Computational Hardness of Optimal Fair Computation: Beyond Minicrypt

Abstract. Secure multi-party computation allows mutually distrusting parties to compute securely over their private data. However, guaranteeing output delivery to honest parties when the adversarial parties may abort the protocol has been a challenging objective. As a representative task, this work considers two-party coin-tossing protocols with guaranteed output delivery, a.k.a., fair coin-tossing.

In the information-theoretic plain model, as in two-party zero-sum games, one of the parties can force an output with certainty. In the commitment-hybrid, any r-message coin-tossing protocol is $1/\sqrt{r}$-unfair, i.e., the adversary can change the honest party's output distribution by $1/\sqrt{r}$ in the statistical distance. Moran, Naor, and Segev (TCC–2009) constructed the first $1/r$-unfair protocol in the oblivious transfer-hybrid. No further security improvement is possible because Cleve (STOC–1986) proved that $1/r$-unfairness is unavoidable. Therefore, Moran, Naor, and Segev's coin-tossing protocol is optimal. However, is oblivious transfer necessary for optimal fair coin-tossing?

Maji and Wang (CRYPTO–2020) proved that any coin-tossing protocol using one-way functions in a black-box manner is at least $1/\sqrt{r}$-unfair. That is, optimal fair coin-tossing is impossible in Minicrypt. Our work focuses on tightly characterizing the hardness of computation assumption necessary and sufficient for optimal fair coin-tossing within Cryptomania, outside Minicrypt. Haitner, Makriyannia, Nissim, Omri, Shaltiel, and Silbak (FOCS–2018 and TCC–2018) proved that better than $1/\sqrt{r}$-unfairness, for any constant r, implies the existence of a key-agreement protocol.

We prove that any coin-tossing protocol using public-key encryption (or, multi-round key agreement protocols) in a black-box manner must be $1/\sqrt{r}$-unfair. Next, our work entirely characterizes the additional power of secure function evaluation functionalities for optimal fair coin-tossing. We augment the model with an idealized secure function evaluation of f, a.k.a., the f-hybrid. If f is complete, that is, oblivious transfer is possible in the f-hybrid, then optimal fair coin-tossing is also possible in the f-hybrid. On the other hand, if f is not complete, then a coin-tossing protocol using public-key encryption in a black-box manner in the f-hybrid is at least $1/\sqrt{r}$-unfair.

Keywords: Fair computation, Optimal fair coin-tossing, Cryptomania, Black-box separation, Hardness of computation results, Secure function evaluation functionalities.

1 Introduction

Secure multi-party computation [75, 32] allows mutually distrusting parties to compute securely over their private data. However, guaranteeing output delivery to honest parties when the adversarial parties may abort during the protocol execution has been a challenging objective. A long line of highly influential works has undertaken the task of defining security with guaranteed output delivery (i.e., *fair computation*) and fairly computing functionalities [34, 11, 35, 10, 1, 5, 40, 3, 60, 4, 2, 14]. This work considers the case when honest parties are not in the majority. In particular, as is standard in this research, the sequel relies on the representative task of two-party secure coin-tossing, an elegant functionality providing uncluttered access to the primary bottlenecks of achieving security in any specific adversarial model.

In the *information-theoretic plain model*, one of the parties can fix the coin-tossing protocol's output (using attacks in two-player zero-sum games, or games against nature [65]). If the parties additionally have access to the commitment functionality (a.k.a., the information-theoretic *commitment-hybrid*), an adversary is forced to follow the protocol honestly (otherwise, the adversary risks being identified), or abort the protocol execution prematurely. Against such adversaries, referred to as *fail-stop adversaries* [20], there are coin-tossing protocols [12, 13, 6, 19] where a fail-stop adversary can change the honest party's output distribution by at most $\mathcal{O}(1/\sqrt{r})$, where r is the round-complexity of the protocol. That is, these protocols are $\mathcal{O}(1/\sqrt{r})$-*insecure*. In a ground-breaking result, Moran, Naor, and Segev [61] constructed the first secure coin-tossing protocol in the oblivious transfer-hybrid [67, 68, 25] that is $\mathcal{O}(1/r)$-insecure. No further security improvements are possible because Cleve [19] proved that $\mathcal{O}(1/r)$-insecurity is unavoidable; hence, the protocol by Moran, Naor, and Segev is *optimal*.

	Secure Construction	Adversarial Attack
Pessiland	Fail-stop Adversary:	In General: constant-unfair [38]
		Fail-stop Adversary: $1/\sqrt{r}$-unfair [20]
Minicrypt	One-way Functions: $1/\sqrt{r}$-unfair [12, 13, 6, 19]	$1/\sqrt{r}$-unfair [59]
Cryptomania	Public-key Encryption:	$1/\sqrt{r}$-unfair [This work]
	PKE + f-hybrid, $f \not\to$ OT:	$1/\sqrt{r}$-unfair [This work]
	Oblivious Transfer: $1/r$-unfair [61]	$1/r$-unfair [19]

Fig. 1. The first column summarizes of the most secure fair coin-tossing protocols in Impagliazzo's worlds [43]. Corresponding to each of these worlds, the second column has the best attacks on these fair coin-tossing protocols.

Incidentally, all fair computation protocols (not just coin-tossing, see, for example, [34, 11, 35, 10, 1, 5, 40, 3, 60, 4, 2, 14]) rely on the oblivious transfer functionality to achieve $\mathcal{O}(1/r)$-insecurity. A fundamental principle in theoretical cryptography is to securely realize cryptographic primitives based on the minimal computational hardness assumptions. Consequently, the following question is natural.

Is oblivious transfer necessary for optimal fair computation?

Towards answering this fundamental research inquiry, recently, Maji and Wang [59] proved that any coin-tossing protocol that uses one-way functions in a *black-box manner* [45, 69, 7] must incur $\Omega(1/\sqrt{r})$-insecurity. This result proves the qualitative optimality of the coin tossing protocols of [12, 13, 6, 19] in Minicrypt [43] because the commitment functionality is securely realizable by the black-box use of one-way functions [62, 63, 39]. Consequently, the minimal hardness of computation assumption enabling optimal fair coin-tossing must be outside Minicrypt.

Summary of our results. This work studies the insecurity of fair coin-tossing protocols outside Minicrypt, within (various levels of) Cryptomania [43]. Our contributions are two-fold.

1. First, we generalize the (fully) black-box separation of Maji and Wang [59] to prove that any coin-tossing protocol using public-key encryption in a fully black-box manner must be $\Omega(1/\sqrt{r})$-insecure.
2. Finally, we prove a dichotomy for two-party secure (possibly, *randomized* output) function evaluation functionalities. For any secure function evaluation functionality f, either (A) optimal fair coin-tossing exists in the information-theoretic f-hybrid, or (B) any coin-tossing protocol in the f-hybrid, even using public-key encryption algorithms in a black-box manner, is $\Omega(1/\sqrt{r})$-insecure.

Our hardness of computation results hold even for a game-theoretic definition of fairness as well (which extends to the stronger simulation-based security definition). Section 1.1 summarizes our contributions. As shown in Figure 1, our results further reinforce the widely-held perception that oblivious transfer is necessary for optimal fair coin-tossing. Our work nearly squeezes out the entire remaining space left open in the state-of-the-art after the recent breakthrough of [59], which was the first advancement on the quality of the attacks on fair coin-tossing protocols since [20] after almost three decades. However, there are fascinating problems left open by our work; Section 6 discusses one.

Positioning the technical contributions. Information-theoretic lower-bounding techniques that work in the plain model and also extend to the f-hybrid are rare. Maji and Wang [59] proved that optimal coin-tossing is impossible in the information-theoretic model even if parties can access a random oracle. This work extends the potential-based approach of [59] to f-hybrid information-theoretic models, such that oblivious transfer is impossible in the f-hybrid and parties additionally have access to a public-key encryption oracle.

	0	1	2
0	z_0	z_0	z_1
1	z_3	z_4	z_1
2	z_3	z_2	z_2

Fig. 2. The Kushilevitz Function, where Alice holds input $x \in \{0, 1, 2\}$ and Bob holds input $y \in \{0, 1, 2\}$. For example, the output is z_0 if $x = 0$ and $y \in \{0, 1\}$.

For the discussion below, consider f to be the Kushilevitz function (see Figure 2). One cannot realize this function securely in the information-theoretic plain model even against honest-but-curious adversaries [51, 9, 57, 50]. Furthermore, oblivious transfer is impossible in the f-hybrid [47, 48]. The characterization of the *exact power* of making ideal f-invocations is not entirely well-understood.

Invocations of the ideal f-functionality are *non-trivially useful*. For example, one can realize the commitment functionality in the f-hybrid model [58] (even with Universally Composable (UC) security [15, 16] against malicious adversaries). The f-functionality is also known to securely implement other secure function evaluation functionalities as well [71]. All these functionalities would otherwise be impossible to securely realize in the plain model [17, 52, 66]. Consequently, it is plausible that one can even implement optimal fair coin-tossing without implementing oblivious transfer in the f-hybrid model.

Our technical contribution is an information-theoretic lower-bounding technique that precisely characterizes the power of any f-hybrid vis-à-vis its ability to implement optimal fair coin-tossing. The authors believe that these techniques shall be of independent interest to characterize the power of performing ideal f-invocations in general.

1.1 Our Contribution

This section provides an informal summary of our results and positions our contributions relative to the state-of-the-art. To facilitate this discussion, we need to introduce a minimalistic definition of coin-tossing protocols. An (r, X)-*coin-tossing* protocol is a two-party r-message interactive protocol where parties agree on the final output $\in \{0, 1\}$, and the expected output of an honest execution of the protocol is X. A coin-tossing protocol is ϵ-*unfair* if one of the parties can change the honest party's output distribution by ϵ (in the statistical distance).

Maji and Wang [59] proved that the existence of optimal coin-tossing protocols is outside Minicrypt [43], where one-way functions and other private-key cryptographic primitives exist (for example, pseudorandom generator [44, 41, 42], pseudorandom function [30, 31], pseudorandom permutation [55], statistically binding commitment [62], statistically hiding commitment [63, 39],

zero-knowledge proof [33], and digital signature [64, 70]). Public-key cryptographic primitives like public-key encryption, (multi-message) key-agreement protocols, and secure oblivious transfer protocol are in Cryptomania [45] (outside Minicrypt). Although the existence of a secure oblivious transfer protocol suffices for optimal fair coin-tossing, it was unknown whether weaker hardness of computation assumptions (like public-key encryption and (multi-message) key-agreement protocols [28]) suffice for optimal fair coin-tossing or not. Previously, Haitner, Makriyannis, Nissim, Omri, Shaltiel, and Silbak [37, 36], for any constant r, prove that r-message coin-tossing protocols imply key-agreement protocols, if they are less than $1/\sqrt{r}$-insecure.

Result I. Towards this objective, we prove the following result.

Corollary 1 (Separation from Public-key Encryption). *Any (r, X)-coin-tossing protocol that uses a public-key encryption scheme in a fully black-box manner is $\Omega(X(1-X)/\sqrt{r})$-unfair.*

We emphasize that X may depend on the message complexity r of the protocol, which, in turn, depends on the security parameter. For example, consider an ensemble of fair coin-tossing protocols with round complexity r and expected output $X = 1/r$. This result shows a fail-stop adversary that changes the honest party's output distribution by $1/r^{3/2}$ in the statistical distance.

This hardness of computation result extends to the fair computation of any multi-party functionality (possibly with inputs) such that the output has some entropy, and honest parties are not in the majority (using a standard partition argument). At a high level, this result implies that relying on stronger hardness of computation assumptions like the existence of public-key cryptography provides no "fairness-gains" for coin-tossing protocols than only using one-way functions.

This result's heart is the following *relativized separation* in the information-theoretic setting (refer to Theorem 5). There exists an oracle PKE_n [56] that enables the secure public-key encryption of n-bit messages. However, we prove that any (r, X)-coin-tossing protocol where parties have oracle access to the PKE_n oracle (with polynomial query complexity) is $\Omega(X(1-X)/\sqrt{r})$-unfair. This relativized separation translates into a fully black-box separation using by-now-standard techniques in this field [69]. Conceptually, this black-box separation indicates that optimal fair coin-tossing requires a hardness of computation assumption that is *stronger* than the existence of a secure public-key encryption scheme.

Gertner, Kannan, Malkin, Reingold, and Vishwanathan [28] showed that the existence of a public-key encryption scheme with additional (seemingly innocuous) properties (like the ability to efficiently sample a public-key without knowing the private-key) enables oblivious transfer. Consequently, our oracles realizing public-key encryption must avoid any property enabling oblivious transfer (even unforeseen ones). This observation highlights the subtlety underlying our technical contributions. For example, our set of oracles permit testing whether a public-key or cipher-text is valid or not. Without this test, oblivious transfer and, in turn, optimal fair coin-tossing is possible. Surprisingly, these test oracles are also sufficient to rule out the possibility of oblivious transfer.

Since public-key encryption schemes imply key agreement protocols, our results prove that optimal fair coin-tossing is black-box separated from key agreement protocols as well.

Result II. Let $f\colon X \times Y \to \mathbb{R}^Z$ be a two-party secure symmetric function evaluation functionality, possibly with randomized output. The function takes private inputs x and y from the parties and samples an output $z \in Z$ according to the probability distribution $p_f(z|x,y)$. The *information-theoretic f-hybrid* is an information-theoretic model where parties have additional access to the (unfair) f-functionality.[1] As an aside, we highlight that the fair f-hybrid (where the adversary cannot block output delivery to the honest parties), for any f where both parties influence the output, straightforwardly yields *perfectly or statistically secure* fair coin-tossing protocol.[2]

Observe that if f is the (symmetrized) oblivious transfer functionality,[3] then the Moran, Naor, and Segev protocol [61] is an optimal fair coin-tossing protocol in the (unfair) f-hybrid. More generally, if f is a functionality such that there is an oblivious transfer protocol in the f-hybrid, one can emulate the Moran, Naor, and Segev optimal coin-tossing protocol; consequently, optimal coin-tossing exists in the f-hybrid. Kilian [48] characterized all functions f such that there exists a secure oblivious transfer protocol in the f-hybrid, referred to as *complete* functions.

Our work explores whether a function f that is *not complete* may enhance the security of fair coin-tossing protocols.

Corollary 2 (Dichotomy of Functions). *Let f be an arbitrary 2-party symmetric function evaluation functionality, possibly with randomized output. Then, exactly one of the following two statements holds.*

---

[1] The functionality delivers the output to the adversary first. If the adversary wants, it can abort the protocol and block the output delivery to the honest parties. Otherwise, if the adversary wants, it can permit the delivery of the output to the honest parties and continue with the protocol execution.

[2] Suppose $f = \text{XOR}$. In a fair f-hybrid, the adversary cannot block the output delivery to the honest parties. So, parties input random bits to the f-functionality and agree on the output. This protocol has 0-insecurity. A similar protocol (using a deterministic extractor for independent small-bias sources) can extract the fair output from any f where both parties have influence on the output distribution. Consider the following "collaborative randomness generation" followed by "extraction" protocol. (a) Invoke (in parallel) a bidirectional influence functionality multiple times with random inputs. The output of each invocation in *not* entirely determined by one of the parties. Consequently, these samples have average min-entropy. (b) Non-interactively, parties use these fair output samples to extract this entropy to obtain the (common) fair coin toss (using convolution/XOR, or traversal of an appropriate expander graph).

[3] In the symmetrized oblivious transfer functionality, the sender has input $(x_0, x_1) \in \{0,1\}^2$, and the receiver has input $(b, r) \in \{0,1\}^2$. The symmetric oblivious transfer functionality returns $x_b \oplus r$ to both the parties. If the receiver picks $r \xleftarrow{\$} \{0,1\}$, then this functionality hides the receiver's choice bit b from the sender.

1. *For all $r \in \mathbb{N}$ and $X \in [0,1]$, there exists an optimal (r, X)-coin-tossing protocol in the f-hybrid (a.k.a., $\mathcal{O}(1/r)$-unfair protocol).*
2. *Any (r, X)-coin-tossing protocol that uses public-key encryption protocols in a black-box manner in the f-hybrid is $\Omega(X(1-X)/\sqrt{r})$-unfair.*

For example, Corollary 1 is implied by the stronger version of our result by using a constant-valued f, a trivial function evaluation. For more details, refer to Theorem 6. In our model, we emphasize that parties can perform an arbitrary number of f-invocations in parallel in every round.

Let us further elaborate on our results. Consider a function f that has a secure protocol in the information-theoretic plain model, referred to as *trivial* functions. For deterministic output, trivial functions' full characterization is known [51, 9, 57, 50]. For randomized output, the characterization of trivial functions is a long-standing open problem.[4] Observe that trivial functions are definitely not complete; otherwise, a secure oblivious transfer protocol shall exist in the information-theoretic plain model, which is impossible. For every $t \in \mathbb{N}$, there are functions f_t such that any secure protocol for f_t requires t rounds of interactive communication in the information-theoretic plain model. For the randomized output case, the authors know of functions such that $|X| = |Y| = 2$ and $|Z| = (t+1)$ that need t-round protocols for secure computation, which is part of ongoing independent research. Compiling out the f_t-hybrid using such a t-round secure computation protocol allows only for an $\Theta\big(X(1-X)/\sqrt{rt}\big)$-insecurity, which yields a useless bound for $t = \Omega(r)$. Consequently, compiling out the trivial functions is inadequate.

It is also well-known that functions of *intermediate* complexity exist [51, 9, 57, 50], which are neither complete nor trivial (for example, the Kushilevitz function, refer to Figure 2). In fact, there are randomized functions of intermediate complexity such that $|X| = |Y| = 2$ and $|Z| = 3$ [24]. For example,

$$\frac{1}{54}\begin{pmatrix}(18,18,18) & (36,12,6)\\ (21,3,30) & (42,2,10)\end{pmatrix}.$$

Our result claims that even an intermediate function f is useless for optimal fair coin-tossing; it is as useless as one-way functions or public-key encryption. Therefore, our results' technical approach must treat each f-hybrid invocation as one step in the protocol. We highlight that the intermediate functions are useful in securely realizing other non-trivial functionalities as well [58, 71]. However, for fair coin-tossing, they are useless.

Before we move ahead, the authors feel that it is instructive to elaborate on what our paper *does not* prove. Let f be an intermediate function, and "sh-f" represents the hardness of computation assumption that there exists a semi-honest secure protocol for function f. We *do not* rule out the possibility that sh-f implies optimal coin-tossing protocols. Our result only proves that the f-hybrid cannot help construct optimal coin-tossing protocols. The existence of a protocol for an intermediate f may have significantly additional implicit consequences, which may, in turn, imply optimal coin-tossing protocol construction.

[4] Even for perfect security, the characterization of randomized trivial functions is open.

In particular, such a result would imply the separation of sh-f and sh-OT, which is one of the most fundamental open problems in this field.[5] Refer to Section 6 for further elaboration.

1.2 Prior Works

Deterministic secure function evaluation. In this paper, we focus on two-party secure function evaluation functionalities that provide the same output to the parties. Consider a deterministic function $f\colon X \times Y \to Z$. The *unfair ideal functionality* implementing f takes as input x and y from two parties and delivers the output $f(x, y)$ to the adversary. The adversary may choose to block the output delivery to the honest party, or permit the delivery of the output to the honest party.

In this document, we consider security against a semi-honest information-theoretic adversary, i.e., the adversary follows the protocol description honestly but is curious to find additional information about the other party's private input. There are several natural characterization problems in this scenario. The functions that have perfectly secure protocols in the information-theoretic plain model, a.k.a., the *trivial functions*, are identical to the set of *decomposable functions* [51, 9]. For every $t \in \mathbb{N}$, there are infinitely many functions that require t-rounds for their secure evaluation. Interestingly, relaxing the security from perfect to statistical security, does not change this characterization [57, 50].

Next, Kilian [47] characterized all deterministic functions f that enable oblivious transfer in the f-hybrid, the *complete functions*. Any functions that has an "embedded OR-minor" (refer to Definition 4) is complete. Such functions, intuitively, are the most powerful functions that enable general secure computation of arbitrary functionalities.

The sets of trivial and complete functions are not exhaustive (for $|Z| > 3$ [18, 49]). There are functions of *intermediate* complexity, which are neither trivial nor complete (see, for example, Figure 2). The power of the f-hybrid, for an intermediate f, was explored by [71] using restricted forms of protocols.

Randomized secure function evaluation. A two-party randomized function $f(x, y)\colon X \times Y \to \mathbb{R}^Z$ is a function that, upon receipt of the inputs x and y, samples an output according to the distribution $p_f(z|x, y)$ over the samples space Z. Kilian [48] characterized all complete randomized functions. Any function that has an "embedded generalized OR-minor" (refer to Definition 4) is

[5] For an insight into some of the bottlenecks encountered for this problem, consider an oracle that allows the computation of f using t interactive rounds. If f is a function where both parties influence the output, then there exists a round where one party can predict the final output with $1/t$ additional advantage than the other party. The primary origin of the non-triviality is the fact that the oblivious transfer protocol can prescribe the parties to partially run the oracle-protocol evaluating f up to this round. This additional advantage in output prediction of one party, for example, may be amplified into an oblivious transfer protocol using the techniques of [23]. Consequently, this problem is extremely subtle and one of the most challenging open problems in this field.

complete. Recently, [24] characterized functions with 2-round protocols. Beyond these characterization, not much is known in the literature and most fundamental characterization problems in this field are essentially open. However, there is sufficient evidence that the landscape of randomized secure function evaluation is extremely rich and fascinating. For example, even when $|X| = |Y| = 2$ the authors know of functions (with $|Z| = (t+1)$) that require t rounds of communication, for any $t \in \mathbb{N}$. Furthermore, even for $|X| = |Y| = 2$ and $|Z| = 3$, there are random function evaluations that are of intermediate complexity [24].

In the field of black-box separation, the seminal work of Impagliazzo and Rudich [45] first proposed the notion of black-box separation between cryptographic primitives. Since then, there has been many influential works [72, 74, 28, 27, 29, 26, 69] in this line of research. Below, we elaborate on a few works that are most relevant to us.

Firstly, for the fair coin-tossing in the random oracle model, the work of Dachman-Soled, Lindell, Mahmoody, and Malkin [21] showed that when the message complexity is small, random oracle can be compiled away and hence is useless for fair coin-tossing. In another work, Dachman-Soled, Mahmoody, and Malkin [22] studied a restricted type of protocols that they called "function-oblivious" and showed that for this particular type of protocols, random oracles cannot yield optimal fair coin-tossing. Recently, Maji and Wang [59] resolved this problem in the full generality. They showed that any r-message coin-tossing protocol in the random oracle model must be $\Omega(1/\sqrt{r})$-unfair.

In a recent work of Haitner, Nissim, Omri, Shaltiel, and Silbak [37] and Haitner, Makriyannis, and Omri [36], they proved that, for any constant r, the existence of an r-message fair coin-tossing protocol that is more secure than $1/\sqrt{r}$ implies the existence of (infinitely often) key agreement protocols.

1.3 Technical Overview

In this section, we present a high-level overview of our proofs. We start by recalling the proofs of Maji and Wang [59].

Before we begin, we need to introduce the notion of Alice and Bob's *defense coins*. At any instance of the protocol evolution, Alice has a private defense coin $\in \{0,1\}$, referred to as the Alice defense coin, which she outputs if Bob aborts the protocol. Similarly, Bob has a Bob defense coin. When Alice prepares a next message of the protocol, she updates her defense coin. However, when Bob prepares a next message of the protocol, Alice's defense coin remains unchanged. Analogously, Bob updates his defense coin when preparing his next messages in the protocol.

Abstraction of Maji and Wang [59] Technique. Consider an arbitrary fair coin-tossing protocol $\pi^{\mathcal{O}}$ where Alice and Bob have black-box access to some oracle $\mathcal{O}$. In their setting, $\mathcal{O}$ is a random oracle. Let r and X be the message complexity and the expected output of this protocol. They used an inductive approach to prove this protocol is $(c \cdot X(1-X)/\sqrt{r})$-insecure as follows ($c$ is a universal constant).

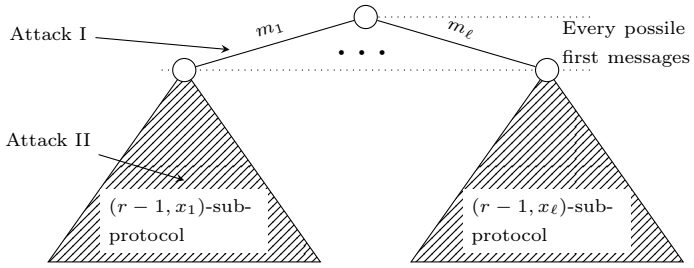

Fig. 3. An intuitive illustration of the approach of Maji and Wang [59].

For every possible first message of this protocol, they consider two attacks (refer to Figure 3). Firstly, parties can attack by immediately abort upon this first message. Secondly, parties can defer their attack to the remaining sub-protocol, which has only $r-1$ messages. Suppose when the first message is m_i, the remaining sub-protocol has expected output x_i. Additionally, the expectation of Alice and Bob defense is a_i and b_i. The effectiveness of the first attack is precisely

$$|x_i - a_i| + |x_i - b_i|,$$

where $|x_i - a_i|$ is the change of Alice's output if Bob aborts, and analogously, $|x_i - b_i|$ is the change of Bob's output if Alice aborts. On the other hand, by the inductive hypothesis, we know the effectiveness of the second attack is at least

$$c \cdot x_i(1-x_i)/\sqrt{r-1}.$$

Now, they employed the results of [46] (refer to Imported Lemma 1) and show that the maximum of these two quantities is lower bounded by

$$\frac{c}{\sqrt{r}} \cdot \left(x_i(1-x_i) + (x_i - a_i)^2 + (x_i - b_i)^2\right).$$

Define potential function $\Phi(x, a, b) := x(1-x) + (x-a)^2 + (x-b)^2$. Maji and Wang noted that if Jensen's inequality holds, i.e.,

$$\mathop{\mathrm{E}}_i\left[\Phi(x_i, a_i, b_i)\right] \geq \Phi\left(\mathop{\mathrm{E}}_i\left[x_i\right], \mathop{\mathrm{E}}_i\left[a_i\right], \mathop{\mathrm{E}}_i\left[b_i\right]\right), \tag{1}$$

then the proof is complete. This is because the overall effectiveness of the attack is lower bounded by

$$\begin{aligned}
&\mathop{\mathrm{E}}_i\left[\frac{c}{\sqrt{r}} \cdot \left(x_i(1-x_i) + (x_i - a_i)^2 + (x_i - b_i)^2\right)\right] \\
\geq &\frac{c}{\sqrt{r}} \cdot \Phi\left(\mathop{\mathrm{E}}_i\left[x_i\right], \mathop{\mathrm{E}}_i\left[a_i\right], \mathop{\mathrm{E}}_i\left[b_i\right]\right) \\
\geq &\frac{c}{\sqrt{r}} \cdot \mathop{\mathrm{E}}_i\left[x_i\right]\left(1 - \mathop{\mathrm{E}}_i\left[x_i\right]\right)
\end{aligned}$$

$$\geq \frac{c}{\sqrt{r}} \cdot X(1-X).$$

To prove Equation 1, they noted that $\Phi(x, a, b)$ could be rewritten as

$$\Phi(x, a, b) = x + (x - a - b)^2 - 2ab.$$

Observe that x and $(x - a - b)^2$ are convex functions, and hence Jensen's inequality holds. The only problematic term is ab. To resolve this, they noted that suppose we have the following guarantee.

Conditioned on the partial transcript, Alice private view and Bob private view are (close to) independent.

Then we shall have $\mathop{\mathrm{E}}_{i}[a_i b_i] \approx \mathop{\mathrm{E}}_{i}[a_i] \mathop{\mathrm{E}}_{i}[b_i]$ (refer to Claim 1).[6] Consequently, Equation 1 shall hold and the proof is done.

Note that the argument thus far is oblivious to the fact that the oracle in use is a random oracle. For any oracle $\mathcal{O}$, if we have the guarantee above, this proof will follow.

In particular, when the oracle in use is the random oracle, Maji and Wang observed that, standard techniques (namely, the heavy querier [8]) do ensure that Alice private view and Bob private view are (close to) independent. This completes their proof.

Extending to f-hybrid. When f is a complete function, one can build oblivious protocol in the f-hybrid model and, consequently, by the MNS protocol [61], optimal fair coin-tossing does exist in the f-hybrid model.

On the other hand, if f is not complete, Kilian [48] showed that f must satisfy the cross product rule (refer to Definition 4). This implies that conditioned on the partial transcript, which includes ideal calls to f, Alice and Bob private view are (perfectly) independent (refer to Lemma 3). Therefore, the proof strategy of Maji and Wang [59] is applicable.

Extending to Public-key Encryption. Our proof for the public-key encryption follows from the ideas of Mahmoody, Maji, and Prabhakaran [56]. First, we define a collection of oracles PKE_n (refer to Section 5.1), with respect to which public-key encryption exists. To prove that optimal fair coin-tossing protocol does not exist, it suffices to ensure that Alice and Bob private view are (close to) independent. However, since with the help of PKE_n oracle, Alice and Bob can agree on a secret key such that a third party, Eve, who sees the transcript and may ask polynomially many queries to the oracle, cannot learn any information about the key. It is impossible to ensure the independence of the private views by only invoking a public algorithm.

To resolve this, [56] showed that one could compile any protocol π in the PKE_n oracle to be a new protocol π' in the PKE_n oracle where parties never

[6] In particular, if Alice private view and Bob private view are *perfectly* independent, we shall have $\mathop{\mathrm{E}}_{i}[a_i b_i] = \mathop{\mathrm{E}}_{i}[a_i] \mathop{\mathrm{E}}_{i}[b_i]$.

query the decryption oracle (refer to Imported Theorem 1). This compiler satisfies that given a local view of Alice (resp., Bob) in protocol π, one could simulate the local view of Alice (resp., Bob) in protocol π' and vice versa. Therefore, instead of considering a fair coin-tossing protocol in the PKE_n oracle model, one could consider a fair coin-tossing protocol in the PKE_n oracle model where parties never query the decryption oracle. And [56] showed that, when the parties do not call the decryption oracle, there does exist a public algorithm, namely the common information learner, who can find all the correlation between Alice and Bob (refer to Imported Theorem 2). And conditioned on the partial transcript with the additional information from the common information learner, Alice and Bob private view are (close to) independent. Therefore, we can continue with the proof-strategy of Maji and Wang [59].

2 Preliminaries

For a random function $f\colon \mathcal{X} \to \mathcal{Y}$, we shall use $f(x;s)$ for f evaluated with input x and randomness s.

We use uppercase letters for random variables, (corresponding) lowercase letters for their values, and calligraphic letters for sets. For a joint distribution (A, B), A and B represent the marginal distributions, and $A \times B$ represents the product distribution where one samples from the marginal distributions A and B independently. For two random variables A and B distributed over a (discrete) sample space Ω, their *statistical distance* is defined as $\mathsf{SD}(A, B) := \frac{1}{2} \cdot \sum_{\omega \in \Omega} |\Pr[A = w] - \Pr[B = w]|$.

For a sequence $(X_1, X_2, \ldots)$, we use $X_{\leq i}$ to denote the joint distribution $(X_1, X_2, \ldots, X_i)$. Similarly, for any $(x_1, x_2, \ldots) \in \Omega_1 \times \Omega_2 \times \cdots$, we define $x_{\leq i} := (x_1, x_2, \ldots, x_i) \in \Omega_1 \times \Omega_2 \times \cdots \times \Omega_i$. Let $(M_1, M_2, \ldots, M_r)$ be a joint distribution over sample space $\Omega_1 \times \Omega_2 \times \cdots \times \Omega_r$, such that for any $i \in \{1, 2, \ldots, n\}$, M_i is a random variable over Ω_i. A (real-valued) random variable X_i is said to be $M_{\leq i}$ *measurable* if there exists a deterministic function $f\colon \Omega_1 \times \cdots \times \Omega_i \to \mathbb{R}$ such that $X_i = f(M_1, \ldots, M_i)$. A random variable $\tau\colon \Omega_1 \times \cdots \times \Omega_r \to \{1, 2, \ldots, r\}$ is called a *stopping time*, if the random variable $\mathbb{1}_{\tau \leq i}$ is $M_{\leq i}$ measurable, where $\mathbb{1}$ is the indicator function. For a more formal treatment of probability spaces, σ-algebras, filtrations, and martingales, refer to, for example, [73].

The following inequality shall be helpful for our proof.

Theorem 1 (Jensen's inequality). *If f is a multivariate convex function, then $\mathrm{E}\left[f\left(\vec{X}\right)\right] \geq f\left(\mathrm{E}\left[\vec{X}\right]\right)$, for all probability distributions $\vec{X}$ over the domain of f.*

In particular, $f(x, y, z) = (x - y - z)^2$ is a tri-variate convex function where Jensen's inequality applys.

3 Fair Coin-tossing Protocol in the f-hybrid Model

Let $f\colon \mathcal{X} \times \mathcal{Y} \to \mathcal{Z}$ be an arbitrary (possibly randomized) function. As standard in the literature, we shall restrict to f such that the input domain $\mathcal{X}$ and $\mathcal{Y}$ and

the range $\mathcal{Z}$ are of constant size. A two-party protocol in the f-hybrid model is defined as follows.

Definition 1 (f-hybrid Model [15, 53]). *A protocol between Alice and Bob in the f-hybrid model is identical to a protocol in the plain model except that both parties have access to a trusted party realizing f. At any point during the execution, the protocol specifies which party is supposed to speak.*

- ***Alice/Bob message.*** *If Alice is supposed to speak, she shall prepare her next message as a deterministic function of her private randomness and the partial transcript. If Bob is supposed to speak, his message is prepared in a similar manner.*
- ***Trusted party message.*** *At some point during the execution, the protocol might specify that the trusted party shall speak next. In this case, the protocol shall also specify a natural number ℓ, which indicates how many instances of f should the trusted party compute. Alice (resp., Bob) will prepare her inputs $\vec{x} = (x_1, \ldots, x_\ell)$ (resp., $\vec{y} = (y_1, \ldots, y_\ell)$) and send it privately to the trusted party. The trusted party shall compute $(f(x_1, y_1), \ldots, f(x_\ell, y_\ell))$ and send it as the next message.*

In this paper, we shall restrict to fail-stop adversarial behavior.

Definition 2 (Fail-stop Attacker in the f-hybrid Model). *A fail-stop attacker follows the protocol honestly and might prematurely abort. She might decide to abort when it is her turn to speak. Furthermore, during the trusted party message, she shall always receive the trusted party message first and, based on this message, decide whether to abort or not. If she decides to abort, this action prevents the other party from receiving the trusted party message.*

In particular, we shall focus on fair coin-tossing protocols in the f-hybrid model.

Definition 3 (Fair Coin-tossing in the f-hybrid Model). *An (X_0, r)-fair coin-tossing in the f-hybrid model is a two-party protocol between Alice and Bob in the f-hybrid model such that it satisfies the following.*

- ***X_0-Expected Output.*** *At the end of the protocol, parties always agree on the output $\in \{0, 1\}$ of the protocol. The expectation of the output of an honest execution is $X_0 \in (0, 1)$.*
- ***r-Message Complexity.*** *The total number of messages of the protocol is (at most) r. This includes both the Alice/Bob message and the trusted party message.*
- ***Defense Preparation.*** *Anytime a party speaks, she shall also prepare a defense coin based on her private randomness and the partial transcript. Her latest defense coin shall be her output when the other party decides to abort. To ensure that parties always have a defense to output, they shall prepare a defense before the protocol begins.*

– ***Insecurity.*** *The insecurity is defined as the maximum change a fail-stop adversary can cause to the expectation of the other party's output.*

For any (randomized) functionality f, Kilian [48] proved that if f *does not* satisfy the following cross product rule, f is complete for information-theoretic semi-honest adversaries. That is, for any functionality g, there is a protocol in the f-hybrid model that realizes g, which is secure against information-theoretic semi-honest adversaries. In particular, this implies that there is a protocol in the f-hybrid model that realizes oblivious transfer.

Definition 4 (Cross Product Rule). *A (randomized) functionality $f\colon \mathcal{X} \times \mathcal{Y} \to \mathcal{Z}$ is said to satisfy the cross product rule if for all $x_0, x_1 \in \mathcal{X}$, $y_0, y_1 \in \mathcal{Y}$, and $z \in \mathcal{Z}$ such that*

$$\Pr[f(x_0, y_0) = z] > 0 \quad \textit{and} \quad \Pr[f(x_1, y_0) = z] > 0,$$

we have

$$\Pr[f(x_0, y_0) = z] \cdot \Pr[f(x_1, y_1) = z] = \Pr[f(x_1, y_0) = z] \cdot \Pr[f(x_0, y_1) = z].$$

We recall the MNS protocol by Moran, Naor, and Segev [61]. The MNS protocol makes black-box uses of the oblivious transfer as a subroutine to construct optimal-fair coin-tossing protocols. In particular, their protocol enjoys the property that any fail-stop attack during the oblivious transfer subroutine is an entirely ineffective attack. Therefore, the MNS protocol, combined with the results of Kilian [48], gives us the following theorem.

Theorem 2 ([48, 61]). *Let f be a (randomized) functionality that is complete. For any $X_0 \in (0, 1)$ and $r \in \mathbb{N}^*$, there is an (X_0, r)-fair coin-tossing protocol in the f-hybrid model that is (at most) $\mathcal{O}(1/r)$-insecure against fail-stop attackers.*

Remark 1 (On the necessity of the unfairness of f). We emphasize that it is necessary that in the f-hybrid model, f is realized *unfairly*. That is, the adversary receives the output of f before the honest party does. If f is realized fairly, i.e., both parties receive the output simultaneously, it is possible to construct perfectly-secure fair coin-tossing. For instance, let f be the XOR function. Consider the protocol where Alice samples $x \xleftarrow{\$} \{0, 1\}$, Bob samples $y \xleftarrow{\$} \{0, 1\}$, and the trusted party broadcast $f(x, y)$, which is the final output of the protocol. Trivially, one can verify that this protocol is perfectly-secure.

Intuitively, the results of Kilian [48] and Moran, Naor, and Segev [61] showed that when f is a functionality that does not satisfy the cross product rule, a secure protocol realizing f can be used to construct optimal-fair coin-tossing.

In this work, we complement the above results by showing that when f is a functionality that does satisfy the cross product rule, a fair coin-tossing protocol in the f-hybrid model is (qualitatively) as insecure as a fair coin-tossing protocol in the information-theoretic model. In other words, f is completely useless for fair coin-tossing. Our results are summarized as the following theorem.

Theorem 3 (Main Theorem for f-hybrid). *Let f be a randomized functionality that is* not *complete. Any (X_0, r)-fair coin-tossing protocol in the f-hybrid model is (at least) $\Omega\left(\frac{X_0(1-X_0)}{\sqrt{r}}\right)$-insecure.*

4 Proof of Theorem 3

4.1 Properties of Functionalities

Let f be a functionality that satisfies the cross product rule. We start by observing some properties of f. Firstly, let us recall the following definition.

Definition 5 (Function Isomorphism [57]). *Let $f\colon \mathcal{X} \times \mathcal{Y} \to \mathcal{Z}$ and $g\colon \mathcal{X} \times \mathcal{Y} \to \mathcal{Z}'$ be any two (randomized) functionalities. We say $f \leq g$ if there exist deterministic mappings $M_{\mathsf{A}}\colon \mathcal{X} \times \mathcal{Z}' \to \mathcal{Z}$ and $M_{\mathsf{B}}\colon \mathcal{Y} \times \mathcal{Z}' \to \mathcal{Z}$ such that, for all $x \in \mathcal{X}$, $y \in \mathcal{Y}$, and randomness s,*

$$M_{\mathsf{A}}\left(x, g(x, y; s)\right) = M_{\mathsf{B}}\left(y, g(x, y; s)\right)$$

and

$$\mathsf{SD}\left(f(x, y)\ ,\ M_{\mathsf{A}}\left(x, g(x, y)\right)\right) = 0.$$

We say f and g are isomorphic (i.e., $f \cong g$) if $f \leq g$ and $g \leq f$.

Intuitively, f and g are isomorphic if securely computing f can be realized by one ideal call to g without any further communication and vise versa. As an example, the (deterministic) XOR functionality $\left[\begin{smallmatrix} 0 & 1 \\ 1 & 0 \end{smallmatrix}\right]$ is isomorphic to $\left[\begin{smallmatrix} 0 & 1 \\ 2 & 3 \end{smallmatrix}\right]$.

Given two isomorphic functionalities f and g, it is easy to see that there is a natural bijection between protocols in the f-hybrid model and g-hybrid model.

Lemma 1. *Let f and g be two functionalities such that $f \cong g$. For every fair coin-tossing protocol π in the f-hybrid model, there is a fair coin-tossing protocol π' in the g-hybrid model such that*

- *π and π' have the same message complexity r and expected output X_0.*
- *For every fail-stop attack strategy for π, there exists a fail-stop attack strategy for π' such that the insecurities they cause are identical and vice versa.*

Proof (Sketch). Given any protocol π in the f-hybrid model between A and B, consider the protocol π' in the g-hybrid model between A' and B'. In π', A' simply simulates A and does what A does. Except when the trusted party sends the output of g, A' uses the mapping M_{A} to recover the output of f and feeds it to A. B' behaves similarly. Easily, one can verify that these two protocols have the same message complexity and expected output. Additionally, for every fail-stop adversary A^* for π, there is a fail-stop adversary $(\mathsf{A}^*)'$ for π' that simulates A^* in the same manner, which deviates the output of Bob by the same amount.

We are now ready to state our next lemma.

Lemma 2 (Maximally Renaming the Outputs of f). *Let $f\colon \mathcal{X} \times \mathcal{Y} \to \mathcal{Z}$ be a (randomized) functionality that is not complete. There exists a functionality $f'\colon \mathcal{X} \times \mathcal{Y} \to \mathcal{Z}'$ such that $f \cong f'$ and f' satisfies the following* strict cross product rule. *That is, for all $x_0, x_1 \in \mathcal{X}$, $y_0, y_1 \in \mathcal{Y}$, and $z' \in \mathcal{Z}'$, we have*

$$\Pr[f'(x_0, y_0) = z'] \cdot \Pr[f'(x_1, y_1) = z'] = \Pr[f'(x_1, y_0) = z'] \cdot \Pr[f'(x_0, y_1) = z'].$$

Following the example above, the XOR functionality $\left[\begin{smallmatrix} 0 & 1 \\ 1 & 0 \end{smallmatrix}\right]$ satisfies the cross product rule, i.e., XOR is not complete, but it does not satisfy the strict cross product rule since

$$\Pr[\mathsf{XOR}(0,0) = 1] \cdot \Pr[\mathsf{XOR}(1,1) = 1] \neq \Pr[\mathsf{XOR}(1,0) = 1] \cdot \Pr[\mathsf{XOR}(0,1) = 1].$$

On the other hand, functionality $\left[\begin{smallmatrix} 0 & 1 \\ 2 & 3 \end{smallmatrix}\right]$ is isomorphic to XOR and does satisfy the strict cross product rule.

Proof (Proof of Lemma 2). We shall rename the output of f as follows. For all $z \in \mathcal{Z}$, define

$$\mathcal{S}_z := \{(x, y)\colon x \in \mathcal{X},\ y \in \mathcal{Y},\ \Pr[f(x, y) = z] > 0\}.$$

By the cross product rule, we know that there does not exist $x_0, x_1 \in \mathcal{X}$ and $y_0, y_1 \in \mathcal{Y}$ such that

$$(x_0, y_0), (x_0, y_1), (x_1, y_0) \in \mathcal{S}_z \quad \text{but} \quad (x_1, y_1) \notin \mathcal{S}_z.$$

Therefore, we can always partition $\mathcal{S}_z$ as a collection of combinatorical rectangles. That is, there exists subsets $\mathcal{X}_1, \ldots, \mathcal{X}_\ell \subseteq \mathcal{X}$ and $\mathcal{Y}_1, \ldots, \mathcal{Y}_\ell \subseteq \mathcal{Y}$ such that

$$\mathcal{S}_z = \bigcup_{i=1}^{\ell} \mathcal{X}_i \times \mathcal{Y}_i,$$

and

$$\forall 1 \leq i < j \leq \ell \quad \mathcal{X}_i \cap \mathcal{X}_j = \emptyset \quad \text{and} \quad \mathcal{Y}_i \cap \mathcal{Y}_j = \emptyset.$$

Now define randomized functionality $f'\colon \mathcal{X} \times \mathcal{Y} \to \mathcal{Z}'$ as follows. Given input x and y with randomness s, let $z = f(x, y; s)$. Let i be the index such that $(x, y) \in \mathcal{X}_i \times \mathcal{Y}_i$. Define

$$f'(x, y; s) := z^{(i)}.$$

Here, $z^{(i)}$ is an (arbitrarily picked) distinct output.

It is trivial to verify that, given $f'(x, y)$, Alice and Bob can recover the same sample, which is identically distributed as $f(x, y)$. On the other hand, given private input x (resp., y) and a sample of $f(x, y)$, Alice (resp., Bob) can recover a sample of $f'(x, y)$. Additionally, they shall always recover the same sample, which is identically distributed as $f'(x, y)$. This proves that f and f' are isomorphic.

Next, we verify that f' satisfies the strict cross product rule. Given any $x_0, x_1 \in \mathcal{X}$, $y_0, y_1 \in \mathcal{Y}$, and $z^{(i)} \in \mathcal{Z}'$, if either $x_0 \notin \mathcal{X}_i$ or $x_1 \notin \mathcal{X}_i$, it is trivially true. Similarly, if either $y_0 \notin \mathcal{Y}_i$ or $y_1 \notin \mathcal{Y}_i$, it is also trivial. Otherwise, when both $x_0, x_1 \in \mathcal{X}_i$ and $y_0, y_1 \in \mathcal{Y}_i$, strict cross product rule follows from cross product rule.

This completes the proof.

By Lemma 1, the insecurity of a fair coin-tossing protocol in the f-hybrid model is identical to a fair coin-tossing protocol in the f'-hybrid model when $f \cong f'$. Therefore, in the rest of this section, without loss of generality, we shall always assume f is maximally renamed according to Lemma 2 such that it satisfies the strict cross product rule.

4.2 Notations and the Technical Theorem

Let π be an (X_0, r)-fair coin-tossing protocol in the f-hybrid model. We shall use R^{A} and R^{B} to denote the private randomness of Alice and Bob. We use random variable M_i to denote the i^{th} message of the protocol, which could be either an Alice/Bob message or a trusted party message. Let X_i be the expected output of the protocol conditioned on the first i messages of the protocol. In particular, this definition is consistent with the definition of X_0.

For an arbitrary i, we consider both Alice aborts and Bob aborts the i^{th} message. Suppose the i^{th} message is Alice's message. Alice abort means that she aborts without sending this message to Bob. Conversely, Bob abort means he aborts in his next message immediately after receiving this message. On the other hand, if this is a trusted party message, then both a fail-stop Alice and a fail-stop Bob can abort this message. This prevents the other party from receiving the message. We refer to the defense output of Alice when Bob aborts the i^{th} message as Alice's i^{th} defense. Similarly, we define the i^{th} defense of Bob. Let D_i^{A} (resp., D_i^{B}) be the expectation of Alice's (resp., Bob's) i^{th} defense conditioned on the first i messages.

Now, we are ready to define our score function.

Definition 6. *Let π be a fair coin-tossing protocol in the f-hybrid model with message complexity r. Let τ be a stopping time. Let $\mathsf{P} \in \{\mathsf{A}, \mathsf{B}, \mathsf{T}\}$ be the party who sends the last message.*[7] *We define the score function as follows.*

$$\mathsf{Score}(\pi, \tau) := \mathrm{E}\left[\mathbb{1}_{(\tau \neq r) \vee (\mathsf{P} \neq \mathsf{A})} \cdot \left|X_\tau - D_\tau^{\mathsf{A}}\right| + \mathbb{1}_{(\tau \neq r) \vee (\mathsf{P} \neq \mathsf{B})} \cdot \left|X_\tau - D_\tau^{\mathsf{B}}\right|\right].$$

The following remarks, similar to [46, 59], provide additional perspectives toward this definition.

Remark 2.

1. In the information-theoretic plain model, for every message of the protocol, one usually only consider the attack by the sender of this message. The attack by the receiver, who may abort immediately after receiving this message,

[7] We use A, B, and T to stand for Alice, Bob, and the trusted party, respectively.

usually is ineffective. This is because the sender is not lagging behind in terms of the progress of the protocol. However, in the f-hybrid model, we have trusted party messages, which reveal information regarding both parties' private randomness. Therefore, both parties' defenses may lag behind, and both parties' attacks could be effective. Hence, in our definition of the score function, for every message we pick in the stopping time, we consider the effectiveness of both parties' attacks.

2. The last message of the protocol is a boundary case of the above argument. Suppose Alice sends the last message of the protocol, Bob does not have the opportunity to abort after receiving this message. Similarly, if this is a Bob message, Alice cannot attack this message. On the other hand, if the last message is a trusted party message, then both parties could potentially attack this message. This explains the indicator function in our definition.
3. Finally, given a stopping time τ^* that witnesses a high score. We can always find a fail-stop attack strategy that deviates the expected output of the other party by $\frac{1}{4} \cdot \mathsf{Score}(\pi, \tau^*)$ in the following way. For Alice, we shall partition the stopping time τ^* by considering whether $X_\tau \geq D_\tau^{\mathsf{B}}$ or not. Similarly, we partition τ^* for Bob. These four attacks correspond to either Alice or Bob favoring either 0 or 1. The quality of these four attacks sums up to be $\mathsf{Score}(\pi, \tau^*)$. Hence, one of these four fail-stop attacks might be at least $\frac{1}{4} \cdot \mathsf{Score}(\pi, \tau^*)$ effective.

The score function measures the effectiveness of a fail-stop attack corresponds to a stopping time τ. We are interested in the effectiveness of the most devastating fail-stop attacks. This motivates the following definition.

Definition 7. *Let π be a fair coin-tossing protocol in the f-hybrid model. Define*

$$\mathsf{Opt}(\pi) := \max_{\tau} \mathsf{Score}(\pi, \tau).$$

Now, we are ready to state our main theorem, which shows that the most devastating fail-stop attack is guaranteed to achieve a high score. In light of the remarks above, Theorem 4 directly implies Theorem 3.

Theorem 4. *For any (X_0, r)-fair coin-tossing protocol π in the f-hybrid model, we have*

$$\mathsf{Opt}(\pi) \geq \Gamma_r \cdot X_0 (1 - X_0),$$

where $\Gamma_r := \sqrt{\frac{\sqrt{2}-1}{r}}$.

4.3 Inductive Proof of Theorem 4

In this section, we shall prove Theorem 4 by using mathematical induction on the message complexity r. Let us first state some useful lemmas.

Firstly, we note that in the f-hybrid model, where f is a (randomized) functionality that satisfies the strict cross product rule, Alice view and Bob view are always independent conditioned on the partial transcript.

Lemma 3 (Independence of Alice and Bob view). *For any i and partial transcript $m_{\leq i}$, conditioned on this partial transcript, the joint distribution of Alice and Bob private randomness is identical to the product of the marginal distribution. That is,*

$$\mathsf{SD}\Big(\big(R^{\mathsf{A}}, R^{\mathsf{B}}\big) \big| M_{\leq i} = m_{\leq i} \ , \ \big(R^{\mathsf{A}} \big| M_{\leq i} = m_{\leq i}\big) \times \big(R^{\mathsf{B}} \big| M_{\leq i} = m_{\leq i}\big) \Big) = 0.$$

In particular, this lemma implies the following claim.

Claim 1 *Let π be an arbitrary fair coin-tossing protocol in the f-hybrid model. Suppose there are ℓ possible first messages, namely, $m_1^{(1)}, m_1^{(2)}, \ldots, m_1^{(\ell)}$, each happens with probability $p^{(1)}, p^{(2)}, \ldots, p^{(\ell)}$. Suppose conditioned on the first message being $M_1 = m_1^{(i)}$, the expected defense of Alice and Bob are $d_1^{\mathsf{A},(i)}$ and $d_1^{\mathsf{B},(i)}$ respectively. Then we have*

$$\sum_{i=1}^{\ell} p^{(i)} \cdot d_1^{\mathsf{A},(i)} d_1^{\mathsf{B},(i)} = D_0^{\mathsf{A}} \cdot D_0^{\mathsf{B}}.$$

Lemma 3 and Claim 1 can be proven in a straightforward manner. Hence, we defer them to Supporting Material A. Finally, the following lemma from [46] shall be helpful as well.

Imported Lemma 1 ([46]) *For all $P \in [0,1]$ and $Q \in [0,1/2]$, if P and Q satisfy that*

$$Q \leq \frac{P}{1+P^2},$$

then for all $x, \alpha, \beta \in [0,1]$, we have

$$\max\left(P \cdot x(1-x) \ , \ |x-\alpha| + |x-\beta|\right) \geq Q \cdot \left(x(1-x) + (x-\alpha)^2 + (x-\beta)^2\right).$$

In particular, for any integer $r \geq 1$, the constraints are satisfied, if we set $P = \Gamma_r$ and $Q = \Gamma_{r+1}$, where $\Gamma_r := \sqrt{\frac{\sqrt{2}-1}{r}}$.

Base case: $r = 1$ We are now ready to prove Theorem 4. Let us start with the base case. In the base case, the protocol consists of only one message. Recall that the last message of the protocol is a boundary case of our score function. It might not be the case that both parties can attack this message. Hence, we prove it in different cases.

Case 1: Alice message. Suppose this message is an Alice message. In this case, we shall only consider the attack by Alice. By definition, with probability X_0, Alice will send a message, conditioned on which the output shall be 1. And with probability $1 - X_0$, Alice will send a message, conditioned on which the output shall be 0. On the other hand, the expectation of Bob's defense will remain the same as D_0^{B}. Therefore, the maximum of the score shall be

$$X_0 \cdot \left|1 - D_0^{\mathsf{B}}\right| + (1 - X_0) \cdot \left|0 - D_0^{\mathsf{B}}\right|,$$

which is

$$\geq X_0 (1 - X_0) .$$

In particular, this is

$$\geq \Gamma_1 \cdot X_0 (1 - X_0) .$$

Case 2: Bob message. This case is entirely analogous to case 1.

Case 3: Trusted party message. In this case, we shall consider the effectiveness of the attacks by both parties. Suppose there are ℓ possible first message by the trusted party, namely, $m_1^{(1)}, m_1^{(2)}, \ldots, m_1^{(\ell)}$, each happens with probability $p^{(1)}, p^{(2)}, \ldots, p^{(\ell)}$. Conditioned on first message being $M_1 = m_1^{(i)}$, the output of the protocol is $x_1^{(i)}$. We must have $x_1^{(i)} \in \{0, 1\}$ since the protocol has ended and parties shall agree on the output. Furthermore, let the expected defense of Alice and Bob be $d_1^{\mathsf{A},(i)}$ and $d_1^{\mathsf{B},(i)}$. Therefore, the maximum of the score will be

$$\sum_{i=1}^{\ell} p^{(i)} \cdot \left(\left| x_1^{(i)} - d_1^{\mathsf{A},(i)} \right| + \left| x_1^{(i)} - d_1^{\mathsf{B},(i)} \right| \right) .$$

We have

$$\begin{aligned}
&\sum_{i=1}^{\ell} p^{(i)} \cdot \left(\left| x_1^{(i)} - d_1^{\mathsf{A},(i)} \right| + \left| x_1^{(i)} - d_1^{\mathsf{B},(i)} \right| \right) \\
\geq &\sum_{i=1}^{\ell} p^{(i)} \cdot \left(x_1^{(i)} \left(1 - x_1^{(i)} \right) + \left(x_1^{(i)} - d_1^{\mathsf{A},(i)} \right)^2 + \left(x_1^{(i)} - d_1^{\mathsf{B},(i)} \right)^2 \right) \\
&\qquad \text{(Since } x_1^{(i)} \in \{0, 1\}\text{)} \\
= &\sum_{i=1}^{\ell} p^{(i)} \cdot \left(x_1^{(i)} + \left(x_1^{(i)} - d_1^{\mathsf{A},(i)} - d_1^{\mathsf{B},(i)} \right)^2 - 2 d_1^{\mathsf{A},(i)} d_1^{\mathsf{B},(i)} \right) \\
&\qquad \text{(Identity Transformation)} \\
\geq &X_0 + \left(X_0 - D^{\mathsf{A}} - D^{\mathsf{B}} \right)^2 - \sum_{i=1}^{\ell} p^{(i)} \cdot 2 d_1^{\mathsf{A},(i)} d_1^{\mathsf{B},(i)} \\
&\qquad \text{(Jensen's inequality on convex function } F(x, y, z) := (x - y - z)^2\text{)} \\
= &X_0 + \left(X_0 - D^{\mathsf{A}} - D^{\mathsf{B}} \right)^2 - 2 D_0^{\mathsf{A}} \cdot D_0^{\mathsf{B}} \qquad \text{(Claim 1)} \\
= &X_0 (1 - X_0) + \left(X_0 - D_0^{\mathsf{A}} \right)^2 + \left(X_0 - D_0^{\mathsf{B}} \right)^2 \qquad \text{(Identity Transformation)} \\
\geq &X_0 (1 - X_0) \\
\geq &\Gamma_1 \cdot X_0 (1 - X_0)
\end{aligned}$$

This completes the proof of the base case.

Inductive Step Suppose the statement is true for message complexity r. Let π be an arbitrary protocol with message complexity $r+1$. Suppose there are ℓ possible first messages, namely, $m_1^{(1)}, m_1^{(2)}, \ldots, m_1^{(\ell)}$, each happens with probability

$p^{(1)}, p^{(2)}, \ldots, p^{(\ell)}$. Conditioned on first message being $M_1 = m_1^{(i)}$, the output of the protocol is $x_1^{(i)}$ and the expected defense of Alice and Bob are $d_1^{\mathsf{A},(i)}$ and $d_1^{\mathsf{B},(i)}$ respectively. Note that conditioned on the first message being $M_1 = m_1^{(i)}$, the remaining protocol $\pi^{(i)}$ becomes a protocol with expected output $x_1^{(i)}$ and message complexity r. By our inductive hypothesis, we have

$$\mathsf{Opt}\left(\pi^{(i)}\right) \geq \Gamma_r \cdot x_1^{(i)}\left(1 - x_1^{(i)}\right).$$

On the other hand, we could also pick the first message $m_1^{(i)}$ as our stopping time, which yields a score of

$$\left|x_1^{(i)} - d_1^{\mathsf{A},(i)}\right| + \left|x_1^{(i)} - d_1^{\mathsf{B},(i)}\right|.$$

Therefore, the stopping time that witnesses the largest score yields (at least) a score of

$$\begin{aligned}
&\max\left(\Gamma_r \cdot x_1^{(i)}\left(1 - x_1^{(i)}\right),\ \left|x_1^{(i)} - d_1^{\mathsf{A},(i)}\right| + \left|x_1^{(i)} - d_1^{\mathsf{B},(i)}\right|\right) \\
\geq &\Gamma_{r+1} \cdot \left(x_1^{(i)}\left(1 - x_1^{(i)}\right) + \left(x_1^{(i)} - d_1^{\mathsf{A},(i)}\right)^2 + \left(x_1^{(i)} - d_1^{\mathsf{B},(i)}\right)^2\right) \\
&\qquad\qquad \text{(Imported Lemma 1)}
\end{aligned}$$

Therefore, $\mathsf{Opt}(\pi)$ is lower bounded by

$$\begin{aligned}
&\sum_{i=1}^{\ell} p^{(i)} \cdot \Gamma_{r+1} \cdot \left(x_1^{(i)}\left(1 - x_1^{(i)}\right) + \left(x_1^{(i)} - d_1^{\mathsf{A},(i)}\right)^2 + \left(x_1^{(i)} - d_1^{\mathsf{B},(i)}\right)^2\right) \\
= &\Gamma_{r+1} \cdot \sum_{i=1}^{\ell} p^{(i)} \cdot \left(x_1^{(i)} + \left(x_1^{(i)} - d_1^{\mathsf{A},(i)} - d_1^{\mathsf{B},(i)}\right)^2 - 2 d_1^{\mathsf{A},(i)} d_1^{\mathsf{B},(i)}\right) \\
&\qquad\qquad \text{(Identity Transformation)} \\
\geq &\Gamma_{r+1} \cdot \left(X_0 + \left(X_0 - D^{\mathsf{A}} - D^{\mathsf{B}}\right)^2 - \sum_{i=1}^{\ell} p^{(i)} \cdot 2 d_1^{\mathsf{A},(i)} d_1^{\mathsf{B},(i)}\right) \\
&\qquad \text{(Jensen's inequality on convex function } F(x,y,z) := (x-y-z)^2) \\
= &\Gamma_{r+1} \cdot \left(X_0 + \left(X_0 - D^{\mathsf{A}} - D^{\mathsf{B}}\right)^2 - 2 D_0^{\mathsf{A}} \cdot D_0^{\mathsf{B}}\right) \qquad \text{(Claim 1)} \\
= &\Gamma_{r+1} \cdot \left(X_0\left(1 - X_0\right) + \left(X_0 - D_0^{\mathsf{A}}\right)^2 + \left(X_0 - D_0^{\mathsf{B}}\right)^2\right) \\
&\qquad\qquad \text{(Identity Transformation)} \\
\geq &\Gamma_{r+1} \cdot X_0\left(1 - X_0\right)
\end{aligned}$$

This completes the proof of the inductive step.

5 Black-box uses of Public-key Encryption is Useless for Optimal Fair Coin-tossing

In this section, we prove that public-key encryption used in a black-boxed manner shall not enable optimal fair coin-tossing. Our objective is to prove the existence

of an oracle, with respect to which public-key encryption exists, but optimal fair coin-tossing does not.

5.1 Public-key Encrytion Oracles

Let n be the security parameter. We follow the work of [56] and define the following set of functions.

- $\mathsf{Gen}\colon \{0,1\}^n \to \{0,1\}^{3n}$. This function is a random injective function.
- $\mathsf{Enc}\colon \{0,1\}^{3n} \times \{0,1\}^n \to \{0,1\}^{3n}$. This function is uniformly randomly sampled among all functions that are injective with respect to the second input. That is, when the first input is fixed, this function is injective.
- $\mathsf{Dec}\colon \{0,1\}^n \times \{0,1\}^{3n} \to \{0,1\}^n \cup \{\bot\}$. This function is the uniquely determined by functions Gen and Enc as follows. Dec takes as inputs a secret-key $sk \in \{0,1\}^n$ and a ciphertext $c \in \{0,1\}^{3n}$. If there exists a message $m \in \{0,1\}^n$ such that $\mathsf{Enc}(\mathsf{Gen}(sk), m) = c$, define $\mathsf{Dec}(sk, c) := m$. Otherwise, define $\mathsf{Dec}(sk, c) := \bot$. Note that such message m, if exists, must be unique, because Enc is injective with respect to the second input.
- $\mathsf{Test}_1\colon \{0,1\}^{3n} \to \{0,1\}$. This function is uniquely determined by function Gen. It takes as an input a public-key $pk \in \{0,1\}^{3n}$. If there exists a secret-key $sk \in \{0,1\}^n$ such that $\mathsf{Gen}(sk) = pk$, define $\mathsf{Test}_1(pk) := 1$. Otherwise, define $\mathsf{Test}_1(pk) := 0$.
- $\mathsf{Test}_2\colon \{0,1\}^{3n} \times \{0,1\}^{3n} \to \{0,1\}$. This function is uniquely determined by function Enc. It takes as inputs a public-key $pk \in \{0,1\}^{3n}$ and a ciphertext $c \in \{0,1\}^{3n}$. If there exists a message m such that $\mathsf{Enc}(pk, m) = c$, define $\mathsf{Test}_2(pk, c) := 1$. Otherwise, define $\mathsf{Test}_2(pk, c) := 0$.

We shall refer to this collection of oracles the PKE oracle. Trivially, the PKE oracle enables public-key encryption. We shall prove that it does not enable optimally-fair coin-tossing. We remark that it is necessary to include the test functions Test_1 and Test_2. Otherwise, it can be used to construct oblivious transfer protocols against semi-honest adversaries [28, 54], which can be further used to construct optimally-fair coin-tossing protocols [61].

5.2 Our Results

We shall prove the following theorem.

Theorem 5 (Main theorem for PKE Oracle). *There exists a universal polynomial $p(\cdot,\cdot,\cdot,\cdot)$ such that the following holds. Let π be any fair coin-tossing protocol in the PKE oracle model, where Alice and Bob make at most m queries. Let X_0 be the expected output, and r be the message complexity of π. There exists an (information-theoretic) fail-stop attacker that deviates the expected output of the other party by (at least)*

$$\Omega\left(\frac{X_0\,(1-X_0)}{\sqrt{r}}\right).$$

This attacker shall ask at most $p\left(n, m, r, \frac{1}{X_0(1-X_0)}\right)$ *additional queries.*

It is instructive to understand why Theorem 3 does not imply Theorem 5. One may be tempted to model the public-key encryption primitive as an idealized secure function evaluation functionality to prove this implication. The idealized functionality for public-key encryption delivers sender's message to the receiver, while hiding it from the eavesdropper. So, the "idealized public-key encryption" functionality is a three-party functionality where the sender's input is delivered to the receiver; the eavesdropper has no input or output. This idealized effect is easily achieved given secure point-to-point communication channels, which we assume in our work. The non-triviality here is that our result is with respect to an *oracle* that implements the public-key encryption functionality. An oracle for public-key encryption is not necessarily used just for secure message passing. Section 6 has a discussion elaborating the difference between an "ideal functionality" and an "oracle implementing the ideal functionality."

Remark 3. As usual in the literature [21, 22, 59], we shall only consider *instant protocols*. That is, once a party aborts, the other party shall not make any additional queries to defend, but directly output her current defense coin. We refer the reader to [21] for justification and more details on this assumption.

In fact, our proof technique is sufficient to prove the following stronger theorem.

Theorem 6. *There exists a universal polynomial* $p(\cdot,\cdot,\cdot,\cdot)$ *such that the following holds. Let* f *be any (randomized) functionality that is not complete. Let* π *be any fair coin-tossing protocol in the* f*-hybrid model where parties have access to the PKE oracle model. Assume Alice and Bob make at most* m *queries. Let* X_0 *be the expected output, and* r *be the message complexity of* π*. There exists an (information-theoretic) fail-stop attacker that deviates the expected output of the other party by (at least)*

$$\Omega\left(\frac{X_0\,(1-X_0)}{\sqrt{r}}\right).$$

This attacker shall ask at most $p\left(n, m, r, \frac{1}{X_0(1-X_0)}\right)$ *additional queries.*

Our proof strategy is similar to that of [56]. It consists of the following two steps.

1. Given a protocol in the PKE oracle model, we shall first convert it into a protocol where parties do not invoke the decryption queries. By Imported Theorem 1 proven in [56], we can convert it in a way such that the insecurity of these two protocols in the presence of a semi-honest adversary is (almost) identical. In particular, this ensures that the insecurity of fair coin-tossing protocol in the presence of a fail-stop adversary is (almost) identical.

2. Next, we shall extend the results of [59], where they proved a fair coin-tossing protocol in the random oracle model is highly insecure, to the setting of PKE oracles without decryption oracle. Intuitively, The proof of [59] only relied on the fact that in the random oracle model, there exists a public algorithm [8] that asks polynomially many queries and decorrelate the private view of Alice and Bob. Mahmoody, Maji, and Prabhakaran [56] proved that (summarized as Imported Theorem 2) the PKE oracles without the decryption oracle satisfies the similar property. Hence, the proof of [59] extends naturally to this setting.

Together, these two steps prove Theorem 5. The first step is summarized in Section 5.3. The second step is summarized in Section 5.4.

5.3 Reduction from PKE Oracle to Image Testable Random Oracle

A (keyed version of) *image-testable random oracles* is a collection of pairs of oracles $(R^{\mathsf{key}}, T^{\mathsf{key}})$ parameterized by a key, such that for every key, the following holds.

- $R^{\mathsf{key}} \colon \{0,1\}^n \to \{0,1\}^{3n}$ is a randomly sampled injective function.
- $T^{\mathsf{key}} \colon \{0,1\}^{3n} \to \{0,1\}$ is uniquely determined by function R^{key} as follows. Define $T^{\mathsf{key}}(\beta) := 1$ if there exists an $\alpha \in \{0,1\}^n$ such that $R^{\mathsf{key}}(\alpha) = \beta$. Otherwise, define $T^{\mathsf{key}}(\beta) = 0$.

Observe that the PKE oracle without the decryption oracle Dec is exactly a (keyed version of) image-testable random oracles with the keys drawn from $\{\bot\} \cup \{0,1\}^{3n}$. If the key is $\bot$, it refers to the pair of oracles $(\mathsf{Gen}, \mathsf{Test}_1)$. If the $\mathsf{key} \in \{0,1\}^{3n}$, it refers to the pair of oracles $(\mathsf{Enc}(\mathsf{key}, \cdot), \mathsf{Test}_2(\mathsf{key}, \cdot))$. We shall refer to the PKE oracle without the decryption oracle Dec as ITRO.

We shall use the following imported theorem, which is implicitly proven in [56].

Imported Theorem 1 ([56]) *There exists a universal polynomial $p(\cdot,\cdot)$ such that the following holds. Let π be a fair coin-tossing protocol in the PKE oracle model. Let X_0 and r be the expected output and message complexity. Suppose Alice and Bob ask (at most) m queries. For any $\epsilon > 0$, there exists a fair coin-tossing protocol π' in the ITRO model such that the following holds.*

- *Let X_0' and r' be the expected output and message complexity of π'. Then, $r' = r$ and $|X_0' - X_0| < \epsilon$.*
- *Parties asks at most $p(m, 1/\epsilon)$ queries in protocol π'.*
- *For any semi-honest adversary $\mathcal{A}'$ for protocol π', there exists a semi-honest adversary $\mathcal{A}$ for protocol π, such that the view of $\mathcal{A}$ is ϵ-close to the view of $\mathcal{A}'$. And vice versa. In particular, this implies that if π' is α-insecure. π is (at least) $(\alpha - \epsilon)$-insecure.*

The intuition behind this theorem is the following. To avoid the uses of decryption oracle, parties are going to help each other decrypt. In more detail, suppose Alice generates a ciphertext using Bob's public key. Whenever the probability that Bob invokes the decryption oracle on this ciphertext is non-negligibly high, Alice will directly reveal the message to Bob. Hence, Bob does not need to use the decryption oracle. This shall not harm the security as a semi-honest Bob can recover the message by asking polynomially many additional queries. We refer the readers to [56] for more details.

Looking forward, we shall prove that any fair coin-tossing protocol in the ITRO model is $\Omega\left(\frac{X_0'(1-X_0')}{\sqrt{r}}\right)$-insecure. By setting ϵ to be $1/\mathsf{poly}$ for some sufficiently large polynomial, we shall guarantee that

$$\epsilon = \mathrm{o}\left(\frac{X_0\left(1-X_0\right)}{\sqrt{r}}\right).$$

This guarantees that the insecurity of the protocol in the PKE oracle model is (qualitatively) identical to the insecure of the protocol in the ITRO model.

5.4 Extending the proof of [59] to Image Testable Random Oracle

We first recall the following theorem from [56].

Imported Theorem 2 (Common Information Learner [56]) *There exists a universal polynomial $p(\cdot,\cdot)$ such that the following holds. Let π be any two-party protocol in the ITRO model, in which both parties make at most m queries. For all threshold $\epsilon \in (0,1)$, there exists a public algorithm, called the* common information learner*, who has access to the transcript between Alice and Bob. After receiving each message, the common information learner performs a sequence of queries and obtain its corresponding answers from the ITRO. Let M_i denote the i^{th} message of the protocol. Let H_i denote the sequence of query-answer pairs asked by the common information learner after receiving the message M_i. Let T_i be the union of the i^{th} message M_i and the i^{th} common information learner message H_i. Let V_i^{A} (resp., V_i^{B}) denote Alice's (resp., Bob's) private view immediately after message T_i, which includes her private randomness, private queries, and the public partial transcript. , The common information learner guarantees that the following conditions are simultaneously satisfied.*

– ***Cross-product Property.*** *Fix any round i,*

$$\mathop{\mathrm{E}}_{t_{\le i}\leftarrow T_{\le i}}\left[\mathsf{SD}\left(\left(V_i^{\mathsf{A}},V_i^{\mathsf{B}}\middle|T_{\le i}=t_{\le i}\right),\left(V_i^{\mathsf{A}}\middle|T_{\le i}=t_{\le i}\right)\times\left(V_i^{\mathsf{B}}\middle|T_{\le i}=t_{\le i}\right)\right)\right]\le\epsilon.$$

Intuitively, it states that on average*, the statistical distance between (1) the joint distribution of Alice's and Bob's private view, and (2) the product of the marginal distributions of Alice's private views and Bob's private views is small.*

– ***Efficient Property.*** *The expected number of queries asked by the common information learner is bounded by* $p(m, 1/\epsilon)$.

This theorem, combined with proof of [59] gives the following theorem.

Theorem 7. *There exists a universal polynomial* $p(\cdot,\cdot,\cdot,\cdot)$ *such that the following holds. Let* π *be a protocol in the ITRO model, where Alice and Bob make at most* m *queries. Let* X_0 *and* r *be the expected output and message complexity. Then, there exists an (information-theoretic) fail-stop adversary that deviates the expected output of the other party by*

$$\Omega\left(\frac{X_0\left(1-X_0\right)}{\sqrt{r}}\right).$$

This attacker asks at most $p\left(n, m, r, \frac{1}{X_0(1-X_0)}\right)$ *additional queries.*

Below, we briefly discuss why Imported Theorem 2 is sufficient to prove this theorem. The full proof is analogous to [59] and the proof of the results in the f-hybrid model. Hence we omit it here.

On a high level, the proof goes as follows. We prove Theorem 7 by induction. Conditioned on the first message, the remaining protocol becomes an $(r-1)$-message protocol, and one can apply the inductive hypothesis. For every possible first message i, we consider whether to abort immediately or defer the attack to the remaining sub-protocol. By invoking Imported Lemma 1, we obtain a potential function, which characterizes the insecurity of the protocol with first message being i. This potential function will be of the form

$$\Phi(x_i, a_i, b_i) = x_i(1-x_i) + (x_i - a_i)^2 + (x_i - b_i)^2,$$

where x_i, a_i, and b_i stands for the expected output, expected Alice defense, and expected Bob defense, respectively. To complete the proof, [59] showed that it suffices to prove the following Jensen's inequality.

$$\mathop{\mathrm{E}}_{i}\left[\Phi(x_i, a_i, b_i)\right] \geq \Phi\left(\mathop{\mathrm{E}}_{i}\left[x_i\right], \mathop{\mathrm{E}}_{i}\left[a_i\right], \mathop{\mathrm{E}}_{i}\left[b_i\right]\right).$$

To prove this, one can rewrite $\Phi(x, a, b)$ as

$$\Phi(x, a, b) = x + (x - a - b)^2 - 2ab.$$

We note that x and $(x-a-b)^2$ are convex functions, and hence Jensen's inequality holds. As for the term ab, we shall have

$$\mathop{\mathrm{E}}_{i}\left[a_i b_i\right] \approx \mathop{\mathrm{E}}_{i}\left[a_i\right] \cdot \mathop{\mathrm{E}}_{i}\left[b_i\right]$$

as long as, conditioned on every possible first message i, Alice's private view is (almost) independent to Bob's private view. This is exactly what Imported Theorem 2 guarantees except for a small error depending on ϵ, which we shall set to be sufficiently small. Therefore, the proof shall follow.

6 Open Problems

In this work, we proved that access to ideal invocations to the secure function evaluation functionalities like the Kushilevitz function (Figure 2) does not enable optimal fair coin-tossing. However, we do *not* resolve the following stronger statement. Suppose there exists an oracle relative to which there exists a secure protocol for the Kushilevitz function. Is optimal fair coin-tossing impossible relative to this oracle?

To appreciate the distinction between these two statements, observe that there may be additional ways to use the "oracle implementing Kushilevitz function" than *merely* facilitating the secure computing of the Kushilevitz function. More generally, there may be implicit consequences implied by the existence of such an oracle. For example, "the existence of an efficient algorithm for 3SAT" not only allows solving 3SAT problems, but it also allows efficiently solving any problem in PH because the entire PH collapses to P.

This problem is incredibly challenging and one of the major open problems in this field. The technical tools developed in this paper also bring us closer to resolving this problem.

References

1. Shashank Agrawal and Manoj Prabhakaran. On fair exchange, fair coins and fair sampling. In *CRYPTO*, 2013.
2. Bar Alon and Eran Omri. Almost-optimally fair multiparty coin-tossing with nearly three-quarters malicious. In *TCC*, 2016.
3. Gilad Asharov. Towards characterizing complete fairness in secure two-party computation. In *TCC*, 2014.
4. Gilad Asharov, Amos Beimel, Nikolaos Makriyannis, and Eran Omri. Complete characterization of fairness in secure two-party computation of boolean functions. In *TCC*, 2015.
5. Gilad Asharov, Yehuda Lindell, and Tal Rabin. A full characterization of functions that imply fair coin tossing and ramifications to fairness. In *TCC*, 2013.
6. Baruch Awerbuch, Manuel Blum, Benny Chor, Shafi Goldwasser, and Silvio Micali. How to implement bracha's o (log n) byzantine agreement algorithm. 1985.
7. Paul Baecher, Christina Brzuska, and Marc Fischlin. Notions of black-box reductions, revisited. In *ASIACRYPT*, 2013.
8. Boaz Barak and Mohammad Mahmoody-Ghidary. Merkle puzzles are optimal—an o (n 2)-query attack on any key exchange from a random oracle. In *CRYPTO*, 2009.
9. Donald Beaver. Perfect privacy for two-party protocols. In *DIMACS*, 1989.
10. Amos Beimel, Yehuda Lindell, Eran Omri, and Ilan Orlov. 1/p-secure multiparty computation without honest majority and the best of both worlds. In *CRYPTO*, 2011.
11. Amos Beimel, Eran Omri, and Ilan Orlov. Protocols for multiparty coin toss with dishonest majority. In *CRYPTO*, 2010.
12. Manuel Blum. Coin flipping by telephone - A protocol for solving impossible problems. 1982.

13. Andrei Z Broder and Danny Dolev. Flipping coins in many pockets (byzantine agreement on uniformly random values). In *FOCS*, 1984.
14. Niv Buchbinder, Iftach Haitner, Nissan Levi, and Eliad Tsfadia. Fair coin flipping: Tighter analysis and the many-party case. In *SODA*, 2017.
15. Ran Canetti. Security and composition of multiparty cryptographic protocols. *J. Cryptology*, 2000.
16. Ran Canetti. Universally composable security: A new paradigm for cryptographic protocols. In *FOCS*, 2001.
17. Ran Canetti, Eyal Kushilevitz, and Lindell Yehuda. On the limitations of universally composable two-party computation without set-up assumptions. In *EUROCRYPT*, 2003.
18. Benny Chor and Eyal Kushilevitz. A zero-one law for boolean privacy (extended abstract). In *STOC*, 1989.
19. Richard Cleve. Limits on the security of coin flips when half the processors are faulty. In *STOC*, 1986.
20. Richard Cleve and Russell Impagliazzo. Martingales, collective coin flipping and discrete control processes (extended abstract). 1993.
21. Dana Dachman-Soled, Yehuda Lindell, Mohammad Mahmoody, and Tal Malkin. On the black-box complexity of optimally-fair coin tossing. In *TCC*, 2011.
22. Dana Dachman-Soled, Mohammad Mahmoody, and Tal Malkin. Can optimally-fair coin tossing be based on one-way functions? In *TCC*, 2014.
23. Ivan Damgård, Joe Kilian, and Louis Salvail. On the (im) possibility of basing oblivious transfer and bit commitment on weakened security assumptions. In *EUROCRYPT*, 1999.
24. Deepesh Data and Manoj Prabhakaran. Towards characterizing securely computable two-party randomized functions. In *PKC*, 2018.
25. Shimon Even, Oded Goldreich, and Abraham Lempel. A randomized protocol for signing contracts. In *CRYPTO*, 1982.
26. Rosario Gennaro, Yael Gertner, and Jonathan Katz. Lower bounds on the efficiency of encryption and digital signature schemes. In *STOC*, 2003.
27. Rosario Gennaro and Luca Trevisan. Lower bounds on the efficiency of generic cryptographic constructions. In *FOCS*, 2000.
28. Yael Gertner, Sampath Kannan, Tal Malkin, Omer Reingold, and Mahesh Viswanathan. The relationship between public key encryption and oblivious transfer. In *FOCS*, 2000.
29. Yael Gertner, Tal Malkin, and Omer Reingold. On the impossibility of basing trapdoor functions on trapdoor predicates. In *FOCS*, 2001.
30. Oded Goldreich, Shafi Goldwasser, and Silvio Micali. How to construct random functions. In *FOCS*, 1984.
31. Oded Goldreich, Shafi Goldwasser, and Silvio Micali. How to construct random functions. *Journal of the ACM*, 33(4):792–807, 1986.
32. Oded Goldreich, Silvio Micali, and Avi Wigderson. How to play any mental game, or a completeness theorem for protocols with honest majority. In *STOC*, 1987.
33. Oded Goldreich, Silvio Micali, and Avi Wigderson. Proofs that yield nothing but their validity or all languages in np have zero-knowledge proof systems. *Journal of the ACM (JACM)*, 38(3):690–728, 1991.
34. S Dov Gordon, Carmit Hazay, Jonathan Katz, and Yehuda Lindell. Complete fairness in secure two-party computation. In *STOC*, 2008.
35. S Dov Gordon and Jonathan Katz. Partial fairness in secure two-party computation. In *EUROCRYPT*, 2010.

36. Iftach Haitner, Nikolaos Makriyannis, and Eran Omri. On the complexity of fair coin flipping. In *TCC*, 2018.
37. Iftach Haitner, Kobbi Nissim, Eran Omri, Ronen Shaltiel, and Jad Silbak. Computational two-party correlation: A dichotomy for key-agreement protocols. In *FOCS*, 2018.
38. Iftach Haitner and Eran Omri. Coin flipping with constant bias implies one-way functions. In *FOCS*, 2011.
39. Iftach Haitner and Omer Reingold. Statistically-hiding commitment from any one-way function. In *STOC*, 2007.
40. Iftach Haitner and Eliad Tsfadia. An almost-optimally fair three-party coin-flipping protocol. In *STOC*, 2014.
41. Johan Håstad. Pseudo-random generators under uniform assumptions. In *STOC*, 1990.
42. Johan Håstad, Russell Impagliazzo, Leonid A Levin, and Michael Luby. A pseudorandom generator from any one-way function. *SIAM Journal on Computing*, 28(4):1364–1396, 1999.
43. Russell Impagliazzo. A personal view of average-case complexity. In *Proceedings of the Tenth Annual Structure in Complexity Theory Conference*, 1995.
44. Russell Impagliazzo, Leonid A. Levin, and Michael Luby. Pseudo-random generation from one-way functions (extended abstracts). In *STOC*, 1989.
45. Russell Impagliazzo and Steven Rudich. Limits on the provable consequences of one-way permutations. In *STOC*, 1989.
46. Hamidreza Amini Khorasgani, Hemanta K. Maji, and Mingyuan Wang. Coin tossing with lazy defense: Hardness of computation results. Cryptology ePrint Archive, Report 2020/131. https://eprint.iacr.org/2020/131.
47. Joe Kilian. A general completeness theorem for two-party games. In *STOC*, 1991.
48. Joe Kilian. More general completeness theorems for secure two-party computation. In *STOC*, 2000.
49. Gunnar Kreitz. A zero-one law for secure multi-party computation with ternary outputs. In *TCC*, 2011.
50. Robin Künzler, Jörn Müller-Quade, and Dominik Raub. Secure computability of functions in the IT setting with dishonest majority and applications to long-term security. In *TCC*, 2009.
51. Eyal Kushilevitz. Privacy and communication complexity. In *FOCS*, 1989.
52. Yehuda Lindell. Lower bounds for concurrent self composition. In *TCC*, 2004.
53. Yehuda Lindell. How to simulate it - A tutorial on the simulation proof technique. In *Tutorials on the Foundations of Cryptography*. 2017.
54. Yehuda Lindell, Eran Omri, and Hila Zarosim. Completeness for symmetric two-party functionalities - revisited. In *ASIACRYPT*, 2012.
55. Michael Luby and Charles Rackoff. How to construct pseudorandom permutations from pseudorandom functions. *SIAM Journal on Computing*, 17(2):373–386, 1988.
56. Mohammad Mahmoody, Hemanta K Maji, and Manoj Prabhakaran. On the power of public-key encryption in secure computation. In *TCC*, 2014.
57. Hemanta K Maji, Manoj Prabhakaran, and Mike Rosulek. Complexity of multi-party computation problems: The case of 2-party symmetric secure function evaluation. In *TCC*, 2009.
58. Hemanta K Maji, Manoj Prabhakaran, and Mike Rosulek. A zero-one law for cryptographic complexity with respect to computational uc security. In *CRYPTO*, 2010.
59. Hemanta K Maji and Mingyuan Wang. Black-box use of one-way functions is useless for optimal fair coin-tossing. In *CRYPTO*, 2020.

60. Nikolaos Makriyannis. On the classification of finite boolean functions up to fairness. In *SCN*, 2014.
61. Tal Moran, Moni Naor, and Gil Segev. An optimally fair coin toss. In *TCC*, 2009.
62. Moni Naor. Bit commitment using pseudorandomness. *J. Cryptology*, 1991.
63. Moni Naor, Rafail Ostrovsky, Ramarathnam Venkatesan, and Moti Yung. Perfect zero-knowledge arguments for *NP* using any one-way permutation. *J. Cryptology*, 1998.
64. Moni Naor and Moti Yung. Universal one-way hash functions and their cryptographic applications. In *STOC*, 1989.
65. Christos H. Papadimitriou. Games against nature (extended abstract). In *FOCS*, 1983.
66. Manoj Prabhakaran and Mike Rosulek. Cryptographic complexity of multi-party computation problems: Classifications and separations. In *CRYPTO*, 2008.
67. Michael O. Rabin. How to exchange secrets by oblivious transfer. *Technical Memo TR-81*, 1981.
68. Michael O. Rabin. How to exchange secrets with oblivious transfer. Cryptology ePrint Archive, Report 2005/187, 2005. https://eprint.iacr.org/2005/187.
69. Omer Reingold, Luca Trevisan, and Salil Vadhan. Notions of reducibility between cryptographic primitives. In *TCC*, 2004.
70. John Rompel. One-way functions are necessary and sufficient for secure signatures. In *STOC*, 1990.
71. Mike Rosulek and Morgan Shirley. On the structure of unconditional uc hybrid protocols. In *TCC*, 2018.
72. Steven Rudich. The use of interaction in public cryptosystems (extended abstract). In *CRYPTO*, 1991.
73. René L Schilling. *Measures, integrals and martingales*. 2017.
74. Daniel R. Simon. Finding collisions on a one-way street: Can secure hash functions be based on general assumptions? In *EUROCRYPT*, 1998.
75. Andrew Chi-Chih Yao. Protocols for secure computations (extended abstract). In *FOCS*, 1982.

Supporting Materials.

A Missing Proofs

A.1 Proof of Lemma 3

Proof. Let $\vec{X} = (X_1, \ldots, X_\ell)$ and $\vec{Y} = (Y_1, \ldots, Y_\ell)$ be the random variables of the private inputs that Alice and Bob send to the trusted party until partial transcript $m_{\le i}$. Clearly, $\vec{X}$ is a deterministic function of $M_{\le i}$ and R^{A}. Similarly, $\vec{Y}$ is a deterministic function of $M_{\le i}$ and R^{B}. Fix any r^{A} and r^{B}, let $\vec{x}$ and $\vec{y}$ be the unique inputs that is consistent with r^{A} and r^{B}. Then, we have

$$
\begin{aligned}
&\Pr\left[\left(R^{\mathsf{A}}, R^{\mathsf{B}}\right) = \left(r^{\mathsf{A}}, r^{\mathsf{B}}\right) \middle| M_{\le i} = m_{\le i}\right] \\
&= \Pr\left[\vec{X} = \vec{x}, \vec{Y} = \vec{y} \middle| M_{\le i} = m_{\le i}\right] \cdot \Pr\left[\left(R^{\mathsf{A}}, R^{\mathsf{B}}\right) = \left(r^{\mathsf{A}}, r^{\mathsf{B}}\right) \middle| M_{\le i} = m_{\le i}, \vec{X} = \vec{x}, \vec{Y} = \vec{y}\right] \\
&= \Pr\left[\vec{X} = \vec{x} \middle| M_{\le i} = m_{\le i}\right] \cdot \Pr\left[\vec{Y} = \vec{y} \middle| M_{\le i} = m_{\le i}\right] \\
&\qquad \cdot \Pr\left[R^{\mathsf{A}} = r^{\mathsf{A}} \middle| M_{\le i} = m_{\le i}, \vec{X} = \vec{x}\right] \cdot \Pr\left[R^{\mathsf{B}} = r^{\mathsf{B}} \middle| M_{\le i} = m_{\le i}, \vec{Y} = \vec{y}\right] \\
&= \Pr\left[R^{\mathsf{A}} = r^{\mathsf{A}} \middle| M_{\le i} = m_{\le i}\right] \cdot \Pr\left[R^{\mathsf{B}} = r^{\mathsf{B}} \middle| M_{\le i} = m_{\le i}\right],
\end{aligned}
$$

where in the second identity, we use the fact that f satisfies the strict cross product rule. Hence the input of f, given the output, can be sampled independently.

A.2 Proof of Claim 1

Proof. Consider the probability that both Alice's first defense and Bob's first defense are 1. On the one hand, since Alice view and Bob view are independent, this equals to the product of the probability that Alice's first defense is 1 and the probability that Bob's first defense is 1, i.e., $D_0^{\mathsf{A}} \cdot D_0^{\mathsf{B}}$. On the other hand, conditioned on the first message being $M_1 = m_1^{(i)}$, Alice view and Bob view are still independent. Hence, by the same reasoning, the probability that both Alice's first defense and Bob's first defense are 1 is $d_1^{\mathsf{A},(i)} d_1^{\mathsf{B},(i)}$. Therefore,

$$
\sum_{i=1}^{\ell} p^{(i)} \cdot d_1^{\mathsf{A},(i)} d_1^{\mathsf{B},(i)} = D_0^{\mathsf{A}} \cdot D_0^{\mathsf{B}}.
$$