

Improved Device Independent Weak Coin Flipping Protocols

7th May 2021

Abstract

[OUTDATED: Needs to be rewritten]

We report a device independent weak coin flipping protocol¹ with $P_A^* \leq \cos^2(\pi/8)$ and $P_B^* \leq 0.667\dots$, by making seemingly minor changes to the best known protocol due to SCAKPM'11 [10.1103/PhysRevLett.106.220501], with $P_A^* \leq \cos^2(\pi/8) \approx 0.85$ and $P_B^* \leq 3/4 = 0.75$. In terms of bias, we improve the SCAKPM'11 result from ≈ 0.336 to ≈ 0.3199 . This improvement is due to two ingredients: a self-testing (of GHZ) step and an extra cheat detection step for Bob. We also introduce a new bias suppression technique that ekes out further security from the abort probability to obtain ... Note that the SCAKPM'11 result held for both strong and weak coin flipping; ours holds only for the latter. TODO: Fix me!

Contents

| | | |
|----------|---|-----------|
| 1 | Introduction | 2 |
| 1.1 | About Weak Coin Flipping | 2 |
| 1.2 | Contributions | 2 |
| 1.3 | Proof Technique | 2 |
| 2 | Device Independent Weak Coin Flipping protocols State Of The Art | 4 |
| 2.1 | Device Independence and the Box Paradigm | 4 |
| 2.2 | The GHZ Test | 5 |
| 2.3 | The Protocol | 6 |
| 3 | First Technique: Self-testing (single shot, unbalanced) | 6 |
| 3.1 | Cheat Vectors | 7 |
| 3.2 | Alice self-tests Protocol \mathcal{P} | 7 |
| 3.3 | Bob self-tests Protocol \mathcal{Q} | 8 |
| 4 | Second Technique: Bias Suppression | 8 |
| 4.1 | Standard Analysis | 9 |
| 4.2 | Cheat Vector Analysis | 9 |
| 5 | Application: Strong Coin Flipping | 11 |
| 6 | Security Proof Asymptotic | 11 |
| 6.1 | Cheat vectors optimisation using Semi Definite Programming | 12 |
| 6.1.1 | SDP when Alice self-tests | 12 |
| 6.1.2 | SDP when Bob self-tests | 14 |
| 6.2 | The self-testing step [Discuss with Tom before writing] | 15 |
| 6.3 | The continuity argument [Enter Jamie] | 15 |

¹which are analysed

1 Introduction

INTERNAL/Atul: Colour coding—Purple is for informal discussions, black is for formal statements and blue is for proofs. We can remove these from the final version; I put it to minimise verbiage.

1.1 About Weak Coin Flipping

Secure two-party computation is a cryptographic setting where two parties, conventionally called Alice and Bob, receive inputs x and y and their goal is to compute some function $f_A(x, y)$ and $f_B(x, y)$ respectively which depends on both their inputs. However, they do not wish to reveal their inputs. Coin flipping (CF) is a cryptographic primitive in this setting, i.e. a building block for constructing more applicable secure two-party cryptographic schemes, where Alice and Bob wish to exchange messages and agree on a random bit, without trusting each other. A protocol that implements coin flipping must protect an honest player from a malicious² player.

A weaker primitive, unsurprisingly, known as *weak coin flipping* (WCF) is where a zero corresponds to Alice winning and one corresponds to Bob winning. It is weaker because now the protocol has to protect Alice from a malicious Bob who tries to bias the outcome towards one (and not towards zero) and conversely, it must protect Bob from a malicious Alice who tries to bias the outcome towards zero (and not towards one). To emphasise the distinction, the former primitive is often termed *strong coin flipping* (SCF).

We primarily focus on WCF in this article and begin with introducing some notation. We denote by P_A^* the highest probability of a malicious Alice convincing an honest Bob that she won (i.e. in the WCF protocol, Alice uses her best cheating strategy against Bob who in turn is following the protocol as described, to convince him that the outcome is zero). Analogously, P_B^* is the highest probability of a malicious Bob convincing an honest Alice that he won. The bias of a WCF protocol is defined as $\epsilon := \max\{P_A^*, P_B^*\} - \frac{1}{2}$. A protocol that is completely secure, has $\epsilon = 0$ and one that is completely insecure has $\epsilon = \frac{1}{2}$.

Using a classical channel of communication between Alice and Bob, unless one makes further assumptions such as computational hardness of certain problems or relativistic assumptions,³ coin flipping (even weak) is impossible to implement with any security, to wit: for all classical protocols at least one of the parties, viz. a malicious Alice or a malicious Bob, can win with certainty because one can show $\epsilon = \frac{1}{2}$ (viz. $\max\{P_A^*, P_B^*\} = 1$). Using a quantum channel of communication, it was shown that WCF can be implemented with vanishing bias. These works, however, do not account for noise in their implementation. One path towards more robust security is device independence wherein the players do not even trust their devices (recall, they already do not trust the other party). This is in contrast to the device independent setting considered in key distribution where the two parties trust each other but neither their devices nor the communication channel (TODO: is the classical communication channel trusted?).

1.2 Contributions

[TODO: fix it—this is outdated] In this work, we start with a device independent (DI) coin flipping (CF) protocol introduced⁴ in [SCA⁺11] which has $P_A^* = \cos^2(\pi/8) \approx 0.854$ and $P_B^* = 3/4 = 0.75$. They then compose these protocols to give a balanced protocol, i.e. with $P_A^* = P_B^* \approx \frac{1}{2} + 0.336$. To the best of our knowledge, this DI CF protocol has the best security guarantee. While Kitaev’s bound for CF rules out perfect DI CF, no lower bounds on the bias are known for DI WCF. In this work, however, we focus on improving the upper bound on the bias, viz. we give DI WCF protocols with biases ≈ 0.319 .

We introduce two key new ideas which result in better protocols. The first, is the use of self-testing by one party before initiating the protocol and the second, is a more general technique to convert unbalanced protocols (i.e. ones in which the probability of maliciously winning for Alice and Bob are unequal) into balanced ones.

1.3 Proof Technique

Notation and Cheat Vectors

²(or cheating, we use these adjectives interchangeably)

³in terms of the spatial locations of the observers; not to be confused with the term *relativising* from computational complexity.

⁴In fact, they introduced a device independent bit commitment protocol which they in turn use to construct a strong coin flipping protocol with the same cheating probabilities for Alice and Bob, ≈ 0.854 and 0.75 respectively.

We introduce some notation to facilitate the discussion here. Denote the DI CF protocol introduced in [SCA⁺11] by \mathcal{I} and let $p_A^*(\mathcal{I}) \approx 0.853 \dots$ denote the maximum probability with which a malicious Alice can win against honest Bob who is following the protocol \mathcal{I} and similarly, let $p_B^*(\mathcal{I}) \approx 0.75$ denote the maximum probability with which a malicious Bob can win against an honest Alice who is following the protocol \mathcal{I} .

One of the key observations we make in this work is the use of what we call “cheat vectors”—it is any tuple of probabilities which can arise in a CF protocol when one player is honest. More precisely, suppose Alice is (possibly) malicious and Bob follows the protocol \mathcal{I} . Then, the cheat vectors for Alice constitute the set

$$\mathbb{C}_A(\mathcal{I}) := \{(\alpha, \beta, \gamma) : \exists \text{ a strategy for } A \text{ s.t. an honest } B \text{ outputs } 0, 1, \text{ and } \perp \text{ with probabilities } \alpha, \beta \text{ and } \gamma\}. \quad (1)$$

We analogously define $\mathbb{C}_B(\mathcal{I})$. Cheat vectors become useful when we try to compose protocols. The observation then, is that the abort event can be taken to abort the full protocol instead of being treated as the honest player winning. The latter gives the malicious player further opportunity to cheat and so preventing it improves the security.

Protocols

We introduce two variants of protocol \mathcal{I} , which we call \mathcal{P} and \mathcal{Q} .

- \mathcal{P} is essentially the same as \mathcal{I} except that Alice self-tests her boxes before starting the protocol and performs an additional test to ensure Bob doesn’t cheat. We show that $p_A^*(\mathcal{P}) \lesssim 0.853 \dots$ and $p_B^*(\mathcal{P}) \lesssim 0.667 \dots$. We also show that $\mathbb{C}_B(\mathcal{P})$ can be cast as an SDP.
- \mathcal{Q} is also essentially the same as \mathcal{I} except that Bob self-tests his boxes before starting the protocol. In this case, $p_X^*(\mathcal{Q}) = p_X^*(\mathcal{I})$ for both values of $X \in \{A, B\}$ so the advantage isn’t manifest. However, now $\mathbb{C}_A(\mathcal{Q})$ can be cast as an SDP which, as we shall see, yields an advantage when \mathcal{Q} is composed.

Compositions

As the protocols $\mathcal{X} \in \{\mathcal{I}, \mathcal{P}, \mathcal{Q}\}$ all have skewed security—either $p_A^*(\mathcal{X}) > p_B^*(\mathcal{X})$ or the other way—and therefore the bias is determined by $p_{\max}^*(\mathcal{X}) := \max\{p_A^*(\mathcal{X}), p_B^*(\mathcal{X})\}$. Note that, $p_{\max}^*(\mathcal{X}) = p_{\max}^*(\mathcal{Y})$ for all $\mathcal{X}, \mathcal{Y} \in \{\mathcal{I}, \mathcal{P}, \mathcal{Q}\}$, which means that we don’t immediately obtain an advantage. However, the most obvious method of composing these protocols to obtain a new protocol, which we describe later, “balances” the advantage. After this composition procedure is applied to some protocol \mathcal{X} , we denote the resulting protocol by $C_{\text{stand}}(\mathcal{X})$. Applying this technique to \mathcal{P} , we already obtain a more secure protocol.

- For all $X \in \{A, B\}$ the cheating probabilities for protocol \mathcal{I} under the standard composition is given by

$$p_X^*(C_{\text{stand}}(\mathcal{I})) \approx \frac{1}{2} + 0.336 \dots$$

while for the improved protocol \mathcal{P} , these are given by

$$p_X^*(C_{\text{stand}}(\mathcal{P})) \approx \frac{1}{2} + 0.3199 \dots \quad (2)$$

The standard composition technique doesn’t yield any improvement for \mathcal{Q} because the cheating probabilities are identical to those of \mathcal{I} . We can extract an advantage by using a composition technique that uses “cheat vectors” and the abort event. We describe it in detail later but for now, we simply denote the new protocol obtained using this improved “abort augmented” composition (of protocol \mathcal{X}) by $C_{\text{AAC}}(\mathcal{X})$.

- Using this technique on \mathcal{Q} , the cheating probabilities become

$$p_X^*(C_{\text{AAC}}(\mathcal{Q})) \approx \frac{1}{2} + 0.317 \dots$$

for all $X \in \{A, B\}$, which is even better than Equation (2).

- Finally, we combine both these protocols to obtain (again, for all $X \in \{A, B\}$)

$$p_X^*(C_{\text{AAC}}(Q, Q, \dots, Q, \mathcal{P})) \approx \frac{1}{2} + 0.2908 \dots$$

where we use the same composition technique except that at the last “level” we use \mathcal{P} instead of Q .

TODO: Obvious questions (answers to which I don’t have anymore; stupid memory): what about $p_X^*(C_{\text{AAC}}(\mathcal{P}))$ and $p_X^*(C_{\text{AAC}}(\mathcal{P}, \mathcal{P}, \dots, \mathcal{P}, Q))$.

aoeu

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. Subsequently we recall the GHZ test and finally we use these to delineate the DI-CF due to [SCA⁺11].

2.1 Device Independence and the Box Paradigm

We describe device independent protocols as classical protocols with the 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).⁵

Definition 1 (Box). A *box* is a device that takes an input $x \in X$ and yields an outputs $a \in \mathcal{A}$ where X and \mathcal{A} are finite sets. Typically, a set of n boxes, taking inputs x_1, x_2, \dots, x_n and yielding outputs a_1, a_2, \dots, a_n are *characterised* by a joint conditional probability distribution, denoted by

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

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

Henceforth, we restrict ourselves to quantum boxes.

Definition 2 (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 $\square_1^A, \square_2^A \dots \square_p^A$ and Bob is given boxes $\square_1^B, \square_2^B, \dots, \square_q^B$.
- (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 3 (Setup of Device Independent Two-Party Protocols). *Alice and Bob*

⁵TODO: Verify if this notion is in fact correct; I hope I’m not making a major mistake somehow. I should be able to take the POVMs as tensor products right, because I can change them at will, independent of the others (and ensuring that there’s no communication between them; could they be somehow entangled, i.e. could it be that somehow the measurement operators are themselves quantum correlated?); I would like to reach the conclusion starting from the locality assumption.

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 4 (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, \dots, A_p together with POVMs⁶

$$\{M_{a|x}^{A_1}\}_a, \{M_{a|x}^{A_2}\}_a, \dots, \{M_{a|x}^{A_p}\}_a$$

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

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

Alice shields A_1, A_2, \dots, A_p (and the POVMs) from each other and from Bob’s lab. Bob similarly shields B_1, B_2, \dots, 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 2) which uses only quantum boxes (Definition 1) can be implemented as a measure and exchange protocol (Definition 4).

2.2 The GHZ Test

Before we define the current best DI CF protocol, we briefly remind the reader of the GHZ test, upon which the aforementioned protocol depends, and set up some conventions.

Definition 5. 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 $x \oplus y \oplus z = 1$.

Claim 6. 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 measurement⁷ $\frac{\sigma_x + \mathbb{I}}{2}$ 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 $|\pm\rangle = \frac{|0\rangle \pm |1\rangle}{\sqrt{2}}$).⁸

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. $\sigma_x \sigma_y + \sigma_y \sigma_x = 0$.

⁶For concreteness, take the case of binary measurements. By $\{M_{a|x}^{A_1}\}_a$, for instance, we mean $\{M_{0|x}^{A_1}, M_{1|x}^{A_1}\}$ is a POVM for $x \in \{0, 1\}$.

⁷we added the identity so that the eigenvalues associated become 0, 1 instead of $-1, 1$.

⁸TODO: Think: Should I add the classical value? This would require me to add what it means to have a classical box.

2.3 The Protocol

The best DI CF protocol known is the one introduced in [SCA⁺11]. While this is a protocol for SCF, and so also works as a WCF protocol, we do not know of any better protocol for the latter.

Algorithm 7 (SCF, original). *Alice has one box and Bob has two boxes (in the security analysis, we let the cheating player distribute the boxes). Each box takes one binary input and gives one binary output.*

1. Alice chooses $x \in_R \{0, 1\}$ and inputs it into her box to obtain a . She chooses $r \in_R \{0, 1\}$ to compute $s = a \oplus x \cdot r$ and sends s to Bob.
2. Bob chooses $g \in_R \{0, 1\}$ (for “guess”) and sends it to Alice.
3. Alice sends x and a to Bob. They both compute the output $x \oplus g$.
4. Test round
 - (a) Bob tests if $s = a$ or $s = a \oplus x$. If the test fails, he aborts. Bob chooses $b, c \in_R \{0, 1\}$ such that $a \oplus b \oplus c = 1$ and then performs a GHZ using a, b, c as the inputs and x, y, z as the output from the three boxes. He aborts if this test fails.

From Claim 6, it is clear that when both players follow Algorithm 7 using GHZ boxes (Definition 5), Bob never aborts and they win with equal probabilities. The security of the protocol is summarised next.

Lemma 8 (Security of SCF). [SCA⁺11] *Let \mathcal{I} denote the protocol corresponding to Algorithm 7. 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.

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

TODO: Assumption: No honest abort.

We make two observations.

First, in Algorithm 7 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 extent) 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 (see TODO: write this down somewhere) but mention this 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. One way of proceeding is that N copies of the supposedly correct boxes are distributed. Alice now picks one out of these N boxes at random and asks Bob to send the associated two boxes to each $N - 1$ box that Alice possesses. Alice runs the GHZ test on each box and if even one test fails, she declares that Bob cheated. This way, for a large N , Alice can ensure with near certainty, that she has a box containing the correct state and (which performs the correct) measurements. 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 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 $P_A^* \leq 3/4$, which is the same as before, but $P_B^* \lesssim 0.667$

3.1 Cheat Vectors

As alluded to in Section 1.3, 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 9 (Cheat Vectors). Given a protocol \mathcal{I} , denote by $\mathbb{C}_B(\mathcal{I})$ the set of *cheat vectors* for Bob, which is defined as follows :

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

and analogously, denote by $\mathbb{C}_A(\mathcal{I})$ the set of cheat vectors for Alice (see Equation (1)).

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 6).

Algorithm 10 (Alice self-tests her boxes). *There are N trios of boxes; Alice has the first part and Bob has the remaining two parts, of each trio.*

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

We restrict ourselves to the i th trio. Alice has one box and Bob has two boxes. Each box takes one binary input and gives one binary output.

1. Alice chooses $x \in_R \{0, 1\}$ and inputs it into her box to obtain a . She chooses $r \in_R \{0, 1\}$ to compute $s = a \oplus x \cdot r$ and sends s to Bob.
 2. Bob chooses $g \in_R \{0, 1\}$ (for “guess”) and sends it to Alice.
 3. Alice sends x [EDIT: maybe not send a] and a to Bob. They both compute the output $x \oplus g$.
 4. Test rounds:
 - (a) If $x \oplus g = 0$:
[EDIT: Send a]
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.

Lemma 11. Let \mathcal{P} denote the protocol corresponding to Algorithm 10. 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 $N \rightarrow \infty$, the solution to the optimisation problem $\max(c_0\alpha + c_1\beta + c_\perp\gamma)$ over $\mathbb{C}_B(\mathcal{P})$ approaches that of a semi definite programme. In particular, i.e. for $c_0 = c_\perp = 0$ and $c_1 = 1$, $p_B^*(\mathcal{P}) \lesssim 0.667\dots$ (in the limit).

We defer the proof to Section 6.1.1. 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 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.

Algorithm 12 (Bob self-tests his boxes). *Proceed exactly as in Algorithm 10, 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 $\{1, 2 \dots N\} \setminus i$, 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 $\{1, 2 \dots N\} \setminus i$, i.e. all the trios except the i th.

Henceforth, proceed as in Algorithm 10 after the self-testing step.

As already indicated in Section 1.3, we don't expect the cheating probabilities to improve but we do expect an SDP characterisation of Alice's cheat vectors.

Lemma 13. *Let Q denote the protocol corresponding to Algorithm 12. Then, Alice's cheating probability, $p_A^*(Q) \leq 3/4$ and Bob's cheating probability, $p_B^*(Q) \leq \cos^2(\pi/8)$ (which are the same as those in Lemma 8). Further, let $c_0, c_1, c_\perp \in \mathbb{R}$, and $\mathbb{C}_A(Q)$ be the set of cheat vectors for Alice. Then, as $N \rightarrow \infty$, the solution to the optimisation problem $\max(c_0\alpha + c_1\beta + c_\perp\gamma)$ over $(\alpha, \beta, \gamma) \in \mathbb{C}_A(Q)$ approaches that of a semi definite programme.*

The proof is again deferred; see Section 6.1.2.

4 Second Technique: Bias Suppression

In this section, we use the convention that \mathcal{I}, \mathcal{P} and Q correspond to the protocols described in Algorithm 7, Algorithm 10 and Algorithm 12, respectively. Notice that $p_A^*(\mathcal{X}) \geq p_B^*(\mathcal{X})$ where $\mathcal{X} \in \{\mathcal{I}, \mathcal{P}, Q\}$. 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.

Definition 14 (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 .

TODO: It might help to explain “roles of Alice and Bob” are interchanged a little bit using the “flip and declare” protocol.

TODO: Winner gets Polarity

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

Algorithm 15 ($C(.,.)$ and $C(.)$). *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 14). 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 $\mathcal{Z}^{i+1} := C_{\text{AAC}}(\mathcal{X}, \mathcal{Z}^i)$ for $i \geq 1$, and $\mathcal{Z}^1 := \mathcal{X}$. Then, formally, define $C(\mathcal{X}) := \lim_{i \rightarrow \infty} \mathcal{Z}^i$.⁹

Example 16. TODO: Perhaps show how \mathcal{Z}_B is constructed from $\mathcal{Z} = C_{\text{stand}}(\mathcal{X}, \mathcal{Y})$. Perhaps have this in the appendix.

In the following sections, we work out some examples and analyse their security and see where having a neat characterisation of cheat vectors helps.

4.1 Standard Analysis

In the standard analysis, we only use bounds on the cheating probabilities, p_A^* and p_B^* , of the protocols being composed to obtain the corresponding bounds for the resultant protocol. Let us take an example. Consider protocol \mathcal{P} (see Algorithm 10) and recall (see Lemma 11)

$$\begin{aligned} p_A^*(\mathcal{P}_A) &\leq \alpha \approx 0.852 \dots, \\ p_B^*(\mathcal{P}_A) &\leq \beta \approx 0.667 \dots \end{aligned}$$

Note that therefore $p_A^*(\mathcal{P}_B) \leq \beta$ and $p_B^*(\mathcal{P}_B) \leq \alpha$. Further, let $\mathcal{P}' := C(\mathcal{P}, \mathcal{P})$, i.e. Alice and Bob first execute \mathcal{P}_A and if the outcome is A , they execute \mathcal{P}_A , otherwise they execute \mathcal{P}_B (see Figure 1). Then,

$$\begin{aligned} p_A^*(\mathcal{P}'_A) &\leq \alpha\alpha + (1 - \alpha)\beta =: \alpha^{(1)}, \\ p_B^*(\mathcal{P}'_B) &\leq \beta\alpha + (1 - \beta)\beta =: \beta^{(1)}. \end{aligned} \tag{3}$$

To see this, focus on Equation (3). 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 at most α . Assuming that with all the remaining probability, the outcome is B (which it is not because Bob will abort with some probability) gives us a bound on the best that Alice can do. Evidently, this bound may not be tight and we address that in the following subsection.

A side remark: one consequence of this simplified analysis is that¹⁰ $\alpha^{(1)} > \beta^{(1)}$. Proceeding similarly, i.e. defining $\mathcal{P}'' := C_{\text{stand}}(\mathcal{P}, \mathcal{P}')$, and repeating $k + 1$ times overall, one obtains¹¹

$$\begin{aligned} \alpha^{(k+1)} &= \alpha\alpha^{(k)} + (1 - \alpha)\beta^{(k)} \\ \beta^{(k+1)} &= \beta\alpha^{(k)} + (1 - \beta)\beta^{(k)}. \end{aligned}$$

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

$$p_A^*(C_{\text{stand}}(\mathcal{P})) = p_B^*(C_{\text{stand}}(\mathcal{P})) \leq \lim_{k \rightarrow \infty} \alpha^{(k)} = \lim_{k \rightarrow \infty} \beta^{(k)} \approx 0.8199 \dots$$

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

$$p_X^*(C_{\text{stand}}(\mathcal{X})) \lesssim 0.836 \dots$$

We thus have the following.

Theorem 17. Let $X \in \{A, B\}$ and $X \in \{I, Q\}$. Then $p_X^*(C(\mathcal{P})) \lesssim 0.8199 \dots$ and $p_X^*(C(\mathcal{X})) \lesssim 0.836 \dots$ [TODO: remove approx eq]

4.2 Cheat Vector Analysis

Remark 18. Again, even though it seems that Alice might be favoured, C may result in a protocol with the opposite polarity—let $\mathcal{Z} := C(\mathcal{X}, \mathcal{Y})$, then \mathcal{Z}_A is not necessarily the same as \mathcal{Z} .

⁹This 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.

¹⁰ $\alpha^{(1)} - \beta^{(1)} = (\alpha - \beta)\alpha - (\alpha - \beta)\beta = (\alpha - \beta)^2 > 0$

¹¹Note that $\alpha^{(k+1)} - \beta^{(k+1)} = (\alpha^{(k)} - \beta^{(k)})(\alpha - \beta) > 0$, which proves Claim ??.

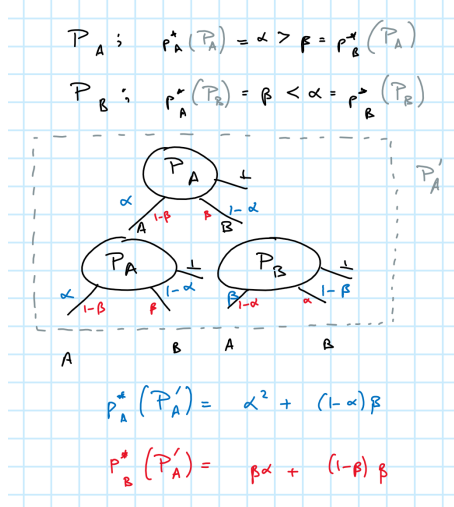


Figure 1: Standard analysis (TODO: improve the caption)

As before, let us workout an example. Consider protocol Q this time (see Algorithm 12) and let $(v_A, v_B, v_\perp) \in \mathbb{C}_A(Q_A)$ be a cheat vector for Q . Further, recall that

$$p_A^*(Q_A) \leq \alpha = 3/4 = 0.75$$

$$p_B^*(Q_A) \leq \beta = \cos^2(\pi/8) = 0.8535 \dots$$

and note that, therefore, $p_A^*(Q_B) \leq \beta$ and $p_B^*(Q_B) \leq \alpha$. Let $Q' := C(Q, Q)$, i.e. Alice and Bob first execute Q_A and if the outcome is A , they execute Q_A , otherwise they execute Q_B (see Figure 2). Our goal is again to obtain an upper bound on $p_A^*(Q')$ and $p_B^*(Q')$, but this time, using cheat vectors. We start from the bottom of Figure 2.

Suppose Alice is malicious. Conditioned on the event that the protocol reaches the bottom left step, it is clear that Alice's highest probability of winning is α . In this case, the abort probability doesn't play a role. Similarly, conditioned on the event that the protocol reaches the bottom right step, it is clear that Alice's highest probability of winning is β . At the first step then, it is clear that the highest probability of Alice winning is

$$p_A^*(Q') \leq \max_{(v_A, v_B, v_\perp) \in \mathbb{C}_A} \alpha v_A + \beta v_B \quad (4)$$

$$\leq \alpha^2 + \beta(1 - \alpha). \quad (5)$$

The first inequality is saturated if α and β are tight bounds. The second inequality holds because $\alpha > \beta$ so one cannot do better than putting all the weight on α (i.e. α) and the remaining on β (i.e. $1 - \alpha$), as we did earlier. [DISCUSS/TODO: Remove repetition; I already explained this.] Operationally, this means that Alice is forcing the outcome A and treating the outcomes B and \perp as proceeding to the Q_B step (i.e. the bottom right step). However, Bob only agrees to proceed to the bottom right step if the outcome is B and hence the maximization, Equation (4), potentially yields a better bound.

The case for malicious Bob would have been similar if we had a good characterisation for $\mathbb{C}_B(Q)$. Since we don't, we content ourselves with a weaker bound, the one analogous to Equation (5):

$$p_B^*(Q') \leq \beta\alpha + (1 - \beta)\beta.$$

TODO: put the numbers (they seem to be off).

Thus to obtain $p_A^*(Q')$, one solves an SDP (numerically) and $p_B^*(Q')$ is obtained readily. Using this, one can consider $Q'' := C(Q, Q')$ and evaluate the cheating probabilities $p_A^*(Q'')$ again as an SDP and $p_B^*(Q'')$ directly. This leads us to the following result.

Theorem 19. *Let $X \in \{A, B\}$. Then*

$$p_X^*(C(Q)) \lesssim 0.8175 \dots$$

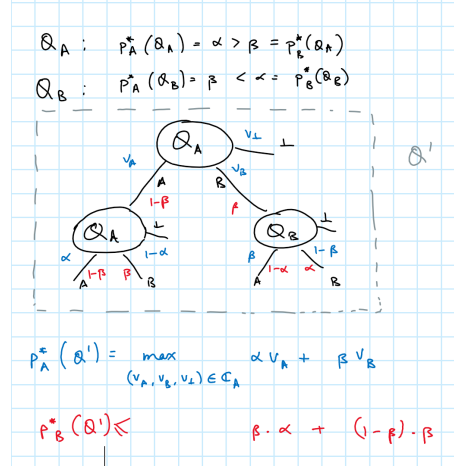


Figure 2: Cheat vector analysis. (TODO: improve the caption)

$(v_A, v_B, v_L) \in \mathbb{C}_A$; because we don't have a good characterisation of \mathbb{C}_B , we use (red) $1 - \beta$ and β for obtaining a bound on $p_B^*(Q')$.

and

$$p_X^*(C(Q)) \lesssim$$

(TODO: calculate).

To suppress the bias further, we note that at the very last step, only the cheating probabilities $p_A^*(Q) \leq \alpha$ and $p_B^*(Q) \leq \beta$ played a role (i.e. the fact that the cheating vectors \mathbb{C}_A for Q had an SDP characterisation was not used). Further, we know that $p_A^*(\mathcal{P}) = p_A^*(Q)$ but $p_B^*(\mathcal{P}) < p_B^*(Q)$, i.e. using \mathcal{P} at the very last step will result in a strictly better protocol.

Theorem 20. Let $X \in \{A, B\}$,

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

Then

$$\lim_{i \rightarrow \infty} p_X^*(\mathcal{Z}^i) \lesssim 0.7908 \dots$$

TODO: Argue why $C(\mathcal{P})$ was not such a great idea? I think this was because the cheat vector with the first component, e.g., maxed out, the abort probability was tiny for \mathcal{P} . Perhaps do the simulation and add the results.

5 Application: Strong Coin Flipping

Using ...

6 Security Proof | Asymptotic

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

Assumption 21. In protocol $\mathcal{P}(Q)$, 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 22. Let $a, b, c, x, y, z \in \{0, 1\}$. Consider a trio of quantum boxes, specified by projectors $\{M_{a|x}^A, M_{b|y}^B, M_{c|z}^C\}$ 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

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

such that

$$\begin{aligned} \Phi(|\psi\rangle) &= |\chi\rangle \otimes |\text{junk}\rangle, \\ \Phi\left(M_{d|t}^D |\psi\rangle\right) &= \Pi_{d|t}^D |\text{GHZ}\rangle \otimes |\text{junk}\rangle \quad \forall D \in \{A, B, C\}, \text{ and } d, t \in \{0, 1\} \end{aligned}$$

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 $\{\Pi_{a|x}^A, \Pi_{b|y}^B, \Pi_{c|z}^C\}$ 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 6.

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.

6.1 Cheat vectors optimisation using Semi Definite Programming

6.1.1 SDP when Alice self-tests

Asymptotic proof of Lemma 11. We prove Lemma 11 under Assumption 21. 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

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

where X represents the random input and A the output. Here, $|\Phi(x,a)\rangle_{IJ}$ are Bell states (see Equation (7)) 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

$$|\Psi_1\rangle := \frac{1}{2} \sum_{x,a,r \in \{0,1\}} |xa\rangle_{XA} |\Phi(x,a)\rangle_{IJ} |r\rangle_R |a \oplus x.r\rangle_S. \quad (6)$$

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

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

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.

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

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

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

where

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

Now we harness the freedom of applying an isometry to the post measured state (as observed above). We choose the local isometry in Lemma 22. 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,

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

where

$$|\Phi(x,a)\rangle_{IJ} = \frac{|00\rangle + (-1)^a (i)^x |11\rangle}{\sqrt{2}} \quad (7)$$

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

$$\begin{aligned} \Pi_i &:= \sum_{x,y \in \{0,1\}: x \oplus g = i} |xg\rangle \langle xg|_{XG} \otimes \mathbb{I}_{AIJRYD}, \\ \Pi^{\text{GHZ}} &:= \sum_{\substack{x,y \in \{0,1\}, \\ a,d \in \{0,1\}: a \oplus d \oplus 1 = xy \cdot (1 \oplus x \oplus y)}} |xyad\rangle \langle xyad|_{XYAD}. \end{aligned} \quad (8)$$

Then, we can write the cheat vector for Alice, i.e. the tuple of probabilities that Alice outputs 0, 1 and abort (see Definition 9), 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 (6), $\rho_2 \in XARG$ 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 (8). □

6.1.2 SDP when Bob self-tests

Proof of Algorithm 12. Denote by \mathcal{I} the protocol corresponding to Algorithm 7.

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 8, 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 21 holds. Then we can assume that Bob starts with the state

$$\rho_0 := \text{tr}_H(|\text{GHZ}\rangle \langle \text{GHZ}|_{HIJ}) \quad (9)$$

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 $\{M_{b|y}^Q, M_{c|z}^R\}$. Then using the isometry in Lemma 22, Bob can relabel his state (and without loss of generality, we can suppose Alice also relabels according to the aforementioned isometry) to get $\Phi_{PQR} |\psi\rangle_{PQR} = |\text{GHZ}\rangle_{HIJ} \otimes |\text{junk}\rangle_{PQR}$. Further, since $\Phi_{PQR}(M_{b|y}^Q \otimes M_{c|z}^R |\psi\rangle_{PQR}) = \Pi_{b|y}^I \Pi_{c|z}^J |\text{GHZ}\rangle_{HIJ} \otimes |\text{junk}\rangle_{PQR}$ 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).

1. Bob receiving s from Alice is modelled by introducing $\rho_1 \in SIJ$ satisfying $\text{tr}_S(\rho_1) = \rho_0$.
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 $\text{tr}_A(\rho_2) = \rho_1 \otimes \frac{\mathbb{I}_G}{2}$.
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

$$U := \sum_{y,x \in \{0,1\}} |y\rangle \langle y|_Y |x\rangle \langle x|_X \otimes (\mathbb{I}_B \otimes \Pi_{0|y}^I + X_B \otimes \Pi_{1|y}^I) \otimes (\mathbb{I}_C \otimes \Pi_{0|(1 \oplus y \oplus x)}^J + X_C \otimes \Pi_{1|(1 \oplus y \oplus x)}^J) \otimes \mathbb{I}_{ASG}. \quad (10)$$

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

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

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

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

to model the GHZ test. Let

$$\Pi^{\text{Test}} := \Pi^{\text{GHZ}} \Pi^{\text{sTest}}, \quad \bar{\Pi}^{\text{Test}} := \mathbb{I} - \Pi^{\text{Test}}. \quad (11)$$

One can then write the cheat vector for Bob, i.e. the tuple of probabilities that Bob outputs 0, 1 and abort (see Definition 9), as

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

To summarise, the final SDP is as follows: let $\rho_0 \in IJ$ be as defined in Equation (9), $\rho_1 \in SIJ$ and $\rho_2 \in AXSIJG$. Then,

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

subject to

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

where U is as defined in Equation (10) and the projectors as in Equation (11). □

6.2 The self-testing step [Discuss with Tom before writing]

Proposition 23. *For any implementation of the boxes and choice of $\delta > 0$, the joint probability that the test Ω passes and that the conclusion T is false is small, i.e. $\Pr[\Omega \cap \bar{T}] \leq \frac{1}{1-\delta+n\delta} \leq \frac{1}{n\delta}$ where the first upper-bound is tight.*

6.3 The continuity argument [Enter Jamie]

References

[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.