Improving the security of device-independent weak coin flipping

Extended abstract

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*

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.

Completeness 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 completeness 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].

Fact 1 ([SCA+11])**.** *There exists a DI-WCF protocol with $p_A^* = \cos^2 \pi/8$ and $p_B^* = 3/4$. Composing this protocol with itself yields a protocol with bias $\epsilon \leq 0.33664$.*

1 Contributions

In this work, we modify the above protocol to decrease its bias. We do this is three steps that we discuss below.

1.1 Changing the protocol in [SCA+11] to weak coin-flipping

Technically, the protocol in [SCA+11] is for strong coin flipping. As such, Bob is the only one who tests for cheating. That is, he always tests if Alice is cheating. Since weak coin flipping has the concept of a "winner", one can easily modify it such that only the winner gets tested. (This is a typical change to create a weak coin flipping protocol from a strong one.)

1.2 Pre-processing step: Self-testing

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 then *device-dependent* protocols, where for instance, the measurements are known by the honest player. However, we employ the concept of self-testing to stop Bob *or Alice* from applying such a strategy.

In the Silman2011 protocol, Alice and Bob are *supposed* to start with three boxes that implement the optimal GHZ strategy. We now illustrate the self-testing pre-processing step where Alice self-tests Bob.

Protocol 2 (Protocol with Alice self-testing)**.** *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 the protocol.*

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 3 (Informal. See the submitted manuscript for a formal statement)**.** *When Alice self-tests Bob, the cheating probabilities are*

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

1.3 Post-processing step: 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 polarized. It is relatively easy to see that composing a polarized coin-flipping 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. For instance, in a simple composition, Bob should not really accept to continue onto another subroutine 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! 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 4 (Alice and Bob's cheat vectors)**.** Given a protocol, 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.

Cheat vectors for (dishonest) Alice and $\mathbb{C}_A(\mathcal{I})$ are analogously defined.

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

$$v_B \cdot p_A^* + v_A \cdot p_B^* + v_\perp \cdot 0. \tag{2}$$

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.

1.4 Main result

Theorem 5. *There exists device-independent weak coin flipping protocols with bias approaching $\epsilon \approx 0.3148$. Under a convergence assumption (explained in the manuscript), the bias can be lowered to $\epsilon \approx 0.29104$.*

I think I should first fix the introduction in the main article and then import it here and prune as necessary.

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.

[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).

[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.