Ideas | Device Independent Weak Coin Flipping

Jamie Sikora, Thomas Van Himbeeck, Atul Singh Arora

March–... 2020

Abstract

(in progress)

Contents

1 Notes from the meetings

We follow two approaches with the common goal of constructing better protocols than the ones currently known.

1.1 Meeting/afterthoughts 1 | Monday

Jamie cooked up a bunch of protocols. Here's the rough list of things to pursue.

- Explore the viability of the J0 protocol, proposed by Jamie.

- We also had to determine if the "fully distrustful" article had a better bias for device independent (strong) coin flipping (EDIT: it did) than the device independent weak coin flipping protocol, by essentially the same authors.

- My task was to try to formalise the "formalism".

1.2 Meeting/afterthoughts 2 | Tuesday

There were a few ideas which were floated around by Jamie.

- F1 $\overset{?}{\Longleftrightarrow}$ F2 where F1 is the first framework and F2 is this one: since in DI protocols, we can start with any arbitrary state, consider these, followed by only classical information, then one last step where a part of the system is sent to the other.

- Understand the GHZ based protocols and re-express them using the F1 framework.

1.3 Meeting/afterthoughts 3 | Thursday [with Tom]

I read and presented the PRL protocol (without the security analysis). Then I tried presenting the equivalence of the two frameworks, intuitively.

- It looks like we can use Jamie's idea; call it the J3 protocol. It basically adds an additional verification step and since it is for weak CF, it doesn't affect the bias for Bob.

- Tom said that allowing quantum communication can be used to break the security because the quantum device can encode and send the input it was fed and since we allow the adversary to control the quantum channels, this makes them too powerful.

  - True but does learning the other player's input (maybe not always but sometimes) necessarily mean they have too much power? The only sane conclusion I can draw from this is that one has to necessarily allow classical information to be communicated but perhaps allowing quantum communication is also relevant.

  - Why isn't teleportation enough? In the honest case it is clearly enough. We want to show that in the dishonest case, the two models are equivalent. The two models being, one with classical + quantum communication the other with only classical communication. In both, we allow an arbitrary number of devices to be shared (which may be shielded to enforce the required structure for non-locality).
    Question: I have to show that a strategy in one can be equivalently expressed in the other.

1.4 Meeting/afterthoughts 4 | Monday

- I explained the framework; with things much better formalised and with higher clarity about how to impose the constraints for a cheating Bob.

  - Jamie pointed out how some "self-testing of unitaries" like scheme wouldn't quite work.

  - I forgot to ask him, thereafter, how to treat post measurement states of boxes (if at all this makes sense).

- Jamie then explained his abort based protocol which seemingly improves the bias; intuitively, this helps when you compose protocols: if you catch a player cheating, you just abort, instead of playing further and getting cheated further.

1.5 Meeting/afterthoughts 5 | Tuesday [with Tom]

- Jamie presented his new abort based protocols. Tom seemed to be ok with the ideas.

- I presented the proofs of Alice's and Bob's security. We compared it with Jamie's direct analysis and saw where they differed. We worked towards characterising the set $\mathcal{A}_c$, where in addition to the probability that Alice succeeds in convincing Bob, we keep track of the probability of her getting caught cheating.

- Tom also pointed out an oversight about testing Bob; we were sending both s_A and r_A to Bob, before testing him. We must delay the sending of r_A else he can pass the test with certainty.

2 F1 | A framework for Distrustful DI protocols

For the moment, consider referring to Collaborations->DI_WCF->F1 (and its subsections). TODO: Add the description here.

3 Sikora's Suppression Technique

We consider a somewhat restricted and simultaneous slightly more general class of weak coin flipping protocols.

why can't a disagreement be an abort?

Definition 1 (WCF with cheating abort). Weak Coin Flipping is a two player (Alice and Bob) interactive protocol wherein they wish to generate a random bit.

- After the interactions, each player outputs either A, B or $\perp$ (for abort). The two players, together, can output $\{AA, BB, AB, BA, * \perp, \perp *\}$ where $*$ is either A or B. Whenever the outcomes differ, the players agree to abort the protocol.
  (This is needed because to proceed, they must agree on what to do next; thus even though after the interaction their outputs may vary but they must, still, agree on how to proceed).

  - *Both Honest.* If the outputs are AA (or BB), Alice and Bob both conclude that Alice (or Bob) won. These two outputs, AA and BB, appear with probability $\frac{1}{2}$.
  - *Nomenclature for Security.* Suppose Alice is cheating (i.e. deviating from the protocol in some way). Let $\vec{q}_A = (\alpha_A, \alpha_B, \alpha_\perp = 1 - \alpha_A - \alpha_B)$, what we call a *cheat vector,* where

    * α_A denotes the probability of the outcome AA
    * α_B denotes the probability of the outcome BB
    * $\alpha_\perp$ denotes the probability of any other outcome, i.e. $\alpha_\perp = 1 - \alpha_A - \alpha_B$.

    in WCF, it can perhaps be ∵ the preferences are known. In SCF, you don't know if verify A helps the other one or not. If you find them cheating you abort.

    Let *cheat vector set* for Alice be $\mathcal{A}_c := \{\vec{q}_A :$ The probabilities correspond to some cheating strategy of Alice$\}$. Analogously, define $\vec{q}_B = (\beta_A, \beta_B, \beta_\perp = 1 - \beta_A - \beta_B)$ and the *cheat vector set* for Bob be $\mathcal{B}_c$.
  - *Standard WCF Security.* A WCF protocol has bias $\max\{\epsilon_A, \epsilon_B\}$ if $P_A^* = \frac{1}{2} + \epsilon_A$ where $P_A^* = \max_{(\alpha_A, \alpha_B, \alpha_\perp) \in \mathcal{A}_c} \alpha_A$ and $P_B^* = \frac{1}{2} + \epsilon_B$ where $P_B^* = \max_{(\beta_A, \beta_B, \beta_\perp) \in \mathcal{B}_c} \beta_B$.
    (This formalises the fact that the players have preferences; i.e. we only care about protecting Bob from Alice trying to bias towards outcome A and analogously, about protecting Alice from Bob trying to bias towards outcome B).

For clarity, we will be slightly redundant.

Definition 2 ((Standard) WCF). (Standard) Weak Coin Flipping is a two player (Alice and Bob) interactive protocol wherein they wish to generate a random bit.

- After the interactions, each player outputs either A or B. (If they catch the other player cheating, they declare themselves the winner). The two players, together, can output $\{AA, BB, AB, BA\}$. Whenever the outcomes differ, the honest player is declared the winner (assuming at least one is honest; when both are cheating, we can't say anything).

  - *Both Honest.* If the outputs are AA then Alice wins and if they are BB then Bob wins. These outputs, AA and BB, appear with probability $\frac{1}{2}$.
  - *Security.* Suppose Alice is cheating and Bob is honest. Then the highest probability (over all of Alice's cheating strategies) of Bob outputting A is given by P_A^*; Alice is assumed to output A. Analogously, the highest probability (over all of Bob's cheating strategies) of Alice outputting B (and thus winning) is given by P_B^*; Bob is assumed to output B.

TODO: I should be able to handle abort in standard WCF as well, just like we do in standard SCF; because if not, any non-trivial noise will break the protocol.

by this I mean no output for abort.

the device

Remark 3. A WCF protocol with cheating abort can be converted into a (standard) WCF protocol by mapping the outputs (of the protocol) A to A, B to B and $\perp$ to the honest player winning.

why did it WCF deliberty to abort irrelevant.

One might argue that since we don't repeat, abort is not really an issue ... but outputting, say, A might still increase the prob of the other success.

Definition 4 ((Standard) SCF). Strong Coin Flipping is a two player (Alice and Bob) interactive protocol wherein they wish to generate a random bit.

- After the interactions, each player outputs either 0, 1 or $\perp$ (for abort). The two players, together, can output $\{00, 11, 01, 10, * \perp, \perp *\}$ where $*$ is either 0 or 1. Whenever the outcomes differ, the players agree to abort the protocol.

  - *Both Honest.* If the outputs are 00 (or 11), Alice and Bob both conclude that Alice (or Bob) won. These two outputs, 00 and 11, appear with probability $\frac{1}{2}$. *(each).*
  - *(Standard SCF) Security.* Suppose Alice is cheating and Bob is honest. Then the maximum (over all possible cheating strategies of Alice) probability that Bob outputs x is given by p_{*x}. Analogously, when Bob is cheating and Alice is honest, then the maximum (over all possible cheating strategies of Bob) probability that Alice outputs y is given by p_{y*}.
    (E.g. p_{*x} indicates the probability that the protocol ends with both players agreeing to the outcome x because Alice can convince Bob of the outcome x and she, being the cheating player, can simply lie about her outcome being x).

Remark 5. A (standard) SCF protocol with security p_{*x}, p_{y*} can be converted into a (standard) WCF protocol with $P_A^* = p_{*0}$ and $P_B^* = p_{1*}$, by mapping 0 to Alice winning, 1 to Bob winning and $\perp$ to the honest player winning.

Lemma 6 (Single Suppression). *Consider a SCF protocol (see Definition 4) with cheating probabilities (p_{*x}, p_{y*}) for both x and y in $\{0, 1\}$ (assume the honest probabilities are $\frac{1}{2}$ and $\frac{1}{2}$). Suppose that $p_{*0} = p_{*1} > p_{0*} = p_{1*}$, i.e. the SCF protocol is unbalanced. Consider a WCF with cheating abort (see Definition 1) and the cheat vector sets $\mathcal{A}_c$ and $\mathcal{B}_c$. Using these, one can compose them to construct a SCF protocol with*

$$p'_{*x} = \max_{(\alpha_A, \alpha_B, \alpha_\perp) \in \mathcal{A}_c} \alpha_A p_{*x} + \alpha_B p_{x*}$$

and

$$p'_{y*} = \max_{(\beta_A, \beta_B, \beta_\perp) \in \mathcal{B}_c} \beta_A p_{y*} + \beta_B p_{*y}.$$

Further, $p'_{*x} \leq P_A^* p_{*x} + (1 - P_A^*) p_{x*}$ and $p'_{y*} \leq P_B^* p_{*y} + (1 - P_B^*) p_{y*}$, i.e. the new SCF protocol necessarily performs at least as good as the one obtained using a standard composition technique (which neglects aborts in WCF).

Proof. First, we construct a SCF protocol using the standard composition technique of a (standard) WCF protocol and a (standard) SCF protocol as in Figure 1. Suppose Alice cheats (Bob is honest). For the WCF protocol, she will try to bias towards A because this increases her chance of running the SCF protocol where she has the advantage (we assume $p_{*x} > p_{y*}$). Her success probability (i.e. of convincing Bob that the output is x), is thus, $p''_{*x} := P_A^* p_{*x} + (1 - P_A^*) p_{x*}$ where we used $\bar{p}_{*x} = p_{x*}$. Suppose Bob cheats (Alice is honest). Bob, similarly, would bias towards B to gain an advantage. His success probability (i.e. of convincing Alice that the output is y), is thus, $p''_{y*} := P_B^* p_{*y} + (1 - P_B^*) p_{y*}$ where we used $\bar{p}_{y*} = p_{*y}$.

We now compose a WCF protocol (with cheat abort) with a (standard) SCF protocol as in Figure 2. Suppose Alice cheats (Bob is honest). Further, suppose she uses some strategy corresponding to the cheat vector $(\alpha_A, \alpha_B, \alpha_\perp) \in \mathcal{A}_c$. Then, her success probability (i.e. of convincing Bob that the output is x) is $p'_{*x} = \alpha_A p_{*x} + \alpha_B p_{x*}$. The corresponding expression of success probability for a cheating Bob is $p'_{y*} = \beta_B p_{*y} + \beta_A p_{y*}$.

It remains to establish that $p'_{*x} \leq p''_{*x}$ and that $p'_{y*} \leq p''_{*y}$. To this end, from Remark 3 we know that $P_A^* = \max_{(\alpha_A, \alpha_B, \alpha_\perp) \in \mathcal{A}_c} \alpha_A$ and similarly $P_B^* = \max_{(\beta_A, \beta_B, \beta_\perp) \in \mathcal{B}_c} \beta_B$. We therefore have

$$\begin{aligned}
p''_{*x} = P_A^* p_{*x} + (1 - P_A^*) p_{x*} &= \left(P_A^* - (P_A^* - \alpha_A) + (P_A^* - \alpha_A)\right) p_{*x} + (1 - P_A^*) p_{x*} \\
&\geq \alpha_A p_{*x} + \left((P_A^* - \alpha_A) + (1 - P_A^*)\right) p_{x*} \qquad \because p_{*x} \geq p_{x*} \\
&= \alpha_A p_{*x} + (1 - \alpha_A) p_{*x} \\
&\geq \alpha_A p_{*x} + \alpha_B p_{*x} = p'_{*x}.
\end{aligned}$$

$\square$

4 Sikora's DI coin flipping protocols

Remark 7. TODO: Explain the standard box exchange description. | something like this: Alice and Bob shield each box so that it can't communicate with any other box.

Algorithm 8 (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.*
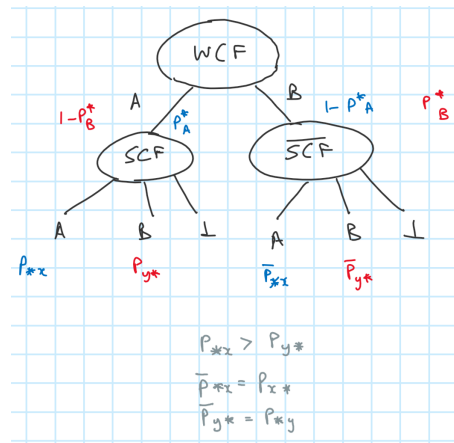

Figure 1: Standard composition of a WCF protocol and unbalanced SCF protocol produces a more balanced SCF protocol.

Figure 2: Composition of a WCF protocol with abort and an unbalanced SCF protocol produces a (possibly better and) more balanced SCF protocol.

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 $c = x \oplus a.r$ and sends c 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 $a \oplus g$.

4. Test round

   (a) Bob tests if $c = x$ or $c = x \oplus a$. 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.

Lemma 9 (Security of SCF). *(Silman et al. 2011) For Algorithm 8, $P_B^* \leq \frac{3}{4}$ and $P_A^* \leq \cos^2(\pi/8)$. 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.*

We now consider three variants of this protocol which should be at least as secure as the original.

Algorithm 10 (WCF). *Alice has one box and Bob has two boxes (in the security analysis, we let the cheating player distribute the boxes). All the boxes take one input and give one output.*

1. Alice chooses $s_A \in_R \{0, 1\}$ and inputs it into her box to obtain r_A. She chooses $a \in_R \{0, 1\}$ to compute $c = r_A \oplus s_A.a$ and sends c to Bob.

2. Bob chooses $b \in_R \{0, 1\}$ and sends it to Alice.

3. Alice sends s_A to Bob. They both compute the output $s_A \oplus b$.

4. Test Rounds.

   (a) If $s_A \oplus b = 0$ then Alice sends r_A to Bob.

       i. Bob checks if $c = r_A$ or $c = r_A \oplus s_A$. He aborts if the test fails.

       ii. Bob chooses $s_B, s_C \in_R \{0, 1\}$ such that $s_A \oplus s_B \oplus s_C = 1$ and then performs a GHZ using s_A, s_B, s_C as the inputs and r_A, r_B, r_C as the output from the three boxes (he measures his two boxes to obtain the two outcomes). He aborts if the test fails.

   (b) If $s_A \oplus b = 1$ then Bob sends his boxes to Alice (Alice tries to ensure he didn't tamper with the boxes).

       i. Alice chooses $s_B, s_C \in_R \{0, 1\}$ such that $s_A \oplus s_B \oplus s_c = 1$ and then performs a GHZ (like Bob did in the previous case). She aborts if the test fails.

Lemma 11 (Security of WCF). *Standard Security: Need to do Bob's part again; Alice's should carry over. Need to do both Alice and Bob to characterise the sets $\mathcal{A}_c$ and $\mathcal{B}_c$ in addition to the usual things.*

5   Acknowledgements

References

Silman, J. et al. (June 2011). "Fully Distrustful Quantum Bit Commitment and Coin Flipping". In: *Physical Review Letters* 106.22. DOI: 10.1103/physrevlett.106.220501.