,649	39,808,629,441
F1	131,625	2,666,430	23,348,549	414,160,047	3,423,681,189	24,851,338,155
F2	14,300	115,752	3,273,662	26,075,045	470,028,582	2,216,082,560
F3	2,700	0	1,065,271	3,484,092	153,932,946	432,754,976
F4	2,639	0	1,041,339	3,405,532	149,523,487	421,903,500
F5	2,639	0	1,041,317	3,405,403	149,520,361	421,889,260
F6	2,639	0	1,041,317	3,405,403	142,916,565	416,869,327
F7	0	0	0	0	1,053,222	1,809,800
F8	0	0	0	0	0	0

Four-round search observations

The first observation is that we were able to search larger spaces than if we used the SDP formulations alone. For example, suppose we took the 2.74×10^{16} protocols from the $d = 9$, $p = 1/8$ search and checked to see if any of these had bias less than 0.7499 by solving only the reduced SDPs. Since each SDP takes at least 0.08 seconds to solve, this search would take at least 69 million years to finish. By applying the techniques in this chapter, we are able to run this search in a matter of days.

We see that symmetry helped to dramatically reduce the number of protocols needed to be tested. In the largest search, we were able to cut down the 2.74×10^{16} protocols down to 3.98×10^{10}.

F1 and F2 perform very well, together cutting down the number of protocols by a factor of about 10. F3, being the first strategy to rely on all four probability vectors, performs very well by reducing the number of protocols by another factor of 10. F4, F5, and F6 do not perform well with F5, being the same as F4 but with β_0 swapped with β_1, cutting down the number of protocols by a very small amount. F7 and F8 perform so well that no SDPs were needed to be solved. This suggests that

$$\min_{\alpha_0,\alpha_1,\beta_0,\beta_1 \in \text{Prob}^9} \max \{\text{F1}, \ldots, \text{F8}\} = 3/4.$$

From Theorem 3.1.9, we see that Bob's eigenstrategy, F8, has the highest cheating probability among Bob's four-round strategies with the exception of possibly the returning strategies F4 and F5. Alice's eigenstrategy, F6, performs better than her three-round strategy F3. Therefore, it seems that the F8 and F6 strategies suffice to prove that the bias of any four-round BCCF-protocol is at least 1/4.

3.4.2 Six-round search

We list the filter cheating strategies in the table below and give an estimate of how long the success probability for each strategy takes to compute (averaging over 1000 randomly generated $\alpha_0, \alpha_1, \beta_0, \beta_1$). We then give tables of how well the filter performs for six-round protocols with $d \in \{2, 3\}$ and p ranging up to 1/14 (depending on d).

Table 3.11: Running times for filter strategies in a six-round protocol for $d = 3$ (1 of 2)

Success Probability	Comp. Time (s)	Code
$\frac{1}{2} + \frac{1}{2}\sqrt{\mathrm{F}(\beta_0, \beta_1)}$	0.000029071	G1
$\frac{1}{2} + \frac{1}{2}\Delta(\mathrm{Tr}_{A_2}(\alpha_0), \mathrm{Tr}_{A_2}(\alpha_1))$	0.000003983	G2
$\frac{1}{2}\lambda_{\max}\left(\kappa\sqrt{\mathrm{Tr}_{A_2}(\beta_0)}\sqrt{\mathrm{Tr}_{A_2}(\beta_0)}^{\top} + \omega\sqrt{\mathrm{Tr}_{A_2}(\beta_1)}\sqrt{\mathrm{Tr}_{A_2}(\beta_1)}^{\top}\right)$ where $\kappa := \sum_{x:\alpha_{0,x} \geq \alpha_{1,x}} \alpha_{0,x}$ and $\omega := \sum_{x:\alpha_{0,x} < \alpha_{1,x}} \alpha_{1,x}$	0.000013010	G3
$\left(\frac{1}{2} + \frac{1}{2}\sqrt{\mathrm{F}(\mathrm{Tr}_{A_2}(\alpha_0), \mathrm{Tr}_{A_2}(\alpha_1))}\right)\left(\frac{1}{2} + \frac{1}{2}\Delta(\mathrm{Tr}_{B_2}(\beta_0), \mathrm{Tr}_{B_2}(\beta_1))\right)$	0.000022501	G4
$\frac{1}{2}\lambda_{\max}\left(\left(\sum_{y:\beta_{0,y} \geq \beta_{1,y}} \beta_{0,y}\right)\sqrt{\alpha_0}\sqrt{\alpha_0}^{\top} + \left(\sum_{y:\beta_{0,y} < \beta_{1,y}} \beta_{1,y}\right)\sqrt{\alpha_1}\sqrt{\alpha_1}^{\top}\right)$	0.000040928	G5
$\frac{1}{2}\lambda_{\max}\left(\eta'\sqrt{\mathrm{Tr}_{A_2}(\alpha_0)}\sqrt{\mathrm{Tr}_{A_2}(\alpha_0)}^{\top} + \tau'\sqrt{\mathrm{Tr}_{A_2}(\alpha_1)}\sqrt{\mathrm{Tr}_{A_2}(\alpha_1)}^{\top}\right)$ where $\eta' := \sum_{y_1 \in B_1:[\mathrm{Tr}_{B_2}(\beta_0)]_{y_1} \geq [\mathrm{Tr}_{B_2}(\beta_1)]_{y_1}} [\mathrm{Tr}_{B_2}(\beta_0)]_{y_1}$, and $\tau' := \sum_{y_1 \in B_1:[\mathrm{Tr}_{B_2}(\beta_0)]_{y_1} < [\mathrm{Tr}_{B_2}(\beta_1)]_{y_1}} [\mathrm{Tr}_{B_2}(\beta_1)]_{y_1}$	0.000036908	G6

Table 3.12: Running times for filter strategies in a six-round protocol for $d = 3$ (2 of 2)

Success Probability	Comp. Time (s)	Code
$\frac{1}{2}\sum_{a\in A_0'} \mathrm{F}\left(\sum_{x\in A} \alpha_{a,x}\tilde{p_2}^{(x)}, \beta_a\right)$	0.002548535	G7
$\frac{1}{2}\sum_{a\in A_0'} \mathrm{F}\left(\sum_{x\in A} \alpha_{a,x}\tilde{p_2}^{(x)}, \beta_{\bar{a}}\right)$ where $\tilde{p_2}^{(x)}$ is as defined in Theorem 3.1.9	0.002363596	G8
$\sum_{y\in B} \operatorname{conc}\left\{\frac{1}{2}\beta_{0,y}\mathrm{F}(\cdot,\alpha_0), \frac{1}{2}\beta_{1,y}\mathrm{F}(\cdot,\alpha_1)\right\}(v)$	0.232172156	G9
$\sum_{y\in B} \operatorname{conc}\left\{\frac{1}{2}\beta_{1,y}\mathrm{F}(\cdot,\alpha_0), \frac{1}{2}\beta_{0,y}\mathrm{F}(\cdot,\alpha_1)\right\}(v)$	0.229131167	G10
$P^*_{\mathrm{B},0}$	0.101631766	SDPB0
$\dfrac{1}{2P^*_{\mathrm{B},0}}$	0.000000655	G11
$P^*_{\mathrm{A},0}$	0.250714766	SDPA0
$P^*_{\mathrm{B},1}$	0.101428438	SDPB1
$\dfrac{1}{2P^*_{\mathrm{B},1}}$	0.000000710	G12
$P^*_{\mathrm{A},1}$	0.245836334	SDPA1

Again, we have to choose which strategy to put first, G1 or G2. Preliminary tests show that placing G1 first is much faster, similar to the four-round case. Even though G5 takes longer to compute than G6, tests show that is better to have G5 first. We calculate SDPB0 before SDPA0 since G9 and G10 are close approximations of SDPA0 and SDPA1, respectively.

We note here a few omissions that are present in the four-round tests. First, we have removed the two returning strategies, F4 and F5. These did not perform well in the four-round tests and preliminary tests show that they did not perform well in the six-round tests either. Also, we do not have all the lower bounds for the eigenstrategies. Preliminary tests show that the lower bounds omitted take just as long or longer to compute than the corresponding upper bound, thus we just use the upper bounds in the filter. Also, the marginal probabilities take approximately 5.49×10^{-6} seconds to compute which is negligible compared to the other times. Thus, we need not be concerned whether the strategies rely on the full probability vectors or their marginals.

Table 3.13: The number of six-round protocols that get through each strategy in the filter for $d = 2$

d = 2	p = 1/3	p = 1/4	p = 1/5	p = 1/6	p = 1/7	p = 1/8
Protocols	160,000	1,500,625	9,834,496	49,787,136	207,360,000	7.41e+08
Symmetry	6,400	59,049	280,900	1,517,824	5,683,456	19,713,600
G1	3,200	20,412	82,680	389,312	1,397,024	4,115,880
G2	2,320	12,516	67,548	272,392	1,112,228	3,057,246
G3	1,725	9,627	52,424	223,034	899,450	2,526,712
G4	714	4,206	27,965	105,050	430,454	1,240,106
G5	210	684	7,743	20,373	112,435	228,274
G6	210	684	7,743	20,373	110,401	228,274
G7	30	48	1,285	1,856	10,979	17,831
G8	0	0	226	0	1,907	1,320
G9	0	0	226	0	1,901	1,212
G10	0	0	226	0	1,901	1,212
SDPB0	0	0	0	0	0	0

Table 3.14: The number of six-round protocols that get through each strategy in the filter for $d = 2$

d = 2	p = 1/9	p = 1/10	p = 1/11	p = 1/12	p = 1/13	p = 1/14
Protocols	2.34e+09	6.69e+09	1.75e+10	4.28e+10	9.83e+10	2.13e+11
Symmetry	58,247,424	155,276,521	401,080,729	973,502,401	2,052,180,601	4,632,163,600
G1	11,020,608	23,862,815	60,761,918	140,154,892	240,820,116	555,641,840
G2	8,944,136	18,717,210	50,337,094	110,274,108	204,522,468	444,537,964
G3	7,335,617	15,503,308	41,447,668	93,222,286	167,717,637	380,238,435
G4	3,477,093	8,534,326	20,503,550	45,888,192	91,991,055	185,971,770
G5	696,601	1,367,115	3,435,390	6,577,917	12,425,039	23,210,979
G6	688,613	1,367,115	3,435,390	6,577,917	12,258,117	23,097,713
G7	57,598	87,303	232,382	355,057	678,384	1,051,339
G8	9,116	9,699	31,812	45,625	85,333	130,765
G9	7,669	7,375	19,874	35,204	56,267	103,598
G10	7,657	7,166	19,592	34,064	54,398	99,126
SDPB0	0	0	8	4	64	20
G11	0	0	8	4	64	20
SDPA0	0	0	0	0	0	0

Table 3.15: The number of six-round protocols that get through each strategy in the filter for $d = 3$

d = 3	p = 1/1	p = 1/2	p = 1/3	p = 1/4
Protocols	6,561	4,100,625	741,200,625	60,037,250,625
Symmetry	81	68,121	6,395,841	279,324,369
G1	72	42,282	5,222,385	180,500,400
G2	24	8,748	3,324,650	86,151,600
G3	20	5,643	1,958,070	58,038,667
G4	2	161	714,393	30,773,918
G5	0	0	464,538	15,310,116
G6	0	0	464,538	15,310,116
G7	0	0	310,518	6,557,007
G8	0	0	26,460	0
G9	0	0	26,460	0
G10	0	0	26,460	0
SDPB0	0	0	0	0

Six-round observations

We find again that the symmetry arguments cut down the number of protocols significantly, this time by a factor of roughly 100. Note that in the four-round case it was a factor of $10,000$ (for the $d = 9$ case). This can be explained by the weaker index symmetry in the six-round version. G1 and G2 cut down the number of protocols by a factor of 10 similar to the four-round case. G5 also performed well, but after this G6 was not much help. G7 and G8 cut down the number of protocols by a factor of 10 each in the $d = 2$ case, but not as much in the $d = 3$ case. The next notable strategy was G10, being the swapped version of G9, which performed very poorly. It seems that the swapped strategies do not help much in the filters, that is, there is not much discrepancy between cheating towards 0 or 1. SDPB0 almost filtered out the rest of the protocols, relying on SDPA0 to stop the rest. The implicit strategy from Kitaev's bound, G11, did not perform well after SDPB0, unfortunately we need SDPB0 to be computed first. Again, we notice that no protocols with bias less than 0.7499 were found.

Another observation is the performance of G9 and G10, the eigenstrategies for Alice. For low-precision tests, we notice that they hardly filter out any protocols, if any at all. In these strategies, we compute a value on the concave hull

$$\operatorname{conc}\left\{\frac{1}{2}\beta_{0,y}\,\mathrm{F}(\cdot,\alpha_0),\frac{1}{2}\beta_{1,y}\,\mathrm{F}(\cdot,\alpha_1)\right\},$$

for every value of y. In the eigenstrategy, we are approximating the concave hull with whichever of the two has the larger constant. When we choose these constants according to a rough mesh, e.g., $p = 1/3$ or $p = 1/4$, the one with the larger constant is a very good approximation of the concave hull. Thus, we need finer precisions to bring out the power of this strategy in the filter.

Since in the four and six-round tests, we did not find any protocols with bias less than 0.7499, we have the following conjecture.

Conjecture 3.4.1. *The minimum bias of* BCCF*-protocols is* $\varepsilon = 1/4$.

3.4.3 Random offset

Since the six-round searches take a long time, we need another way to test more protocols. We could increase the mesh precision, but this increases the search time dramatically. To test more protocols, we offset all of the values in the search by some random factor $\delta > 0$. For example, the entries of α_0, α_1, β_0, and β_1 have been selected from the set

$$\{0, p, 2p, \ldots, 1-p, 1\}.$$

With an offset parameter $\delta \in (0, p/2)$, we use the range

$$\{\delta, \delta+p, \delta+2p, \ldots, \delta+1-p\}.$$

Note that this destroys the index symmetry. The simplest way to see this is to consider the two-dimensional probability distributions created in this way. They are

$$\left\{ \begin{bmatrix} \delta \\ 1-\delta \end{bmatrix}, \begin{bmatrix} \delta+p \\ 1-\delta-p \end{bmatrix}, \begin{bmatrix} \delta+2p \\ 1-\delta-2p \end{bmatrix}, \ldots, \begin{bmatrix} \delta+1-p \\ p-\delta \end{bmatrix} \right\}.$$

We see that the set of first entries is not the same as the set of second entries when $\delta > 0$. We arbitrarily choose the last entry in each vector to be such that the entries add to 1. Also, since we are generating all four of the probability distributions in the same manner, we can still apply the symmetry arguments to suppose α_0 has the largest entry out of both α_0 and α_1 and simultaneously β_0 has the largest entry out of both β_0 and β_1.

The tables below show how well each strategy in the filter performs in the worst case and average case over 100 random choices of offset parameter $\delta \in [0, 1/100]$.

Table 3.16: The percentage of protocols that get stopped by each strategy in the worst case over 100 random instances of offset parameter δ

d = 2	p = 1/3	p = 1/4	p = 1/5	p = 1/6
G1	71.87 %	82.35 %	84.06 %	86.63 %
G2	17.18 %	29.80 %	15.80 %	24.15 %
G3	8.17 %	10.73 %	13.46 %	12.12 %
G4	51.45 %	49.68 %	53.99 %	48.44 %
G5	70.00 %	83.29 %	78.02 %	82.96 %
G6	0 %	0 %	0 %	0 %
G7	79.16 %	92.43 %	87.46 %	94.39 %
G8	100.00 %	100.00 %	43.83 %	100.00 %
G9	–	–	0 %	–
G10	–	–	0 %	–
SDPB0	–	–	100.00 %	–

Table 3.17: The percentage of protocols that get stopped by each strategy in the average case over 100 random instances of offset parameter δ

d = 2	p = 1/3	p = 1/4	p = 1/5	p = 1/6
G1	85.51 %	86.65 %	88.82 %	90.69 %
G2	17.18 %	29.80 %	15.80 %	24.15 %
G3	10.95 %	13.50 %	14.56 %	12.36 %
G4	62.14 %	51.73 %	55.39 %	53.25 %
G5	70.00 %	86.83 %	92.23 %	94.12 %
G6	0 %	0 %	0 %	0 %
G7	99.04 %	98.91 %	95.75 %	99.13 %

Random offset observations

We see that G6 performs very poorly on these tests. We need a finer mesh to see the effects of G6 in the filter. Also, G1 performs generally better as the filter precision increases and, as the previous tables suggest, it should stay at roughly 90%. We see that G5 and G7 perform very well. G8 performs well most of the time, except in the worst case the percentage was quite low in the $p = 1/5$ column. No protocols with bias $\varepsilon < 0.7499$ were found, with only SDPB0 needed to be solved of the four SDPs. This supports the conjecture that $\varepsilon = 1/4$ is the smallest attainable bias for BCCF-protocols.

Chapter 4

A protocol and lower bounds for quantum oblivious transfer

The content of this chapter is based on my work with André Chailloux and Iordanis Kerenidis in [CKS10].

In the last two chapters, we have been examining coin-flipping protocols based on bit-commitment. As we have seen, bit-commitment and coin-flipping are very closely related primitives. It turns out that some of the knowledge about bit-commitment can be applied to oblivious transfer as discussed in this chapter. This chapter is organized as follows.

- Section 4.1: We show a two-message protocol for oblivious transfer which has bias $\varepsilon_{\text{OT}} = 1/4$. This protocol is similar to the bit-commitment protocol in [KN04] (which was used for strong coin-flipping). We notice that this maintains the same bias as the bit-commitment protocol, however this is a coincidence since there is no known way to reduce bit-commitment to oblivious transfer in a way that preserves the bias.
- Section 4.2: We study a reduction from oblivious transfer to bit-commitment and show how the cheating probabilities are related to each other. We then present a Learning-In-Sequence Lemma which roughly states: If there are is a projective measurement to learn the value of x_0 with probability p from an encoding ρ_{x_0,x_1} and another projective measurement to learn the value of x_1 with probability q, then

there is a measurement to learn both x_0 and x_1 with probability $a(2a-1)^2$, where $a := \frac{p+q}{2}$. Using this lemma and lower bounds for bit-commitment [CK11], we obtain the lower bound on the bias $\varepsilon_{\mathrm{OT}} \geq 0.0852$ for any oblivious transfer protocol.

- Section 4.3: We prove the Learning-In-Sequence Lemma.
- Section 4.4: We conclude with studying forcing oblivious transfer which is a variant of oblivious transfer and a generalization of coin-flipping. Using SDP formulations of cheating strategies, we prove optimal lower bounds on the bias. This analysis illustrates the variety of security notions in quantum cryptography and relates oblivious transfer with the analysis in earlier parts of the thesis.

4.1 An oblivious transfer protocol with bias 1/4

The protocol starts with the same starting states as the bit-commitment protocol in [KN04]. This helps Bob conceal the value of b from Alice. In addition, both these states have the property that under the action of changing a local phase, the image and preimage become orthogonal. We use this property to encode Alice's message into a local phase. The protocol is given below.

Protocol 4.1.1 (An OT protocol with bias 1/4)**.**

(i) *Bob randomly chooses $b \in \{0,1\}$ and sends Alice the $\mathbb{C}^A$ part of the following two-qutrit state*

$$|\phi_b\rangle := \frac{1}{\sqrt{2}}|bb\rangle + \frac{1}{\sqrt{2}}|22\rangle \in \mathbb{C}^{A \times B},$$

where $A = B = \{0,1,2\}$.

(ii) *Alice randomly chooses $x_0, x_1 \in \{0,1\}$ and applies the following unitary U_{x_0,x_1} to $\mathbb{C}^A$:*

$$|0\rangle \to (-1)^{x_0}|0\rangle, \quad |1\rangle \to (-1)^{x_1}|1\rangle, \quad |2\rangle \to |2\rangle.$$

(iii) *Alice returns $\mathbb{C}^A$ to Bob. Bob now has the two-qutrit state*

$$\frac{(-1)^{x_b}}{\sqrt{2}}|bb\rangle + \frac{1}{\sqrt{2}}|22\rangle \in \mathbb{C}^{A \times B}.$$

(iv) *Bob performs the measurement* $\{\Pi_0 := |\phi_b\rangle\langle\phi_b|,\ \Pi_1 := \mathrm{I} - \Pi_0\}$ *on the state.*

(v) *If the outcome is "0" then* $x_b = 0$*. If the outcome is "1" then* $x_b = 1$*.*

Note that if he sends half of the state $\frac{1}{\sqrt{2}}|00\rangle + \frac{1}{\sqrt{2}}|11\rangle$ in the first message, then he can learn $x_0 \oplus x_1$ perfectly (although he does not learn either x_0 or x_1). Thus, this protocol is not secure if we want to stop Bob from learning $x_0 \oplus x_1$.

We now show that it is impossible for Bob to perfectly learn both x_0 and x_1 and also that his bit is not completely revealed to a cheating Alice.

Theorem 4.1.2. *In the protocol described above, we have* $P^*_{\mathrm{A,OT}} = P^*_{\mathrm{B,OT}} = 3/4$*.*

Proof. We analyze the cheating probabilities of Alice and Bob.

Cheating Alice: Let $\sigma_b := \mathrm{Tr}_B|\phi_b\rangle\langle\phi_b|$ denote the reduced states Alice may receive in the first message. An optimal strategy for Alice to learn b is to perform the optimal measurement to distinguish between σ_0 and σ_1. In this case, she succeeds with probability

$$\frac{1}{2} + \frac{1}{2}\Delta(\sigma_0, \sigma_1) = \frac{3}{4}.$$

Alice's optimal measurement is, in fact, a measurement in the computational basis. If she gets outcome "0" or "1" then she knows b with certainty. If she gets outcome "2" then she randomly guesses. Note that Bob never aborts.

Cheating Bob: Bob wants to learn both bits (x_0, x_1). We now describe a general strategy for Bob:

- Bob creates $|\psi\rangle := \sum_{i \in A} \alpha_i |i\rangle_A |\xi_i\rangle_{B'} \in \mathbb{C}^{A \times B'}$, where $\mathbb{C}^{B'}$ is a space used by Bob to cheat. He sends the $\mathbb{C}^A$ part to Alice. The $|\xi_i\rangle$ states need not be orthogonal but $\sum_{i \in A} |\alpha_i|^2 = 1$.
- Alice applies U_{x_0,x_1} on $\mathbb{C}^A$ and sends it back to Bob. He now has the state

  $$|\psi_{x_0,x_1}\rangle := \sum_{i \in A} \alpha_i (-1)^{x_i} |i\rangle|\xi_i\rangle,$$

  where $x_2 := 0$.

At the end of the protocol, Bob applies a four-outcome measurement on $|\psi_{x_0,x_1}\rangle$ to obtain his guess for (x_0, x_1).

From this strategy, we create another strategy with the same cheating probability where Bob sends a pure state. We define this strategy as follows:

- Bob creates $|\psi'\rangle := \sum_{i \in A} \alpha_i |i\rangle_A \in \mathbb{C}^A$ and sends the whole state to Alice.
- Alice applies U_{x_0,x_1} on $\mathbb{C}^A$ and sends it back to Bob. He now has the state
$$|\psi'_{x_0,x_1}\rangle := \sum_{i \in A} \alpha_i (-1)^{x_i} |i\rangle \in \mathbb{C}^A.$$
- Bob applies the unitary $U : |i\rangle|0\rangle \to |i\rangle|\xi_i\rangle$ to $|\psi'_{x_0,x_1}\rangle|0\rangle$ and obtains $|\psi_{x_0,x_1}\rangle$.

To determine (x_0, x_1), Bob applies the same measurement as in the original strategy.

Clearly both strategies have the same success probability. When Bob uses the second strategy, Alice and Bob are unentangled after the first message and Alice sends back a qutrit to Bob. Since Bob has an encoding of two randomly chosen bits in a qutrit state, we have
$$\Pr[\text{Bob correctly guesses } (x_0, x_1)] \leq 3/4$$
from information bounds (see [Nay99]).

Note that there is a strategy for Bob to learn both bits (x_0, x_1) with probability $3/4$. Suppose he creates the state
$$|\psi\rangle := \frac{1}{\sqrt{3}}|0\rangle + \frac{1}{\sqrt{3}}|1\rangle + \frac{1}{\sqrt{3}}|2\rangle$$
and sends it to Alice. The state Bob has after Alice's message is
$$\frac{1}{\sqrt{3}}(-1)^{x_0}|0\rangle + \frac{1}{\sqrt{3}}(-1)^{x_1}|1\rangle + \frac{1}{\sqrt{3}}|2\rangle.$$
Then, Bob performs a projective measurement in the four-dimensional basis
$$\{|\Psi_{x_0,x_1}\rangle : x_0, x_1 \in \{0,1\}\},$$
where
$$|\Psi_{x_0,x_1}\rangle := \frac{1}{2}(-1)^{x_0}|0\rangle + \frac{1}{2}(-1)^{x_1}|1\rangle + \frac{1}{2}|2\rangle + \frac{1}{2}(-1)^{x_0 \oplus x_1}|3\rangle.$$

The probability that Bob guesses the two bits x_0 and x_1 correctly is

$$\sum_{x_0,x_1} \frac{1}{4} \Pr[\text{Bob guesses } (x_0, x_1)] = \sum_{x_0,x_1} \frac{1}{4} |\langle \Psi_{x_0,x_1} | \psi_{x_0,x_1} \rangle|^2 = \frac{3}{4}.$$

Note that Alice never aborts. □

4.2 A lower bound on any oblivious transfer protocol

In this section, we prove that the bias of any OT protocol is bounded below by a constant. We start from an OT protocol and show how to construct a bit-commitment protocol. Then, we prove a relationship between the cheating probabilities of the bit-commitment protocol and those for OT. Lastly, we use the lower bound for quantum bit-commitment [CK11] (reproduced as Proposition 4.2.4) to derive a lower bound on the bias of any OT protocol.

We create a bit-commitment protocol from an OT protocol as follows.

Protocol 4.2.1 (Bit-commitment protocol via OT)**.**

(i) *Commit phase: We invert the roles of Alice and Bob so that Bob is the one who commits. He wants to commit to a bit a. Alice and Bob perform the* OT *protocol such that Alice has (x_0, x_1) and Bob has (b, x_b). Bob sends $c := a \oplus b$ to Alice.*

(ii) *Reveal phase: Bob reveals (b, x_b) to Alice. If (b, x_b) from Bob is consistent with Alice's bits, then Alice accepts $c \oplus b = a$. Otherwise, Alice aborts.*

Note that the OT outputs could be generated from within the OT protocol for this construction and the following arguments to work.

We now analyze how much Alice and Bob can cheat in the bit-commitment protocol and compare these quantities to those of OT. Our goal is to show the following proposition.

Proposition 4.2.2. *For the protocol above, we have*

$$P^*_{\text{A,OT}} = P^*_{\text{A,BC}} \quad \textit{and} \quad P^*_{\text{B,OT}} \geq f(P^*_{\text{B,BC}}) \ \ \textit{where} \ \ f(x) = x(2x-1)^2.$$

Proof. Let $\neg\bot_A^{\text{BC}}$ (resp. $\neg\bot_B^{\text{BC}}$) denote the event "Alice (resp. Bob) does not abort during the entire bit-commitment protocol." Let $\neg\bot_A^{\text{OT}}$ (resp. $\neg\bot_B^{\text{OT}}$) denote the event "Alice (resp. Bob) does not abort during the OT subroutine."

Cheating Alice: Since Alice knows $c := a \oplus b$, the probability of Alice guessing a in the bit-commitment protocol is the same as the probability of her guessing b in the OT protocol. Therefore $P^*_{\text{A,OT}} = P^*_{\text{A,BC}}$.

Cheating Bob: By definition, we have

$$\begin{aligned} P^*_{\text{B,OT}} &= \sup\{\Pr[(\text{Bob guesses } (x_0, x_1)) \wedge \neg\bot_A^{\text{OT}}]\} \\ &= \sup\{\Pr[\neg\bot_A^{\text{OT}}] \cdot \Pr[(\text{Bob guesses } (x_0, x_1))|\neg\bot_A^{\text{OT}}]\}, \end{aligned}$$

where the suprema are taken over all strategies for Bob. If Bob wants to reveal 0 in the bit-commitment protocol (a similar argument works if he wants to reveal 1), then first, Alice must not abort in the OT protocol and second, Bob must send $b = c$ as well as the correct x_c such that Alice does not abort in the last round of the bit-commitment protocol. This is equivalent to saying that Bob succeeds if he guesses x_c and Alice does not abort in the OT protocol. Since Bob randomly chooses which bit he wants to reveal, we can write the probability of Bob cheating as

$$\begin{aligned} & \max\left\{\frac{1}{2}\Pr[(\text{Bob guesses } x_0) \wedge \neg\bot_A^{\text{OT}}] + \frac{1}{2}\Pr[(\text{Bob guesses } x_1) \wedge \neg\bot_A^{\text{OT}}]\right\} \\ = \; & \max\left\{\Pr[\neg\bot_A^{\text{OT}}] \cdot \right. \\ & \qquad \left. \left(\frac{1}{2}\Pr[(\text{Bob guesses } x_0)|\neg\bot_A^{\text{OT}}] + \frac{1}{2}\Pr[(\text{Bob guesses } x_1)|\neg\bot_A^{\text{OT}}]\right)\right\}. \end{aligned}$$

Notice that we use "max" instead of "sup" above. This is because an optimal strategy exists for every bit-commitment protocol. To see this, we can construct a coin-flipping protocol from any bit-commitment protocol and an optimal strategy always exists for a coin-flipping protocol. This is a consequence of strong duality in the semidefinite programming formulation of coin-flipping cheating strategies, see the SDPs and their duals in Subsection 1.4.6.

Let us now fix Bob's optimal cheating strategy in the bit-commitment protocol. For this strategy, let

$$p := \Pr[(\text{Bob guesses } x_0)|\neg\bot_A^{\text{OT}}], \quad q := \Pr[(\text{Bob guesses } x_1)|\neg\bot_A^{\text{OT}}], \quad \text{and} \quad a := \frac{p+q}{2}.$$

We use the following lemma to relate $P^*_{\text{B,BC}}$ and $P^*_{\text{B,OT}}$ which we prove in Subsection 4.3.1.

Lemma 4.2.3 (Learning-In-Sequence Lemma)**.** *Let $p, q \in [1/2, 1]$ and suppose Alice and Bob share a joint pure state. Suppose Alice performs on her space a projective measurement $M = \{M_{x_0,x_1}\}_{x_0,x_1 \in \{0,1\}}$ to determine the value of (x_0, x_1). Suppose there is a projective measurement $P = \{P_0, P_1\}$ on Bob's space that allows him to guess bit x_0 with probability p and a projective measurement $Q = \{Q_0, Q_1\}$ on his space that allows him to guess bit x_1 with probability q. Then, there exists a measurement on Bob's space that allows him to guess (x_0, x_1) with probability at least $a(2a-1)^2$ where $a = \frac{p+q}{2}$.*

Note that we can assume Alice determines x_0 and x_1 at the end of the OT protocol even if she has inputs. She can instead input $|+\rangle$ states, run the protocol with the appropriate controlled operations, then measure them at the end in the computational basis to obtain random values for x_0 and x_1. However, the result applies to the case where the outputs are generated from within the protocol as well.

We now construct a cheating strategy for Bob for the OT protocol: Run the optimal bit-commitment strategy and look at Bob's state after the commit phase conditioned on Alice not aborting. Note that this event happens with nonzero probability in the optimal bit-commitment strategy since otherwise the success probability would be 0. The optimal bit-commitment strategy gives measurements that allow Bob to guess x_0 with probability p and x_1 with probability q. Bob uses these measurements and the procedure of Lemma 4.2.3 to guess (x_0, x_1). Let m be the probability he guesses (x_0, x_1) using this strategy. From Lemma 4.2.3, we have that $m \geq a(2a-1)^2$. By definition of $P^*_{\mathrm{B,OT}}$ and $P^*_{\mathrm{B,BC}}$, we have:

$$m = \Pr[(\text{Bob guesses } (x_0, x_1)) | \neg\bot_A^{\mathrm{OT}}] \leq \frac{P^*_{\mathrm{B,OT}}}{\Pr[\neg\bot_A^{\mathrm{OT}}]} \quad \text{and} \quad a = \frac{P^*_{\mathrm{B,BC}}}{\Pr[\neg\bot_A^{\mathrm{OT}}]}.$$

This gives us

$$\frac{P^*_{\mathrm{B,OT}}}{\Pr[\neg\bot_A^{\mathrm{OT}}]} \geq \frac{P^*_{\mathrm{B,BC}}}{\Pr[\neg\bot_A^{\mathrm{OT}}]} \left(2\frac{P^*_{\mathrm{B,BC}}}{\Pr[\neg\bot_A^{\mathrm{OT}}]} - 1 \right)^2 \implies P^*_{\mathrm{B,OT}} \geq P^*_{\mathrm{B,BC}} \left(2P^*_{\mathrm{B,BC}} - 1 \right)^2,$$

where the implication holds since $P^*_{\mathrm{B,BC}} \geq 1/2$. □

We make use of the following lower bound for quantum bit-commitment.

Proposition 4.2.4 ([CK11])**.** *For any quantum bit-commitment protocol, there is a parameter $t \in [0,1]$ such that*

$$P^*_{\mathrm{A,BC}} \geq \left(1 - \left(1 - \frac{1}{\sqrt{2}}\right) t\right)^2 \quad \text{and} \quad P^*_{\mathrm{B,BC}} \geq \frac{1}{2} + \frac{t}{2}.$$

Using Proposition 4.2.2 and the lower bound for quantum bit-commitment above, we can show the following lower bound on oblivious transfer.

Theorem 4.2.5. *In any quantum oblivious transfer protocol, at least one party can cheat with probability* 0.5852*.*

Proof. From Proposition 4.2.2, we have $P^*_{\mathrm{A,BC}} = P^*_{\mathrm{A,OT}}$ and $P^*_{\mathrm{B,OT}} \geq f(P^*_{\mathrm{B,BC}})$ (where $f(x) = x(2x-1)^2$). From the bit-commitment lower bound, we know there exists a parameter $t \in [0,1]$ such that

$$P^*_{\mathrm{B,BC}} \geq \left(1 - \left(1 - \frac{1}{\sqrt{2}}\right) t\right)^2 \geq \frac{1}{2} \quad \text{and} \quad P^*_{\mathrm{A,BC}} \geq \frac{1}{2} + \frac{t}{2} \geq \frac{1}{2},$$

noting that we have reversed the roles of Alice and Bob in the bit-commitment protocol. From Proposition 4.2.4, we know there exists a parameter $t \in [0,1]$ such that

$$P^*_{\mathrm{B,OT}} \geq f\left(\left(1 - \left(1 - \frac{1}{\sqrt{2}}\right) t\right)^2\right) \quad \text{and} \quad P^*_{\mathrm{A,OT}} \geq \frac{1}{2} + \frac{t}{2},$$

since f is nondecreasing on the interval $[1/2, 1]$. We get a lower bound on the maximum of $P^*_{\mathrm{A,OT}}$ and $P^*_{\mathrm{B,OT}}$ by equating the lower bounds above and solving for $t \approx 0.1705$ (t is a solution of a degree six polynomial). At this value of t, we have $P^*_{\mathrm{A,OT}}, P^*_{\mathrm{B,OT}} \approx 0.5852$, yielding the desired bound. □

4.3 Proof of the Learning-In-Sequence Lemma

In this section, we prove the Learning-In-Sequence Lemma which is reproduced below.

Lemma 4.3.1 (Learning-In-Sequence Lemma). *Let $p, q \in [1/2, 1]$ and suppose Alice and Bob share a joint pure state. Suppose Alice performs on her space a projective measurement $M = \{M_{x_0,x_1}\}_{x_0,x_1 \in \{0,1\}}$to determine the values of (x_0, x_1). Suppose there is a* *projective measurement $P = \{P_0, P_1\}$ on Bob's space that allows him to guess bit x_0 with probability p and a projective measurement $Q = \{Q_0, Q_1\}$ on his space that allows him to guess bit x_1 with probability q. Then, there exists a measurement on Bob's space that allows him to guess (x_0, x_1) with probability at least $a(2a-1)^2$ where $a = \frac{p+q}{2}$.*

Before giving a proof, we start with a few technical lemmas.

Lemma 4.3.2. *Let $|X\rangle$ be a pure state, Q a projection, and $|Y\rangle$ a pure state such that $Q|Y\rangle = |Y\rangle$. Then we have*

$$\|Q|X\rangle\|_2^2 \geq |\langle X|Y\rangle|^2.$$

Proof. Using Cauchy-Schwarz, we have

$$|\langle X|Y\rangle|^2 = |\langle X|Q|Y\rangle|^2 \leq \|Q|X\rangle\|_2^2 \, \||Y\rangle\|_2^2 = \|Q|X\rangle\|_2^2 \,,$$

as desired. □

Lemma 4.3.3. *Suppose $\theta, \theta' \in [0, \pi/4]$. If $|\langle\psi|\phi\rangle| \geq \cos(\theta)$ and $|\langle\phi|\xi\rangle| \geq \cos(\theta')$, then*

$$|\langle\psi|\xi\rangle| \geq \cos(\theta + \theta').$$

Proof. Define the angle between two pure states $|\psi\rangle$ and $|\phi\rangle$ as $A(\psi, \phi) := \arccos|\langle\psi|\phi\rangle|$. This is a metric (see [NC00] page 413). Thus, we have

$$\arccos|\langle\psi|\xi\rangle| = A(\psi, \xi) \leq A(\psi, \phi) + A(\phi, \xi) = \arccos|\langle\psi|\phi\rangle| + \arccos|\langle\phi|\xi\rangle| \leq \theta + \theta'.$$

Taking the cosine of both sides yields the result. □

Lemma 4.3.4. *Let $\theta, \theta' \in [0, \pi/4]$. Then*

$$\cos(\theta + \theta') \geq \cos^2(\theta) + \cos^2(\theta') - 1.$$

Proof. Without loss of generality, suppose that $\theta \geq \theta'$. Consider the function

$$f(\theta) = \cos(\theta + \theta') - \cos^2(\theta) + \sin^2(\theta')$$

for fixed θ'. Taking its derivative, we get

$$f'(\theta) = -\sin(\theta + \theta') + \sin(2\theta)$$

which is nonnegative for $\theta \in [\theta', \pi/4]$. Since $f(\theta') = 0$, we conclude that $f(\theta) \geq 0$ for $\theta \in [\theta', \pi/4]$, which gives the desired result. □

We combine these lemmas to prove the following lemma.

Lemma 4.3.5. *Let $|\psi\rangle$ be a pure state and let $\{C, I-C\}$ and $\{D, I-D\}$ be two projective measurements such that*

$$\cos(\theta) := \|C|\psi\rangle\|_2 \geq \frac{1}{\sqrt{2}} \quad \textit{and} \quad \cos(\theta') := \|D|\psi\rangle\|_2 \geq \frac{1}{\sqrt{2}}.$$

Then we have

$$\|DC|\psi\rangle\|_2^2 \geq \cos^2(\theta)\cos^2(\theta+\theta').$$

Proof. Define the following states

$$|X\rangle := \frac{C|\psi\rangle}{\|C|\psi\rangle\|_2}, \quad |X'\rangle := \frac{(I-C)|\psi\rangle}{\|(I-C)|\psi\rangle\|_2}, \quad |Y\rangle := \frac{D|\psi\rangle}{\|D|\psi\rangle\|_2}, \quad |Y'\rangle := \frac{(I-D)|\psi\rangle}{\|(I-D)|\psi\rangle\|_2},$$

which are well-defined (otherwise, if any of the denominators are 0, then the result holds trivially). We can write

$$|\psi\rangle = \cos(\theta)|X\rangle + \sin(\theta)|X'\rangle \quad \text{and} \quad |\psi\rangle = \cos(\theta')|Y\rangle + \sin(\theta')|Y'\rangle$$

yielding

$$\begin{aligned} \|DC|\psi\rangle\|_2^2 &= \cos^2(\theta)\,\|D|X\rangle\|_2^2 \\ &\geq \cos^2(\theta)|\langle Y|X\rangle|^2 \quad \text{using Lemma 4.3.2} \\ &\geq \cos^2(\theta)\cos^2(\theta+\theta') \quad \text{using Lemma 4.3.3,} \end{aligned}$$

as desired. □

We now prove Lemma 4.3.1.

Proof. Let $|\Omega\rangle \in \mathbb{C}^{A\times B}$ be the joint pure state shared by Alice and Bob, where $\mathbb{C}^A$ is the space controlled by Alice and $\mathbb{C}^B$ is the space controlled by Bob.

Let $M = \{M_{x_0,x_1}\}_{x_0,x_1\in\{0,1\}}$ be Alice's projective measurement on $\mathbb{C}^A$ to determine her bits x_0, x_1. Let $P = \{P_0, P_1\}$ be Bob's projective measurement that allows him to guess x_0 with probability $p = \cos^2(\theta)$ and $Q = \{Q_0, Q_1\}$ be Bob's projective measurement that allows him to guess x_1 with probability $q = \cos^2(\theta')$. These measurements are on $\mathbb{C}^B$ only. Recall that $a = \frac{p+q}{2} = \frac{\cos^2(\theta)+\cos^2(\theta')}{2}$. We consider the following projections on $\mathbb{C}^{A\times B}$:

$$C = \sum_{x_0,x_1} M_{x_0,x_1} \otimes P_{x_0} \quad \text{and} \quad D = \sum_{x_0,x_1} M_{x_0,x_1} \otimes Q_{x_1}.$$

C (resp. D) is the projection on the subspace where Bob guesses correctly the first bit (resp. the second bit) after applying P (resp. Q).

A strategy for Bob to learn both bits is simple: Apply the two measurements P and Q one after the other, where the first one is chosen uniformly at random. The measurement operator on the subspace where Bob guesses (x_0, x_1) when applying P then Q is

$$E = \sum_{x_0,x_1} M_{x_0,x_1} \otimes Q_{x_1} P_{x_0} = DC.$$

Similarly, the measurement operator on the subspace where Bob guesses (x_0, x_1) when applying Q then P is

$$F = \sum_{x_0,x_1} M_{x_0,x_1} \otimes P_{x_0} Q_{x_1} = CD.$$

With this strategy, Bob can guess both bits with probability

$$\begin{aligned}
& \frac{1}{2}\left(||E|\Omega\rangle||_2^2 + ||F|\Omega\rangle||_2^2\right) \\
= & \frac{1}{2}\left(||DC|\Omega\rangle||_2^2 + ||CD|\Omega\rangle||_2^2\right) \\
\geq & \frac{1}{2}\left(\cos^2(\theta) + \cos^2(\theta')\right)\cos^2(\theta+\theta') \quad \text{using Lemma 4.3.5} \\
\geq & \frac{1}{2}\left(\cos^2(\theta) + \cos^2(\theta')\right)\left(\cos^2(\theta) + \cos^2(\theta') - 1\right)^2 \quad \text{using Lemma 4.3.4} \\
= & a(2a-1)^2.
\end{aligned}$$

Note that we can use Lemma 4.3.5 since Bob's optimal measurements to guess x_0 and x_1 each succeed with probability at least 1/2. □

4.4 Forcing oblivious transfer

Here we discuss a variant of oblivious transfer which is a generalization of coin-flipping. Like coin-flipping, it can be analyzed using Kitaev's semidefinite programming formulation of cheating strategies. Although this primitive is not as interesting as the standard definition of OT from a cryptographic standpoint, it helps illustrate the different security notions in quantum cryptography. In addition, it shows how OT can be analyzed using the techniques from earlier chapters.

We reproduce the definition of forcing oblivious transfer below.

Definition 4.4.1 (**Forcing oblivious transfer** (Forcing-OT))**.** *A k-out-of-n forcing oblivious transfer protocol, denoted as* Forcing-OT$_{(n,k)}$*, with* forcing bias $\varepsilon_{\mathrm{FOT}}$*, is a protocol satisfying:*

- *Alice and Bob start uncorrelated,*
- *Alice and Bob have* no inputs*,*
- *Alice outputs n random bits $x := (x_1, \ldots, x_n)$,*
- *Bob outputs a random k-index set b and bit string x_b consisting of x_i for $i \in b$,*
- $P^*_{\mathrm{A},(b,x_b)} := \sup\{\Pr[\textit{Alice can force Bob to output } (b, x_b)]\} = \dfrac{\varepsilon_{\mathrm{A}}}{\binom{n}{k} \cdot 2^k}$,
- $P^*_{\mathrm{B},x} := \sup\{\Pr[\textit{Bob can force Alice to output } x]\} = \dfrac{\varepsilon_{\mathrm{B}}}{2^n}$,
- $\varepsilon_{\mathrm{FOT}} = \max\{\varepsilon_{\mathrm{A}}, \varepsilon_{\mathrm{B}}\}$,

where the suprema are taken over all cheating strategies of the party indicated.

Note that we are interested in how much each party can force a desired output, as in coin-flipping. In the language of Forcing-OT, strong coin-flipping is a Forcing-OT$_{(1,1)}$ protocol. This is in contrast to the standard definition of oblivious transfer studied in the first part of this chapter, where the goal was to *learn* inputs.

4.4.1 Extending Kitaev's lower bound for strong coin-flipping

We now extend Kitaev's formalism from the setting of coin-flipping to the more general setting of Forcing-OT$_{(n,k)}$. Very little modifications are needed to the coin-flipping setting described in Subsection 1.4.6, we can still use the setting for the spaces and unitaries.

However, we need different projections since the outputs of the protocol are different in the case of Forcing-OT. We can suppose Alice has projections

$$\{\Pi_{\text{A,abort}}\} \cup \{\Pi_{\text{A},x} : x \in \mathbb{Z}_2^n\} \subset \Sigma_+^A$$

and Bob has the projections

$$\{\Pi_{\text{B,abort}}\} \cup \left\{\Pi_{\text{B},(b,x_b)} : b \text{ a k-element subset of n indices}, x_b \in \mathbb{Z}_2^k\right\} \subset \Sigma_+^B.$$

If we let $|\psi\rangle \in \mathbb{C}^{A \times M \times B}$ be the state at the end of the protocol when Alice and Bob are honest, the requirements of a Forcing-OT$_{(n,k)}$ protocol are satisfied when

$$\langle \Pi_{\text{A},x} \otimes \text{I}_M \otimes \Pi_{\text{B},(b,x_b)}, |\psi\rangle\langle\psi| \rangle = \frac{1}{\binom{n}{k} 2^n} \text{ for all consistent } (x, b, x_b).$$

Similar to coin-flipping, we can capture cheating strategies as semidefinite programs. Bob can force honest Alice to output a specific $x \in \mathbb{Z}_2^n$ with maximum probability equal to the optimal value of the following semidefinite program

$$\begin{array}{rrcll}
P^*_{\text{B},x} = \max & \langle \Pi_{\text{A},x}, \rho_{A,n} \rangle & & & \\
\text{subject to} & \rho_{A,0} & = & |\psi_{A,0}\rangle\langle\psi_{A,0}|, & \\
& \rho_{A,i} & = & \rho_{A,i-1}, & \text{for all } i \text{ even}, \\
& \text{Tr}_M \tilde{\rho}_{A,i} & = & \rho_{A,i}, & \text{for all } i \text{ even}, \\
& \rho_{A,i} & = & \text{Tr}_M \left[U_{\text{A},i} \tilde{\rho}_{A,i-1} U^*_{\text{A},i} \right], & \text{for all } i \text{ odd}, \\
& \rho_{A,i} & \in & \Sigma_+^A, & \text{for all } i, \\
& \tilde{\rho}_{A,i} & \in & \Sigma_+^{A \times M}, & \text{for all } i \text{ even}.
\end{array}$$

Similarly, Alice can force honest Bob to output a specific (b, x_b) with maximum probability equal to the optimal value of the following semidefinite program

$$\begin{array}{rrcll}
P^*_{\text{A},(b,x_b)} = \max & \langle \Pi_{\text{B},(b,x_b)}, \rho_{B,n} \rangle & & & \\
\text{subject to} & \rho_{B,0} & = & |\psi_{B,0}\rangle\langle\psi_{B,0}|, & \\
& \rho_{B,i} & = & \rho_{B,i-1}, & \text{for all } i \text{ odd}, \\
& \text{Tr}_M \tilde{\rho}_{B,i} & = & \rho_{B,i}, & \text{for all } i \text{ odd}, \\
& \rho_{B,i} & = & \text{Tr}_M \left[U_{\text{B},i} \tilde{\rho}_{B,i-1} U^*_{\text{B},i} \right], & \text{for all } i \text{ even}, \\
& \rho_{B,i} & \in & \Sigma_+^B, & \text{for all } i, \\
& \tilde{\rho}_{B,i} & \in & \Sigma_+^{M \times B}, & \text{for all } i \text{ odd}.
\end{array}$$

Using Kitaev's proof for coin-flipping, we get the following theorem.

Theorem 4.4.2. *In any* Forcing-OT$_{(n,k)}$ *protocol, for consistent* (x, b, x_b)*, we have*

$$P^*_{\mathrm{B},x}\, P^*_{\mathrm{A},(b,x_b)} \geq \Pr[\textit{Alice honestly outputs } x \textit{ and Bob honestly outputs } (b, x_b)] = \frac{1}{\binom{n}{k}2^n}.$$

In particular, the forcing bias satisfies $\varepsilon_{\mathrm{FOT}} \geq 2^{k/2}$*.*

The only difference in the proof is that the honest outcome probabilities are different. From the last line of Kitaev's proof, we get

$$P^*_{\mathrm{B},x} \cdot P^*_{\mathrm{A},(b,x_b)} \geq \langle \Pi_{\mathrm{A},x} \otimes \mathrm{I}_M \otimes \Pi_{\mathrm{B},(b,x_b)}, |\psi\rangle\langle\psi| \rangle = \frac{1}{\binom{n}{k}2^n},$$

as desired.

4.4.2 A protocol with optimal forcing bias

First, consider the following protocol which achieves the bound in Theorem 4.4.2 but is asymmetric. Alice sends n random bits to Bob who then outputs b, a random k-index subset of n indices, and x_b. In this protocol, Bob can force a desired outcome with probability $\frac{1}{2^n}$ and Alice can force a desired outcome with probability $\frac{1}{\binom{n}{k}}$. Thus, the product of the cheating probabilities is optimal, that is it achieves the lower bound in Theorem 4.4.2, however the protocol is asymmetric. This can be easily remedied using coin-flipping. We present an optimal protocol below.

Protocol 4.4.3 (An optimal Forcing-OT$_{(n,k)}$ protocol with forcing bias $2^{k/2}$)**.**

(i) *Bob randomly chooses a k-index set b and sends b to Alice.*

(ii) *Alice and Bob generate each bit in x_b using a strong coin-flipping protocol where neither party can force the outcome with probability greater than $\frac{1}{\sqrt{2}} + \delta$, for some small constant $\delta > 0$.*

(iii) *Alice randomly chooses her bits not in b.*

Theorem 4.4.4. *For any $\gamma > 0$, we can choose a $\delta > 0$ such that the* Forcing-OT$_{(n,k)}$ *protocol above satisfies*

$$P^*_{\mathrm{A},(b,x_b)} \leq \frac{2^{k/2}(1+\gamma)}{\binom{n}{k} \cdot 2^k} \quad \text{and} \quad P^*_{\mathrm{B},x} \leq \frac{2^{k/2}(1+\gamma)}{2^n}.$$

The bias satisfies $\varepsilon_{\text{FOT}} \leq 2^{k/2}(1+\gamma)$.

Proof. Fix $\gamma > 0$ and a coin-flipping parameter $\delta > 0$ sufficiently small enough so that $\left(\frac{1}{\sqrt{2}} + \delta\right)^k \leq \frac{2^{k/2}(1+\gamma)}{2^k}$. This sets an upper bound on the probability of forcing a k bit string using k coin-flipping protocols each with a maximum cheating probability of $\frac{1}{\sqrt{2}} + \delta$. We now analyze each party cheating. Alice has no control over the index set but she can try to force a particular bit string for x_b. Her maximum cheating probability is

$$P^*_{\text{A},(b,x_b)} \leq \frac{1}{\binom{n}{k}} \cdot \left(\frac{1}{\sqrt{2}} + \delta\right)^k \leq \frac{1}{\binom{n}{k}} \cdot \frac{2^{k/2}(1+\gamma)}{2^k} = \frac{2^{k/2}(1+\gamma)}{\binom{n}{k}2^k}.$$

Bob has no control over Alice's $n-k$ remaining bits, so he can cheat with maximum probability

$$P^*_{\text{B},x} \leq \frac{1}{2^{n-k}} \cdot \left(\frac{1}{\sqrt{2}} + \delta\right)^k \leq \frac{1}{2^{n-k}} \cdot \frac{2^{k/2}(1+\gamma)}{2^k} = \frac{2^{k/2}(1+\gamma)}{2^n}.$$

□

For the special case of Forcing-OT$_{(1,2)}$, we have the following corollary.

Corollary 4.4.5 (Optimal Forcing-OT$_{(1,2)}$)**.** *For any* $\gamma > 0$*, there exists a protocol for* Forcing-OT$_{(1,2)}$ *where each party has honest outcome probabilities of* $1/4$ *and neither party can cheat with probability greater than* $\frac{1}{\sqrt{8}} + \gamma \approx 0.3535 + \gamma$.

We remark that this protocol is completely classical with the exception of the quantum coin-flipping subroutines. This is similar to the optimal strong coin-flipping protocol in [CK09] designed using classical messages and quantum weak coin-flipping subroutines.

Chapter 5

Concluding remarks

In this thesis, we analyzed two primitives using convex optimization techniques. In Chapters 2 and 3, we studied a family of coin-flipping protocols which are based on quantum bit-commitment and Chapter 4 involved the analysis of oblivious transfer using ideas from bit-commitment and coin-flipping. Below, we summarize the ideas and concepts used in this thesis and some open questions that arise.

Coin-flipping

We studied the security of quantum (and classical) coin-flipping protocols based on bit-commitment utilizing SDP formulations of cheating strategies. These SDPs allowed us to use concepts from convex optimization to further our understanding of the security of such protocols. In particular, using a reduction of the SDPs and duality theory, we were able to develop a family of point games corresponding to the protocols.

We constructed the classical counterpart to the quantum protocols and showed how the analysis is very similar to the quantum version due to the similarities between the cheating strategy formulations. Figure 5.1 (on the next page) summarizes the connections between the classical and quantum protocols. Using these connections, we were able to show that a bias of $\varepsilon = 1/\sqrt{2} - 1/2$ is impossible for BCCF-protocols using a modified version of Kitaev's lower bound.

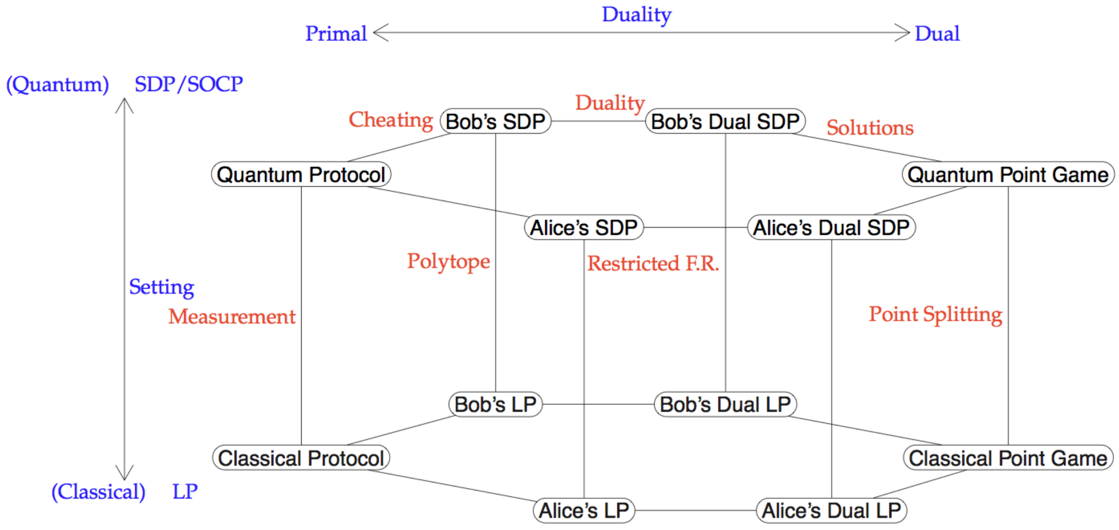


Figure 5.1: Classical/quantum BCCF-protocol relationship.

Finding optimal cheating strategies for protocols with multiple rounds of communication turns out to be a very difficult problem. To further our understanding of the structure of optimal protocols, we designed a search algorithm that seeks protocols with small bias. Using cheating strategies that arise from studying the cheating SDPs, we designed a protocol filter to eliminate protocols with high bias from the search. Using the protocol filter and other heuristics, we checked 10^{16} protocols for optimality, a task that would take millions of years without using the optimization techniques developed in this thesis. We conclude with the conjecture that the smallest attainable bias for BCCF-protocols is $\varepsilon = 1/4$.

Coin-flipping open questions

An open problem is to find the optimal cheating strategies for a general n-round BCCF-protocol. This can be accomplished by finding closed-form optimal solutions to the cheating SDPs or the reduced cheating SDPs. Very few highly interactive protocols, such as BCCF-protocols, have descriptions of optimal cheating strategies and therefore having such for this family of protocols would be very interesting.

A related open problem is to find an explicit construction of optimal protocols for weak coin-flipping, strong coin-flipping, or bit-commitment. We can accomplish all three of these tasks by finding a construction of optimal weak coin-flipping protocols, so this would be very rewarding. Technically, such a construction is implicit in [Moc07], however it involves many reductions and is quite complicated. As for strong coin-flipping, it seems that we may need to look further than BCCF-protocols to find a way to construct an optimal protocol.

Oblivious transfer

We described a new protocol for oblivious transfer such that Alice can learn Bob's index with maximum probability 3/4 and Bob can learn Alice's input with maximum probability 3/4. As for lower bounds, we use a construction of bit-commitment from oblivious transfer and related the corresponding cheating strategies. We then exhibited a cheating strategy for Bob using a Learning-In-Sequence Lemma which is a way for Bob to sequentially learn Alice's two input bits. Using this strategy and known bounds for bit-commitment, we derived a lower bound on the bias of any oblivious transfer protocol.

To relate oblivious transfer to the analysis of Chapters 2 and 3, we studied a "forcing" variant of oblivious transfer. This primitive allows for SDP formulations of cheating strategies in the same way as coin-flipping. As in the case of coin-flipping, these SDPs led to optimal lower bounds for the bias.

Oblivious transfer open questions

An open problem is to formulate cheating strategies for oblivious transfer as semidefinite programs in a way that gives a nontrivial lower bound on the bias. The difficulty is that in oblivious transfer, the cheating goal is to *learn* the other party's input (as opposed to the goal in coin-flipping which is to *force* an outcome). This subtle difference makes the task much more challenging. If it were possible, it would be interesting to see if a point game analog can be made for oblivious transfer.

Solving for the optimal bias of oblivious transfer protocols is still an open problem. The lower bound we obtained for the bias originally relied on Kitaev's bound for strong coin-flipping, but then we used the stronger bound for bit-commitment to get a better result. Perhaps it is possible to find a different reduction between oblivious transfer and another well-understood primitive to give an improved lower bound on the bias or a better protocol.

Bibliography

[ABDR04] A. Ambainis, H. Buhrman, Y. Dodis, and H. Rohrig. Multiparty quantum coin flipping. In *Proceedings of the 19th IEEE Annual Conference on Computational Complexity*, pages 250–259, 2004.

[AG03] F. Alizadeh and D. Goldfarb. Second-order cone programming. *Math. Program.*, 95:3–51, 2003.

[Alb83] P. Alberti. A note on the transition probability over C^*-algebras. *Letters in Mathematical Physics*, 7(1):25–32, 1983.

[Amb01] A. Ambainis. A new protocol and lower bounds for quantum coin flipping. In *Proceedings of 33rd Annual ACM Symposium on the Theory of Computing*, pages 134 – 142, 2001.

[Amb02] A. Ambainis. Lower bound for a class of weak quantum coin flipping protocols. Available as arXiv.org e-Print quant-ph/0204063, 2002.

[ATVY00] D. Aharonov, A. Ta-Shma, U. Vazirani, and A. Yao. Quantum bit escrow. In *Proceedings of 32nd Annual ACM Symposium on the Theory of Computing*, pages 705–714, 2000.

[BB84] C. Bennett and G. Brassard. Quantum cryptography: Public key distribution and coin tossing. In *Proceedings of the IEEE International Conference on Computers, Systems, and Signal Processing*, pages 175–179, 1984.

[BBBW83] C. Bennett, G. Brassard, S. Breidbard, and S. Wiesner. Quantum cryptography, or unforgeable subway tokens. In *Advances in Cryptology CRYPTO 1982*, pages 267–275, 1983.

[BCvD05] D. Bacon, A. Childs, and W. van Dam. From optimal measurement to efficient quantum algorithms for the hidden subgroup problem over semidirect product groups. In *Proceedings of 46th IEEE Symposium on Foundations of Computer Science*, pages 469–478, 2005.

[BCvD06] D. Bacon, A. Childs, and W. van Dam. Optimal measurements for the dihedral hidden subgroup problem. In *Chicago Journal of Theoretical Computer Science*, article 2, 2006.

[BG89] M. Bellare and S. Goldwasser. New paradigms for digital signatures and message authentication based on non-interactive zero knowledge proofs. In *Advances in Cryptology CRYPTO 1989*, pages 194–211, 1989.

[Blu81] M. Blum. Coin flipping by telephone. In *Advances in Cryptology CRYPTO 1981*, pages 11–15, 1981.

[BSS03] H. Barnum, M. Saks, and M. Szegedy. Quantum query complexity and semidefinite programming. In *IEEE Conference on Computational Complexity*, pages 179–193, 2003.

[BV97] E. Bernstein and U. Vazirani. Quantum complexity theory. *Special issue on Quantum Computation of the Siam Journal of Computing*, 26(5):1411–1473, 1997.

[CK09] A. Chailloux and I. Kerenidis. Optimal quantum strong coin flipping. In *Proceedings of 50th IEEE Symposium on Foundations of Computer Science*, pages 527–533, 2009.

[CK11] A. Chailloux and I. Kerenidis. Optimal bounds for quantum bit commitment. In *Proceedings of 52nd IEEE Symposium on Foundations of Computer Science*, pages 354–362, 2011.

[CKS10] A. Chailloux, I. Kerenidis, and J. Sikora. Lower bounds for quantum oblivious transfer. In *Proceedings of IARCS Annual Conference on Foundations of Software Technology and Theoretical Computer Science*, volume 8, pages 157–168, 2010.

[Cré87] C. Crépeau. Equivalence between two flavours of oblivious transfers. In *Advances in Cryptology CRYPTO 1987*, pages 350–354, 1987.

[DFSS08] I. Damgård, S. Fehr, L. Salvail, and C. Schaffner. Cryptography in the bounded quantum-storage model. *SIAM Journal of Computing*, 37(6):1865–1890, 2008.

[DiV00] D. DiVincenzo. The physical implementation of quantum computation. *Fortschr. Phys.*, 48:771–783, 2000.

[EGL85] S. Even, O. Goldreich, and A. Lempel. A randomized protocol for signing contracts. *Communications of the ACM*, 28(6):637–647, 1985.

[Eis05] J. Eisert. Optimizing linear optics quantum gates. *Phys. Rev. Lett.*, 95:040502, 2005.

[EMV03] Y. Eldar, A. Megretski, and G. Verghese. Designing optimal quantum detectors via semidefinite programming. *IEEE Transactions on Information Theory*, 49:1007–1012, 2003.

[ESH04] Y. Eldar, M. Stojnic, and B. Hassibi. Optimal quantum detectors for unambiguous detection of mixed states. *Phys. Rev. A*, 69(6):062318, 2004.

[FvdG99] C. Fuchs and J. van de Graaf. Cryptographic distinguishability measures for quantum mechanical states. *IEEE Transactions on Information Theory*, 45:1216–1227, 1999.

[GSU11] S. Gharibian, J. Sikora, and S. Upadhyay. QMA variants with polynomially many provers. Available as arXiv.org e-Print quant-ph/1108.0617, 2011.

[Gut09] G. Gutoski. *Quantum strategies and local operations.* PhD Thesis, University of Waterloo, 2009. Available at arXiv.org e-Print quant-ph/1003.0038.

[Gut10] G. Gutoski. On a measure of distance for quantum strategies. Available as arXiv.org e-Print quant-ph/1008.4636, 2010.

[GW95] M. Goemans and D. Williamson. Improved approximation algorithms for maximum cut and satisfiability problems using semidefinite programming. *Journal of the ACM*, pages 1115–1145, 1995.

[GW05] G. Gutoski and J. Watrous. Quantum interactive proofs with competing provers. *Lecture Notes in Computer Science*, 3404:605–616, 2005.

[GW07] G. Gutoski and J. Watrous. Toward a general theory of quantum games. In *Proceedings of the 39th ACM Symposium on Theory of Computing*, pages 565–574, 2007.

[Hel69] C. Helstrom. Quantum detection and estimation theory. *Journal of Statistical Physics*, 1(2):231–252, 1969.

[HLS07] P. Høyer, T. Lee, and R. Spalek. Negative weights make adversaries stronger. In *Proceedings of 39th Annual ACM Symposium on the Theory of Computing*, pages 526–535, 2007.

[Hol73a] A. Holevo. Some estimates of the information transmitted by quantum communication channels. *Problemy Peredachi Informatsii*, 9:3–11, 1973.

[Hol73b] A. Holevo. Statistical decisions in quantum theory. *Journal of Multivariate Analysis*, 3:337–394, 1973.

[Ip03] L. Ip. Shor's algorithm is optimal. Manuscript, 2003.

[JJUW11] R. Jain, Z. Ji, S. Upadhyay, and J. Watrous. QIP = PSPACE. *Journal of ACM*, 58(6), Article no. 30, 2011.

[JRS09] R. Jain, J. Radhakrishnan, and P. Sen. A new information-theoretic property about quantum states with an application to privacy in quantum communication. *Journal of ACM*, 56(6), Article no. 33, 2009.

[Ker83] A. Kerckhoffs. La cryptographie militaire. *Journal des sciences militaires*, 9:5–38, 1883.

[Kil88] J. Kilian. Founding cryptography on oblivious transfer. In *Proceedings of the 20th Annual ACM Symposium on Theory of Computing*, pages 20–31, 1988.

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

[KN04] I. Kerenidis and A. Nayak. Weak coin flipping with small bias. *Inf. Process. Lett.*, 89(3):131–135, 2004.

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

[LC97] H.-K. Lo and H. Chau. Why quantum bit commitment and ideal quantum coin tossing are impossible. *Physica D: Nonlinear Phenomena*, 120:177–194, 1997.

[LC99] H.-K. Lo and H. Chau. Unconditional security of quantum key distribution over arbitrarily long distances. *SCIENCE*, 283:2050–2056, 1999.

[LMR$^+$11] T. Lee, R. Mittal, B. Reichardt, R. Spalek, and M. Szegedy. Quantum complexity of state conversion. In *Proceedings of 52nd IEEE Symposium on Foundations of Computer Science*, pages 344–353, 2011.

[Lo97] H.-K. Lo. Insecurity of quantum secure computations. *Phys. Rev. A*, 56(2):1154–1162, 1997.

[LR11] T. Lee and J. Roland. A strong direct product theorem for quantum query complexity. Available as arXiv.org e-Print quant-ph/1104.4468, 2011.

[May97] D. Mayers. Unconditionally secure quantum bit commitment is impossible. *Phys. Rev. Lett.*, 78(17):3414–3417, 1997.

[May01] D. Mayers. Unconditional security in quantum cryptography. *Journal of the ACM*, 48(3):351–406, 2001.

[Mit03] H. Mittelmann. An independent benchmarking of SDP and SOCP solvers. Computational semidefinite and second order cone programming: the state of the art. *Math. Program.*, 95(2 Ser. B):407–430, 2003.

[Moc04] C. Mochon. Quantum weak coin-flipping with bias of 0.192. In *Proceedings of the 45th Annual IEEE Symposium on Foundations of Computer Science*, pages 2–11, 2004.

[Moc05] C. Mochon. A large family of quantum weak coin-flipping protocols. *Phys. Rev. A*, 72(2):022341, 2005.

[Moc07] C. Mochon. Quantum weak coin flipping with arbitrarily small bias. Available as arXiv.org e-Print quant-ph/0711.4114, 2007.

[MS07] R. Mittal and M. Szegedy. Product rules in semidefinite programming. *Lecture Notes in Computer Science*, 4639:435–445, 2007.

[Nay99] A. Nayak. Optimal lower bounds for quantum automata and random access codes. In *Proceedings of 40th IEEE Symposium on Foundations of Computer Science*, pages 369–376, 1999.

[NC00] M. Nielsen and I. Chuang. *Quantum Computation and Quantum Information.* Cambridge University Press, 2000.

[NN94] Y. Nesterov and A. Nemirovskii. *Interior Point Polynomial Methods in Convex Programming: Theory and Applications*, SIAM. Philadelphia, 1994.

[NS03] A. Nayak and P. Shor. On bit-commitment based quantum coin flipping. *Phys. Rev. A*, 67(1):012304, 2003.

[PS00] J. Preskill and P. Shor. Simple proof of security of the BB84 quantum key distribution protocol. *Phys. Rev. Lett.*, 85(2):441–444, 2000.

[PW09] M. Piani and J. Watrous. All entangled states are useful for channel discrimination. *Phys. Rev. Lett.*, 102(25):250501, 2009.

[Rab81] M. Rabin. How to exchange secrets by oblivious transfer. In *Technical Report TR-81, Aiken Computation Laboratory, Harvard University*, 1981.

[Rei09] B. Reichardt. Span programs and quantum query complexity: The general adversary bound is nearly tight for every boolean function. In *Proceedings of 50th IEEE Symposium on Foundations of Computer Science*, pages 544–551, 2009.

[Sac05] M. Sacchi. Optimal discrimination of quantum operations. *Phys. Rev. A*, 71(6):062340, 2005.

[Sch10] C. Schaffner. Simple protocols for oblivious transfer and secure identification in the noisy-quantum-storage model. *Phys. Rev. A*, 82:032308, 2010.

[Sho94] P. Shor. Algorithms for quantum computation: Discrete logarithms and factoring. In *Proceedings of 35th IEEE Symposium on Foundations of Computer Science*, pages 124–134, 1994.

[SR01] R. Spekkens and T. Rudolph. Degrees of concealment and bindingness in quantum bit commitment protocols. *Phys. Rev. A*, 65:012310, 2001.

[SR02] R. Spekkens and T. Rudolph. Quantum protocol for cheat-sensitive weak coin flipping. *Phys. Rev. Lett.*, 89(22):1–4, 2002.

[SS06] R. Spalek and M. Szegedy. All quantum adversary methods are equivalent. *Theory of Computing*, 2(1):1–18, 2006.

[SSS09] L. Salvail, C. Schaffner, and M. Sotáková. On the power of two-party quantum cryptography. In *International Conference on the Theory and Application of Cryptology and Information Security - ASIACRYPT 2009*, pages 70–87, 2009.

[Stu99] J. Sturm. Using SeDuMi 1.02, a MATLAB toolbox for optimization over symmetric cones. *Optimization Methods and Software*, 11:625–653, 1999.

[Stu02] J. Sturm. Implementation of interior point methods for mixed semidefinite and second order cone optimization problems. *Optimization Methods and Software*, 17(6):1105–1154, 2002.

[Tun10] L. Tunçel. *Polyhedral and Semidefinite Programming Methods in Combinatorial Optimization*, volume FIM/27. Fields Institute Monograph Series, AMS, 2010.

[Uhl76] A. Uhlmann. The "transition probability" in the state space of a *-algebra. *Reports on Mathematical Physics*, 9(2):273–279, 1976.

[Wat09] J. Watrous. Semidefinite programs for completely bounded norms. *Theory of Computing*, 5:217–238, 2009.

[Wie83] S. Wiesner. Conjugate coding. *SIGACT News*, 15(1):78–88, 1983.

[WSV00] H. Wolkowicz, R. Saigal, and L. Vandenberghe, editors. *Handbook of Semidefinite Programming.* Kluwer Academic Publishers, 2000.

[Yao93] A. Yao. Quantum circuit complexity. In *Proceedings of the 34th Annual IEEE Symposium on Foundations of Computer Science*, pages 352–361, 1993.

[YKL75] H. Yuen, R. Kennedy, and M. Lax. Optimum testing of multiple hypotheses in quantum detection theory. *IEEE Transactions on Information Theory*, 21:125–134, 1975.