Improving the security of device-independent weak coin flipping

Atul Singh Arora[1], Jamie Sikora[2], and Thomas Van Himbeeck[3,4]

[1] *California Institute of Technology, USA*
[2] *Virginia Polytechnic Institute and State University, USA*
[3] *University of Toronto, Canada*
[4] *Institute of Quantum Computing, University of Waterloo, Canada*

May 11, 2021

Abstract

Weak coin flipping is the cryptographic task where Alice and Bob remotely flip a coin but want opposite outcomes. This work studies this task in the device-independent regime where Alice and Bob neither trust each other, nor their quantum devices. The best protocol was devised ten years ago by Silman, Chailloux, Aharon, Kerenidis, Pironio, and Massar with bias $\epsilon \leq 0.33664$, where the bias is a commonly adopted security measure for coin flipping protocols. This work presents some techniques to lower the bias of device-independent weak coin flipping protocols, namely self-testing and abort-phobic compositions. By applying these techniques to the SCAKPM '11 protocol above, we are able to lower the bias to $\epsilon \approx 0.3148$. Under a continuity assumption, we can lower it still to $\epsilon \approx 0.29104$. In our analysis, we study the estimation of expectation value of a single GHZ game from known statistics to apply self-testing results in our case. We also study the continuity of semidefinite programs. These may be of independent interest.

Contents

1 Introduction

Coin-flipping is the two-party cryptographic primitive where two parties, henceforth called Alice and Bob, wish to flip a coin, but, to make things interesting, they do not trust each other. This primitive was introduced by Blum [Blu83] who also introduced the first (classical) protocol. In this work, we concentrate on *weak* coin flipping (WCF) protocols where Alice and Bob desire opposite outcomes. Since then, a series of quantum protocols were introduced which kept improving the security. Mochon finally settled the question about the limits of the security in the quantum regime by proving the existence of quantum protocols with security approaching the ideal limit [Moc07]. Mochon's work was based on the notion of point games, a concept introduced by Kitaev. Since then, a sequence of works have continued the study of point games. In particular, the proof has been simplified [ACG+14] and made explicit [ARW, ARW19, ARV] Interestingly, Miller [Mil20] used Mochon's proof to show that protocols approaching the ideal limit must have an exponentially increasing number of messages. We note that all of this work is in the *device-dependent* setting where *Alice and Bob trust their quantum devices*. In this work, we *revise* the security definitions such that when Alice or Bob cheat, they have control of each other's quantum devices, opening up a plethora of new cheating strategies that were not considered in the previously mentioned references.

The prefix *weak* in weak coin flipping refers to the situation where Alice and Bob desire opposite outcomes of the coin. (We have occasion to discuss *strong* coin flipping protocols, where Alice or Bob could try to bias the coin towards either outcome, but it is not the focus of this work.) When designing weak coin flipping protocols, the security goals are as follows.

Correctness for honest parties:	If Alice and Bob are honest, then they share the same outcome of a protocol $c \in \{0,1\}$, and c is generated uniformly at random by the protocol.
Soundness against cheating Bob:	If Alice is honest, then a dishonest (i.e., cheating) Bob cannot force the outcome $c = 1$.
Soundness against cheating Alice:	If Bob is honest, then a dishonest (i.e., cheating) Alice cannot force the outcome $c = 0$.

The commonly adopted goal of two-party protocol design is to assume perfect correctness and then minimize the effects of a cheating party, i.e., to make it as sound as possible. This way, if no parties cheats, then the protocol at least does what it is meant to still. With this in mind, we need a means to quantify the effects of a cheating party. It is often convenient to have a single measure to determine if one protocol is better than another. For this purpose, we use *cheating probabilities* (denoted p_B^* and p_A^*) and *bias* (denoted ϵ), defined as follows.

- p_B^*: The maximum probability with which a dishonest Bob can force an honest Alice to accept the outcome $c = 1$.
- p_A^*: The maximum probability with which a dishonest Alice can force an honest Bob to accept the outcome $c = 0$.
- ϵ: The maximum amount with which a dishonest party can bias the probability of the outcome away from uniform. Explicitly, $\epsilon = \max\{p_B^*, p_A^*\} - 1/2$.

These definitions are not complete in the sense that we have not yet specified what a cheating Alice or a cheating Bob are allowed to do, or of their capabilities. In this work, we study *information theoretic security* meaning that Alice and Bob are only bounded by the laws of quantum mechanics. For example, they are not bounded by polynomial-time quantum computations. In addition to this, we study the security in the *device-independent* regime where we assume Alice and Bob have complete control over the quantum devices when they decide to "cheat".

When studying device-independent (DI) protocols, one should first consider whether or not there are decent classical protocols (since these are not affected by the DI assumption). Indeed, Kitaev [Kit03] proved that any classical WCF protocol has bias $\epsilon = 1/2$, which is the worst possible value. Thus, it makes sense to study quantum WCF

protocols in the DI setting, especially if one with bias $\epsilon < 1/2$ can be found. Indeed, Silman, Chailloux, Aharon, Kerenidis, Pironio, and Massar presented a protocol in [SCA+11] which has bias $\epsilon \approx 0.33664$.

In this work, we provide two techniques which can be applied to a wide range of protocols (including [SCA+11], mentioned above) which can improve the bias. To illustrate our ideas, we now present the protocol in [SCA+11].

Protocol 1 (Protocol $\mathcal{I}$; a DI-WCF protocol with $p_A^* = \cos^2 \pi/8$ and $p_B^* = 3/4$ [SCA+11]; see also Section 2.2)**.** *Alice has one box and Bob has two boxes. Each box takes one binary input and gives one binary output and are designed to play the optimal GHZ game strategy. (Who creates and distributes the boxes is not important in the DI setting.)*

1. *Alice chooses a uniformly random input to her box $x \in \{0,1\}$ and obtains the outcome a. She chooses another uniformly random bit $r \in \{0,1\}$ and computes $s = a \oplus (x \cdot r)$. She sends s to Bob.*
2. *Bob chooses a uniformly random bit $g \in \{0,1\}$ and sends it to Alice. (We may think of g as Bob's "guess" for the value of x.)*
3. *Alice sends x and a to Bob. They both compute the output $c = x \oplus g$. This is the outcome of the protocol assuming neither Alice nor Bob abort.*
4. *Bob now tests to see if Alice was honest using the following two tests.*

   Test 1*: Bob sees if $s = a$ or $s = a \oplus x$. If this is not the case, he knows Alice cheated and aborts.*

   Test 2*: Bob chooses a uniformly random bit $y \in \{0,1\}$ and computes $z = x \oplus y \oplus 1$. He inputs y and z into his two boxes and obtains respective outcomes b and c. He aborts if (a, b, c, x, y, z) does not satisfy the winning conditions of the GHZ game.*
5. *If Bob does not abort, they both accept the value of c as the outcome of the protocol.*

To obtain a bias $\epsilon \leq 0.33664$ using the protocol above, they compose the protocol many times.

Remark. Note that the above protocol is actually a strong coin-flipping protocol, but as such, we can always treat it as a WCF protocol.

In this work, we build on this protocol using two techniques, which we discuss next.

1.1 Our main result

We now state the main result of our work.

Theorem 2. *There exists device-independent weak coin flipping protocols with bias approaching $\epsilon \approx 0.3148$. Under a convergence assumption, the bias can be lowered to $\epsilon \approx 0.29104$.*

Author's note: *We believe the convergence assumption above to be true, we just did not have enough time to rigourously prove it before the QCRYPT submission deadline. Hence, we state it as an assumption. However, it does lead to a better bias, and thus we have decided to state it here.*

We now discuss the proof of our main theorem, above. The first step is that we turn the strong coin flipping protocol into a weak coin flipping protocol in a routine manner. Basically, since weak coin flipping has the notion of a "winner" (if $c = 0$ Alice wins and if $c = 1$ Bob wins) we have the person who does not win do the testing. We illustrate this new protocol, below.

Protocol 3 (Weak version of Protocol 1)**.** *Alice has one box and Bob has two boxes. Each box takes one binary input and gives one binary output and are designed to play the optimal GHZ game strategy. (Who creates and distributes the boxes is not important in the DI setting.)*

1. *Alice chooses a uniformly random input to her box $x \in \{0,1\}$ and obtains the outcome a. She chooses another uniformly random bit $r \in \{0,1\}$ and computes $s = a \oplus (x \cdot r)$. She sends s to Bob.*

2. *Bob chooses a uniformly random bit $g \in \{0,1\}$ and sends it to Alice. (We may think of g as Bob's "guess" for the value of x.)*

3. *Alice sends x to Bob. They both compute the output $c = x \oplus g$. This is the outcome of the protocol assuming neither Alice nor Bob abort.*

4. *Test rounds:*

   (a) *If $x \oplus g = 0$:*
   Alice sends a to Bob.
   Bob tests if $s = a$ or $s = a \oplus x$. If the test fails, he aborts. Bob chooses $y, z \in_R \{0,1\}$ such that $x \oplus y \oplus z = 1$ and then performs a GHZ using x, y, z as the inputs and a, b, c as the output from the three boxes. He aborts if this test fails.

   (b) *Else, if $x \oplus g = 1$:*

      i. *Alice chooses $y, z \in_R \{0,1\}$ s.t. $x \oplus y \oplus z = 1$ and sends them to Bob.*

      ii. *Bob inputs y, z into his boxes, obtains and sends b, c to Alice.*

   Alice tests if x, y, z as inputs and a, b, c as outputs, satisfy the GHZ test. She aborts if this test fails.

5. *If Alice and Bob do not abort, they both accept the value of c as the outcome of the protocol.*

We now add a pre-processing step to this protocol which *self-tests* the boxes Alice and Bob share. This concept is not new, but it applies well to this setting.

We also use the revised protocol in a new way of composing protocols which we subbed an *abort-phobic* composition.

1.2 First technique: Self-testing

In Protocols 1 and 3, a cheating party may control what measurement is performed in the boxes of other party and how the state of the boxes is correlated to its own quantum memory. This is more general than *device-dependent* protocols, where for instance, the measurements are known to the honest player. However, we employ the concept of self-testing to stop Bob (or Alice) from applying such a strategy. The case where Alice self-tests Bob is illustrated below.

Protocol 4 (Protocol $\mathcal{P}$, where Alice self-tests)**.** *Alice starts with n boxes, indexed from 1_1 to 1_n. Bob starts with $2n$ boxes, the first half indexed by 2_1 to 2_n and the last half indexed by 3_1 to 3_n. The triple of boxes $(1_i, 2_i, 3_i)$ is meant to play the optimal GHZ game strategy.*

1. *Alice selects a uniformly random index $i \in \{1, \ldots, n\}$ and asks Bob to send her all the boxes* except *those indexed by 2_i and 3_i.*

2. *Alice plays $n-1$ GHZ games using the $n-1$ triples of boxes she has, making sure she has a space-like separation between the boxes. (She has long arms.)*

3. *Alice aborts if* any *of the GHZ games lose. Otherwise, she announces to Bob that they can use the remaining boxes for Protocol 3.*

The idea is that if n is chosen large enough, then this forces a dishonest Bob to not tamper with the boxes too much. Indeed, this step already allows us to reduce the cheating probabilities.

Lemma 5 (Informal. See Lemma 17 for a formal statement)**.** *For protocol $\mathcal{P}$ where Alice self tests Bob (i.e. Protocol 4 above), the cheating probabilities, in the limit of large n, are*

$$p_A^* = \cos^2(\pi/8) \approx 0.85355 \quad \text{and} \quad p_B^* \approx 0.6667. \tag{1}$$

Recall that for Protocol 1 [SCA+11], $p_A^* = \cos^2(\pi/8)$ and $p_B^* = 3/4$. To prove this lemma, we have to dive into two technical concepts, which we briefly discuss below. Note that for device-dependent protocols, where Alice and Bob trust their devices, cheating probabilities can be cast as semidefinite programs [Kit03, Moc07]. In the DI regime, we cannot even bound the dimension of the state within the boxes, making it much harder to analyze and bound Alice and Bob's cheating probabilities.

Self-testing the GHZ game. We build on known self-testing results to prove that when Alice self-tests Bob and all $n-1$ rounds of GHZ tests succeed, then the remaining triple of boxes has to be approximately performing the optimal GHZ strategy. The differences between this approximation and the optimal strategy disappear in the limit of large n. See Section 6.1 for details.

Continuity of semidefinite programs. When Alice self-tests Bob, we are able to formulate Bob's cheating probability as a semidefinite program in the limit of large n. However, we cannot have a protocol with an infinite number of messages. Consequently, we study a family of protocols where Bob's cheating probabilities approach certain thresholds. Therefore, we need the semidefinite program values to capture the behaviour of the cheating probabilities as they approach the limit of large n. See Section 6.3 for details.

Remark. Both of these technical steps may find use in independent applications. In particular, the continuity of semidefinite programs section is written for general semidefinite programs for the most part.

1.3 Second technique: abort-phobic composition

It can happen, that for a given WCF protocol, $p_B^* \neq p_A^*$, in which case we say the protocol is polarised. It is known (e.g. [SCA+11]) that composing a polarised protocol with itself (or other protocols) can effectively reduce the bias. Our second improvement is a modified way of composing protocols, when there is a positive probability that the honest player catches the cheating player. Let us start by recalling the standard way of composing protocols.

Standard composition. For a protocol with cheating probabilities p_B^* and p_A^*, we say that it has polarity towards Alice when it satisfies $p_A^* > p_B^*$. Similarly, we say that it has polarity towards Bob when $p_B^* > p_A^*$. Given a polarized protocol $\mathcal{R}$, we may switch the roles of Alice and Bob since the definition of coin-flipping is symmetric. To make the polarity explicit, we define $\mathcal{R}_A$ to be the version of the protocol with $p_A^* > p_B^*$ and $\mathcal{R}_B$ to be the version with $p_B^* > p_A^*$. With this in mind, we can now define a simple composition.

Protocol 6 (Winner-gets-polarity composition)**.** *Alice and Bob agree on a protocol $\mathcal{R}$.*

1. *Alice and Bob perform protocol $\mathcal{R}$.*
2. *If Alice wins, she polarizes the second protocol towards herself, i.e., they now use the protocol $\mathcal{R}_A$ to determine the outcome of the (entire) protocol.*
3. *If Bob wins, he polarizes the second protocol towards himself, i.e., they now use the protocol $\mathcal{R}_B$ to determine the outcome of the (entire) protocol.*

The standard composition above is a decent way to balance the cheating probabilities of a protocol. For instance, if $\mathcal{R}$ has cheating probabilities p_A^* and p_B^* with $p_A^* > p_B^*$, then the composition gets to decide "who gets to be Alice" in the second run. We can easily compute Alice's cheating probability in the composition as

$$(p_A^*)^2 + (1 - p_A^*)p_B^* < p_A^* \tag{2}$$

and Bob's as

$$p_B^* p_A^* + (1 - p_B^*)p_B^* < p_A^*. \tag{3}$$

This does indeed reduce the bias since the maximum cheating probability is now smaller.

Abort-phobic composition. The "traditional" way of considering WCF protocols is to view them as only having two outcomes "Alice wins" (when $c = 0$) or "Bob wins" ($c = 1$). This is because Alice can declare herself the winner if she catches Bob cheating. Similarly, Bob can declare himself the winner if he catches Alice cheating. This is completely fine when we consider "one-shot" versions of these protocols, but we lose something when we compose them. For instance, in the simple composition used in Protocol 6, Bob should not really accept to continue onto the second protocol if he catches Alice cheating in the first. That is, if he knows Alice cheated, he can declare himself the winner of the entire protocol! In other words, the cheating probabilities (2) and (3) may get reduced even further. For purposes of this discussion, suppose Bob adopts a cheating strategy which has a probability v_B of him winning ($c = 1$), a probability v_A of him losing ($c = 0$), and a probability $v_\perp$ of Alice catching him cheating. Then his cheating probability in the (abort-phobic) version of the simple composition is now

$$v_B \cdot p_A^* + v_A \cdot p_B^* + v_\perp \cdot 0. \tag{4}$$

This quantity may be a strict improvement if $v_\perp > 0$ when $v_B = p_B^*$.

The concept of abort-phobic composition is simple. Alice and Bob keep using WCF protocols and the winner (at that round) gets to choose the polarity of the subsequent protocol. However, if either party *ever aborts*, then it is game over and the cheating player loses *the entire composition protocol*.

One may think it is tricky to analyze abort-phobic compositions, but we may do this one step at time. To this end, we introduce the concept of *cheat vectors*.

Definition 7 ($\mathbb{C}_A, \mathbb{C}_B$; Alice and Bob's cheat vectors)**.** Given a protocol $\mathcal{R}$, we say that $(v_A, v_B, v_\perp)$ is a cheat vector for (dishonest) Bob if there exists a cheating strategy where:

- v_B is the probability with which Alice accepts the outcome $c = 1$,
- v_A is the probability with which Alice accepts the outcome $c = 0$,
- $v_\perp$ is the probability with which Alice aborts.

We denote the set of cheat vectors for (dishonest) Bob by $\mathbb{C}_B(\mathcal{R})$. Cheat vectors for (dishonest) Alice and $\mathbb{C}_A(\mathcal{R})$ are analogously defined keeping the notation v_A for her winning, v_B for her losing, and $v_\perp$ for Bob aborting.

In this work, we show how to capture cheat vectors as the feasible region of a semidefinite program, from which we can optimize

$$v_B \cdot p_A^* + v_A \cdot p_B^* + v_\perp \cdot 0. \tag{5}$$

For this to work, we assume we have p_A^* and p_B^* for the protocol that comes in the second round. The neat thing is that once we solve for the optimal cheating probabilities in the abort-phobic composition in this way, we can then fix those probabilities and compose again! In other words, we are recursively composing the abort-phobic composition. Therefore, we calculate the cheating probabilities from the *bottom-up*.

By using protocols where Alice self-tests and abort-phobic compositions, we are able to find protocols which converge onto a bias of $\epsilon \approx 0.3148$ proving the main result of this work. We give more details below. By composing protocols where Alice self-tests with ones where Bob self-tests, we are able to reduce the bias further[1] to $\epsilon \approx 0.29104$, as also stated in our main theorem.

1.4 Applications

The concept of polarity extends beyond finding WCF protocols and, as such, the "winner-gets-polarity" concept allows for WCF to be used in other compositions. Indeed, we can use it to balance the cheating probabilities in *any* polarized protocol for any symmetric two-party cryptographic task for which such notions can be properly defined.

For instance, many *strong* coin-flipping protocols can be thought of as polarized. For an example, the protocol **??** is indeed a polarized strong coin-flipping protocol. Thus, by balancing the cheating probabilities of that protocol using our DI WCF protocol, we get the following theorem.

Theorem 8. *There exists DI strong coin-flipping protocols where no party can cheat with probability greater than* 0.33439. *Under a continuity assumption, the bound can be lowered to* 0.33192.

To contrast, for [SCA+11], the bound on cheating probabilities was 0.336637. There are likely more examples of protocols which can be balanced in a DI way using this idea.

[1] Under our convergence assumption.

1.5 Paper Organisation

We begin with clarifying the connection between the language of "boxes" and physical implementation of protocols described using them and state known relevant results (Section 2??). In Section 3??, we define and state the security of protocols $\mathcal{P}$ (where Alice self-tests; Protocol 4) and $\mathcal{Q}$ (where Bob self-tests; ????). In Section 4??, we formally define standard compositions, C^{LL} and abort-phobic compositions, $C^{\perp L}$ and $C^{L\perp}$. For instance, given a protocol $\mathcal{R}$, $C^{LL}(\mathcal{R})$ means one applies standard composition multiple times on $\mathcal{R}$. The meaning of L and $\perp$ is also clarified there. Using these, below we summarise the build-up to our main results. In Section 5??, we give the security proofs in the asymptotic limit and in Section 6??? we argue why we expect the appropriate notion of continuity to hold.

Section 3 | Protocols

- We show that $p_A^*(\mathcal{P}) \approx 0.853\ldots$ and $p_B^*(\mathcal{P}) \approx 0.667\ldots$ (see Lemma **??**; informally stated above). We also show that the set of cheat vectors $\mathbb{C}_B(\mathcal{P})$ can be cast as an SDP.
- Next, we show, $p_A^*(\mathcal{Q}) = p_A^*(\mathcal{I})$ and $p_B^*(\mathcal{Q}) = p_B^*(\mathcal{I})$ so the advantage is not immediate. However, now $\mathbb{C}_A(\mathcal{Q})$ can be cast as a semidefinite program which, as we shall see, yields an advantage when $\mathcal{Q}$ is composed.

Section 4 | Compositions

Let $\epsilon(\mathcal{R})$ denote the bias of protocol $\mathcal{R}$.

- The bias of protocol $\mathcal{I}$ under the standard composition is given by

  $$\epsilon(C^{LL}(\mathcal{I})) \approx 0.33664$$

  while for our improved protocol $\mathcal{P}$, it is given by

  $$\epsilon(C^{LL}(\mathcal{P})) \approx 0.3199. \tag{6}$$

The standard composition does not yield any improvement for $\mathcal{Q}$ because the cheating probabilities are identical to those of $\mathcal{I}$. We can extract an advantage by using abort-phobic compositions.

- Composing the protocol $\mathcal{P}$ with itself many times, using the abort-phobic compositions gives a bias

  $$\epsilon(C^{\perp L}(\mathcal{P})) \approx 0.3148$$

  which is a further improvement.

- Composing the protocol $\mathcal{Q}$ with itself many times, using the abort-phobic composition gives a bias

  $$\epsilon(C^{L\perp}(\mathcal{Q})) \approx 0.3226$$

  which is worse than Equation (6).

- However, when we compose the protocol $\mathcal{Q}$ many times with itself, followed by protocol $\mathcal{P}$, we find

  $$\epsilon(C^{L\perp}(\mathcal{Q}, \mathcal{Q}, \ldots, \mathcal{Q}, \mathcal{P})) \approx 0.29104\ldots$$

  where we use the same composition technique except that at the last "level" we use[2] $\mathcal{P}$ instead of $\mathcal{Q}$.

(Jamie: this notation needs to be explained since it's not clear what is above and what is in the footnote.)

2 Device Independent Weak Coin Flipping protocols | State Of The Art

In the following, we first discuss how one can describe DI WCF protocols in terms of the players exchanging "boxes"—devices which take classical inputs and give classical outputs. This discussion mostly formalises rather obvious notions in the interest of clarity and parts of it may be appropriately skipped. Subsequently we give some details about Protocol 1 [SCA+11] upon which we build our results.

[2] $C^{\perp L}(\mathcal{P}, \mathcal{P}, \ldots, \mathcal{P}, \mathcal{Q})$ is strictly worse than considering $C^{\perp L}(\mathcal{P}, \mathcal{P}, \ldots, \mathcal{P}, \mathcal{P})$; this should become evident later.

2.1 Device Independence and the Box Paradigm

We describe device independent protocols as classical protocols with one modification: we assume that the two parties can exchange boxes and that the parties can shield their boxes (from the other boxes i.e. the boxes can't communicate with each other once shielded).

Definition 9 (Box)**.** A *box* is a device that takes an input $x \in \mathcal{X}$ and yields an outputs $a \in \mathcal{A}$ where $\mathcal{X}$ and $\mathcal{A}$ are finite sets. Typically, a set of n boxes, taking inputs $x_1, x_2, \ldots x_n$ and yielding outputs $a_1, a_2 \ldots a_n$ are *characterised* by a joint conditional probability distribution, denoted by

$$p(a_1, a_2 \ldots a_n | x_1, x_2 \ldots x_n).$$

Further, if $p(a_1, a_2 \ldots a_n | x_1, x_2 \ldots x_n) = \text{tr}\left[M^1_{a_1|x_1} \otimes M^2_{a_2|x_2} \cdots \otimes M^n_{a_n|x_n} \rho\right]$ then we call the set of boxes, *quantum boxes*, where $\{M^i_{a'|x'}\}_{a' \in \mathcal{A}_i}$ constitute a POVM for a fixed i and x', ρ is a density matrix and their dimensions are mutually consistent.

Henceforth, we restrict ourselves to quantum boxes.

Definition 10 (Protocol in the box formalism)**.** A generic two-party protocol in the box formalism has the following form:

1. Inputs:
   (a) Alice is given boxes $\square^A_1, \square^A_2 \ldots \square^A_p$ and Bob is given boxes $\square^B_1, \square^B_2, \ldots \square^B_q$.
   (b) Alice is given a random string r^A and Bob is given a random string r^B (of arbitrary but finite length).
2. Structure: At each round of the protocol, the following is allowed.
   (a) Alice and Bob can locally perform arbitrary but finite time computations on a Turing Machine.
   (b) They can exchange classical strings/messages and boxes.

A protocol in the box formalism is readily expressed as a protocol which uses a (trusted) classical channel (i.e. they trust their classical devices to reliably send/receive messages), untrusted quantum devices and an untrusted quantum channel (i.e. a channel that can carry quantum states but may be controlled by the adversary).

Assumption 11 (Setup of Device Independent Two-Party Protocols)**.** *Alice and Bob*

1. *both have private sources of randomness,*
2. *can send and receive classical messages over a (trusted) classical channel,*
3. *can prevent parts of their untrusted quantum devices from communicating with each other, and*
4. *have access to an untrusted quantum channel.*

We restrict ourselves to a "measure and exchange" class of protocols—protocols where Alice and Bob start with some pre-prepared states and subsequently, only perform classical computation and quantum measurements locally in conjunction with exchanging classical and quantum messages. More precisely, we consider the following (likely restricted) class of device independent protocols.

Definition 12 (Measure and Exchange (Device Independent Two-Party) Protocols)**.** A *measure and exchange (device independent two-party) protocol* has the following form:

1. Inputs:
   (a) Alice is given quantum registers $A_1, A_2, \ldots A_p$ together with POVMs[3]

$$\{M^{A_1}_{a|x}\}_a, \{M^{A_2}_{a|x}\}_a, \ldots \{M^{A_p}_{a|x}\}_a$$

[3]For concreteness, take the case of binary measurements. By $\{M^{A_1}_{a|x}\}_a$, for instance, we mean $\{M^{A_1}_{0|x}, M^{A_1}_{1|x}\}$ is a POVM for $x \in \{0, 1\}$.

which act on them and Bob is, analogously, given quantum registers $B_1, B_2, \ldots B_q$ together with POVMs

$$\{M_{b|y}^{B_1}\}_b, \{M_{b|y}^{B_2}\}_b, \ldots, \{M_{b|y}^{B_q}\}_b.$$

Alice shields $A_1, A_2, \ldots A_p$ (and the POVMs) from each other and from Bob's lab. Bob similarly shields $B_1, B_2 \ldots B_q$ (and the POVMs) from each other and from Alice's lab.

(b) Alice is given a random string r^A and Bob is given a random string r^B (of arbitrary but finite length).

2. Structure: At each round of the protocol, the following is allowed.

(a) Alice and Bob can locally perform arbitrary but finite time computations on a Turing Machine.

(b) They can exchange classical strings/messages.

(c) Alice (for instance) can

i. send a register A_l and the encoding of her POVMs $\{M_i^{A_l}\}_i$ to Bob, or

ii. receive a register B_m and the encoding of the POVMs $\{M_i^{B_m}\}_i$.

Analogously for Bob.

It is clear that a protocol in the box formalism (Definition 10) which uses only quantum boxes (Definition 9) can be implemented as a measure and exchange protocol (Definition 12).

2.2 Security of Protocol 1

We defined Protocol 1 in the introduction and it referred to a GHZ test informally. We briefly remind the reader of this test and setup some notation we intend to use.

Definition 13. Suppose we are given three boxes, $\square^A, \square^B$ and $\square^C$, which accept binary inputs $a, b, c \in \{0, 1\}$ and produces binary output $x, y, z \in \{0, 1\}$ respectively. The boxes pass the GHZ test if $a \oplus b \oplus c = xyz \oplus 1$, given the inputs satisfy $x \oplus y \oplus z = 1$.

It is known that no classical triple of boxes can pass the GHZ test with certainty but quantum boxes can. **(Tom: Add self-testing)**

Claim 14. Quantum boxes pass the GHZ test with certainty (even if they cannot communicate), for the state $|\psi\rangle_{ABC} = \frac{|000\rangle_{ABC}+|111\rangle_{ABC}}{\sqrt{2}}$, and measurement[4] $\frac{\sigma_x+\mathbb{I}}{2}$ for input 0 and $\frac{\sigma_y+\mathbb{I}}{2}$ for input 1 (in the notation introduced earlier, $M_{0|0}^A = |+\rangle\langle+|, M_{1|0}^A = |-\rangle\langle-|$ and so on, where $|\pm\rangle = \frac{|0\rangle\pm|1\rangle}{\sqrt{2}}$).

The proof is easier to see in the case where the outcomes are ± 1; it follows from the observations that $\sigma_y \otimes \sigma_y \otimes \sigma_y |\psi\rangle = -|\psi\rangle$, $\sigma_x \otimes \sigma_x \otimes \sigma_x |\psi\rangle = |\psi\rangle$ and the anti-commutation of σ_x and σ_y matrices, i.e. $\sigma_x\sigma_y + \sigma_y\sigma_x = 0$.

While [SCA+11] do not rely on the self-testing property of the GHZ test for their analysis of Protocol 1, our work does. We discuss self-testing in Section 5 and Section 6.

We now discuss the correctness and soundness of Protocol 1. From Claim 14, it is clear that when both players follow the protocol using GHZ boxes (Definition 13), Bob never aborts and they win with equal probabilities. As for the security, [SCA+11] proved the following.

Lemma 15 (Security of SCF). *[SCA+11] Let $\mathcal{I}$ denote the protocol corresponding to Protocol 1. Then, the success probability of cheating Bob, $p_B^*(\mathcal{I}) \leq \frac{3}{4}$ and that of cheating Alice, $p_A^*(\mathcal{I}) \leq \cos^2(\pi/8)$. Further, both bounds are saturated by a quantum strategy which uses a GHZ state and the honest player measures along the σ_x/σ_y basis corresponding to input 0/1 into the box. Cheating Alice measures along $\sigma_{\hat{n}}$ for $\hat{n} = \frac{1}{\sqrt{2}}(\hat{x} + \hat{y})$ while cheating Bob measures his first box along σ_x and second along σ_y.*

Note that both players can cheat maximally assuming they share a GHZ state and the honest player measures along the associated basis. This entails that even though the cheating player could potentially tamper with the boxes before handing them to the honest player, surprisingly, exploiting this freedom does not offer any advantage to the cheating player.

[4] we added the identity so that the eigenvalues associated become 0, 1 instead of −1, 1.

3 First Technique: Self-testing (single shot, unbalanced)

While we already motivated the idea in the introduction, we state it with some more detail. We make two observations.

First, in Protocol 1 only Bob performs the test round. In WCF, there is a notion of Alice winning and Bob winning. Thus, if $x \oplus g = 0$, i.e. the outcome corresponding to "Alice wins", we can imagine that Bob continues to perform the test to ensure (at least to some extend) that Alice did not cheat. However, if $x \oplus g = 1$, i.e. the outcome corresponding to "Bob wins", we can require Alice to now complete the GHZ test to ensure that Bob did not cheat. It turns out that this does not lower p_B^*. Interestingly, the best cheating strategy deviates from the GHZ state and measurements for the honest player. We omit the details here but mention it anyway to motivate the following.

Second, Alice (say) can harness the self-testing property of GHZ states and measurements to ensure that Bob has not tampered with her boxes. Note that no such scheme can be concocted which simultaneously self-tests Alice and Bob's boxes. More precisely, no such procedure can ensure that Alice and Bob share a GHZ state (Alice one part, Bob the other two, for instance) because this would mean perfect (or near perfect) SCF is possible which is forbidden even in the device dependent case. Kitaev [Kit03] showed that for any SCF protocol, $\epsilon \geq \frac{1}{\sqrt{2}} - \frac{1}{2}$.

Combining these two observations, results in an improvement in the security for Alice. We obtain a protocol with $P_A^* = 3/4$, which is the same as before, but $P_B^* \approx 0.6667$ which is lower than 0.75, that of protocol 1.

3.1 Cheat Vectors

As alluded to in Section 1.5, using cheat vectors, it is sometimes possible to compose protocols and obtain a lower bias compared to protocols which are composed without using cheat vectors. We describe such procedures in the next section, Section 4. Here, we simply define cheat vectors and show that self-testing allows one to express relevant optimisation problems over cheat vectors as semi definite programmes.

Definition 16 ($\mathbb{C}_A, \mathbb{C}_B$)**.** Given a protocol $\mathcal{I}$, denote by $\mathbb{C}_B(\mathcal{I})$ the set of *cheat vectors* for Bob, which is defined as follows :

$$\mathbb{C}_B(\mathcal{I}) := \{(v_A, v_B, v_\perp) : \exists \text{ a strategy of } B \text{ s.t. an honest } A \text{ outputs 0, 1, and } \perp \text{ with probabilities } v_A, v_B \text{ and } v_\perp\}$$

and analogously, denote by $\mathbb{C}_A(\mathcal{I})$ the set of cheat vectors for Alice.

3.2 Alice self-tests | Protocol $\mathcal{P}$

We begin with the case where Alice self-tests. In the honest implementation, the *trio* of boxes used in the following are characterised by the GHZ setup (see Claim 14).

(Tom: I'm going to remove the repetition of Protocol 1**)** **(Atul:** I was also convinced that's a good idea but now that there are bits and pieces of different protocols in the introduction, I think it'll be much easier to read if in the main sections, we restate things (we should obviously say, "restated" or something)**)** **(Tom:** Could we have separate number of protocols, theorems etc ?**)**

(Tom: something about assuming perfect self-testing**)**

Lemma 17. *Let $\mathcal{P}$ denote the protocol corresponding to Protocol 4. Then Alice's cheating probability $p_A^*(\mathcal{P}) \leq \cos^2(\pi/8) \approx 0.852$. Further, let $c_0, c_1, c_\perp \in \mathbb{R}$, and $\mathbb{C}_B(\mathcal{P})$ be the set of cheat vectors for Bob. Then, as $N \to \infty$, the solution to the optimisation problem $\max(c_0\alpha + c_1\beta + c_\perp\gamma)$ over $\mathbb{C}_B(\mathcal{Q})$ approaches that of a semi definite programme. In particular, i.e. for $c_0 = c_\perp = 0$ and $c_1 = 1$, $p_B^*(\mathcal{P}) \approx 0.667\ldots$ (in the limit).*

We defer the proof to Section **??**. The value for $p_B^*(\mathcal{P})$ was obtained by numerically solving the corresponding semi definite programme while the analysis for cheating Alice is the same as that of the original protocol.

3.3 Bob self-tests | Protocol $\mathcal{Q}$

What if we modified the protocol and had Bob self-test his boxes? Does that yield a better protocol? We address the first question now and the second in the subsequent section.

Protocol 18 (Bob self-tests his boxes)**.** *Proceed exactly as in Protocol 4, except for the self-testing where the rolls of Alice and Bob are reversed. More explicitly, suppose there are N trios of boxes; Alice has the first part and Bob has the remaining two parts, of each trio.*

1. *Bob selects a number $i \in_R \{1, 2 \dots N\}$ and sends it to Alice.*
2. *Alice sends her part of the trio of boxes corresponding to $\{1, 2 \dots N\} \backslash i$, i.e. she sends all the boxes, except the ones corresponding to the trio i.*
3. *Bob performs a GHZ test on all the trios labelled $\{1, 2 \dots N\} \backslash i$, i.e. all the trios except the ith.*

Henceforth, proceed as in Protocol 4 after the self-testing step.

As already indicated in Section 1.5, we don't expect the cheating probabilities to improve but we do expect an SDP characterisation of Alice's cheat vectors.

Lemma 19. *Let $\mathcal{Q}$ denote the protocol corresponding to Protocol 18. Then, Alice's cheating probability, $p^*_A(\mathcal{Q}) \leq 3/4$ and Bob's cheating probability, $p^*_B(\mathcal{Q}) \leq \cos^2(\pi/8)$ (which are the same as those in Lemma 15). Further, let $c_0, c_1, c_\perp \in \mathbb{R}$, and $\mathbb{C}_A(\mathcal{Q})$ be the set of cheat vectors for Alice. Then, as $N \longrightarrow \infty$, the solution to the optimisation problem $\max(c_0\alpha + c_1\beta + c_\perp\gamma)$ over $(\alpha, \beta, \gamma) \in \mathbb{C}_A(\mathcal{Q})$ approaches that of a semi definite programme.*

The proof is again deferred; see Section 5.2.

4 Second Technique: Abort-phobic composition

In this section, we use the convention that $\mathcal{I}, \mathcal{P}$ and $\mathcal{Q}$ correspond to the protocols described in Protocol **??**, Protocol 4 and Protocol 18, respectively. Notice that $p^*_A(\mathcal{X}) > p^*_B(\mathcal{X})$ where $\mathcal{X} \in \{\mathcal{I}, \mathcal{P}, \mathcal{Q}\}$. We call such protocols "unbalanced". In this section we start from unbalanced WCF protocols and compose them to construct balanced WCF protocols. To this end, we introduce some notation and the term "polarity", to capture which among A and B is favoured. **(Tom:** Definition is a repetition, so I will reference to the introduction instead**)**

Definition 20 (Unbalanced protocols, Polarity)**.** Given a WCF protocol $\mathcal{X}$, we say that it is unbalanced if $p^*_A(\mathcal{X}) \neq p^*_B(\mathcal{X})$. We say that $\mathcal{X}$ has polarity A if $p^*_A(\mathcal{X}) > p^*_B(\mathcal{X})$ and polarity B if $p^*_A(\mathcal{X}) < p^*_B(\mathcal{X})$.

Finally, let $X, Y \in \{A, B\}$ be distinct and suppose that $\mathcal{R}$ is an unbalanced protocol. Then, we define $\mathcal{R}_X$ to be protocol $\mathcal{R}$ where Alice's and Bob's roles are possibly interchanged so that $\mathcal{R}_X$ has polarity X, i.e. $p^*_X(\mathcal{R}_X) > p^*_Y(\mathcal{R}_X)$. We refer to $\mathcal{R}_X$ as $\mathcal{R}$ polarised towards X.

We now describe how these protocols can be composed such that the "winner gets polarity".

4.1 Composition

Definition 21 ($C(\cdot,\cdot)$ and $C(\cdot)$)**.** Given two unbalanced WCF protocols, $\mathcal{X}$ and $\mathcal{Y}$, let $\mathcal{X}_A, \mathcal{X}_B$ and $\mathcal{Y}_A, \mathcal{Y}_B$ be their polarisations (see Definition 20). Define $C(\mathcal{X}, \mathcal{Y})$ as follows:

1. Alice and Bob execute $\mathcal{X}_A$ and obtain outcome $X \in \{A, B, \perp\}$.
2. If
   (a) $X = A$, execute $\mathcal{Y}_A$ and obtain outcome $Y \in \{A, B, \perp\}$, else if
   (b) $X = B$, execute $\mathcal{Y}_B$ and obtain outcome $Y \in \{A, B, \perp\}$, and finally if
   (c) $X = \perp$, set $Y = \perp$.

   Output Y.

Let $\mathcal{Z}^{i+1} := C(\mathcal{X}, \mathcal{Z}^i)$ for $i \geq 1$, and $\mathcal{Z}^1 := \mathcal{X}$. Then, formally, define $C(\mathcal{X}) := \lim_{i\to\infty} \mathcal{Z}^i$.[5]

[5] This is just to facilitate notation. This way the cheating probabilities p^*_A and p^*_B converge and numerically this only takes a few compositions to reach in our case.

The study of such composed protocols is simplified by assuming that in an honest run, neither player outputs $\perp$ (abort), i.e. they either output A or B. We take a moment to explain this.

Consider any protocol $\mathcal{R}$ where, when both players are honest, the probability of abort is zero. The protocols we have described so far, satisfy this property, so long as we assume that honest players can prepare perfect GHZ boxes. Such protocols are readily converted into protocols where an honest player never outputs abort.

For instance, suppose that in the execution of the aforementioned protocol $\mathcal{R}$ (with no-honest-abort), Alice behaves honestly but Bob is malicious. Suppose after interacting with Bob, Alice reaches the conclusion that she must abort. Since she knows that if Bob was honest, the outcome abort could not have arisen, she concludes that Bob is cheating and declares herself the winner, i.e. she outputs A. Similarly, when Bob is honest and after the interaction, reaches the outcome abort, he knows Alice cheated and can therefore declare himself the winner, i.e. output B.

Whenever we modify a protocol so that an honest Alice (Bob) replaces the outcome abort with Alice (Bob) winning, we say Alice (Bob) is *lenient*. This is motivated by the fact that when we compose protocols, if Alice can conclude that Bob is cheating, and she still outputs A instead of aborting, she is giving Bob further opportunity to cheat—she is being lenient.

Definition 22 ($\mathcal{R}$ with lenient players)**.** Suppose $\mathcal{R}$ is a WCF protocol such that when both players are honest, the probability of outcome abort, $\perp$, is zero. Then by "$\mathcal{R}$ *with lenient Alice (Bob)*", which we denote by $\mathcal{R}^{L\perp}$ ($\mathcal{R}^{\perp L}$), we mean that Alice (Bob) follows $\mathcal{R}$ except that the outcome $\perp$ replaced with A (B). Finally, by "*lenient* $\mathcal{R}$", which we denote by $\mathcal{R}^{LL}$, we mean $\mathcal{R}$ with lenient Alice and Bob.

For clarity and conciseness, we define C^{LL} to be compositions with lenient variants of the given protocols. We work out some examples of such protocols and analyse their security in the following section. These can be improved by considering $C^{L\perp}$ and $C^{\perp L}$—compositions where only one player is lenient. We discuss those afterwards.

Definition 23 (C^{LL}, $C^{\perp L}$ and $C^{L\perp}$)**.** Suppose a WCF protocol $\mathcal{X}$ can be transformed into its *lenient* variants (see Definition 22). Then define

$$\begin{aligned} C^{LL}(\mathcal{X},\mathcal{Y}) &:= C(\mathcal{X}^{LL},\mathcal{Y}),\\ C^{\perp L}(\mathcal{X},\mathcal{Y}) &:= C(\mathcal{X}^{\perp L},\mathcal{Y}), \quad \text{and}\\ C^{L\perp}(\mathcal{X},\mathcal{Y}) &:= C(\mathcal{X}^{L\perp},\mathcal{Y}). \end{aligned}$$

In words, C^{LL} is referred to as a *standard* composition, while $C^{\perp L}$ and $C^{L\perp}$ are referred to as *abort-phobic* compositions.

4.2 Standard Composition | C^{LL}

We begin with the simplest case, standard composition, C^{LL}. Let us take an example. Consider protocol $\mathcal{P}$ (see Protocol 4) and recall (see Lemma 17)

$$\begin{aligned} p^*_A(\mathcal{P}_A) &=: \alpha \approx 0.852\ldots,\\ p^*_B(\mathcal{P}_A) &=: \beta \approx 0.667\ldots. \end{aligned}$$

Note that therefore $p^*_A(\mathcal{P}_B) = \beta$ and $p^*_B(\mathcal{P}_B) = \alpha$. Further, let $\mathcal{P}' := C^{LL}(\mathcal{P},\mathcal{P})$, i.e. Alice and Bob (who are both lenient) first execute $\mathcal{P}_A$ and if the outcome is A, they execute $\mathcal{P}_A$, while if the outcome is B, they execute $\mathcal{P}_B$. This is illustrated in Figure **??** where note that the event abort doesn't appear due to the leniency assumption. This allows us to evaluate the cheating probabilities for the resulting protocol as

$$\begin{aligned} p^*_A(\mathcal{P}') &= \alpha\alpha + (1-\alpha)\beta =: \alpha^{(1)}, \quad \text{and}\\ p^*_B(\mathcal{P}') &= \beta\alpha + (1-\beta)\beta =: \beta^{(1)}. \end{aligned} \tag{7}$$

To see this, consider Equation (7). Alice knows that if she wins the first round, her probability of winning is $\alpha > \beta$. She knows that in the first round, she can force the outcome A with probability α. From leniency, she knows that Bob would output B with the remaining probability.[6]

[6]Without leniency, this probability could have been shared between the outcomes $\perp$ (abort) and B. Consequently, only upper bounds could be obtained on $p^*_A(\mathcal{P}')$ and $p^*_B(\mathcal{P}')$ using only α and β as security guarantees for $\mathcal{P}_A$. Upper bounds, however, would not be enough to determine the polarity of $\mathcal{P}'$ and an stymie unambiguous repetition of the composition procedure (at least as it is defined). One could nevertheless compose by hoping that the upper bounds can be used to accurately represent the polarity. This would still yield a protocol and the same calculation would yield correct bounds but the composition itself might be sub-optimal.

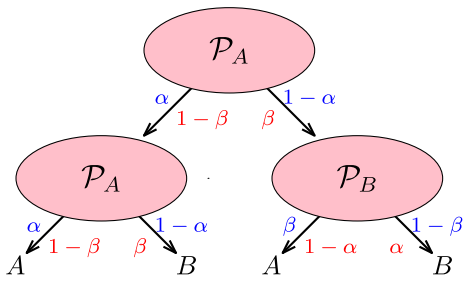

Figure 1: Standard composition of Coin flipping protocols. Subprotocols only have two outcomes depending on the coin flip. Labels indicate probabilities of outcomes for cheating Alice (blue) and cheating Bob (red)

A side remark: one consequence of this simplified analysis is that[7] $\alpha^{(1)} > \beta^{(1)}$. Intuitively, it means that plority is preserved by the composition procedure. Proceeding similarly, i.e. defining $\mathcal{P}'' := C^{LL}(\mathcal{P}, \mathcal{P}')$, and repeating $k+1$ times overall, one obtains[8]

$$\alpha^{(k+1)} = \alpha\alpha^{(k)} + (1-\alpha)\beta^{(k)}$$
$$\beta^{(k+1)} = \beta\alpha^{(k)} + (1-\beta)\beta^{(k)}.$$

In the limit of $k \to \infty$, one obtains

$$p_A^*(C^{LL}(\mathcal{P})) = p_B^*(C^{LL}(\mathcal{P})) = \lim_{k\to\infty} \alpha^{(k)} = \lim_{k\to\infty} \beta^{(k)} \approx 0.8199\ldots.$$

Proceeding similarly, one obtains for $X \in \{A, B\}$ and $\mathcal{X} \in \{\mathcal{I}, \mathcal{Q}\}$,

$$p_X^*(C^{LL}(\mathcal{X})) \approx 0.836\ldots$$

We thus have the following.

Theorem 24. *Let $X \in \{A, B\}$ and $\mathcal{X} \in \{\mathcal{I}, \mathcal{Q}\}$. Then $p_X^*(C^{LL}(\mathcal{P})) \approx 0.8199\ldots$ and $p_X^*(C^{LL}(\mathcal{X})) \approx 0.836\ldots$.*

4.3 Abort Phobic Compositions | $C^{L\perp}, C^{\perp L}$

We now look at the case of abort phobic compositions, $C^{L\perp}$ and $C^{\perp L}$. We work through essentially the same example as above and see what changes in this setting. Consider protocol $\mathcal{P}$ (see ...) and recall that as before

$$p_A^*(\mathcal{P}_A) =: \alpha \approx 0.852\ldots,$$
$$p_B^*(\mathcal{P}_A) =: \beta \approx 0.667\ldots.$$

In addition, we know from Lemma 17 that cheat vectors for Bob, $(\alpha, \beta, \gamma) \in \mathbb{C}_B(\mathcal{P}_A)$ admit a nice characterisation courtesy of the self testing step. Let $\mathcal{P}' := C^{\perp L}(\mathcal{P}, \mathcal{P})$, i.e. Alice and Bob execute $\mathcal{P}_A$ and if the outcome is A, they execute $\mathcal{P}_A$ while if the outcome is B, they execute $\mathcal{P}_B$. Bob is assumed to be lenient so an honest Bob never outputs abort, $\perp$. However, an honest Alice can output abort, $\perp$ so we keep that output in the illustration, Lemma 17. Our goal is to find $p_A^*(\mathcal{P}')$ and $p_B^*(\mathcal{P}')$. The former is the same as before because Bob is lenient:

$$p_A^*(\mathcal{P}') = \alpha \cdot \alpha + (1-\alpha) \cdot \beta.$$

Clearly, $p_B^*(\mathcal{P}') \leq \beta\alpha + (1-\beta)\beta$ but this bound may not be tight because $(1-\beta)$ is the combined probability of Alice aborting and Alice outputting A. However, we can use cheat vectors to obtain

$$p_B^*(\mathcal{P}') = \max_{(v_A, v_B, v_\perp) \in \mathbb{C}_B} v_B\alpha + v_A\beta$$

[7] $\alpha^{(1)} - \beta^{(1)} = (\alpha-\beta)\alpha - (\alpha-\beta)\beta = (\alpha-\beta)^2 > 0$

[8] Again, note that $\alpha^{(k+1)} - \beta^{(k+1)} = (\alpha^{(k)} - \beta^{(k)})(\alpha - \beta) > 0$.

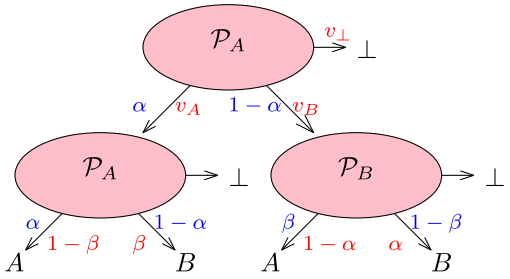

Figure 2: Abort phobic compositing for Coin flipping protocols. Subprotocols have three possible outcomes including an abort symbol. Aborting in any subprotocol directly leads to aborting the whole protocol. Labels indicate probabilities of outcomes for cheating Alice (blue) and cheating Bob (red). In the security analysis of cheating Bob, we need to optimise over the cheat vectors $(v_A, v_B, v_\perp) \in \mathbb{C}_B$.

which is an SDP one can solve numerically. Unlike the previous case, the polarity of the resulting protocol, $\mathcal{P}'$, might have flipped (compared to the polarity of $\mathcal{P}$).

Repeating this procedure, one can consider $\mathcal{P}'' := C^{\perp L}(\mathcal{P}, \mathcal{P}')$ and obtain $p_A^*(\mathcal{P}'')$ directly as illustrated above and numerically solve for $p_B^*(\mathcal{P}'')$ using the cheat vectors. Numerically, we found that ten-fifteen repetitions caused the cheating probabilities to converge to approximately 0.81459. We saw that the abort probabilities associated with $\mathcal{P}$ were quite small and therefore one could hope that $\mathcal{Q}$ fares better. Proceed analogously for protocol and considering $\mathcal{Q}' := C^{L\perp}(\mathcal{Q}, \mathcal{Q})$, $\mathcal{Q}'' := C^{L\perp}(\mathcal{Q}, \mathcal{Q}')$, etc., the cheating probabilities converge to approximately 0.822655.

Theorem 25. *Let $X \in \{A, B\}$. Then*

$$p_X^*(C^{\perp L}(\mathcal{P})) \approx 0.81459$$

and

$$p_X^*(C^{L\perp}(\mathcal{Q})) \approx 0.822655$$

where the latter holds assuming Conjecture ?? is true.

While by itself $\mathcal{Q}$ doesn't seem to help, one can suppress the bias further, by noting that at the very last step, only the cheating probabilities $p_A^*(\mathcal{Q})$ and $p_B^*(\mathcal{Q})$ played a role (i.e. the fact that the cheating vectors $\mathbb{C}_A$ for $\mathcal{Q}$ had an SDP characterisation was not used). Further, we know that $p_A^*(\mathcal{P}) = p_A^*(\mathcal{Q})$ but $p_B^*(\mathcal{P}) < p_B^*(\mathcal{Q})$, i.e. using $\mathcal{P}$ at the very last step will result in a strictly better protocol.

Theorem 26. *Let $X \in \{A, B\}$,*

$$\begin{aligned} \mathcal{Z}^1 &:= C^{L\perp}(\mathcal{Q}, \mathcal{P}), \quad \text{and} \\ \mathcal{Z}^{i+1} &:= C^{L\perp}(\mathcal{Q}, \mathcal{Z}^i) \quad i > 1. \end{aligned}$$

Then

$$\lim_{i\to\infty} p_X^*(\mathcal{Z}^i) \approx 0.791044\ldots$$

assuming Conjecture ?? holds.

5 Security Proof | Asymptotic limit

In this section, we prove the security under the following assumption:

Assumption 27. *In protocol $\mathcal{P}$ ($\mathcal{Q}$), Alice (Bob) does not perform the box verification step and instead it is assumed that her box is (his boxes are) taken from a trio of boxes which win the GHZ game with certainty.*

Later, we drop the assumption and use the box verification step (see ..) to estimate the probability of winning the GHZ game. When the winning probability is exactly one, the states and measurements are the same as the GHZ state and σ_x, σ_y measurements, up to local isometries and this allows us to use semi definite programming.

Lemma 28. *Let $a, b, c, x, y, z \in \{0, 1\}$. Consider a trio of quantum boxes, specified by projectors $\{M^A_{a|x}, M^B_{b|y}, M^C_{c|z}\}$ acting on finite dimensional Hilbert spaces $\mathcal{H}^A, \mathcal{H}^B$ and $\mathcal{H}^C$, and $|\psi\rangle \in \mathcal{H}^A \otimes \mathcal{H}^B \otimes \mathcal{H}^C =: \mathcal{H}^{ABC}$. If the trio pass the GHZ test with certainty, then there exists a local isometry*

$$\Phi = \Phi^A \otimes \Phi^B \otimes \Phi^C : \mathcal{H}^{ABC} \to \mathcal{H}^{ABC} \otimes \mathbb{C}^{2\times 3}$$

such that

$$\Phi\left(|\psi\rangle\right) = |\chi\rangle \otimes |\text{junk}\rangle\,,$$
$$\Phi\left(M^D_{d|t}\,|\psi\rangle\right) = \Pi^D_{d|t}\,|\text{GHZ}\rangle \otimes |\text{junk}\rangle \quad \forall D \in \{A, B, C\},\ \text{and } d, t \in \{0, 1\}$$

where $|\text{GHZ}\rangle = \frac{|000\rangle+|111\rangle}{\sqrt{2}} \in \mathbb{C}^{2\times 3}$, $|\text{junk}\rangle \in \mathcal{H}^{ABC}$ is some arbitrary state and $\{\Pi^A_{a|x}, \Pi^B_{b|y}, \Pi^C_{c|z}\}$ are projectors corresponding to σ_x on the first, second and third qubit of $|\text{GHZ}\rangle$ respectively, for $x = 0$ and corresponding to σ_y for $x = 1$, as in Claim 14.

INTERNAL; (TODO: remove): Isometries can only increase dimensions (they must be injective; that is to ensure they preserve inner products of vectors). Therefore the isometry can't get rid of the $|\text{junk}\rangle$ part.

5.1 SDP when Alice self-tests

Asymptotic proof of Lemma 17. We prove Lemma 17 under Assumption 27. We begin by making two observations.

First, note that in the protocol, if Alice applies an isometry on her box *after* she has inputted x, obtained the outcome a (and has noted it somewhere), the security of the resulting protocol is unchanged because the rest of the protocol only depends on x and a, and Alice's isometry only amounts to relabelling of the post measurement state. This freedom allows us to simplify the analysis.

Second, in the analysis, we cannot model Alice's random choice, say for x, as a mixed state because Bob can always hold a purification and thus know x. Therefore, we model the randomness using pure states and measure them in the end.

Notation: Other than PQR, all other registers store qubits.

We proceed step by step.

1. We can model (justified below) Alice's act of inputting a random x and obtaining an outcome a from her box through the state

$$|\Psi_0\rangle := \frac{1}{2} \sum_{x,a\in\{0,1\}} |xa\rangle_{XA}\,|\Phi(x,a)\rangle_{IJ} \tag{8}$$

where X represents the random input and A the output. Here, $|\Phi(x,a)\rangle_{IJ}$ are Bell states (see Equation (10)) and the registers IJ are held by Bob. Alice's act of choosing r at random, computing $s = a \oplus x.r$ is modelled as

$$|\Psi_1\rangle := \frac{1}{2\sqrt{2}} \sum_{x,a,r\in\{0,1\}} |xa\rangle_{XA}\,|\Phi(x,a)\rangle_{IJ}\,|r\rangle_R\,|a \oplus x.r\rangle_S\,. \tag{9}$$

Finally, Alice's act of sending s is modelled as Alice starting with the state

$$\text{tr}_{IJS}\left[|\Psi_1\rangle\langle\Psi_1|\right] \in XAR.$$

(Tom: Can we call this state ρ_1 ?**)**

Justification for starting with $|\Psi_0\rangle$.
To see why we start with the state $|\Psi_0\rangle$, model Alice's choice of x as $|+\rangle_X$, suppose her measurement result is stored in $|0\rangle_A$, the state of the boxes before measurement is $|\psi\rangle_{PQR}$ and Alice holds P, i.e.

$$\left|\Psi'_0\right\rangle := |+\rangle_X\,|0\rangle_A\,|\psi\rangle_{PQR}\,.$$

Let $\{M^P_{a|x}\}$ be the measurement operators corresponding to Alice's box. The measurement process is unitarily modelled as

$$\left|\Psi'_1\right\rangle := U_{\text{measure}}\left|\Psi'_0\right\rangle = \frac{1}{\sqrt{2}} \sum_{x,a\in\{0,1\}} |x\rangle_X\,|a\rangle_A\,M^P_{a|x}\,|\psi\rangle_{PQR}$$

where

$$U_{\text{measure}} = \sum_{x\in\{0,1\}} |x\rangle\langle x|_X \otimes \left[\mathbb{I}_A \otimes M^P_{0|x} + X_X \otimes M^P_{1|x}\right] \otimes \mathbb{I}_{QR}.$$

Now we harness the freedom of applying an isometry to the post measured state (as observed above). We choose the local isometry in Lemma 28. Without loss of generality, we can assume that Bob had already applied his part of the isometry before sending the boxes (because he can always reverse it when it is his turn). We thus have,

$$\begin{aligned}|\Psi'_2\rangle := \Phi_{PQR}|\Psi'_1\rangle &= \frac{1}{\sqrt{2}}\sum_{x,a\in\{0,1\}} |x\rangle_X |a\rangle_A \Pi^H_{x|a} |\text{GHZ}\rangle_{HIJ} \otimes |\text{junk}\rangle_{PQR} \\ &= \frac{1}{2}\sum_{x,a\in\{0,1\}} |x\rangle_X |a\rangle_A U^H(x,a)|0\rangle_H |\Phi(x,a)\rangle_{IJ} \otimes |\text{junk}\rangle_{PQR}\end{aligned}$$

where

$$|\Phi(x,a)\rangle_{IJ} = \frac{|00\rangle + (-1)^a (i)^x |11\rangle}{\sqrt{2}} \tag{10}$$

and $U^H(x,a)|0\rangle_H$ is $\frac{|0\rangle+(-1)^a(i)^x|1\rangle}{\sqrt{2}}$. Since the state of register H is completely determined by registers X and A, we can drop it from the analysis without loss of generality. Finally, since $|\text{junk}\rangle_{PQR}$ is completely tensored out, we can drop it too without affecting the security. Formally, we can assume that Alice gives Bob the register P at this point.

2. Bob sending g is modelled by introducing $\rho_2 \in XARG$ satisfying $\text{tr}_{IJS}[|\Psi_1\rangle\langle\Psi_1|] = \text{tr}_G(\rho_2)$.

3. At this point, either $x \oplus g$ is zero, in which case Alice's output is fixed or $x \oplus g$ is one, and in that case Bob will already know x because he knows g (he sent it) and Alice will proceed to testing Bob. Formally, therefore, we needn't do anything at this step.

4. Assuming $x \oplus g = 1$, Alice sends y, z to Bob such that $x \oplus y \oplus z = 1$. However, since Bob already knows x, he can deduce z from y. We thus only need to model Alice sending y and Bob responding with $d = b \oplus c$ (because Alice will only use $b \oplus c$ to test the GHZ game, so it suffices for Bob to send d). This amounts to introducing $\rho_3 \in XARGYD$ satisfying $\rho_2 \otimes \frac{\mathbb{I}_Y}{2} = \text{tr}_D(\rho_3)$.

5. Since we postponed the measurements to the end, we add this last step. Alice now measures ρ_3 to determine $x \oplus g$ and if it is one, whether the GHZ test passed. Let

$$\begin{aligned}\Pi_i &:= \sum_{x,y\in\{0,1\}:x\oplus g=i} |xg\rangle\langle xg|_{XG} \otimes \mathbb{I}_{ARYD}, \\ \Pi^{\text{GHZ}} &:= \sum_{\substack{x,y\in\{0,1\},\\ a,d\in\{0,1\}:a\oplus d\oplus 1=xy\cdot(1\oplus x\oplus y)}} |xyad\rangle\langle xyad|_{XYAD} \otimes \mathbb{I}_{RG}.\end{aligned} \tag{11}$$

Then, we can write the cheat vector for Alice, i.e. the tuple of probabilities that Alice outputs 0, 1 and abort (see Definition 16), as

$$(\alpha,\beta,\gamma) = (\text{tr}(\Pi_0\rho_3), \text{tr}(\Pi_1\Pi^{\text{GHZ}}\rho_3), \text{tr}(\Pi_1\bar{\Pi}^{\text{GHZ}}\rho_3))$$

where $\bar{\Pi} := \mathbb{I} - \Pi$.

To summarise, the final SDP is as follows: let $|\Psi_1\rangle \in XAIJRS$ be as given in Equation (9), $\rho_2 \in XARG$ and $\rho_3 \in XARGYD$

$$\max \quad \text{tr}([c_0\Pi_0 + \Pi_1(c_1\Pi^{\text{GHZ}} + c_\perp\bar{\Pi}^{\text{GHZ}})]\rho_3)$$

subject to

$$\begin{aligned}\text{tr}_{IJS}[|\Psi_1\rangle\langle\Psi_1|] &= \text{tr}_G(\rho_2) \\ \rho_2 \otimes \frac{\mathbb{I}_Y}{2} &= \text{tr}_D(\rho_3)\end{aligned}$$

where the projectors are defined in Equation (11). □

5.2 SDP when Bob self-tests

Proof of Lemma 19. Denote by $\mathcal{I}$ the protocol corresponding to Protocol ??.

It is evident that $p^*_B(Q) \leq p^*_B(\mathcal{I})$ because compared to $\mathcal{I}$, in Q Alice performs an extra test. However, it is not hard to see that the inequality is saturated, i.e. $p^*_B(Q) = p^*_B(\mathcal{I})$. Consider ... (TODO: recall/re-construct the cheating strategy for Bob that lets him win with the same 3/4 probability).

From Lemma 15, it is also clear that $p^*_A(Q) = p^*_A(\mathcal{I})$ because the only difference between Bob's actions in Q and $\mathcal{I}$ is that Bob self-tests to ensure his boxes are indeed GHZ. However, the optimal cheating strategy for $\mathcal{I}$ can be implemented using GHZ boxes.

This establishes the first part of the lemma. For the second part, i.e. establishing that optimising $c_0\alpha + c_1\beta + c_\perp\gamma$ over $(\alpha, \beta, \gamma) \in \mathbb{C}_A$ is an SDP, we proceed as follows. Suppose Assumption 27 holds. Then we can assume that Bob starts with the state

$$\rho_0 := \mathrm{tr}_H(|\mathrm{GHZ}\rangle\langle\mathrm{GHZ}|_{HIJ}) \tag{12}$$

and the effect of measuring the two boxes can be represented by the application of projectors of pauli operators X and Z.

The justification is similar to that given in the former proof. Suppose Bob holds registers QR of $|\psi\rangle_{PQR}$ which is the combined state of the three boxes. Suppose his measurement operators are $\{M^Q_{b|y}, M^R_{c|z}\}$. Then using the isometry in Lemma 28, Bob can relabel his state (and without loss of generality, we can suppose Alice also relabels according to the aforementioned isometry) to get $\Phi_{PQR}|\psi\rangle_{PQR} = |\mathrm{GHZ}\rangle_{HIJ} \otimes |\mathrm{junk}\rangle_{PQR}$. Further, since $\Phi_{PQR}(M^Q_{b|y} \otimes M^R_{c|z}|\psi\rangle_{PQR}) = \Pi^I_{b|y}\Pi^J_{c|z}|\mathrm{GHZ}\rangle_{HIJ} \otimes |\mathrm{junk}\rangle_{PQR}$ Bob's act of measurement, in the new labelling, corresponds to simply measuring the GHZ state in the appropriate Pauli basis. (TODO: in the approximate case, the initial state will be close to the one mentioned and the post-measured state will similarly only be close to the one post projectors; There should be some way of showing that this can be absorbed into the initial state). **(Tom:** There is still a TODO here**)**

1. Bob receiving s from Alice is modelled by introducing $\rho_1 \in SIJ$ satisfying $\mathrm{tr}_S(\rho_1) = \rho_0$.
2. Bob sending $g \in_R \{0,1\}$ can be seen as appending a mixed state: $\rho_1 \otimes \frac{1}{2}\mathbb{I}_G$.
3. Alice sending x (and a) can be modelled as introducing $\rho_2 \in AXSIJG$ satisfying $\mathrm{tr}_A(\rho_2) = \rho_1 \otimes \frac{\mathbb{I}_G}{2}$.
4. To model the GHZ test, introduce a register Y in the state $\frac{|0\rangle_Y + |1\rangle_Y}{\sqrt{2}}$. Recall that to perform the GHZ test, we need $x \oplus y \oplus z = 1$ i.e. $z = 1 \oplus y \oplus x$. Further introduce registers B and C to hold the measurement results, define

$$U := \sum_{y,x\in\{0,1\}} |y\rangle\langle y|_Y |x\rangle\langle x|_X \otimes (\mathbb{I}_B \otimes \Pi^I_{0|y} + X_B \otimes \Pi^I_{1|y}) \otimes (\mathbb{I}_C \otimes \Pi^J_{0|(1\oplus y\oplus x)} + X_C \otimes \Pi^J_{1|(1\oplus y\oplus x)}) \otimes \mathbb{I}_{ASG}. \tag{13}$$

   By construction, $\rho_3 := U(|+\rangle\langle+|_Y \otimes |00\rangle\langle 00|_{BC} \otimes \rho_2)U^\dagger \in YBCAXSIJG$ models the measurement process. (TODO: this equality would become approximately true...but perhaps the noise can be absorbed in ρ_0 with some argument)
5. Since we postponed the measurements to the end, we add this step. Define

$$\Pi_i := \sum_{x,g\in\{0,1\}:x\oplus g=i} |xg\rangle\langle xg|_{XG} \otimes \mathbb{I}_{YABSIJ}$$

   to determine who won. Define

$$\Pi^{\mathrm{sTest}} := \sum_{s,a,x\in\{0,1\}:s=a\vee s=a\oplus x} |sax\rangle\langle sax|_{SAX} \otimes \mathbb{I}_{GYBCIJ}$$

   to model the first test, i.e. s should either be a or $a \oplus x$. Define

$$\Pi^{\mathrm{GHZ}} := \sum_{\substack{x,y\in\{0,1\},\\ a,b,c\in\{0,1\}:a\oplus b\oplus c\oplus 1=xy\cdot(1\oplus x\oplus y)}} |xyabc\rangle\langle xyabc|_{XYABC} \otimes \mathbb{I}_{GSIJ}$$

to model the GHZ test. Let

$$\Pi^{\text{Test}} := \Pi^{\text{GHZ}}\Pi^{s\text{Test}}, \quad \bar{\Pi}^{\text{Test}} := \mathbb{I} - \Pi^{\text{Test}}. \tag{14}$$

One can then write the cheat vector for Bob, i.e. the tuple of probabilities that Bob outputs 0, 1 and abort (see Definition 16), as

$$(\alpha, \beta, \gamma) = (\text{tr}(\Pi_0\Pi^{\text{Test}}\rho_3), \text{tr}(\Pi_1\rho_3), \text{tr}(\Pi_0\bar{\Pi}^{\text{Test}}\rho_3)).$$

(Tom: NB: Still old notation for the cheat vectors**)** To summarise, the final SDP is as follows: let $\rho_0 \in IJ$ be as defined in Equation (12), $\rho_1 \in SIJ$ and $\rho_2 \in AXSIJG$. Then,

$$\max \quad \text{tr}\left([\Pi_0(c_0\Pi^{\text{Test}} + c_\perp\bar{\Pi}^{\text{Test}}) + c_1\Pi_1]U\left(|{+}00\rangle\langle{+}00|_{YBC} \otimes \rho_2\right)U^\dagger \right)$$

subject to

$$\text{tr}_S(\rho_1) = \rho_0$$
$$\text{tr}_A(\rho_2) = \frac{1}{2}\rho_1 \otimes \mathbb{I}_G$$

where U is as defined in Equation (13) and the projectors as in Equation (14).

□

6 Security Proof | Finite n

TODO: Write the following properly

In this section, we drop Assumption ??, and estimate the GHZ winning probability from Algorithm ??. We then use the robust variant of the self-testing result to conclude that the SDP of interest must be close to the SDP we considered (with some larger space tensored to it). Finally, we show the continuity of these SDPs and thereby conclude that we converge to the asymptotic result as n is increased.

We show this for the case where Alice self-tests. We expect an analogous result to hold when Bob self-tests.

6.1 Estimation of GHZ winning probability

We assume that the $3n$ boxes are described by some joint quantum state and local measurement operators. After playing the GHZ game with $3(n-1)$ of them, and verifying that they all pass, we want to make a statement about the remaining box, whose state $\tilde{\rho}$ is conditioned on the passing of all the other test.

Protocol 29. *Estimation of the GHZ value.*

1. *Pick a box $J \in [n]$ uniformly at random.*
2. *For $i \in [n]\backslash J$, play the GHZ game with box i, denote outcome of game as $X_i \in \{0, 1\}$*
3. *If*

$$\Omega: \quad X_i = 1, \textit{ for all } i \in [n]\backslash J \tag{15}$$

4. *Then conclude that the remaining box satisfies*

$$T: \quad E[X_J|J, \Omega] \geq 1 - \delta \tag{16}$$

The expectation value of $E[X_J|J, \Omega]$ accurately describes the expected GHZ value associated to the state of the remaining boxes J, conditioned on having measuring some outcome sequence in the other boxes which passes all the GHZ tests. Note that the conditioning in J is important because otherwise we would get a bound on the GHZ averaged over all boxes, but we are only interested in the remaining box.

Proposition 30 (Security of Protocol 29)**.** *For any implementation of the boxes and choice of $\delta > 0$ the joint probability that that the test Ω passes and that the conclusion T is false is small* $\Pr[\Omega \cap \bar{T}] \leq \frac{1}{1-\delta+n\delta} \leq \frac{1}{n\delta}$*, where the first upper-bound is tight.*

This is the correct form of the security statement. It is important to bound the joint distribution of Ω and $\overline{T}$, and not $\Pr[\overline{T}|\Omega]$, conditioning on passing the test Ω. Indeed in the latter case, it would not be possible to conclude anything of value about the remaining box J, as there could be some implementation of the boxes which has a very low expectation value of GHZ, but which passes the test with small but non-zero probability. The present security definition has a nice interpretation in the composable security framework of [ref]. Consider an hypothetical ideal protocol, which after having chosen J, only passes when T is true. In that case, $\Pr[\Omega \cap \overline{T}] = 0$. Then the actual protocol is equivalent the ideal one, except that if fails with probability $\epsilon = \frac{1}{1-\delta+m\delta}$, and so it is ϵ-close to the ideal algorithm.

Proof. For a given implementation of the boxes, let $p(x_1, \cdots x_n)$ denote the joint probability distribution of passing the GHZ games. Let $S = \{j|\ E[X_j|J = j, \Omega] < 1 - \delta\} \subset [n]$ be the set of boxes that have an expectation value for GHZ (conditioned on passing in the other boxes) below our target threshold and let $m = |S|$ be the number of such boxes. The value of m is unknown, so we will need to maximise over it in the end.

Let $\alpha = \Pr(\{X_i\}_i = 1)$ and $\beta_i = \Pr(\{X_i\}_{i\neq j} = 1 \cap X_j = 0)$ be respectively the probabilities of the events where all the tests pass, or they all pass except for the jth test. This allows us to rewrite $E[X_j|J = j, \Omega] = \Pr(\{X_i\}_i = 1)/\Pr(\{X_i\}_{i\neq j} = 1) = \alpha/(\alpha + \beta_j)$, and so, by definition of S, we have $\alpha/(\alpha + \beta_j) < (1 - \delta)$, for $j \in S$, which is equivalent to $\beta_j > \frac{\delta}{1-\delta}\alpha$.

The aim of the proof is to bound the probability $\Pr[\Omega \cap \overline{T}]$. If we condition and summed over the different values of J, we can rewrite it as

$$\Pr(\Omega \cap \overline{T}) = \sum_j \frac{1}{n} \Pr(\Omega \cap \overline{T}|J = j) = \sum_{j\in S} \frac{1}{n} \Pr(\{X_i\}_{i\neq j} = 1) = \frac{1}{n} \sum_{j\in S} (\alpha + \beta_i)\,, \tag{17}$$

where we have kept the round $j \in S$ ones, conditioned on which T is false. We are thus left with the optimisation problem

$$\max_{\alpha\geq 0,(\beta_i)_i\geq 0} \quad \frac{1}{n}\left(\sum_{j\in S} \alpha + \beta_j\right) \tag{18}$$

$$\text{subject to} \quad \alpha + \sum_{j\in S} \beta_j \leq 1 \tag{19}$$

$$\beta_j \geq \frac{\delta}{1-\delta}\alpha, \text{ for } j \in S \tag{20}$$

This is a linear problem. Simplifying it by defining $\Sigma = \sum_{j\in S} \beta_j$, gives

$$\max_{\alpha\geq 0,\Sigma\geq 0} \quad \frac{1}{n}(m\alpha + \Sigma) \tag{21}$$

$$\text{subject to} \quad \alpha + \Sigma \leq 1 \tag{22}$$

$$\Sigma \geq m\frac{\delta}{1-\delta}\alpha \tag{23}$$

It is easily shown that the maximum is attained for $(\alpha, \Sigma) = \left(\frac{1-\delta}{1-\delta+m\delta}, \frac{m\delta}{1-\delta+m\delta}\right)$ which gives the upper-bound

$$\Pr[\Omega \cap \overline{T}] \leq \frac{1}{n} \max_m \frac{m}{1-\delta+m\delta} = \frac{1}{1-\delta+n\delta} \tag{24}$$

We note that the upper-bound is an increasing function of m and so the maximum is attained for $m = n$. This yield the desired upper-bound. From the converse statement, we note that from the present proof we can construct a probability distribution $p(x_1, \cdots x_n)$, which saturates all inequalities, and so the upper-bound $\frac{1}{1-\delta+n\delta}$ is tight. □

6.2 Robust self-testing

Lemma 31. *Let $a, b, c, x, y, z \in \{0, 1\}$. Consider a trio of quantum boxes, specified by projectors $\{M^A_{a|x}, M^B_{b|y}, M^C_{c|z}\}$ acting on finite dimensional Hilbert spaces $\mathcal{H}^A, \mathcal{H}^B$ and $\mathcal{H}^C$, and $|\psi\rangle \in \mathcal{H}^A \otimes \mathcal{H}^B \otimes \mathcal{H}^C =: \mathcal{H}^{ABC}$. If the trio pass the GHZ test with probability $1 - \epsilon$ (for $1 > \epsilon > 0$), then there exists a local isometry,*

$$\Phi = \Phi^A \otimes \Phi^B \otimes \Phi^C : \mathcal{H}^{ABC} \to \mathcal{H}^{ABC} \otimes \mathbb{C}^{2\times 3}$$

and a decreasing function of ϵ, $f(\epsilon)$ such that

$$\|\Phi(|\psi\rangle) - |\chi\rangle \otimes |\text{junk}\rangle\| \leq f(\epsilon), \tag{25}$$

$$\left\|\Phi\left(M^D_{d|t}|\psi\rangle\right) - \Pi^D_{d|t}|\text{GHZ}\rangle \otimes |\text{junk}\rangle\right\| \leq f(\epsilon) \quad \forall D \in \{A, B, C\}, \text{ and } d, t \in \{0, 1\} \tag{26}$$

where $|\text{GHZ}\rangle = \frac{|000\rangle+|111\rangle}{\sqrt{2}} \in \mathbb{C}^{2\times 3}$, $|\text{junk}\rangle \in \mathcal{H}^{ABC}$ is some arbitrary state and $\{\Pi^A_{a|x}, \Pi^B_{b|y}, \Pi^C_{c|z}\}$ are projectors corresponding to σ_x on the first, second and third qubit of $|\text{GHZ}\rangle$ respectively, for $x = 0$ and corresponding to σ_y for $x = 1$, as in Claim 14.

Proof. A proofs of robust self-testing for GHZ can be found in [MS13] and [McK14]. □

6.3 SDP-valued functions and their continuity

A semidefinite program (SDP) is an optimization problem of the form

$$\begin{aligned} f(A, B) = \text{maximize:} \quad & \langle A, X\rangle \\ \text{subject to:} \quad & \Phi(X) = B \\ & X \geq 0. \end{aligned} \tag{27}$$

We call $f(A, B)$ the value of the semidefinite program which is the supremum of $\langle A, X\rangle$ over all X that are feasible ($X \geq 0$ and $\Phi(X) = B$). In this work we wish to view how the value of an SDP changes as you change A and/or B. Ultimately, we wish to know if the value of an SDP is continuous as a function of A and B. To this end, let us consider the function

$$h(A) = \text{maximize:} \quad \langle A, X\rangle \tag{28}$$

$$\text{subject to:} \quad X \in C \tag{29}$$

where C is a nonempty, convex set. This is a generalization of an SDP which is convenient for the upcoming analysis. Notice that when C is unbounded, it may be the case that f takes the value $+\infty$. Since we cannot count that high, we use the following definition.

Definition 32. We define the *support* of the function h, denoted as $\text{supp}(h)$, as

$$\text{supp}(h) := \{A : h(A) \text{ is finite}\}. \tag{30}$$

We now show some elementary properties of this function.

Lemma 33. *The support of h is convex and h is a convex function on its support.*

Proof. For $A_1, A_2 \in \text{supp}(h)$ and $\lambda_1, \lambda_2 \geq 0$ satisfying $\lambda_1 + \lambda_2 = 1$, we have

$$h(\lambda_1 A_1 + \lambda_2 A_2) \leq h(\lambda_1 A_1) + h(\lambda_2 A_2) \tag{31}$$

$$= \lambda_1 h(A_1) + \lambda_2 h(A_2) \tag{32}$$

$$< +\infty \tag{33}$$

where the last inequality follows from $A_1, A_2 \in \text{supp}(h)$. Thus, $\lambda_1 A_1 + \lambda_2 A_2 \in \text{supp}(h)$, proving $\text{supp}(h)$ is a convex set, and h is convex from the above inequalities. □

The following corollary follows from the fact that h is convex.

Corollary 34. *h is continuous on the interior of its support.*

Another well-known corollary is that h is continuous everywhere if C is compact. This follows from the above corollary since the support is the entire space.

Corollary 35. *If C is compact, h is continuous everywhere.*

References

[ACG+14] Dorit Aharonov, André Chailloux, Maor Ganz, Iordanis Kerenidis, and Loïck Magnin, *A simpler proof of existence of quantum weak coin flipping with arbitrarily small bias*, SIAM Journal on Computing **45** (2014), no. 3, 633–679.

[ARV] Atul Singh Arora, Jérémie Roland, and Chrysoula Vlachou, *Analytic quantum weak coin flipping protocols with arbitrarily small bias*, pp. 919–938.

[ARW] Atul Singh Arora, Jérémie Roland, and Stephan Weis, *Quantum weak coin flipping*.

[ARW19] Atul Singh Arora, Jérémie Roland, and Stephan Weis, *Quantum weak coin flipping*, Proceedings of the 51st Annual ACM SIGACT Symposium on Theory of Computing - STOC 2019, ACM Press, 2019.

[Blu83] Manuel Blum, *Coin flipping by telephone a protocol for solving impossible problems*, SIGACT News **15** (1983), no. 1, 23–27.

[Kit03] Alexei Kitaev, *Quantum coin flipping*, Talk at the 6th workshop on Quantum Information Processing, 2003.

[McK14] Matthew McKague, *Self-testing graph states*, Theory of Quantum Computation, Communication, and Cryptography (Berlin, Heidelberg) (Dave Bacon, Miguel Martin-Delgado, and Martin Roetteler, eds.), Springer Berlin Heidelberg, 2014, pp. 104–120.

[Mil20] Carl A. Miller, *The impossibility of efficient quantum weak coin flipping*, Proceedings of the 52nd Annual ACM SIGACT Symposium on Theory of Computing (New York, NY, USA), STOC 2020, Association for Computing Machinery, 2020, pp. 916–929.

[Moc07] Carlos Mochon, *Quantum weak coin flipping with arbitrarily small bias*, arXiv:0711.4114 (2007).

[MS13] Carl A. Miller and Yaoyun Shi, *Optimal Robust Self-Testing by Binary Nonlocal XOR Games*, 8th Conference on the Theory of Quantum Computation, Communication and Cryptography (TQC 2013) (Dagstuhl, Germany) (Simone Severini and Fernando Brandao, eds.), Leibniz International Proceedings in Informatics (LIPIcs), vol. 22, Schloss Dagstuhl–Leibniz-Zentrum fuer Informatik, 2013, pp. 254–262.

[SCA+11] J. Silman, A. Chailloux, N. Aharon, I. Kerenidis, S. Pironio, and S. Massar, *Fully distrustful quantum bit commitment and coin flipping*, Physical Review Letters **106** (2011), no. 22.