entropy

MDPI

Article

Computational Hardness of Collective Coin-Tossing Protocols

Hemanta K. Maji

Department of Computer Science, Purdue University, West Lafayette, IN 47907, USA; hmaji@purdue.edu; Tel.: +1-765-494-6184

Abstract: Ben-Or and Linial, in a seminal work, introduced the full information model to study collective coin-tossing protocols. Collective coin-tossing is an elegant functionality providing uncluttered access to the primary bottlenecks to achieve security in a specific adversarial model. Additionally, the research outcomes for this versatile functionality has direct consequences on diverse topics in mathematics and computer science. This survey summarizes the current state-of-the-art of coin-tossing protocols in the full information model and recent advances in this field. In particular, it elaborates on a new proof technique that identifies the minimum insecurity incurred by any coin-tossing protocol and, simultaneously, constructs the coin-tossing protocol achieving that insecurity bound. The combinatorial perspective into this new proof-technique yields new coin-tossing protocols that are more secure than well-known existing coin-tossing protocols, leading to new isoperimetric inequalities over product spaces. Furthermore, this proof-technique's algebraic reimagination resolves several long-standing fundamental hardness-of-computation problems in cryptography. This survey presents one representative application of each of these two perspectives.

Keywords: collective coin-tossing; full information model; optimal coin-tossing protocols; isoperimetric inequalities; relativized separation; black-box separation

Citation: Maji, H.K. Computational Hardness of Collective Coin-Tossing Protocols. *Entropy* **2021**, *23*, 44. https://doi.org/doi:10.3390/e23010044

Received: 10 November 2020
Accepted: 13 December 2020
Published: 30 December 2020

1. Introduction

Ben-Or and Linial [1,2], in a seminal work, introduced the full information model to study collective coin-tossing protocols. Collective coin-tossing protocols upgrade the local independent private randomness of each of the processors into shared randomness with which all processors agree. In this model, each processor has an unbounded computational power and communicates over a broadcast channel. Collective coin-tossing is an elegant functionality providing uncluttered access to the primary bottlenecks of achieving security in a specific adversarial model. These hardness of computation results for the coin-tossing functionality extend to other general functionalities as well. Furthermore, the research outcomes for this functionality has direct consequences on diverse topics in mathematics and computer science—for example, extremal graph theory [3–5], extracting randomness from imperfect sources [6–9], cryptography [10–16], game theory [17,18], circuit representation [19–21], distributed protocols [22–24], and poisoning and evasion attacks on learning algorithms [25–28].

This survey summarizes the current state-of-the-art of coin-tossing protocols in the full information model and recent advances in this field, settling several long-standing open problems. In particular, it elaborates on a new proof technique introduced in [14] to simultaneously characterize the optimal coin-tossing protocols and prove lower bounds to the insecurity of any coin-tossing protocol. The geometric (or combinatorial) interpretation of this proof technique is inherently constructive; that is, the proof technique identifies the optimal coin-tossing protocols, which have applications in new isoperimetric inequalities in the product spaces over large alphabets [29]. This proof technique's algebraic reimagination lifts these hardness of computation results to more complex relativized settings via a new data processing inequality, central to resolving some of the most fundamental problems in computer science and cryptography [15,16,30].

1. Geometric/combinatorial proof-technique. Section 3 models a coin-tossing protocol as a martingale that evolves from $X_0 \in (0,1)$ to $X_n \in \{0,1\}$ in $n \in \mathbb{N}$ discrete time-steps. Any stopping time $\tau \in \{1,2,\ldots,n,\infty\}$ in this coin-tossing martingale translates into adversarial attacks on the coin-tossing protocol. Khorasgani, Maji, and Mukherjee [14] introduced an inductive approach that characterizes a lower bound on the insecurity $C_n(X_0)$ of any such coin-tossing protocol. Furthermore, their approach is constructive, i.e., it constructs a coin-tossing protocol such that its insecurity is, at most, $C_n(X_0)$. Surprisingly, these secure coin-tossing protocols are more secure than the folklore constructions widely believed to be optimal earlier.
2. Algebraized version. Section 4 presents an algebraic version of the proof technique mentioned above, as introduced by [16,30]. This proof technique sacrifices a small constant factor on the lower bound on insecurity. However, the algebraized proof technique extends to more complicated information-theoretic models where parties have access to oracles. These lower bounds to insecurity in complex relativized settings translate into black-box separation results [31,32] settling several long-standing open problems.
3. Connection to isoperimetric inequalities. Section 5 establishes a connection between the security of optimal coin-tossing protocols in the information-theoretic model and isoperimetric inequalities in product spaces over large alphabets. Isoperimetric inequalities in product spaces of large alphabets are known to be not sufficiently well-behaved. The cryptographic perspective into isoperimetric inequalities makes a case for new "symmetrized" versions of these isoperimetric inequalities. For example, the initial results of [29] demonstrate that these symmetrized isoperimetric inequalities are significantly more well-behaved.

2. Preliminaries and Model

2.1. System Specification

Following Ben-Or and Linial [1,2], the survey considers the standard n-processor coin-tossing protocols in the *full information* setting, i.e., the processors are computationally unbounded and send their messages over a common broadcast channel. The coin-tossing protocol proceeds in *rounds*, where a subset of the processors broadcast their messages in that round, and this subset of processors possibly depends on the messages broadcast in the previous rounds. In a *t-turn* coin-tossing protocol, each processor sends (at most) t messages during the entire protocol execution. When the protocol completes, all processors agree on a common output, their collective coin. Intuitively, a collective coin-tossing protocol upgrades the local private randomness of multiple parties into their shared randomness.

2.2. Extensions

The base system mentioned above is the *information-theoretic plain model*. The survey also encompasses this base system's extension with oracles (for example, a *random oracle*) and other ideal secure computation functionalities (for example, *secure function evaluation* functionalities). These extensions enable studying the complexity of secure coin-tossing relative to various hardness of computation assumptions and the complexity of performing other secure computations. For example, the *random oracle model* provides random oracle access to the parties. That is a random $\{0,1\}^n \to \{0,1\}^n$ function, where n represents the bit-length of the input to the random oracle. Intuitively, a random oracle answers old queries consistently and new queries uniformly and independently at random from the set $\{0,1\}^n$. The *f-hybrid model* provides parties access to the ideal f-functionality.

2.3. Adversary and Security Model

A Byzantine adversary with *corruption threshold* k may corrupt up to k processors statically (i.e., the adversary decides which processors to corrupt before the protocol execution starts), or adaptively (i.e., the adversary corrupts processors depending on the

protocol evolution). A *strong* adaptive adversary [33] can observe a processor's message before corrupting it. The adversary controls all corrupted processors' messages and is *rushing*; that is, all honest processors in a particular round broadcast their message first, and then the adversary determines the messages of the corrupted processors for that round. The adversary may choose to abort the protocol execution prematurely.

The survey considers both *security with abort* and *security with guaranteed output delivery*. Intuitively, if the adversary aborts, security with abort permits the protocol to abort without providing any output to the honest processors. On the other hand, the significantly stringent security notion of guaranteed output delivery insists that the honest processors receive output even if the adversary aborts. A coin-tossing protocol is *ε-insecure* if the adversary can change the honest processors' output distribution by, at most, ε in the total variation distance (or, equivalently, statistical distance).

2.4. Notations and Terminology

In the sequel, the common output of a coin-tossing protocol is 0 or 1. Intuitively, 1 represents heads, and 0 represents tails. The expected output of a coin-tossing protocol represents the probability of its output being heads. A *bias-X* coin-tossing protocol is an interactive protocol whose expected (common) output is $X \in [0,1]$.

This survey considers r-round coin-tossing protocols. The *transcript exposure filtration* reveals the messages of every round sequentially. The partial transcript of a protocol after i rounds is $(T_1, \ldots, T_i)$, i.e., the concatenation of the messages broadcast in rounds $1, \ldots, i$. Conditioned on the transcript, being $T_1, \ldots, T_i$, the random variable X_i represents the expected output of the protocol. Note that $X_r \in \{0,1\}$, that is, all processors agree on the output at the end of the protocol. Furthermore, the random variable X_0 represents the expected output of the protocol before the protocol began, that is, $X_0 = X$ for a bias-X coin-tossing protocol. Observe that $(X_0, X_1, \ldots, X_r)$ is a *martingale* w.r.t. the transcript exposure filtration.

2.5. Coin-Tossing Protocols as Trees

One can represent coin-tossing protocols equivalently as labeled trees. This tree representation helps develop a combinatorial intuition for the results, and enables a succinct and intuitive (yet, precise) presentation of the primary technical ideas. Every node in the tree corresponds to a partial transcript $(T_1, \ldots, T_i)$. For a node v in the tree, $T(v)$ represents its corresponding partial transcript. If a node u represents the transcript $(T_1, \ldots, T_{i-1})$ and a node v represents the transcript $(T_1, \ldots, T_i)$, then u is the *parent* of v. The *root* of the tree corresponds to the empty transcript $\emptyset$, and the leaves correspond to the complete transcripts of the protocol. The label on the edge (u, v) is the value of the random variable T_i (that is, the message sent in round i).

The *color* of a node v, represented by $X(v) \in [0,1]$, is the expected output of the protocol conditioned on the partial transcript being $T(v)$. Therefore, leaves have color 0 or 1, and the root has color X in a bias-X coin-tossing protocol. The coin-tossing protocol $\Pi(v)$ represents the bias-$X(v)$ coin-tossing protocol associated with subtree rooted at v.

For illustrative purposes, Figure 1 presents the tree representation of the "majority protocol" for $n = 3$ processors. In round $i \in \{1, 2, \ldots, n\}$, processor i broadcasts an independent uniformly random bit. The collective output $b \in \{0,1\}$ after n bits have been broadcast is the majority of these bits. The edge going left corresponds to the broadcast message being 0, and the edge going right corresponds to the broadcast message being 1. The nodes are labeled by their color.

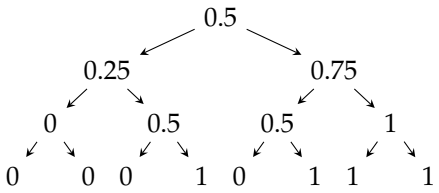

Figure 1. Transcript tree for the majority protocol, where $n = 3$.

3. Optimal Coin-Tossing Protocols: A Geometric Approach

This section introduces the original combinatorial technique of Khorasgani, Maji, and Mukherjee [14] for characterizing the "most secure" coin-tossing protocol.

3.1. A Representative Motivating Application

Consider a distributed collective coin-tossing protocol for n processors, where a processor i broadcasts its message in round i. At the end of the protocol, all processors reconstruct the common output from the public transcript. When all processors are honest, the probability of the final output being 1 is X_0 and the probability of the final output being 0 is $1 - X_0$, i.e., the final output is a *bias-*X_0 *coin*. Suppose there is an adversary who can (adaptively) choose to *restart* one of the processors after seeing her message (i.e., the *strong adaptive* corruptions model introduced by Goldwasser, Kalai, and Park [33]); otherwise her presence is innocuous. Our objective is to design bias-X_0 coin-tossing protocols, such that the adversary cannot significantly change the distribution of the final output.

In summary, we consider single-turn collective coin-tossing protocols where only one processor broadcasts every round. We consider security with abortion against an adversary that is strong [33] and adaptive. The adversary can perform a soft attack where it may restart a processor if it does not like its message.

The Majority Protocol. Against computationally unbounded adversaries, (essentially) the only known protocol is the well-known majority protocol [34–37] for $X_0 = 1/2$. The majority protocol requests one uniformly random bit from each processor and the final output is the majority of these n bits. An adversary can alter the expected output by $1/\sqrt{2\pi n}$ (more specifically, the fractional weight of the central binomial coefficient), i.e., the majority protocol is $1/\sqrt{2\pi n}$-insecure. More generally, one considers *threshold protocols*, where the collective output is 1 if and only if the total number of broadcast bits is more than a fixed threshold.

Figure 2 shows the optimal attack on the majority protocol for $n = 3$ that increases the expected output of the protocol. The shaded nodes in the tree represents the partial transcripts where the adversary intervenes and restarts the last processor that broadcast its message. The insecurity of this protocol is $\binom{n}{\lfloor n/2 \rfloor} \cdot 2^{-n} = 0.1875$. Figure 8, as a consequence of the recent works [14,38], presents a protocol that has higher security than this majority protocol.

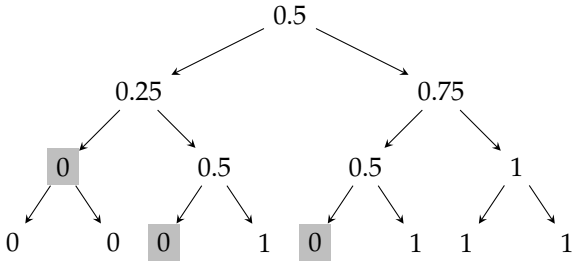

Figure 2. Transcript tree for the majority protocol, where $n = 3$. The shaded nodes represent the partial transcripts where the adversary restarts the processor that broadcast the last message in the protocol.

Towards this objective, first, the survey summarizes the new proof technique introduced by Khorasgani, Maji, and Mukherjee [14] that yields a two-approximation to the

optimal solution of the motivating problem above (Section 3.4 summarizes this proof technique). Section 3.6 includes empirical results summarizing conjectured constructions that have higher security than the threshold protocols.

3.2. Martingale Problem Statement

Given a complete transcript, let $\tau \in \{1, 2, \ldots, n, \infty\}$ represent the round where the adversary intervenes. Observe that, by restarting the processor at $\tau \in \{1, 2, \ldots, n\}$, the adversary changes the expected output of the protocol from X_τ to $X_{\tau-1}$. Therefore, the change in the expected output of the protocol is $X_{\tau-1} - X_\tau$. The intervention strategy of an adversary is equivalently represented as a stopping time $\tau \colon \Omega \to \{1, 2, \ldots, n, \infty\}$, where Ω is the set of all complete transcripts of the coin-tossing protocol. If the associated stopping time for a complete transcript is ∞, then the adversary does not intervene during the generation of that complete transcript. The increase in the expected output corresponding to this adversarial strategy τ is equal to

$$\mathrm{E}[X_{\tau-1} - X_\tau].$$

For the simplicity of presenting the primary technical ideas, it is instructive to consider a related, albeit slightly different, *score function*

$$\mathrm{E}[|X_\tau - X_{\tau-1}|].$$

The inspiration of the approach introduced by Khorasgani, Maji, Mukherjee [14] is best motivated using a two-player game between, namely, the *martingale designer* and the *adversary*. Fix n and X_0. The martingale designer presents a martingale $\mathcal{X} = (X_0, X_1, \ldots, X_n)$ (w.r.t. to the transcript exposure filtration) to the adversary and the adversary finds a stopping time τ that maximizes the score function.

$$\mathrm{E}[|X_\tau - X_{\tau-1}|]$$

Intuitively, the adversary demonstrates the most severe *susceptibility* of the martingale by presenting the corresponding stopping time τ as a witness. The stopping time witnessing the highest susceptibility shall translate into appropriate adversarial strategies. The martingale designer's objective is to design martingales that have less susceptibility. Khorasgani et al. [14] introduce a geometric approach to inductively provide tight bounds on the least susceptibility of martingales for all $n \geq 1$ and $X_0 \in [0, 1]$, that is, the following quantity.

$$C_n(X_0) \coloneqq \inf_{\mathcal{X}} \sup_{\tau} \mathrm{E}[|X_\tau - X_{\tau-1}|]$$

Similar to [10], this precise study of $C_n(X_0)$, for general $X_0 \in [0, 1]$, is motivated by natural applications in discrete process control as illustrated by the representative motivating problem.

3.3. Prior Approaches to the General Martingale Problem

Azuma–Hoeffding inequality [39,40] states that, if $|X_i - X_{i-1}| = o(1/\sqrt{n})$, for all $i \in \{1, \ldots, n\}$, then, essentially, $|X_n - X_0| = o(1)$ with probability 1. That is, the final information X_n remains close to the a priori information X_0. However, in our problem statement, we have $X_n \in \{0, 1\}$. In particular, this constraint implies that the final information X_n is significantly different from the a priori information X_0. So, the initial constraint "for all $i \in \{1, \ldots, n\}$ we have $|X_i - X_{i-1}| = o(1/\sqrt{n})$" must be violated. What is the probability of this violation?

For $X_0 = 1/2$, Cleve and Impagliazzo [10] proved that there exists a round i such that $|X_i - X_{i-1}| \geq \frac{1}{32\sqrt{n}}$ with probability $1/5$. We emphasize that the round i is a random variable and not a constant. However, the definition of the "big jump" and the "probability to encounter big jumps" are both exponentially small functions of X_0. So, the approach of

Cleve and Impagliazzo is only applicable to constant $X_0 \in (0,1)$. Recently, in an independent work, Beimel et al. [41] demonstrate an identical bound for *weak martingales* (that have some additional properties), which is used to model multi-party coin-tossing protocols.

For the upper-bound, on the other hand, Doob's martingale, corresponding to the majority protocol, is the only known martingale for $X_0 = 1/2$ with a small *maximum susceptibility*. In general, to achieve arbitrary $X_0 \in [0,1]$, one considers coin-tossing protocols, where the output is 1 if the total number of heads in n uniformly random coins surpasses an appropriate threshold.

3.4. Inductive Approach

This section presents a high-level overview of the inductive strategy to characterizing optimal coin-tossing protocols. In the sequel, we shall assume that we are working with discrete-time martingales $(X_0, X_1, \ldots, X_n)$ such that $X_n \in \{0,1\}$.

Given a martingale $(X_0, \ldots, X_n)$, its *susceptibility* is represented by the following quantity

$$\sup_{\text{stopping time } \tau} \mathrm{E}[|X_\tau - X_{\tau-1}|]$$

Intuitively, if a martingale has high susceptibility, then it has a stopping time, such that the gap in the martingale while encountering the stopping time is large. Our objective is to characterize the *least susceptibility* that a martingale $(X_0, \ldots, X_n)$ can achieve. More formally, given n and X_0, characterize

$$C_n(X_0) \coloneqq \inf_{(X_0,\ldots,X_n)} \sup_{\text{stopping time } \tau} \mathrm{E}[|X_\tau - X_{\tau-1}|].$$

The approach proceeds by induction on n to exactly characterize the curve $C_n(X)$, and our argument naturally constructs the best martingale that achieves $C_n(X_0)$.

1. Base case. Note that the base case is $C_1(X) = 2X(1-X)$ (see Figure 3 for this argument).
2. Inductive step. Given the curve $C_{n-1}(X)$, one identifies a *geometric transformation* T (see Figure fig:transform-def) that defines the curve $C_n(X)$ from the curve $C_{n-1}(X)$.

   Furthermore, for any $n \geq 1$, there exist martingales such that its susceptibility is exactly $C_n(X_0)$.

We shall prove the following technical result in this section.

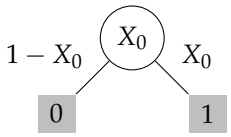

Figure 3. Base case for the inductive approach. Note $C_1(X_0) = \inf_{(X_0,X_1)} \sup_\tau \mathrm{E}[|X_\tau - X_{\tau-1}|]$. The optimal stopping time is shaded and its score is $X_0 \cdot |1 - X_0| + (1 - X_0) \cdot |0 - X_0| = 2X_0(1 - X_0)$.

Theorem 1. *Fix any $X_0 \in (0,1)$ and $n \in \mathbb{N}$. Let $\mathcal{X} = (X_0, X_1, \ldots, X_n)$ be a martingale, such that $X_n \in \{0,1\}$. There exists a stopping time τ in such that*

$$\mathrm{E}[|X_\tau - X_{\tau-1}|] \geq C_n(X).$$

Furthermore, for all $n \in \mathbb{N}$ and $X_0 \in (0,1)$, there exists a martingale $\mathcal{X}^* = (X_0, X_1^*, \ldots, X_n^*)$ such that $X_n^* \in \{0,1\}$ and, for all stopping times τ, we have

$$\mathrm{E}[|X_\tau^* - X_{\tau-1}^*|] = C_n(X_0).$$

Base Case of $n = 1$

Refer to Figure 3 for the following discussion. For a martingale (X_0, X_1) of depth $n = 1$, we have $X_1 \in \{0,1\}$. Thus, without loss of generality, we assume that E_1 takes only

two values. Then, it is easy to verify that the max score is always equal to $2X_0(1-X_0)$. This score is witnessed by the stopping time $\tau = 1$. So, we conclude that $C_1(X_0) = 2X_0(1-X_0)$.

Inductive Step: $n = 2$ (For Intuition).

Suppose that the root $X_0 = x$ in the corresponding martingale tree has t children with values $x^{(1)}, x^{(2)}, \ldots, x^{(t)}$, and the probability of choosing the j-th child is $p^{(j)}$, where $j \in \{1, \ldots, t\}$ (see Figure 4).

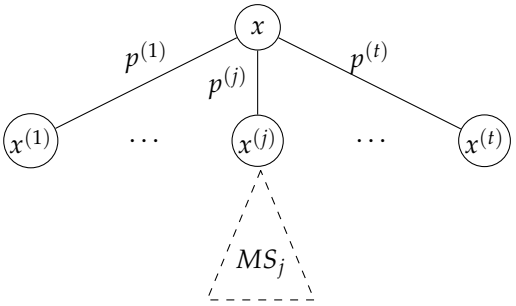

Figure 4. The inductive step of the combinatorial proof. MS_j represents the max-score of the sub-tree of depth $n-1$ which is rooted at $x^{(j)}$. For simplicity, the subtree of $x^{(j)}$ is only shown here.

Given a martingale (X_0, X_1, X_2), the adversary's objective is to find the stopping time τ that maximizes the score $\mathrm{E}[|X_\tau - X_{\tau-1}|]$. If the adversary chooses to stop at $\tau = 0$, then the score $\mathrm{E}[|X_\tau - X_{\tau-1}|] = 0$, which is not a good strategy. So, for each j, the adversary chooses whether to stop at the child $x^{(j)}$, or defer the attack to a stopping time in the sub-tree rooted at $x^{(j)}$. The adversary chooses the stopping time based on which of these two strategies yield a better score. If the adversary stops the martingale at child j, then the contribution of this decision to the score is $p^{(j)} \cdot \left|x^{(j)} - x\right|$. On the other hand, if she does not stop at child j, then the contribution from the sub-tree is guaranteed to be $p^{(j)} \cdot MS_j \geq p^{(j)} \cdot C_1(x^{(j)})$. Overall, from the j-th child, an adversary obtains a score that is at least $p^{(j)} \cdot \max\left\{\left|x^{(j)} - x\right|, C_1(x^{(j)})\right\}$.

Let $h^{(j)} := \max\left\{\left|x^{(j)} - x\right|, C_1(x^{(j)})\right\}$. We represent the points $Z^{(j)} = (x^{(j)}, h^{(j)})$ in a two dimensional plane. Then, clearly, all these points lie on the solid curve defined by $\max\{|X - x|, C_1(X)\}$—see Figure 5.

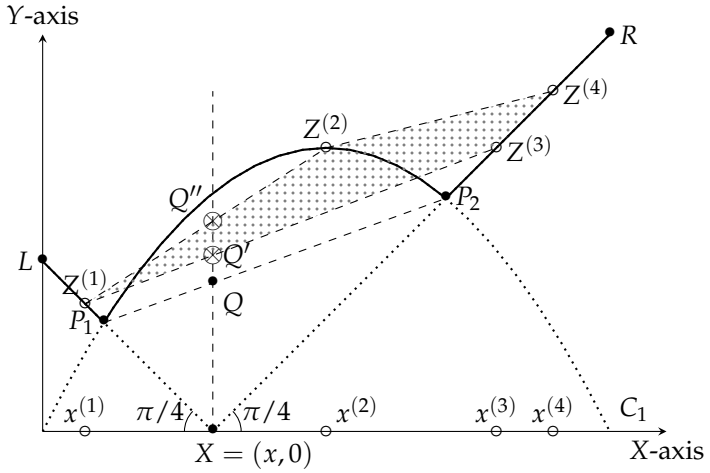

Figure 5. Intuitive summary of the inductive step for $n = 2$.

Since (X, E) is a martingale, we have $x = \sum_{j=1}^{t} p^{(j)} x^{(j)}$ and the adversary's strategy for finding $\tau_{\max}$ gives us $\lambda = \sum_{j=1}^{t} p^{(j)} h^{(j)}$. This observation implies that the coordinate

$(x, \lambda) = \sum_{j=1}^{t} p^{(j)} \cdot Z^{(j)}$. So, the point in the plane giving the adversary the maximum score for a tree of depth $n = 2$ with bias $X_0 = x$ lies in the *intersection* of the convex hull of the points $Z^{(1)}, \ldots, Z^{(t)}$, and the line $X = x$. Let us consider the martingale defined in Figure 5 as a concrete example. Here $t = 4$, and the points $Z^{(1)}, Z^{(2)}, Z^{(3)}, Z^{(4)}$ lie on $\max\{|X - x|, C_1(X)\}$. The martingale designer specifies the probabilities $p^{(1)}, p^{(2)}, p^{(3)}$, and $p^{(4)}$, such that $p^{(1)}x^{(1)} + \cdots + p^{(4)}x^{(4)} = x$. These probabilities are not represented in Figure 5. Note that the point $\left(p^{(1)}x^{(1)} + \cdots + p^{(4)}x^{(4)}, p^{(1)}h^{(1)} + \cdots + p^{(4)}h^{(4)}\right)$ representing the score of the adversary is the point $p^{(1)}Z^{(1)} + \cdots + p^{(4)}Z^{(4)}$. This point lies inside the convex hull of the points $Z^{(1)}, \ldots, Z^{(4)}$ and on the line $X = p^{(1)}x^{(1)} + \cdots + p^{(4)}x^{(4)} = x$. The exact location depends on $p^{(1)}, \ldots, p^{(4)}$.

Point Q' is the point with minimum height. Observe that the height of the point Q' is at least the height of the point Q. So, in any martingale, the adversary shall find a stopping time that scores more than (the height of) the point Q.

On the other hand, the martingale designer's objective is to reduce the score that an adversary can achieve. So, the martingale designer chooses $t = 2$, and the two points $Z^{(1)} = P_1$ and $Z^{(2)} = P_2$ to construct the optimum martingale. We apply this method for each $x \in [0, 1]$ to find the corresponding point Q; that is, the *locus of the point* Q, for $x \in [0, 1]$, which yields the curve $C_2(X = x)$.

Observe that the height of the point Q is the *harmonic-mean* of the heights of the points P_1 and P_2. This observation follows from elementary geometric facts. Let h_1 represent the height of the point P_1, and h_2 represent the height of the point P_2. Observe that the distance of $x - x_S(x) = h_1$ (because the line ℓ_1 has slope $\pi - \pi/4$). Similarly, the distance of $x_L(x) - x = h_2$ (because the line ℓ_2 has slope $\pi/4$). So, using properties of similar triangles, the height of Q turns out to be

$$h_1 + \frac{h_1}{h_1 + h_2} \cdot (h_2 - h_1) = \frac{2h_1h_2}{h_1 + h_2}.$$

This property inspires the definition of the geometric transformation T, see Figure 6. Applying T on the curve $C_1(X)$ yields the curve $C_2(X)$. All bias-X $(n = 2)$ processor coin-tossing protocols are $C_n(X)$-insecure. Furthermore, there exists a coin-tossing protocol that achieves this insecurity bound.

General Inductive Step: $n \geq 2$

Note that a similar approach works for general $n = d \geq 2$. Fix X_0 and $n = d \geq 2$. We assume that the adversary can compute $C_{d-1}(X_1)$, for any $X_1 \in [0, 1]$.

Suppose the root in the corresponding martingale tree has t children with values $x^{(1)}, x^{(2)}, \ldots, x^{(t)}$, and the probability of choosing the j-th child is $p^{(j)}$ (see Figure 4). Let $(X^{(j)}, E^{(j)})$ represent the martingale associated with the sub-tree rooted at $x^{(j)}$.

For any $j \in \{1, \ldots, t\}$, the adversary can choose to stop at the child j. This decision will contribute $\left|x^{(j)} - x\right|$ to the score with weight $p^{(j)}$. On the other hand, if she defers the attack to the subtree rooted at $x^{(j)}$, she will get at least a contribution of (at least) $C_{n-1}(x^{(j)})$, with weight $p^{(j)}$. Therefore, the adversary can obtain the following contribution to her score

$$p^{(j)} \max\left\{\left|x^{(j)} - x\right|, C_{d-1}(x^{(j)})\right\}$$

Similar to the case of $n = 2$, we define the points $Z^{(1)}, \ldots, Z^{(t)}$. For $n > 2$, however, there is one difference from the $n = 2$ case. The point $Z^{(j)}$ need not *lie on the solid curve*, but it can lie on or above it, i.e., they lie in the gray area of Figure 7. This phenomenon is attributable to a suboptimal martingale designer, producing martingales with suboptimal scores, i.e., *strictly above* the solid curve. For $n = 1$, it happens to be the case that there is (effectively) only one martingale that the martingale designer can design (the optimal tree). The adversary obtains a score that is at least the height of the point Q', which is at least the height of Q. On the other hand, the martingale designer can choose $t = 2$, and $Z^{(1)} = P_1$

and $Z^{(2)} = P_2$ to define the optimum martingale. Again, the locus of point Q is defined by the curve $T(C_{d-1})$.

Given. A curve C defined by the zeroes $\subseteq \mathbb{R}^2$ of equation $Y = f(X)$, where $X \in [0,1]$.
Definition of the Transform. The transform of C, represented by $T(C)$, is the curve defined by the zeroes of equation $Y = g(X)$, where, for $x \in [0,1]$, the value of $g(x)$ is defined below.

1. Let $x_S(x) \in [0,1]$ be a solution of the equation $X + f(X) = x$.
2. Let $x_L(x) \in [0,1]$ be a solution of the equation $X - f(X) = x$.
3. Then $g(x) \coloneqq \text{H.M.}(y^{(1)}, y^{(2)})$, where $y^{(1)} = f(x_S(x))$, $y^{(2)} = f(x_L(x))$, and H.M.$(y^{(1)}, y^{(2)})$ represents the harmonic mean of $y^{(1)}$ and $y^{(2)}$, i.e., $\frac{2y^{(1)}y^{(2)}}{y^{(1)}+y^{(2)}}$.

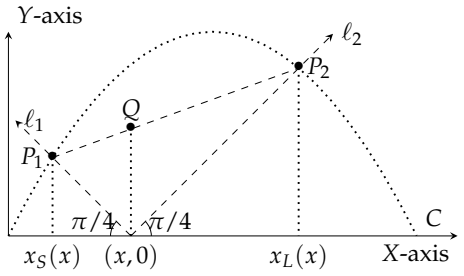

Figure 6. Definition of transform of a curve C, represented by $T(C)$. The locus of the point Q (in the right figure) defines the curve $T(C)$.

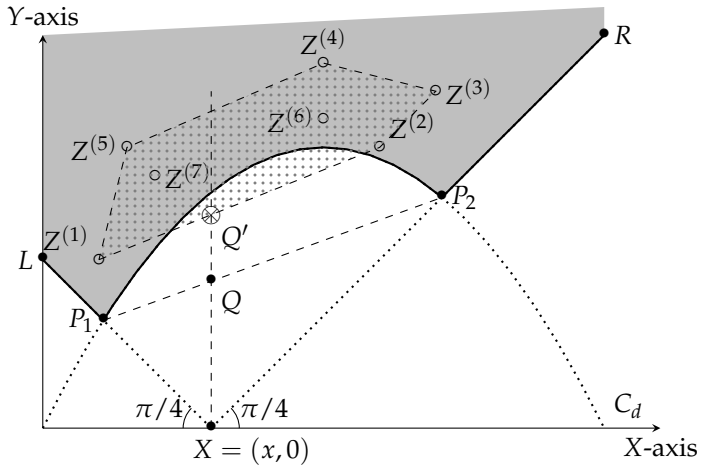

Figure 7. Intuitive summary of the inductive argument. Our objective is to pick the set of points $\{Z^{(1)}, Z^{(2)} \dots\}$ in the gray region to minimize the length of the intercept XQ' of their (lower) convex hull on the line $X = x$. Clearly, the unique optimal solution corresponds to including both P_1 and P_2 in the set.

Conclusion

So, by induction, we have proved that $C_n(X) = T^{n-1}(C_1(X))$. Additionally, note that, during induction, in the optimum martingale, we always have $\left|x^{(0)} - x\right| = C_{n-1}(x^{(0)})$ and $\left|x^{(1)} - x\right| = C_{n-1}(x^{(1)})$. Intuitively, the decision to stop at $x^{(j)}$ or continue to the subtree rooted at $x^{(j)}$ has identical consequence. So, by induction, *all stopping times* in the optimum martingale have score $C_n(x)$.

A close-form characterization of $C_n(X)$ using elementary functions seems challenging. Khorasgani et al. [14] proved the following upper and lower bounds.

$$\min\left\{\frac{2}{\sqrt{n+3}}\cdot\sqrt{X(1-X)}, 2X, 2-2X\right\} \geq C_n(X) \geq \sqrt{\frac{2}{n-1/2}}\cdot X(1-X).$$

3.5. Related Work: Multiple Corruptions

Another line of research characterizes the minimum number of corruptions t that suffices to change the expected output of the coin-tossing protocol by a constant. The presentation below, for simplicity, ignores polylogarithmic factors in the asymptotic notation. The authors in [42] proved that a Byzantine adversary can adaptively corrupt $t = \widetilde{\mathcal{O}}(\sqrt{n})$ processors in any n-processor single-turn protocol, where every processor broadcasts one-bit messages, to change the expected output of the protocol by a constant. Subsequently, [33,43] generalized this result to the case where the processors broadcast arbitrary-length messages. Recently, in a breakthrough result, Haitner and Karidi-Heller [44] extended this result to *multi-turn* coin-tossing protocols, i.e., a processor may send messages in multiple rounds. Essentially, these results imply that the majority protocol (more generally, the threshold protocols) are qualitatively optimal. However, the characterization of the most secure coin-tossing protocols remains open.

A prominent model in distribution computing considers the following adversarial model for coin-tossing protocols. A strong adversary can adaptively corrupt (up to) t processors and the messages of all corrupted processors are erased. Aspnes [22,23] uses an inductive approach to characterize the robustness of such coin-tossing protocols. This approach also uses a geometric approach to perform induction on t, the number of corruptions that the adversary makes, to account for (a) the maximum increase in the expected output of the coin-tossing protocol and (b) the maximum decrease in the expected output of the coin-tossing protocols. [22,23] proves that $t = \mathcal{O}(\sqrt{n})$ suffices to change the expected output of an n-processor coin-tossing protocol by a constant. However, this inductive approach is non-constructive because the recursion does not characterize the evolution of the martingale corresponding to the most secure coin-tossing protocol.

3.6. Experimental Results

The presentation above considers the case where the stopping time representing an adversarial strategy is $\tau\colon \Omega \to \{1, 2, \ldots, n\}$ (where Ω represents the set of all complete transcripts), and the score of a stopping time is $\mathrm{E}[|X_\tau - X_{\tau-1}|]$. Khorasgani, Maji, Mehta, Mukherjee, and Wang [14,38] study a related recursion. In this recursion, the stopping time is $\tau\colon \Omega \to \{1, 2, \ldots, n, \infty\}$. However, the stopping times are restricted as follows. Given a partial transcript u, if the adversary has the following choices: (1) Do not abort for any child of u; (2) Abort at all children v, such that $X(v)$ (i.e., the expected output conditioned on v) is at least a particular threshold; (3) Abort at all children v such that $X(v)$ is at most a particular threshold. The optimal score for such restricted stopping times is represented by $A_n(X)$. The authors in [38] construct an algorithm with running time $\mathsf{poly}(n, 1/\delta)$ for computing $A_n := T^{n-1}(A_1)$, where $A_1(X) = X(1-X)$ with (at most) $n\delta$ error. We highlight that the geometric transformation $T(\cdot)$ is identical to the one presented in Section 3.4. However, the base cases are different; $A_1(X) = X(1-X)$, but $C_1(X) = 2X(1-X)$. Now, consider the optimal protocol corresponding to this recursion. For example, Figure 8 shows the martingale corresponding to $X_0 = 1/2$ and $n = 3$. The optimal attack that increases the expected output is represented by the shaded nodes. Restarting the last processor broadcasting the message resulting in a shaded partial transcript increases the output by 0.1362, which is significantly less than 0.1865, the insecurity of the majority protocol from Figure 2.

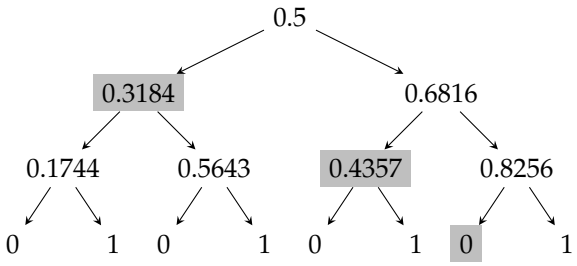

Figure 8. Transcript tree for the optimal protocol corresponding to the recursion $A_1(X) := X(1-X)$ and $A_n = T^{n-1}(A_1)$, where $n = 3$. The shaded partial transcripts represents the attack that increases the expected output by 0.1362, which is also the maximum insecurity of this coin-tossing protocol.

Experimentally, we implement our protocol and show that the insecurity of our protocol is observably smaller than the insecurity of threshold protocols. As a representative example, Figure 9 plots the insecurity of our new protocol, for $n = 101$ processors and $X \in [0, 1/2]$ with accuracy parameter $\delta = 10^{-6}$. This demonstrates the insecurity of bias-X coin-tossing protocols, where $X \in (1/2, 1]$, is identical to the insecurity of bias-$(1-X)$ coin-tossing protocols. So, it suffices to consider bias-X protocols, where $X \in [0, 1/2]$.

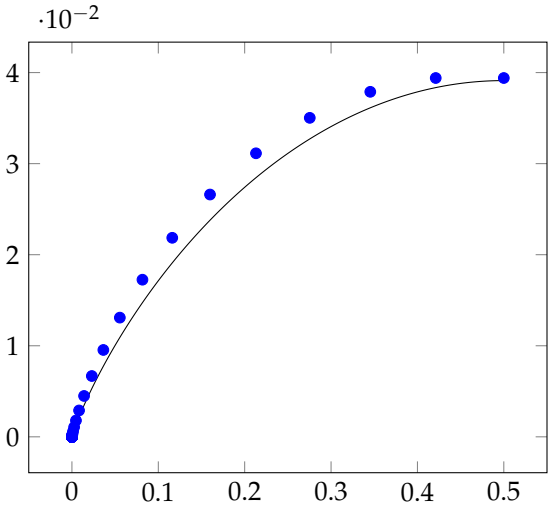

Figure 9. The X-axis represents the bias of the coin-tossing protocols and the Y-axis represents the insecurity of the protocols. For $n = 101$, the blue marks denote the insecurity of bias-X coin-tossing protocols that are implementable using a threshold protocol. The black curve represents the insecurity of our new protocols. A coin-tossing protocol with lower insecurity is more desirable.

Figure 9 also plots the insecurity of all bias-X coin-tossing protocols that can be implemented using a threshold protocol. Note that the insecurity of our protocol is less than the insecurity of threshold protocol. This reduction in insecurity is prominent, especially when $X \in (0, 1/2)$ is simultaneously far from 0 and $1/2$.

Finally, our experiments uncover an exciting phenomenon. As Figure 10 indicates, our experimental results show that the insecurity of our protocols for $X = 1/2$ tends towards the insecurity of the majority protocol, as n tends to infinity. This experiment lends support to the conjecture that the majority protocol is the optimal secure coin-tossing protocol as $n \to \infty$. However, for every finite n and $X \in (0, 1/2)$, there are more secure protocols than the threshold protocols.

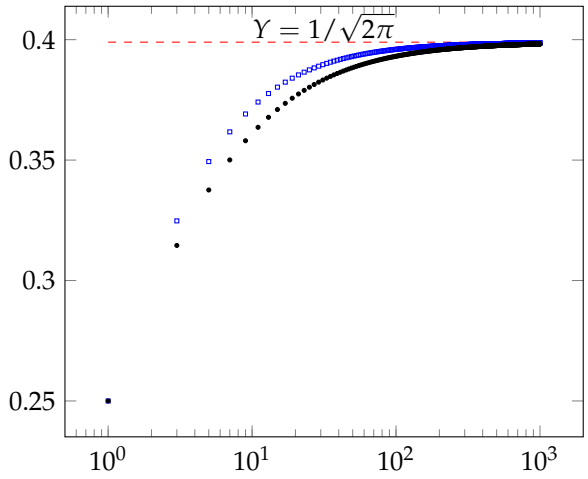

Figure 10. For $n \in \{1, 3, \ldots, 1001\}$, the blue squares show the plot of $\sqrt{n}$-times the insecurity of the majority coin-tossing protocol. The red dashed line shows the limit of the insecurity of majority protocol using Stirling's approximation, when $n \to \infty$. The black dots show the plot of $\sqrt{n}$-times the security of our new optimal protocols. The graph uses log scale on the X-axis.

4. Hardness of Computation Relative to Oracles

This section considers two-processor one-message-per-round coin-tossing protocols with guaranteed output delivery. The first processor, say, Alice sends messages in rounds $\{1, 3, \ldots\}$, and the second processor (Bob) sends messages in rounds $\{2, 4, \ldots\}$. This coin-tossing protocol has n-rounds, and Alice has $\lceil n/2 \rceil$ turns and Bob has $\lfloor n/2 \rfloor$ turns. A Byzantine adversarial processor may abort prematurely during the protocol execution; however, the honest processor still has to output a coin. This coin-tossing protocol is the well-known *fair coin-tossing protocol* [45]. A fair coin-tossing protocol is ε-*unfair* if it is ε-insecure.

4.1. Summary of Relevant Work

4.1.1. Impagliazzo's Worlds

Among the several potential approaches to summarize the state-of-the-art in this field, the author prefers positioning the results in Impagliazzo's multiverse [46] (refer to Table 1). The sequel considers three of the five Impagliazzo's multiverses, which are most relevant to the discussion. In Pessiland, every efficient function is efficiently invertible, i.e., one-way functions do not exist [47]. One-way functions exist in Minicrypt and allow for private-key cryptographic primitives, such as pseudorandom generators [48–50], pseudorandom functions [51,52], pseudorandom permutations [53], statistically binding commitment [54], statistically hiding commitment [55,56], zero-knowledge proofs [57], and digital signatures [58,59]. However, public-key cryptography, as with public-key encryption, key-agreement, and general secure computation, are not possible in Minicrypt [31,60–62]. The secure constructions of these primitives lie in Cryptomania. One should imagine Cryptomania as a conglomeration of infinitely many hardness of computation assumptions that are *not* securely attainable based solely on the existence of one-way functions. Intuitively, Minicrypt enables fast cryptography, while Cryptomania supports more sophisticated, yet comparatively slower, cryptography.

Table 1. State-of-the-art constructions and hardness of computation results in fair coin-tossing. The expression "$f \not\to$ OT" represents the fact that there is no secure protocol for oblivious transfer [63,64] in the f-hybrid model against honest-but-curious adversaries.

	Secure Construction	**Adversarial Attack**
Pessiland		In General: $\Omega(1)$-unfair [65,66]
		Fail-stop Adversary: $\Omega(1/\sqrt{n})$-unfair [10,14]
Minicrypt	One-way Functions: $\mathcal{O}(1/\sqrt{n})$-unfair [34–37]	One-way Functions: $\Omega(1/\sqrt{n})$-unfair [16]
Cryptomania	Public-key Encryption:	Public-key Encryption: $\Omega(1/\sqrt{n})$-unfair [30]
	PKE + f-hybrid, $f \not\to$ OT:	PKE + f-hybrid, $f \not\to$ OT: $\Omega(1/\sqrt{n})$-unfair [30]
	Oblivious Transfer: $\mathcal{O}(1/n)$-unfair [67]	Oblivious Transfer: $\Omega(1/n)$-unfair [37]

4.1.2. Relevant Results

In the information-theoretic setting, in any protocol, one of the processors can always force an output with certainty (using attacks in two-player zero-sum games, or games against nature [68]). However, one can emulate this attack relying on a significantly weaker assumption. For instance, if one-way functions do not exist [47] then Berman, Haitner, Omri, and Tentes [65,66] rely on rejection sampling techniques to construct a Byzantine adversary that forces an output with (near) certainty. (Intuitively, the assumption that "one-way functions do not exist" is slightly weaker than the assumption that "$\mathsf{NP} \subseteq \mathsf{BPP}$," because the latter implies the former. However, these two assumptions are extremely similar in nature as well. The former assumption, roughly states that every efficient function is efficiently invertible on an average. The latter assumption, on the other hand, states that every efficient function is efficiently invertible for every element in the range. For example, the constant-unfairness of any coin-tossing protocol, based on the assumption that $\mathsf{NP} \subseteq \mathsf{BPP}$, is implied by [69].) That is, in summary, if one-way functions do not exist, then every coin-tossing protocol is $\Omega(1)$-unfair. If one restricts to attacks where parties are honest-but-curious but may abort during the protocol execution, referred to as *fail-stop adversaries*, then, if one-way functions do not exist, every coin-tossing protocol is $\Omega(1/\sqrt{n})$-unfair [10,14].

On the other hand, if one-way functions exist, then (using commitments and zero-knowledge proofs) there are protocols that are $\mathcal{O}(1/\sqrt{n})$-insecure [34–37], that is, the adversary can change the honest processor's output distribution by at most $\mathcal{O}(1/\sqrt{n})$ (in the statistical distance). In a ground-breaking result, Moran, Naor, and Segev [67] presented the first fair coin-tossing protocol that is only $\mathcal{O}(1/n)$-unfair based on the existence of (unfair) secure protocols for oblivious transfer functionality. Cleve [37] proved that $\Omega(1/n)$-unfairness is unavoidable; hence, the protocol by Moran, Naor, and Segev is "optimal."

Recent Advances. A recent work [16] proves that any fair coin-tossing protocol using one-way functions in a *black-box manner* is $\Omega(1/\sqrt{n})$-unfair, settling this long-standing open problem and proving the qualitative optimality of the protocol by [34–37]. Black-box separation is a prominent technique introduced by Impagliazzo and Rudich [31], and further nuances in its definition were highlighted in [32,70]. Suppose one "*black-box separates* the cryptographic primitive Q from another cryptographic primitive P." Then, one interprets this result as indicating that the primitive P is unlikely to facilitate the secure construction of Q using black-box constructions. Most constructions in theoretical computer science and cryptography are black-box in nature. That is, they rely only on the input–output behavior of the primitive P, and are oblivious to, for instance, the particular

implementation of the primitive P. The security reduction in cryptographic black-box constructions also uses the adversary in a black-box manner. There are, however, some highly non-trivial nonblack-box constructions in theoretical computer science, for example, [57,71–77]. However, an infeasibility of black-box constructions to realize Q from P indicates the necessity of new nonblack-box constructions, which, historically, have been significantly infrequent. Prior to this result, this black-box separation was known only for restricted families of constructions [11,12,78,79].

Subsequently, using similar techniques, ref. [30] further strengthens this separation beyond Minicrypt by proving that any fair coin-tossing protocol using a public-key encryption algorithm (which is even stronger than one-way functions) in a black-box manner is also $\Omega\left(1/\sqrt{n}\right)$-unfair. In fact, ref. [30] proves a stronger result. Let f be a secure function evaluation functionality, such that (two-choose-one single-bit) oblivious transfer [63,64] can be securely implemented in the f-hybrid model (i.e., a system where parties have access to an idealized implementation of the f-functionality). In this f-hybrid model, one can emulate the optimal fair coin-tossing protocol of Moran, Naor, and Segev [67], which is $\mathcal{O}(1/n)$-unfair. However, consider f, such that oblivious transfer is impossible in the f-hybrid model (all such functions were characterized by [80]). In the f-hybrid model, ref. [30] proves that any fair coin-tossing protocol using a public-key encryption protocol in a black-box manner is $\Omega\left(1/\sqrt{n}\right)$-unfair. These results prove that the set of all secure function evaluation functionalities have the following dichotomy. Given any secure function evaluation f, either there exists an optimal fair coin-tossing protocol in the f-hybrid model, or any coin-tossing protocol (even using public-key encryption in a black-box manner) in the f-hybrid model is $\Omega\left(1/\sqrt{n}\right)$-unfair.

4.2. Augmented Protocols

Discussion

Let Π be a coin-tossing protocol in the information-theoretic plain model. Assume that T is the partial transcript of the protocol. Suppose V_A is the random variable denoting the private view of Alice, and V_B is the random variable denoting the private view of Bob. Then, the following Markov chain is an essential property of protocols in the information-theoretic plain model.

$$V_A \leftrightarrow T \leftrightarrow V_B.$$

That is, conditioned on the partial transcript T, the joint distribution of Alice–Bob private views $(V_A, V_B|T)$ is a product of the marginal distributions of their views, that is $(V_A|T) \times (V_B|T)$. However, when parties have access to an oracle, then this property need not hold.

Augmented Protocols

In the random oracle model, it is not necessary that the joint distribution of Alice–Bob private views $(V_A, V_B|T)$ is a product of the marginal distributions of their views. Their views may be correlated via the random oracle and this correlation is not eliminated by public transcript. For example, if the private queries to the random oracle by Alice and Bob are disjointed, then the joint distribution of their private views is a product distribution. However, if they have common private queries then, conditioned on the public transcript, their views may be correlated. For example, suppose Alice and Bob privately query at 0. In this case, conditioned on the public transcript, the answer to the query has n-bits of entropy. The views of Alice and Bob are perfectly correlated, i.e., the mutual information of their private views is n.

There are standard techniques to circumvent this bottleneck. There exists a *public querying algorithm* (one that depends only on the public transcript of the protocol) and performs additional queries to the random oracle and can reduce this correlation. In particular, for any $\varepsilon \in (0,1)$, there are public algorithms [31,62,81] that perform $\mathsf{poly}(n/\varepsilon)$ additional queries to the random oracle, such that, (with high probability) conditioned on these additional query–answer pairs, the joint distribution of Alice–Bob views is ε-close

(in the statistical distance) to the product of its marginal distributions. We emphasize that the parameter ε may depend on the input-length n of the random oracle and the round complexity r of the protocol.

Consequently, any protocol in the random oracle model can be converted into an augmented protocol where the party sending the next message in the protocol adds the query–answer pairs of the public querying algorithm mentioned above. This compilation preserves the round complexity of the protocol while ensuring that the crucial Markov chain property holds. Therefore, for the rest of this section, we shall assume that the coin-tossing protocol in the random oracle model is an augmented protocol.

4.3. Technical Proof

Recall that, in augmented coin-tossing protocols, the joint distribution of Alice–Bob views is ε close to the product of their marginal distribution with high probability. For the simplicity of presentation, we shall assume that $V_A \leftrightarrow T \leftrightarrow V_B$ for augmented coin-tossing protocols in the random oracle model. The simplified analysis captures all the primary innovations of the new proof strategy of Maji and Wang [16,30]. The actual analysis only introduces a $\mathsf{poly}(\varepsilon)$ slack in the simplified analysis presented below.

Main Result and Inductive Hypothesis

Our objective is to prove the following result.

Theorem 2. *There exists a universal constant $C > 0$, such that any two-party r-message bias-X fair coin-tossing protocol in the random oracle model is at least $\frac{C}{\sqrt{r}} \cdot X(1-X)$-unfair.*

This result, proved in [16], implies that any coin-tossing protocol using one-way functions in a (fully) black-box manner is $\Omega\left(1/\sqrt{r}\right)$-unfair. [30] extends this result to demonstrate a separation from public-key encryption even in an f-hybrid model, where f is a secure function evaluation functionality, such that secure oblivious transfer does not exist in the f-hybrid model.

Towards proving this theorem, the survey shall summarize the proof of the following technical result. Consider a stopping time $\tau \colon \Omega \to \{1, 2, \ldots, n, \infty\}$. Let X_τ represent the expected color conditioned on the partial transcript $(T_1, \ldots, T_\tau)$. Similarly, A_τ represents the expected Alice defense conditioned on the fact that Bob aborts at the partial transcript $(T_1, \ldots, T_\tau)$. Analogously, B_τ represents the expectation of the Bob defense coin conditioned on the fact that Alice aborts at the partial transcript $(T_1, \ldots, T_\tau)$.

Theorem 3. *There exists a universal constant $C > 0$, such that for any two-party r-message bias-X, the fair coin-tossing protocol in the random oracle model the following bound holds.*

$$\sup_{\text{stopping time } \tau} \mathrm{E}[|X_\tau - A_\tau| + |X_\tau - B_\tau|] \geq \frac{4C}{\sqrt{n}} \cdot X(1-X).$$

Given Theorem 3 it is easy to prove Theorem 2 using an averaging argument. The stopping time τ, witnessing the attack inTheorem 3, accounts for four different attacks: Alice/Bob increasing/decreasing the expected output of the coin-tossing protocol. So, one of these four attacks changes the expected output of the coin-tossing protocol by at least $\frac{C}{\sqrt{n}} \cdot X(1-X)$.

Potential Function

Our analysis shall use the following potential function.

$$\Phi(x, a, b) := x(1-x) + (x-a)^2 + (x-b)^2.$$

Looking ahead, the variable x shall represent the expected output conditioned on the partial transcript being T. Furthermore, a shall represent the expected Alice defense coin (i.e., Alice output if Bob aborts) conditioned on the partial transcript being T. Similarly, b shall represent the expected Bob defense coin conditioned on the partial transcript being T. Intuitively, in hindsight, the term $x(1-x)$ in the potential represents the quality of the fail-stop attack, owing to the entropy in the output. For example, if the expected output x is already close to 0 or 1, it is possible that one cannot significantly change the output of the protocol. However, if the expected output x is far from both 0 and 1, then we shall show that a large attack is possible. Furthermore, the term $(x-a)^2$, intuitively, captures the quality of the attack on honest Alice if she defends improperly. For example, if Bob aborts, then the output distribution of Alice changes by $|x-a|$. Similarly, the term $(x-b)^2$ captures the quality of the attack on honest Bob if his expected defense is far from the expected output.

We remark that the function $\Phi(\cdot,\cdot,\cdot)$ is *not* convex, in general. However, Jensen's inequality holds when one considers the evolution of augmented coin-tossing protocols.

Convexity of the Potential Function in Augmented Protocols

Given a partial transcript T of an augmented protocol, conditioned on this partial transcript T, let (1) X represent the expected output of the protocol; (2) A be the expected Alice defense coin; (3) B be the expected Bob defense coin. Consider one step in the evolution of the protocol. Let T' be the random variable representing the transcript that extends the transcript T by one step in the protocol evolution. We represent this randomized evolution as $T' \vdash T$. Let X', A', B' represent the expected output, Alice defense, and Bob defense, respectively, conditioned on the partial transcript T'.

Observe that the following identities, referred to as the *martingale properties*, hold.

$$\begin{aligned} \mathrm{E}_{T' \vdash T}[X'] &= X \\ \mathrm{E}_{T' \vdash T}[A'] &= A, \text{ and} \\ \mathrm{E}_{T' \vdash T}[B'] &= B. \end{aligned}$$

The augmented protocol has the property that the joint distribution of the views of Alice and Bob is *close* to the product of its marginal distributions. In the sequel, for the simplicity of presentation, we shall assume that the joint distribution of the Alice and Bob views is *identical* to the product of its marginal distributions. This simplifying assumption essentially preserves the qualitative nature of our results. The actual analysis incurs a small slack in the lower bound that is linear in the "closeness parameter." Using our simplifying assumption, we have the following identity in augmented fair coin-tossing protocols.

$$\mathrm{E}_{T' \vdash T}[A' \cdot B'] = \mathrm{E}_{T' \vdash T}[A'] \cdot \mathrm{E}_{T' \vdash T}[B'].$$

Now, we can state and prove a Jensen's inequality for our potential function that holds only in augmented fair coin-tossing protocols.

Proposition 1 (Jensen's Inequality for Augmented Protocols)**.** *The following inequality holds in any augmented fair coin-tossing protocol.*

$$E_{T' \vdash T}[\Phi(X', A', B')] \geq \Phi(X, A, B).$$

Proof. Consider the following manipulation.

$$\begin{aligned}
\mathrm{E}_{T' \vdash T}[\Phi(X', A', B')] &= \mathrm{E}_{T' \vdash T}[X'(1-X') + (X'-A')^2 + (X'-B')^2] && \text{(expanding)} \\
&= \mathrm{E}_{T' \vdash T}[X' + (X'-A'-B')^2 - 2A'B'] && \text{(rearranging)} \\
&= \mathrm{E}_{T' \vdash T}[X'] + \mathrm{E}_{T' \vdash T}[(X'-A'-B')^2] - 2\mathrm{E}_{T' \vdash T}[A'B'] \\
& && \text{(linearity of expectation)} \\
&= X + \mathrm{E}_{T' \vdash T}[(X'-A'-B')^2] - 2\mathrm{E}_{T' \vdash T}[A'B'] && \text{(martingale property)} \\
&\geq X + \mathrm{E}_{T' \vdash T}[(X'-A'-B')]^2 - 2\mathrm{E}_{T' \vdash T}[A'B'] \\
& && \text{(Jensen's inequality on the function } Z^2\text{)} \\
&= X + (X-A-B)^2 - 2\mathrm{E}_{T' \vdash T}[A'B'] && \text{(martingale property)} \\
&= X + (X-A-B)^2 - 2\mathrm{E}_{T' \vdash T}[A'] \cdot \mathrm{E}_{T' \vdash T}[B'] \\
& && \text{(augmented protocol property)} \\
&= X + (X-A-B)^2 - 2AB && \text{(martingale property)} \\
&= \Phi(X, A, B). && \text{(rearranging)}
\end{aligned}$$

This observation completes the proof of the claim. □

A Technical Lemma

For our proof, we shall need a technical lemma.

Lemma 1 (Technical Lemma: Quality of Choice)**.** *There exists a universal positive constant C, such that the following identity holds.*

$$\max\left\{ \left|X'-A'\right| + \left|X'-B'\right|, \frac{4C}{\sqrt{n-1}} \cdot X'(1-X') \right\} \geq \frac{4C}{\sqrt{n}} \cdot \Phi(X', A', B').$$

The interested reader may refer to [15] for a proof of this technical result. For $A' = B' = X$, this result is an algebraization of the geometric transformation in Figure 6 while introducing some slack in the constants.

Putting things together: The Inductive Argument

Proof of Theorem 3. We present the proof of the inductive step of the result. Let $T = \emptyset$, the empty transcript, and T' be the first message of the augmented coin-tossing protocol. Below, we use Π' to represent the coin-tossing protocol Π conditioned on the partial transcript, which is T'.

$$\begin{aligned}
\mathsf{opt}(\Pi) &= \mathrm{E}_{\Pi' \vdash \Pi}[\max\{\left|X'-A'\right]| + \left|X'-B'\right|, \mathsf{opt}(\Pi')\} \\
& && \text{(definition of the optimal adversarial strategy)} \\
&\geq \mathrm{E}_{\Pi' \vdash \Pi}[\max\left\{ \left|X'-A'\right]| + \left|X'-B'\right|, \frac{4C}{\sqrt{n-1}} \cdot X'(1-X') \right\} \\
& && \text{(inductive hypothesis)} \\
&\geq \mathrm{E}_{\Pi' \vdash \Pi}[\frac{4C]}{\sqrt{n}} \cdot \Phi(X', A', B') && \text{(technical lemma)} \\
&\geq \frac{4C}{\sqrt{n}} \cdot \Phi(X, A, B) && \text{(Jensen's inequality for our potential function)} \\
&= \frac{4C}{\sqrt{n}} \cdot \left(X(1-X) + (X-A)^2 + (X-B)^2 \right) \\
& && \text{(definition of the potential function)} \\
&\geq \frac{4C}{\sqrt{n}} \cdot X(1-X) && \text{(non-negativity of } (X-A)^2 \text{ and } (X-B)^2\text{)}
\end{aligned}$$

This derivation concludes the proof of our main technical result. □

5. Isoperimetric Inequalities

This section considers n-processors one-turn one-round coin-tossing protocols. The strong [33] Byzantine adversary can see all messages of the processors and then decide to corrupt k processors of its choice. Let $\varepsilon^+(\pi)$ represent the maximum insecurity caused by a Byzantine adversary who increases the expected output of the protocol. Similarly, $\varepsilon^-(\pi)$ represents the maximum insecurity caused by a Byzantine adversary decreasing the expected output.

The connection to isoperimetric inequalities [3–5,82] (via the expansion of fixed density subset of product spaces) establishes the relevance to topics in theoretical computer science, such as expander graphs, complexity theory, and error-correcting codes. Every coin-tossing protocol is equivalent to a unique subset S of an n-dimension product space Σ^n, where the size of the alphabet set $\sigma := |\Sigma|$ depends on the randomness complexity of the coin-tossing protocol. In the full information model, without the loss of generality, one can assume that all interactive protocols are stateless, and processors use a fresh block of private randomness to generate the next message at any point during the evolution of the coin-tossing protocol [83–85]. Furthermore, each processor can directly broadcast its local private randomness as her message because the final output is a deterministic function of the public transcript. Elements of this product space represent the complete transcript of the coin-tossing protocol, the i-th coordinate of an element corresponds to the message sent by processor i, and the subset S contains all elements of the product space on which the coin-tossing protocol outputs 1. One considers the uniform distribution over Σ^n to sample the elements.

The discussion in the sequel extends to the arbitrary corruption threshold k. However, for the simplicity of the presentation, we consider the specific case of $k = 1$. Let ∂S_k^+ be the set of elements in $\overline{S}$ (the complement of S) that are at a Hamming distance $k = 1$ from the set S. Consequently, a Byzantine adversary can change an element from the set $\partial S_k^+ \subseteq \overline{S}$ into some element of S by editing (at most) $k = 1$ coordinates. Note that, if the Byzantine adversary can see *all* the messages and then performs the edits, then it can increase the expected output by exactly $\varepsilon^+ = \left|\partial S_k^+\right|/\sigma^n$.

Analogously, one defines the set $\partial S_k^- \subseteq S$ that contains all elements at a Hamming distance $k = 1$ from the set $\overline{S}$. So, a Byzantine adversary who sees all the messages before editing can reduce the expected output by $\varepsilon^- = \left|\partial S_k^-\right|/\sigma^n$.

Traditional isoperimetric inequalities over product spaces consider either the edge or vertex perimeter of the set S. The vertex perimeter of a set is most relevant to our current discussion. In this extremal graph-theoretic terminology, the (width-k) *vertex perimeter* of the set S, represented by $\partial_{V,k}S$ is the set of all elements in $\overline{S}$ that are at a Hamming distance of at most k from some element in S. Therefore, the perimeter $\partial_{V,k}S$ is identical to the set ∂S_k^+. Similarly, the vertex perimeter of the set $\overline{S}$ (which is $\partial_{V,k}\overline{S}$) is identical to the set ∂S_k^-.

Extremal graph theory studies the vertex perimeter of a dense set S; the density of the set S is X if this set corresponds to a bias-X coin-tossing protocol. The density of the set $\overline{S}$, therefore, is $(1 - X)$. The objective of extremal graph theory is to characterize the optimal set S of a fixed density that minimizes its vertex perimeter. That is, equivalently, the objective is to design a coin-tossing protocol (identified by S) that minimizes ε^+. Note that minimizing ε^+ does *not* automatically entail the simultaneous minimization of ε^- for general Σ.

For small alphabet-size, for example, when $\sigma = 2$, the choice of S that minimizes its perimeter is explicitly known (namely, the set of appropriate density that greedily includes the smallest elements in the simplistic ordering). For this value of σ, *it happens to be the case* that the complementary set of S simultaneously minimizes ε^-. For large alphabets, however, there are explicit counterexamples. Consider the large alphabet set $\Sigma = \{0, 1, \ldots, \sigma - 1\}$, for a suitable choice of σ. Each party broadcast a random element from the set. The AND protocol checks if all messages are non-zero. This protocol realizes $(\varepsilon^+, \varepsilon^-) = (\frac{1-X}{n}, X)$. The

complementary protocol realizes $(\varepsilon^+, \varepsilon^-) = (1 - X, \frac{X}{n})$. Therefore, there are coin-tossing protocols achieving $(\varepsilon^+, \varepsilon^-) = (\frac{1-X}{n}, X)$ and $(\varepsilon^+, \varepsilon^-) = (1 - X, \frac{X}{n})$), demonstrating that minimizing the perimeter of S does not automatically minimize the perimeter of $\overline{S}$.

The *new cryptographic objective* is to minimize $\varepsilon = \max\{\varepsilon^+, \varepsilon^-\}$, i.e., simultaneously minimize the maximum of the vertex perimeters of S and $\overline{S}$. One may choose the *proxy objective* of minimizing the sum of the perimeters of S and $\overline{S}$, that is, the quantity $\varepsilon^+ + \varepsilon^-$. This proxy objective is a two-approximation of the new cryptographic objective.

In particular, this observation motivates studying new isoperimetric inequalities in extremal graph theory that are inspired by natural practical applications. Instead of minimizing the vertex perimeter of a set S of fixed density, one considers the new objective of minimizing the *symmetric perimeter* defined under various norms.

$$\partial^{\mathsf{sym}}_{V,k,\ell}(S) := \left(\left|\partial_{V,k} S\right|^{\ell} + \left|\partial_{V,k} \overline{S}\right|^{\ell}\right)^{1/\ell}.$$

The $\ell = \infty$ case corresponds to our new cryptographic objective, and the $\ell = 1$ corresponds to the case above. The proposed research provides evidence that such symmetric perimeters may be more well-behaved in general. For instance, when $k = 1$ and $\ell = 1$, Khorasgani, Maji, and Wang [29] demonstrate that the density of the symmetric perimeter is $1/\sqrt{n}$ for any dense set S, even in product spaces over large alphabets.

6. Conclusions

There are several fascinating open problems in coin-tossing protocols pertaining to the topics covered in this survey. This section highlights two such problems below.

Consider a coin-tossing protocol where processors have multiple turns. In a t-turn coin-tossing protocol, every processor sends messages in t rounds. Consequently, once the adversary corrupts a particular processor, it can adversarially set all its future messages. In this setting, what is the optimal bias-X coin-tossing protocol? For two processors, we know that one of the processors can force an output $0/1$ with certainty (based on games against nature [68]). For a larger number of processors, characterizing the optimal coin-tossing protocols remains open.

For the relativized separation results, [30] proved that if oblivious transfer is impossible in the f-hybrid model, then any r-round coin-tossing protocol in the f-hybrid model is (roughly) $\frac{C}{\sqrt{r}}$-unfair. Fix one such function f that does not allow for the oblivious transfer for the discussion below. Now, consider the hardness of computation assumption that "there exists a secure protocol for f." Observe that this hardness of computation assumption is as powerful as the f-hybrid model. However, it may implicitly provide additional restrictions on the computation power of the parties and the adversaries. Consider the following analogy from complexity theory to highlight this subtle difference. In the first scenario, consider efficient algorithms with access to the NP-oracle. In the second scenario, consider the assumption that $\mathsf{NP} = \mathsf{P}$. The second scenario is at least as powerful as the first scenario. However, the second scenario has several additional implicit consequences. In particular, the entire PH collapses, and we have $\mathsf{P} = \mathsf{PH}$. Consequently, is it possible that, through the hardness of computation assumption "there exists a secure protocol for f," which is stronger than the f-hybrid model and enables coin-tossing with lower unfairness?

Funding: The NSF CISE Research Initiation Initiative (CRII) Award (CNS–1566499), an NSF SMALL Award (CNS–1618822), the IARPA HECTOR project, MITRE Innovation Program Academic Cybersecurity Research Awards (2019–2020, 2020–2021), a Purdue Research Foundation (PRF) Award, and The Center for Science of Information, an NSF Science and Technology Center, Cooperative Agreement CCF–0939370 support his research.

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: Data sharing not applicable.

Acknowledgments: The survey summarizes results presented in [14–16,29,30,38]. The author would like to thank all his co-authors Hamidreza Amini Khorasgani, Himanshi Mehta, Tamalika Mukherjee, and Mingyuan Wang in these works.

Conflicts of Interest: The author declares no conflict of interest.

References

1. Ben-Or, M.; Linial, N. Collective Coin Flipping, Robust Voting Schemes and Minima of Banzhaf Values. In Proceedings of the 26th Annual Symposium on Foundations of Computer Science, Portland, OR, USA, 21–23 October 1985; pp. 408–416. [CrossRef]
2. Ben-Or, M.; Linial, N. Collective Coin Flipping. *Adv. Comput. Res.* **1989**, *5*, 91–115.
3. Kruskal, J.B. The number of simplices in a complex. *Math. Optim. Tech.* **1963**, *10*, 251–278.
4. Katona, G. *A Theorem for Finite Sets, Theory of Graphs*; Erdös, P., Katona, G., Eds.; Academic Press: Cambridge, MA, USA, 1968.
5. Harper, L.H. Optimal numberings and isoperimetric problems on graphs. *J. Comb. Theory* **1966**, *1*, 385–393. [CrossRef]
6. Santha, M.; Vazirani, U.V. Generating Quasi-Random Sequences from Slightly-Random Sources (Extended Abstract). In Proceedings of the 25th Annual Symposium on Foundations of Computer Science, Singer Island, FL, USA, 24–26 October1984; pp. 434–440. [CrossRef]
7. Chor, B.; Goldreich, O.; Håstad, J.; Friedman, J.; Rudich, S.; Smolensky, R. The Bit Extraction Problem of t-Resilient Functions (Preliminary Version). In Proceedings of the 26th Annual Symposium on Foundations of Computer Science, Portland, OR, USA, 21–23 October 1985; pp. 396–407. [CrossRef]
8. Vazirani, U.V. Towards a Strong Communication Complexity Theory or Generating Quasi-Random Sequences from Two Communicating Slightly-random Sources (Extended Abstract). In Proceedings of the 17th Annual ACM Symposium on Theory of Computing, Providence, RI, USA, 6–8 May 1985; ACM Press: Providence, RI, USA, 1985; pp. 366–378. [CrossRef]
9. Friedman, J. On the Bit Extraction Problem. In Proceedings of the 33rd Annual Symposium on Foundations of Computer Science, Pittsburgh, PA, USA, 24–27 October 1992; pp. 314–319. [CrossRef]
10. Cleve, R.; Impagliazzo, R. Martingales, collective coin flipping and discrete control processes. *Other Words* **1993**, *1*, 5.
11. Dachman-Soled, D.; Lindell, Y.; Mahmoody, M.; Malkin, T. On the Black-Box Complexity of Optimally-Fair Coin Tossing. In Proceedings of the 8th Theory of Cryptography Conference, TCC 2011, Providence, RI, USA, 28–30 March 2011; Volume 6597, pp. 450–467. [CrossRef]
12. Dachman-Soled, D.; Mahmoody, M.; Malkin, T. Can Optimally-Fair Coin Tossing Be Based on One-Way Functions? In Proceedings of the 11th Theory of Cryptography Conference, TCC 2014, San Diego, CA, USA, 24–26 February 2014; Volume 8349, pp. 217–239. [CrossRef]
13. Haitner, I.; Omri, E.; Zarosim, H. Limits on the Usefulness of Random Oracles. *J. Cryptol.* **2016**, *29*, 283–335. [CrossRef]
14. Khorasgani, H.A.; Maji, H.K.; Mukherjee, T. Estimating Gaps in Martingales and Applications to Coin-Tossing: Constructions and Hardness. In Proceedings of the 17th Theory of Cryptography Conference, Part II, Nuremberg, Germany, 1–5 December 2019; Volume 11892, pp. 333–355. [CrossRef]
15. Khorasgani, H.A.; Maji, H.K.; Wang, M. Coin Tossing with Lazy Defense: Hardness of Computation Results. *IACR Cryptol. ePrint Arch.* **2020**, *2020*, 131.
16. Maji, H.K.; Wang, M. Black-Box Use of One-Way Functions is Useless for Optimal Fair Coin-Tossing. In *Advances in Cryptology—CRYPTO 2020, Part II*; Micciancio, D., Ristenpart, T., Eds.; Lecture Notes in Computer Science; Springer: Heidelberg, Germany; Santa Barbara, CA, USA, 2020; Volume 12171, pp. 593–617. [CrossRef]
17. Banzhaf, J.F., III. Weighted voting doesn't work: A mathematical analysis. *Rutgers L. Rev.* **1964**, *19*, 317.
18. Coleman, J.S. Control of collectivities and the power of a collectivity to act. In *Social Choice*; Liebermann, B., Ed.; Springer: Berlin, Germany, 1971; pp. 269–300.
19. Winder, R.O. Chow Parameters in Threshold Logic. *J. ACM* **1971**, *18*, 265–289. [CrossRef]
20. O'Donnell, R.; Servedio, R.A. The Chow parameters problem. In Proceedings of the 40th Annual ACM Symposium on Theory of Computing, Victoria, BC, Canada, 17–20 May 2008; Ladner, R.E., Dwork, C., Eds.; pp. 517–526. [CrossRef]
21. O'Donnell, R.; Servedio, R.A. The Chow Parameters Problem. *SIAM J. Comput.* **2011**, *40*, 165–199. [CrossRef]
22. Aspnes, J. Lower Bounds for Distributed Coin-Flipping and Randomized Consensus. In Proceedings of the 29th Annual ACM Symposium on Theory of Computing, El Paso, TX, USA, 4–6 May 1997; pp. 559–568. [CrossRef]
23. Aspnes, J. Lower Bounds for Distributed Coin-Flipping and Randomized Consensus. *J. ACM* **1998**, *45*, 415–450. doi10.1145/278298.278304. [CrossRef]
24. Bar-Joseph, Z.; Ben-Or, M. A Tight Lower Bound for Randomized Synchronous Consensus. In *ACM Symposium Annual on Principles of Distributed Computing*; Coan, B.A., Afek, Y., Eds.; Association for Computing Machinery: Puerto Vallarta, Mexico, 1998; pp. 193–199. [CrossRef]

25. Diochnos, D.I.; Mahloujifar, S.; Mahmoody, M. Adversarial Risk and Robustness: General Definitions and Implications for the Uniform Distribution. In Proceedings of the Advances in Neural Information Processing Systems 31: Annual Conference on Neural Information Processing Systems 2018, NeurIPS 2018, Montreal, QC, Canada, 3–8 December 2018; pp. 10380–10389.
26. Mahloujifar, S.; Diochnos, D.I.; Mahmoody, M. The Curse of Concentration in Robust Learning: Evasion and Poisoning Attacks from Concentration of Measure. In Proceedings of the Thirty-Third AAAI Conference on Artificial Intelligence, AAAI 2019, The Thirty-First Innovative Applications of Artificial Intelligence Conference, IAAI 2019, The Ninth AAAI Symposium on Educational Advances in Artificial Intelligence, EAAI 2019, Honolulu, HI, USA, 27 January–1 February 2019; pp. 4536–4543. [CrossRef]
27. Mahloujifar, S.; Mahmoody, M. Can Adversarially Robust Learning LeverageComputational Hardness? In Proceedings of the Algorithmic Learning Theory, ALT 2019, Chicago, IL, USA, 22–24 March 2019; Volume 98, pp. 581–609.
28. Etesami, O.; Mahloujifar, S.; Mahmoody, M. Computational Concentration of Measure: Optimal Bounds, Reductions, and More. In Proceedings of the 31st Annual ACM-SIAM Symposium on Discrete Algorithms, Salt Lake City, UT, USA, 5–8 January 2020; pp. 345–363. [CrossRef]
29. Khorasgani, H.A.; Maji, H.K.; Wang, M. Design & Analysis of Optimal Coin-tossing: New Techniques. *IACR Cryptol. ePrint Arch.* **2020**, *2020*, 519.
30. Maji, H.K.; Wang, M. Computational Hardness of Optimal Fair Computation: Beyond Minicrypt. Unpublished work. 2020. Available Online: https://www.cs.purdue.edu/homes/hmaji/papers/MajWan20a.pdf (accessed on 15 December 2020)
31. Impagliazzo, R.; Rudich, S. Limits on the Provable Consequences of One-Way Permutations. In Proceedings of the 21st Annual ACM Symposium on Theory of Computing, Seattle, WA, USA, 14–17 May 1989; pp. 44–61. [CrossRef]
32. Reingold, O.; Trevisan, L.; Vadhan, S.P. Notions of Reducibility between Cryptographic Primitives. In Proceedings of the TCC 2004: 1st Theory of Cryptography Conference, Cambridge, MA, USA, 19–21 February 2004; Volume 2951, pp. 1–20. [CrossRef]
33. Goldwasser, S.; Kalai, Y.T.; Park, S. Adaptively Secure Coin-Flipping, Revisited. In Proceedings of the ICALP 2015: 42nd International Colloquium on Automata, Languages and Programming, Part II, Kyoto, Japan, 6–10 July 2015; Volume 9135, pp. 663–674.. [CrossRef]
34. Blum, M. Coin Flipping by Telephone - A Protocol for Solving Impossible Problems. In Proceedings of the COMPCON'82, Digest of Papers, Twenty-Fourth IEEE Computer Society International Conference, San Francisco, CA, USA, 22–25 February 1982; pp. 133–137.
35. Broder, A.Z.; Dolev, D. Flipping coins in many pockets (Byzantine agreement on uniformly random values). In Proceedings of the 25th Annual Symposium on Foundations of Computer Science, Singer Island, FL, USA, 24–26 October 1984; pp. 157–170. [CrossRef]
36. Awerbuch, B.; Blum, M.; Chor, B.; Goldwasser, S.; Micali, S. How to implement Bracha's O (log n) byzantine agreement algorithm. **1985**, Unpublished.
37. Cleve, R. Limits on the Security of Coin Flips when Half the Processors Are Faulty (Extended Abstract). In Proceedings of the 18th Annual ACM Symposium on Theory of Computing, Berkeley, CA, USA, 28–30 May 1986; pp. 364–369. [CrossRef]
38. Maji, H.K.; Mehta, H.; Wang, M. *On Efficient Distributed Coin-tossing Protocols*; Purdue University: West Lafayette, IN, USA, 2020; Unpublished.
39. Azuma, K. Weighted sums of certain dependent random variables. *Tohoku Math. J.* **1967**, *19*, 357–367. [CrossRef]
40. Hoeffding, W. Probability Inequalities for Sums of Bounded Random Variables. *J. Am. Stat. Assoc.* **1963**, *58*, 13–30. [CrossRef]
41. Beimel, A.; Haitner, I.; Makriyannis, N.; Omri, E. Tighter Bounds on Multi-Party Coin Flipping via Augmented Weak Martingales and Differentially Private Sampling. In Proceedings of the 59th Annual Symposium on Foundations of Computer Science, Paris, France, 7–9 October 2018; pp. 838–849. [CrossRef]
42. Lichtenstein, D.; Linial, N.; Saks, M. Some extremal problems arising from discrete control processes. *Combinatorica* **1989**, *9*, 269–287. [CrossRef]
43. Kalai, Y.T.; Komargodski, I.; Raz, R. A Lower Bound for Adaptively-Secure Collective Coin-Flipping Protocols. In Proceedings of the 32nd International Symposium on Distributed Computing, DISC 2018, New Orleans, LA, USA, 15–19 October 2018; Volume 121, pp. 34:1–34:16.
44. Haitner, I.; Karidi-Heller, Y. A Tight Lower Bound on Adaptively Secure Full-Information Coin Flip. *arXiv* **2020**, arXiv:2005.01565.
45. Cleve, R. Controlled Gradual Disclosure Schemes for Random Bits and Their Applications. In Proceedings of the Advances in Cryptology—CRYPTO'89, Santa Barbara, CA, USA, 11–15 August 1990; Volume 435, pp. 573–588. [CrossRef]
46. Impagliazzo, R. A Personal View of Average-Case Complexity. In Proceedings of the Tenth Annual Structure in Complexity Theory Conference, Minneapolis, MI, USA, 19–22 June 1995; pp.134–147. [CrossRef]
47. Impagliazzo, R.; Luby, M. One-way Functions are Essential for Complexity Based Cryptography (Extended Abstract). In Proceedings of the 30th Annual Symposium on Foundations of Computer Science, Research Triangle Park, NC, USA, 30 October–1 November 1989; pp. 230–235. [CrossRef]
48. Impagliazzo, R.; Levin, L.A.; Luby, M. Pseudo-random Generation from one-way functions (Extended Abstracts). In *21st Annual ACM Symposium on Theory of Computing*; ACM Press: Seattle, WA, USA, 1989; pp. 12–24. [CrossRef]
49. Håstad, J. Pseudo-Random Generators under Uniform Assumptions. In *22nd Annual ACM Symposium on Theory of Computing*; ACM Press: Baltimore, MD, USA, 1990; pp. 395–404. [CrossRef]

50. Håstad, J.; Impagliazzo, R.; Levin, L.A.; Luby, M. A Pseudorandom Generator from any One-way Function. *SIAM J. Comput.* **1999**, *28*, 1364–1396. [CrossRef]
51. Goldreich, O.; Goldwasser, S.; Micali, S. How to Construct Random Functions (Extended Abstract). In Proceedings of the COMPCON'82, Digest of Papers, Twenty-Fourth IEEE Computer Society International Conference, San Francisco, CA, USA, 22–25 February 1982; pp. 464–479. [CrossRef]
52. Goldreich, O.; Goldwasser, S.; Micali, S. How to Construct Random Functions. *J. ACM* **1986**, *33*, 792–807. [CrossRef]
53. Luby, M.; Rackoff, C. How to construct pseudorandom permutations from pseudorandom functions. *SIAM J. Comput.* **1988**, *17*, 373–386. [CrossRef]
54. Naor, M. Bit Commitment Using Pseudorandomness. *J. Cryptol.* **1991**, *4*, 151–158. [CrossRef]
55. Naor, M.; Ostrovsky, R.; Venkatesan, R.; Yung, M. Perfect Zero-Knowledge Arguments for *NP* Using Any One-Way Permutation. *J. Cryptol.* **1998**, *11*, 87–108. [CrossRef]
56. Haitner, I.; Reingold, O. Statistically-hiding commitment from any one-way function. In *39th Annual ACM Symposium on Theory of Computing*; Johnson, D.S., Feige, U., Eds.; ACM Press: San Diego, CA, USA, 2007; pp. 1–10. [CrossRef]
57. Goldreich, O.; Micali, S.; Wigderson, A. Proofs That Yield Nothing But Their Validity Or All Languages in NP Have Zero-Knowledge Proof Systems. *J. ACM* **1991**, *38*, 691–729. [CrossRef]
58. Naor, M.; Yung, M. Universal One-Way Hash Functions and their Cryptographic Applications. In *21st Annual ACM Symposium on Theory of Computing*; ACM Press: Seattle, WA, USA, 1989; pp. 33–43. [CrossRef]
59. Rompel, J. One-Way Functions are Necessary and Sufficient for Secure Signatures. In *22nd Annual ACM Symposium on Theory of Computing*; ACM Press: Baltimore, MD, USA, 1990; pp. 387–394. [CrossRef]
60. Gertner, Y.; Kannan, S.; Malkin, T.; Reingold, O.; Viswanathan, M. The Relationship between Public Key Encryption and Oblivious Transfer. In Proceedings of the 41st Annual Symposium on Foundations of Computer Science, Redondo Beach, CA, USA, 12–14 November 2000; pp. 325–335. [CrossRef]
61. Mahmoody, M.; Maji, H.K.; Prabhakaran, M. Limits of random oracles in secure computation. In *ITCS 2014: 5th Conference on Innovations in Theoretical Computer Science*; Naor, M., Ed.; Association for Computing Machinery: Princeton, NJ, USA, 2014; pp. 23–34. [CrossRef]
62. Mahmoody, M.; Maji, H.K.; Prabhakaran, M. On the Power of Public-Key Encryption in Secure Computation. In Proceedings of the TCC 2014: 11th Theory of Cryptography Conference, San Diego, CA, USA, 24–26 February 2014; Volume 8349, pp. 240–264. [CrossRef]
63. Even, S.; Goldreich, O.; Lempel, A. A Randomized Protocol for Signing Contracts. In *Advances in Cryptology—CRYPTO'82*; Chaum, D., Rivest, R.L., Sherman, A.T., Eds.; Plenum Press: New York, NY, USA; Santa Barbara, CA, USA, 1982; pp. 205–210.
64. Even, S.; Goldreich, O.; Lempel, A. A randomized protocol for signing contracts. *Commun. ACM* **1985**, *28*, 637–647. [CrossRef]
65. Haitner, I.; Omri, E. Coin Flipping with Constant Bias Implies One-Way Functions. In Proceedings of the 52nd Annual Symposium on Foundations of Computer Science, Palm Springs, CA, USA, 22–25 October 2011; pp. 110–119. [CrossRef]
66. Berman, I.; Haitner, I.; Tentes, A. Coin flipping of *any* constant bias implies one-way functions. In *46th Annual ACM Symposium on Theory of Computing*; Shmoys, D.B., Ed.; ACM Press: New York, NY, USA, 2014; pp. 398–407. [CrossRef]
67. Moran, T.; Naor, M.; Segev, G. An Optimally Fair Coin Toss. In Proceedings of the TCC 2009: 6th Theory of Cryptography Conference, San Francisco, CA, USA, 15–17 March 2009; Volume 5444, pp. 1–18. [CrossRef]
68. Papadimitriou, C.H. Games Against Nature (Extended Abstract). In Proceedings of the 24th Annual Symposium on Foundations of Computer Science, Tucson, AZ, USA, 7–9 November 1983; pp. 446–450. [CrossRef]
69. Maji, H.K.; Prabhakaran, M.; Sahai, A. On the Computational Complexity of Coin Flipping. In Proceedings of the 51st Annual Symposium on Foundations of Computer Science, Las Vegas, NV, USA, 23–26 October 2010; pp. 613–622. [CrossRef]
70. Baecher, P.; Brzuska, C.; Fischlin, M. Notions of Black-Box Reductions, Revisited. In Proceedings of the Advances in Cryptology—ASIACRYPT 2013, Bengaluru, India, 1–5 December 2013; Volume 8269, pp. 296–315. [CrossRef]
71. Cook, S.A. The Complexity of Theorem-Proving Procedures. In Proceedings of the 3rd Annual ACM Symposium on Theory of Computing, Shaker Heights, OH, USA, 3–5 May 1971; pp. 151–158. [CrossRef]
72. Karp, R.M. Reducibility Among Combinatorial Problems. In *Complexity of Computer Computations*; Miller, R.E., Thatcher, J.W., Eds.; The IBM Research Symposia Series; Plenum Press: New York, NY, USA, 1972; pp. 85–103. [CrossRef]
73. Yao, A.C.C. How to Generate and Exchange Secrets (Extended Abstract). In Proceedings of the 27th Annual Symposium on Foundations of Computer Science, Toronto, ON, Canada, 27–29 October 1986; pp. 162–167. [CrossRef]
74. Goldreich, O.; Micali, S.; Wigderson, A. How to Play any Mental Game or A Completeness Theorem for Protocols with Honest Majority. In Proceedings of the 19th Annual ACM Symposium on Theory of Computing, New York, NY, USA, 25–27 May 1987; pp. 218–229. [CrossRef]
75. Feige, U.; Shamir, A. Witness Indistinguishable and Witness Hiding Protocols. In Proceedings of the 22nd Annual ACM Symposium on Theory of Computing, Baltimore, MD, USA, 13–17 May 1990; pp. 416–426. [CrossRef]
76. Dolev, D.; Dwork, C.; Naor, M. Nonmalleable Cryptography. *SIAM J. Comput.* **2000**, *30*, 391–437. [CrossRef]
77. Barak, B. Constant-Round Coin-Tossing with a Man in the Middle or Realizing the Shared Random String Model. In Proceedings of the 43rd Annual Symposium on Foundations of Computer Science, Vancouver, BC, Canada, 19 November 2002; pp. 345–355. [CrossRef]

78. Haitner, I.; Omri, E.; Zarosim, H. Limits on the Usefulness of Random Oracles. In Proceedings of the TCC 2013: 10th Theory of Cryptography Conference, Tokyo, Japan, 3–6 March 2013; Volume 7785, pp. 437–456. [CrossRef]
79. Haitner, I.; Makriyannis, N.; Omri, E. On the Complexity of Fair Coin Flipping. In Proceedings of the TCC 2018: 16th Theory of Cryptography Conference, Part I, Panaji, India, 11–14 November 2018; Volume 11239, pp. 539–562. [CrossRef]
80. Kilian, J. More general completeness theorems for secure two-party computation. In Proceedings of the 32nd Annual ACM Symposium on Theory of Computing, Portland, OR, USA, 21–23 May 2000; pp. 316–324. [CrossRef]
81. Barak, B.; Mahmoody-Ghidary, M. Merkle Puzzles Are Optimal—An $O(n^2)$-Query Attack on Any Key Exchange from a Random Oracle. In Proceedings of the Advances in Cryptology—CRYPTO 2009, Santa Barbara, CA, USA, 16–20 August 2009; Volume 5677, pp. 374–390. [CrossRef]
82. Harper, L.H. On an Isoperimetric Problem for Hamming Graphs. *Discret. Appl. Math.* **1999**, *95*, 285–309. [CrossRef]
83. Jerrum, M. Random Generation of Combinatorial Structures from a Uniform Distribution (Extended Abstract). In Proceedings of the Automata, Languages and Programming, 12th Colloquium, Nafplion, Greece, 15–19 July 1985; Volume 194, pp. 290–299. [CrossRef]
84. Jerrum, M.; Valiant, L.G.; Vazirani, V.V. Random Generation of Combinatorial Structures from a Uniform Distribution. *Theor. Comput. Sci.* **1986**, *43*, 169–188. [CrossRef]
85. Bellare, M.; Goldreich, O.; Petrank, E. Uniform Generation of NP-Witnesses Using an NP-Oracle. *Inf. Comput.* **2000**, *163*, 510–526. [CrossRef]