Electronic Colloquium on Computational Complexity, Report No. 84 (2018)

On the Complexity of Fair Coin Flipping

Iftach Haitner*† Nikolaos Makriyannis‡‡ Eran Omri§

April 24, 2018

Abstract

A two-party coin-flipping protocol is ε-fair if no efficient adversary can bias the output of the honest party (who always outputs a bit, even if the other party aborts) by more than ε. Cleve [STOC '86] showed that r-round $o(1/r)$-fair coin-flipping protocols do not exist. Awerbuch et al. [Manuscript '85] constructed a $\Theta(1/\sqrt{r})$-fair coin-flipping protocol, assuming the existence of one-way functions. Moran et al. [Journal of Cryptology '16] constructed an r-round coin-flipping protocol that is $\Theta(1/r)$-fair (thus matching the aforementioned lower bound of Cleve [STOC '86]), assuming the existence of oblivious transfer.

The above gives rise to the intriguing question of whether oblivious transfer, or more generally "public-key primitives", is required for an $o(1/\sqrt{r})$-fair coin flipping. This question was partially answered by Dachman-Soled et al. [TCC '11] and Dachman-Soled et al. [TCC '14], who showed that *restricted* types of fully black-box reductions cannot establish $o(1/\sqrt{r})$-fair coin-flipping protocols from one-way functions.

We make progress towards answering the above question, by showing that, for any (constant) $r \in \mathbb{N}$, the existence of an $1/(c\cdot\sqrt{r})$-fair, r-round coin-flipping protocol implies the existence of an infinitely-often key-agreement protocol, where c denotes some universal constant (independent of r). Our reduction is non black-box and makes a novel use of the recent dichotomy for two-party protocols of Haitner et al. to facilitate a two-party variant of the recent attack of Beimel et al. on multi-party coin-flipping protocols.

*School of Computer Science, Tel Aviv University. E-mail: iftachh@cs.tau.ac.il. Member of the Check Point Institute for Information Security.

†Research supported by ERC starting grant 638121.

‡School of Computer Science, Tel Aviv University. E-mail: n.makriyannis@gmail.com.

§Department of Computer Science, Ariel University. E-mail: omrier@ariel.ac.il. Research supported by ISF grant 152/17.

ISSN 1433-8092

Contents

1 Introduction

In a two-party coin-flipping protocol, introduced by Blum [7], the parties wish to output a common (close to) uniform bit, even though one of the parties may be corrupted and try to bias the output. Slightly more formally, an ε-fair coin-flipping protocol should satisfy the following two properties: first, when both parties behave honestly (i.e., follow the prescribed protocol), they both output the *same* uniform bit. Second, in the presence of a corrupted party that may deviate from the protocol arbitrarily, the distribution of the honest party's output may deviate from the uniform distribution (unbiased bit) by at most ε. We emphasize that the above notion requires an honest party to *always* output a bit, regardless of what the corrupted party does, and, in particular, it is not allowed to abort if a cheat is detected.[1] Coin-flipping is a fundamental primitive with numerous applications, and thus lower bounds on coin-flipping protocols yield analogous bounds for many basic cryptographic primitives, including other inputless primitives and secure computation of functions that take input (e.g., XOR).

In his seminal work, Cleve [9] showed that, for *any* efficient two-party r-round coin-flipping protocol, there exists an efficient adversarial strategy that biases the output of the honest party by $\Theta(1/r)$. The above lower bound on coin-flipping protocols was met for the two-party case by Moran, Naor, and Segev [21] improving over the $\Theta(n/\sqrt{r})$-fairness achieved by the majority protocol of Awerbuch, Blum, Chor, Goldwasser, and Micali [3]. The protocol of [21], however, uses oblivious transfer; to be compared with the protocol of [3] that can be based on any one-way function. An intriguing open question is whether oblivious transfer, or more generally "public-key primitives", is required for an $o(1/\sqrt{r})$-fair coin-flip. The question was partially answered in the black-box setting by Dachman-Soled et al. [11] and Dachman-Soled et al. [12], who showed that *restricted* types of fully black-box reductions cannot establish $o(1/\sqrt{r})$-bias coin-flipping protocols from one-way functions.

1.1 Our Results

Our main result is that constant-round coin-flipping protocols with better bias compared to the majority protocol of [2] imply the existence of infinitely-often key-agreement.

Theorem 1.1 (Main result, informal)**.** *For any (constant) $r \in \mathbb{N}$, the existence of an $1/(c\cdot\sqrt{r})$-fair, r-round coin-flipping protocol implies the existence an infinitely-often key-agreement protocol, for $c > 0$ being a universal constant (independent of r).*

As in [9, 11, 12], our result extends via a simple reduction to general multi-party coin-flipping protocols (with more than two-parties) without an honest majority. Our reduction is non black-box, and makes a novel use of the recent dichotomy for two-party protocols of Haitner et al. [16]. Specifically, assuming that io-key-agreement does not exist and applying Haitner et al.'s dichotomy, we show that a two-party variant of the recent multi-party attack of Beimel et al. [5] yields a $1/(c \cdot \sqrt{r})$-bias attack.

1.2 Our Technique

Let $\pi = (\mathsf{A}, \mathsf{B})$ be a r-round two-party coin-flipping protocol. We show that the nonexistence of key-agreement protocols yields an efficient $\Theta(1/\sqrt{r})$-bias attack on π. We start by describing the

[1] Such protocols are typically addressed as having *guaranteed output delivery*, or, abusing terminology, as *fair*.

$1/\sqrt{r}$-bias *inefficient* attack of Cleve and Impagliazzo [10], and the approach of Beimel et al. [5] towards making this attack efficient. We then explain how to use the recent results by Haitner et al. [16] to obtain an efficient attack (assuming the nonexistence of io-key-agreement protocols).

1.2.1 Cleve and Impagliazzo's Inefficient Attack

We describe the inefficient $1/\sqrt{r}$-bias attack due to Cleve and Impagliazzo [10]. Let $M_1, \ldots, M_r$ denote the messages in a random execution of π, and let out denote the (without loss of generality) always common output of the parties in a random honest execution of π. Let $X_i = \mathbf{E}[\mathsf{out} \mid M_{\leq i}]$. Namely, X_i is the expected outcome of the parties in π given $M_{\leq i} = M_1, \ldots, M_i$. It is easy to see that $X_0, \ldots, X_r$ is a martingale sequence: $\mathbf{E}[X_i \mid X_0, \ldots, X_{i-1}] = X_{i-1}$ for every i. Since the parties in an honest execution of π output a uniform bit, it holds that $X_0 = \Pr[\mathsf{out} = 1] = 1/2$ and $X_r \in \{0, 1\}$. Cleve and Impagliazzo [10] (see Beimel et al. [5] for an alternative simpler proof) prove that, for such a sequence (omitting absolute values and constant factors),

$$\text{Gap:} \qquad \Pr\left[\exists i \in [r] \colon X_i - X_{i-1} \geq 1/\sqrt{r}\right] \geq 1/2 \tag{1}$$

Let the i^{th} *backup value* of party P, denoted Z_i^{P}, be the output of party P if the other party aborts prematurely *after* the i^{th} message was sent (recall that the honest party must always output a bit, by definition). In particular, Z_r^{P} denotes the final output of P (if no abort occurred). We claim that the following holds

$$\text{Backup values approximate outcome:} \quad \Pr\left[\exists i \in [r] \colon \left|X_i - \mathbf{E}\left[Z_i^{\mathsf{P}} \mid M_{\leq i}\right]\right| \geq 1/2\sqrt{r}\right] \leq 1/4 \tag{2}$$

for both $\mathsf{P} \in \{\mathsf{A}, \mathsf{B}\}$. If not, for at least one $z \in \{0, 1\}$, by aborting at the end of round i if $(-1)^{1-z} \cdot (X_i - \mathbf{E}\left[Z_i^{\mathsf{P}} \mid M_{\leq i}\right]) \geq 1/\sqrt{r}$, the (possibly inefficient) adversary controlling $\overline{\mathsf{P}} \in \{\mathsf{A}, \mathsf{B}\} \setminus \mathsf{P}$ biases the output of P towards $1 - z$ by $\Theta(1/\sqrt{r})$, and we are done. Finally, since the coins of the parties are *independent* conditioned on the transcript, if party A sends the $(i+1)$ message then

$$\text{Independence:} \qquad \mathbf{E}\left[Z_i^{\mathsf{B}} \mid M_{\leq i}\right] = \mathbf{E}\left[Z_i^{\mathsf{B}} \mid M_{\leq i+1}\right] \tag{3}$$

Combining the above observations yields that without loss of generality:

$$\Pr\left[\exists i \in [r] \colon \mathsf{A} \text{ sends the } i^{\text{th}} \text{ message} \wedge X_i - \mathbf{E}\left[Z_{i-1}^{\mathsf{B}} \mid M_{\leq i}\right] \geq 1/2\sqrt{r}\right] \geq 1/8 \tag{4}$$

Equation (4) yields the following (possibly inefficient) attack for a corrupted party A biasing B's output towards zero: before sending the i^{th} message M_i, party A aborts if $X_i - \mathbf{E}\left[Z_{i-1}^{\mathsf{B}} \mid M_{\leq i}\right] \geq 1/2\sqrt{r}$. By Equation (4), this attack biases B's output towards zero by $\Omega(1/2\sqrt{r})$.

The clear limitation of the above attack is that, assuming one-way functions exist, the value of $X_i = \mathbf{E}[\mathsf{out} \mid M_{\leq i} = t]$ and of $\mathbf{E}\left[Z_i^{\mathsf{P}} \mid M_{\leq i} = t\right]$ might *not* be efficiently computable as a function of t.[2] Facing this difficulty, Beimel et al. [5] considered the martingale sequence $X_i = \mathbf{E}\left[\mathsf{out} \mid Z_{\leq i}^{\mathsf{P}}\right]$ (recall that Z_i^{P} is the i^{th} backup value of P). It follows that, for constant-round protocols, the value of X_i is only a function of a constant size string, and thus it is efficiently computable ([5] have facilitated this approach for protocols of super-constant round complexity, see Footnote 3). The price of using the alternative sequence $X_1, \ldots, X_r$ is that the independence property (Equation (3)) might no longer hold. Yet, [5] manage to facilitate the above approach into an efficient $\widetilde{\Omega}(1/\sqrt{r})$-attack on *multi-party* protocols. In the following, we show how to use the dichotomy of Haitner et al. [16] to facilitate a two-party variant of the attack from [5].

[2] For instance, the first two messages might contain commitments to the parties' randomness.

1.2.2 Inexistence of Key-Agreement Implies an Efficient Attack

Let U_p denote the Bernoulli random variable taking the value 1 with probability p, and let $P \stackrel{\mathrm{C}}{\approx}_\rho Q$ stand for Q and P are ρ-computationally indistinguishablity (i.e., an efficient distinguisher cannot tell P from Q with advantage better than ρ). We are using two results by Haitner et al. [16]. The first one given below holds for any two-party protocol.

Theorem 1.2 (Haitner et al. [16]'s forecaster, informal)**.** *Let $\Delta = (\mathsf{A}, \mathsf{B})$ be a single-bit output (each party outputs a bit) two-party protocol. Then, for any constant $\rho > 0$, there exists a constant-output length poly-time algorithm (forecaster) F mapping transcripts of Δ into (the binary description of) pairs in $[0,1] \times [0,1]$ such that the following holds: let (X, Y, T) be the parties outputs and transcript in a random execution of Δ, then*

- $(X, T) \stackrel{\mathrm{C}}{\approx}_\rho (U_{p^\mathsf{A}}, T)_{(p^\mathsf{A}, \cdot) \leftarrow \mathsf{F}(T)}$*, and*
- $(Y, T) \stackrel{\mathrm{C}}{\approx}_\rho (U_{p^\mathsf{B}}, T)_{(\cdot, p^\mathsf{B}) \leftarrow \mathsf{F}(T)}$.

Namely, given the transcript, F forecasts the output-distribution for each party in a way that is computationally indistinguishable from the real value.

Consider the $(r+1)$-round protocol $\widetilde{\pi} = (\widetilde{\mathsf{A}}, \widetilde{\mathsf{B}})$, defined by $\widetilde{\mathsf{A}}$ sending a random $i \in [r]$ to $\widetilde{\mathsf{B}}$ as the first message and then the parties interact in a random execution of π for the first i rounds. At the end of the execution, the parties output their i^{th} backup values z_i^A and z_i^B and halt. Let F be the forecaster for $\widetilde{\pi}$ guaranteed by Theorem 1.2 for $\rho = 1/r^2$ (note that ρ is indeed constant). A simple averaging argument yields that

$$(Z_i^\mathsf{P}, M_{\le i}) \stackrel{\mathrm{C}}{\approx}_{1/r} (U_{p^\mathsf{P}}, M_{\le i})_{(p^\mathsf{A}, p^\mathsf{B}) \leftarrow \mathsf{F}(M_{\le i})} \tag{5}$$

for both $\mathsf{P} \in \{\mathsf{A}, \mathsf{B}\}$ and every $i \in [r]$, letting $\mathsf{F}(m_{\le i}) = \mathsf{F}(i, m_{\le i})$. Namely, F is a good forecaster for the partial transcripts of π.

Let $M_1, \ldots, M_r$ denote the messages in a random execution of π and let out denote the output of the parties in π. Let $Y_i = \left(Y_i^\mathsf{A}, Y_i^\mathsf{B}\right) = \mathsf{F}(M_{\le i})$ and let $X_i = \mathbf{E}\left[\mathsf{out} \mid Y_{\le i}\right]$. It is easy to see that $X_1, \ldots, X_r$ is a martingale sequence and that $X_0 = 1/2$. We assume without loss of generality that the last message of π contains the common output. Thus, it follows from Equation (5) that $Y_r \approx (\mathsf{out}, \mathsf{out}) \in \{(0,0), (1,1)\}$ (otherwise, it will be very easy to distinguish the emulated outputs from the real ones, given M_r). Hence, similarly to Section 1.2.1, it holds that

$$\text{Gap:} \qquad \Pr\left[\exists i \in [r] \colon X_i - X_{i-1} \ge 1/\sqrt{r}\right] \ge 1/2 \tag{6}$$

Since Y_i has constant-size support and since π is constant round, it follows that X_i is efficiently computable from $M_{\le i}$.[3]

[3] In the spirit of Beimel et al. [5], we could have modified the definition of the X_i's to make them efficiently computable even for non constant-round protocols. The idea is to define $X_i = \mathbf{E}\left[\mathsf{out} \mid Y_i, X_{i-1}\right]$. While the resulting sequence might not be a martingale, [5] proves that a $1/\sqrt{r}$-gap also occurs with constant probability with respect to such a sequence. Unfortunately, we cannot benefit from this improvement, since the results of Haitner et al. [16] only guarantees indistinguishablity for constant ρ, which makes it useful only for attacking constant-round protocols.

Let Z_i^{P} denote the backup value computed by party P in round i of a random execution of π. The indistinguishablity of F yields that $\mathbf{E}\left[Z_i^{\mathsf{P}} \mid Y_{\leq i}\right] \approx Y_i^{\mathsf{P}}$. Similarly to Section 1.2.1, unless there is a simple $1/\sqrt{r}$-attack, it holds that

$$\text{Backup values approximate outcome:} \quad \Pr\left[\exists i \in [r] \colon \left|X_i - \mathbf{E}\left[Z_i^{\mathsf{P}} \mid Y_{\leq i}\right]\right| \geq 1/2\sqrt{r}\right] \leq 1/4 \quad (7)$$

Thus, for an efficient variant of [10]'s attack, we require that

$$\text{Independence:} \qquad \mathbf{E}\left[Z_i^{\mathsf{P}} \mid Y_{\leq i}\right] \stackrel{\mathrm{C}}{\approx}_{1/r} \mathbf{E}\left[Z_i^{\mathsf{P}} \mid Y_{\leq i+1}\right] \quad (8)$$

for every $\mathsf{P} \in \{\mathsf{A}, \mathsf{B}\}$ and round i in which party $\overline{\mathsf{P}} \in \{\mathsf{A}, \mathsf{B}\} \setminus \{\mathsf{P}\}$ sends the $(i+1)$ message. However, unlike Equation (3) in Section 1.2.1, Equation (8) does not hold unconditionally (in fact, assuming oblivious transfer exists, the implied attack must fail for some protocols, yielding that Equation (8) is false for these protocols). Rather, we relate Equation (8) to the existence of a key-agreement protocol. Specifically, we show that if Equation (8) is not true, then there exists a key-agreement protocol.

Proving that Y_{i+1} and Z_i^b are approximately independent given $Y_{\leq i}$, assuming nonexistence of key-agreement. We are now using a second result by Haitner et al. [16].[4]

Theorem 1.3 (Haitner et al. [16]'s dichotomy, informal)**.** *Let $\Delta = (\mathsf{A}, \mathsf{B})$ be an efficient single-bit output two-party protocol and assume infinitely-often key-agreement protocol does not exist. Then, for any constant $\rho > 0$, there exists a poly-time algorithm (decorrelator) Dcr mapping transcripts of Δ into $[0,1] \times [0,1]$ such that the following holds: let (X, Y, T) be the parties' outputs and transcript in a random execution of Δ, then*

$$(X, Y, T) \stackrel{\mathrm{C}}{\approx}_{\rho} (U_{p^{\mathsf{A}}}, U_{p^{\mathsf{B}}}, T)_{(p^{\mathsf{A}}, p^{\mathsf{B}}) \leftarrow \mathsf{Dcr}(T)}.$$

Namely, assuming io-key-agreement does not exist, the distribution of the parties' output given the transcript seems ρ-close to the product distribution given by Dcr. We assume for simplicity that the theorem holds for *many-bit* output protocols and not merely single bit (we get rid of this assumption in the actual proof).

We define another variant $\widehat{\pi}$ of π that internally uses the forecaster F. We prove that assuming the existence of a decorrelator for $\widehat{\pi}$, it holds that X_{i+1} and Z_i^{P} are approximately independent given $Y_{\leq i}$, and Equation (8) follows. For concreteness, we focus on party $\mathsf{P} = \mathsf{B}$.

Fix i such that A sends the $(i+1)$ message in π. Let $\widehat{\pi} = (\widehat{\mathsf{A}}, \widehat{\mathsf{B}})$ be a protocol in which the parties interact just as in π for the first i rounds. Then, $\widehat{\mathsf{B}}$ outputs the i^{th} backup value of B, and $\widehat{\mathsf{A}}$ internally computes t_{i+1}, and outputs $y_{i+1} = \mathsf{F}(t_{i+1})$. Assume key-agreement protocols do not exist, Theorem 1.3 implies the existence of an efficient decorrelator Dcr for $\widehat{\pi}$ with respect to $\rho = 1/r$. By the definition of the decorrelator, it holds that

$$(Y_{i+1}, Z_i^{\mathsf{B}}, M_{\leq i}) \stackrel{\mathrm{C}}{\approx}_{1/r} (U_{p^{\widehat{\mathsf{A}}}}, U_{p^{\widehat{\mathsf{B}}}}, M_{\leq i})_{(p^{\widehat{\mathsf{A}}}, p^{\widehat{\mathsf{B}}}) \leftarrow \mathsf{Dcr}(M_{\leq i})}, \quad (9)$$

[4]Assuming the nonexistence of key-agreement protocols, Theorem 1.3 implies Theorem 1.2. Yet, we chose to use both results to make the text more modular.

where now $p^{\widehat{\mathsf{A}}}$ describes a non-Boolean distribution, and $U_{p^{\widehat{\mathsf{A}}}}$ denotes an independent sample from this distribution. Hence, to prove that Y_{i+1} and Z_i^{B} are approximately independent given $Y_{\leq i}$, it suffices to prove that $U_{p^{\widehat{\mathsf{A}}}}$ and $U_{p^{\widehat{\mathsf{B}}}}$ are approximately independent given $Y_{\leq i}$.

Since F and Dcr both output an estimate of (the expectation of) $Z_i^{\mathsf{B}}|M_{\leq i}$ in a way that is indistinguishable from the real distribution of Z_i^{B} (given $M_{\leq i}$), both algorithms output essentially the same value. Otherwise, at least one of the algorithms is far from the "real" value, and the other algorithm can be used to distinguish the real distribution from the simulated one. It follows that

$$(U_{p^{\widehat{\mathsf{A}}}}, U_{p^{\widehat{\mathsf{B}}}}, M_{\leq i})_{(p^{\widehat{\mathsf{A}}}, p^{\widehat{\mathsf{B}}}) \leftarrow \mathsf{Dcr}(M_{\leq i})} \overset{\text{C}}{\approx}_{1/r} (U_{p^{\widehat{\mathsf{A}}}}, U_{Y_i^{\mathsf{B}}}, M_{\leq i})_{p^{\widehat{\mathsf{A}}} \leftarrow \mathsf{Dcr}(M_{\leq i})^{\widehat{\mathsf{A}}}} \tag{10}$$

Using a data-processing argument in combination with Equations (9) and (10), we deduce that

$$\left(Y_{i+1}, Z_i^{\mathsf{B}}, Y_{\leq i}\right) \overset{\text{C}}{\approx}_{1/r} \left(U_{p^{\widehat{\mathsf{A}}}}, U_{p^{\widehat{\mathsf{B}}}}, Y_{\leq i}\right)_{(p^{\widehat{\mathsf{A}}}, p^{\widehat{\mathsf{B}}}) \leftarrow \mathsf{Dcr}(M_{\leq i})} \overset{\text{C}}{\approx}_{1/r} \left(U_{p^{\widehat{\mathsf{A}}}}, U_{Y_i^{\mathsf{B}}}, Y_{\leq i}\right)_{p^{\widehat{\mathsf{A}}} \leftarrow \mathsf{Dcr}(M_{\leq i})^{\widehat{\mathsf{A}}}} \tag{11}$$

Finally, conditioned on $Y_{\leq i}$, the distribution of $(U_{p^{\widehat{\mathsf{A}}}}, U_{Y_i^{\mathsf{B}}})$ is a convex combination of product distributions of the form $(\cdot, U_{Y_i^{\mathsf{B}}}) = (U_{p^{\widehat{\mathsf{A}}}}, U_{Y_i^{\mathsf{B}}})|_{M_{\leq i}=t_{\leq i}}$ (for $t_i \leftarrow M_{\leq i}|_{Y_{\leq i}}$), and thus it is a product distribution.

1.3 Related Work

We review some of the relevant work on fair coin-flipping protocols.

Necessary hardness assumptions. This line of work examines the minimal assumptions required to achieve an $o(1/\sqrt{r})$-bias two-party coin-flipping protocols, as done in this paper. The necessity of one-way functions for weaker variants of coin flipping protocol where the honest party is allowed to abort if the other party aborts or deviates from the prescribed protocol, were considered in [18, 19, 13, 6]. More related to our bound is the work of Dachman-Soled et al. [11] who showed that any fully black-box construction of $O(1/r)$-bias two-party protocols based on one-way functions (with r-bit input and output) needs $\Omega(r/\log r)$ rounds, and the work of Dachman-Soled et al. [12] showed that there is no fully black-box and function *oblivious* construction of $O(1/r)$-bias two-party protocols from one-way functions (a protocol is function oblivious if the outcome of protocol is independent of the choice of the one-way function used in the protocol).

Lower bounds. Cleve [9] proved that, for every r-round two-party coin-flipping protocol, there exists an efficient adversary that can bias the output by $\Omega(1/r)$. Cleve and Impagliazzo [10] proved that, for every r-round two-party coin-flipping protocol, there exists an inefficient fail-stop adversary that biases the output by $\Omega(1/\sqrt{r})$. They also showed that a similar attack exists if the parties have access to an ideal commitment scheme. All above bounds extend to the multi-party case (with no honest majority) via a simple reduction. Very recently, Beimel et al. [5] showed that *any* r-round n-parties coin-flipping with $n^k > r$, for some $k \in \mathbb{N}$, can be biased by $1/(\sqrt{r} \cdot (\log r)^k)$. Ignoring logarithmic factors, this means that if the number of parties is $r^{\Omega(1)}$, the majority protocol of [3] is optimal.

Upper bounds. Blum [7] presented a two-party two-round coin-flipping protocol with bias $1/4$. Awerbuch et al. [3] presented an n-party r-round protocol with bias $O(n/\sqrt{r})$ (the two-party case appears also in Cleve [9]). Moran et al. [20] solved the two-party case by giving a two-party r-round coin-flipping protocol with bias $O(1/r)$. Haitner and Tsfadia [14] solved the three-party case up to poly-logarithmic factor by giving a three-party coin-flipping protocol with bias $O(\text{polylog}(r)/r)$. Buchbinder et al. [8] showed an n-party r-round coin-flipping protocol with bias $\widetilde{O}(n^3 2^n / r^{\frac{1}{2}+\frac{1}{2^{n-1}-2}})$. In particular, their protocol for four parties has bias $\widetilde{O}(1/r^{2/3})$, and for $n = \log\log r$ their protocol has bias smaller than Awerbuch et al. [3].

For the case where less than $2/3$ of the parties are corrupt, Beimel et al. [4] showed an n-party r-round coin-flipping protocol with bias $2^{2^k}/r$, tolerating up to $t = (n+k)/2$ corrupt parties. Alon and Omri [1] showed an n-party r-round coin-flipping protocol with bias $\widetilde{O}(2^{2^n}/r)$, tolerating up to t corrupted parties, for constant n and $t < 3n/4$.

Paper Organization

Basic definitions and notation used through the paper, are given in Section 2. The formal statement and proof of the main theorem are given in Section 3.

2 Preliminaries

2.1 Notation

We use calligraphic letters to denote sets, uppercase for random variables and functions, lowercase for values. For $a, b \in \mathbb{R}$, let $a \pm b$ stand for the interval $[a-b, a+b]$. For $n \in \mathbb{N}$, let $[n] = \{1, \ldots, n\}$ and $(n) = \{0, \ldots, n\}$. Let poly denote the set of all polynomials, let PPT stand for probabilistic polynomial time and PPTM denote a PPT algorithm (Turing machine). A function $\nu \colon \mathbb{N} \to [0,1]$ is *negligible*, denoted $\nu(n) = \text{neg}(n)$, if $\nu(n) < 1/p(n)$ for every $p \in \text{poly}$ and large enough n. For a sequence $x_1, \ldots, x_r$ and $i \in [r]$, let $x_{\leq i} = x_1, \ldots, x_i$ and $x_{<i} = x_1, \ldots, x_{i-1}$.

Given a distribution, or random variable, D, we write $x \leftarrow D$ to indicate that x is selected according to D. Given a finite set $\mathcal{S}$, let $s \leftarrow \mathcal{S}$ denote that s is selected according to the uniform distribution over $\mathcal{S}$. The support of D, denoted $\text{Supp}(D)$, be defined as $\{u \in \mathcal{U} : D(u) > 0\}$. The *statistical distance* between two distributions P and Q over a finite set $\mathcal{U}$, denoted as $\text{SD}(P,Q)$, is defined as $\max_{\mathcal{S} \subseteq \mathcal{U}} |P(\mathcal{S}) - Q(\mathcal{S})| = \frac{1}{2}\sum_{u \in \mathcal{U}} |P(u) - Q(u)|$. Distribution ensembles $X = \{X_\kappa\}_{\kappa \in \mathbb{N}}$ and $Y = \{Y_\kappa\}_{\kappa \in \mathbb{N}}$ are *δ-computationally indistinguishable in the set $\mathcal{I}$*, denoted by $X \overset{\text{C}}{\approx}_{\mathcal{I},\delta} Y$, if for every PPTM D and sufficiently large $\kappa \in \mathcal{I}$: $|\Pr[\mathsf{D}(1^\kappa, X_\kappa) = 1] - \Pr[\mathsf{D}(1^\kappa, Y_\kappa) = 1]| \leq \delta$.

2.2 Protocols

Let $\pi = (\mathsf{A}, \mathsf{B})$ be a two-party protocol. The protocol π is PPT if the running time of both A and B is polynomial in their input length (regardless of the party they interact with). We denote by $(\mathsf{A}(x), \mathsf{B}(y))(z)$ a random execution of π with private inputs x and y, and common input z, and sometimes abuse notation and write $(\mathsf{A}(x), \mathsf{B}(y))(z)$ for the parties' output in this execution.

We will focus on no-input two-party single-bit output PPT protocol: the only input of the two PPT parties is the common security parameter given in unary representation. At the end of the execution, each party outputs a single bit. Throughout, we assume without loss of generality that the transcript contains 1^κ as the first message. Let $\pi = (\mathsf{A}, \mathsf{B})$ be such a two-party single-bit output protocol. For $\kappa \in \mathbb{N}$, let $O_\pi^{\mathsf{A},\kappa}$, $O_\pi^{\mathsf{B},\kappa}$ and T_π^κ denote the outputs of A, B and the transcript of π, respectively, in a random execution of $\pi(1^\kappa)$.

2.2.1 Fair Coin Flipping

Since we are concerned with a lower bound, we only give the game-based definition of coin-flipping protocols (see [15] for the stronger simulation-based definition).

Definition 2.1 (Fair coin-flipping protocols)**.** *A* PPT *single-bit output two-party protocol* $\pi = (\mathsf{A}, \mathsf{B})$ *is an* ε*-fair coin-flipping protocol, if the following holds.*

Output delivery: *The honest party always outputs a bit (even if the other party acts dishonestly, or aborts).*

Agreement: *The parties always output the same bit in an honest execution.*

Uniformity: $\Pr\left[O_\kappa^{\mathsf{A}} = b\right] = 1/2$ *(and thus* $\Pr\left[O_\kappa^{\mathsf{B}} = b\right] = 1/2$*), for both* $b \in \{0,1\}$ *and all* $\kappa \in \mathbb{N}$*.*

Fairness: *For any* PPT A^* *and* $b \in \{0,1\}$*, for sufficiently large* $\kappa \in \mathbb{N}$ *it holds that*

$\Pr\left[O_\kappa^{\mathsf{B},(\mathsf{A}^*,\mathsf{B})} = b\right] \leq 1/2 + \varepsilon$*, and the same holds for the output bit of* A*.*

2.2.2 Key-Agreement

We focus on single-bit output key-agreement protocols.

Definition 2.2 (Key-agreement protocols)**.** *A* PPT *single-bit output two-party protocol* $\pi = (\mathsf{A}, \mathsf{B})$ *is* io-key-agreement*, if there exist an infinite* $\mathcal{I} \subseteq \mathbb{N}$*, such that the following hold for* κ*'s in* $\mathcal{I}$*:*

Agreement. $\Pr\left[X_\kappa^\pi = Y_\kappa^\pi\right] \geq 1 - \operatorname{neg}(\kappa)$*.*

Secrecy. $\Pr\left[\mathsf{Eve}(T_\kappa^\pi) = X_\kappa^\pi\right] \leq 1/2 + \operatorname{neg}(\kappa)$*, for every* PPT Eve*.*

2.3 Martingales

Definition 2.3 (Martingales)**.** *Let* $X_0, \ldots, X_r$ *be a sequence of random variables. We say that* $X_0, \ldots, X_r$ *is a martingale sequence if* $\mathbf{E}\left[X_{i+1} \mid X_{\leq i} = x_{\leq i}\right] = x_i$ *for every* $i \in [r-1]$*.*

In plain terms, a sequence is a martingale if the expectation of the next point conditioned on the entire history is exactly the last observed point. One way to obtain a martingale sequence is by constructing a *Doob martingale*. Such a sequence is defined by $X_i = \mathbf{E}\left[f(Z) \mid Z_{\leq i}\right]$, for arbitrary random variables $Z = (Z_1, \ldots, Z_r)$ and a function f of interest. We will use the following fact proven by [10] (we use the variant as proven in [5]).

Theorem 2.4. *Let* $X_0, \ldots, X_r$ *be a martingale sequence such that* $X_i \in [0,1]$*, for every* $i \in [r]$*. If* $X_0 = 1/2$ *and* $\Pr\left[X_r \in \{0,1\}\right] = 1$*, then* $\Pr\left[\exists i \in [r] \text{ s.t. } |X_i - X_{i-1}| \geq \frac{1}{4\sqrt{r}}\right] \geq \frac{1}{20}$*.*

3 Fair Coin-Flipping to Key-Agreement

In this section, we prove our main result: if there exist constant-round coin-flipping protocols which improve over the $1/\sqrt{r}$-bias majority protocol of [2], then infinitely-often key-agreement exists as well. Formally, we prove the following theorem.

Theorem 3.1. *The following holds for any (constant) $r \in \mathbb{N}$: if there exists an r-round, $\frac{1}{25600\sqrt{r}}$-fair two-party coin-flipping protocol, see Definition 2.1, then there exists an infinitely-often key-agreement protocol.*[5]

Before formally proving Theorem 3.1, we briefly recall the outline of the proof as presented in the introduction (we ignore certain constants in this outline). We begin with a good forecaster for the coin-flipping protocol π (which must exist, according to [16]), and we define an efficiently computable conditional expected outcome sequence $X = (X_0, \ldots, X_r)$ for π, conditioned on the forecaster's outputs. Then, we show that (1) the i^{th} backup value (default output in case the opponent aborts) should be close to X_i; otherwise, an efficient attacker can use the forecaster to bias the output of the other party (this attack is applicable regardless of the existence of infinitely-often key-agreement). And (2), since X is a martingale sequence, "large" $1/\sqrt{r}$-gaps are bound to occur in some round, with constant probability. Hence, combining (1) and (2), with constant probability, for some i, there is a $1/\sqrt{r}$-gap between X_i and the forecasters' prediction for one party *at the preceding round* $i-1$. Therefore, unless protocol π implies io-key-agreement, the aforementioned gap can be exploited to bias that party's output by $1/\sqrt{r}$, by instructing the opponent to abort as soon as the gap is detected. In more detail, the success of the attack requires that (3) the event that a gap occurs is (almost) *independent* of the backup value of the honest party. It turns out that if π does not imply io-key-agreement, this third property is guaranteed by the dichotomy theorem of [16]. In summary, if io-key-agreement does not exist, then protocol π is at best $1/\sqrt{r}$-fair.

Moving to the formal proof, fix an r-round, two-party coin-flipping protocol $\pi = (\mathsf{A}, \mathsf{B})$ (we assume nothing about its fairness parameter for now). We associate the following random variables with a random honest execution of $\pi(1^\kappa)$. Let $M^\kappa = (M_1^\kappa, \ldots, M_r^\kappa)$ denote the messages of the protocol and let O^κ denote the (always) common output of the parties. For $i \in \{0, \ldots, r\}$ and $\mathsf{P} \in \{\mathsf{A}, \mathsf{B}\}$, let $Z_i^{\mathsf{P},\kappa}$ be the "backup" value party P outputs, if the other party aborts after the i^{th} message was sent. In particular, $Z_r^{\mathsf{A},\kappa} = Z_r^{\mathsf{B},\kappa} = O^\kappa$ and $\Pr[O^\kappa = 1] = 1/2$.

Forecaster for π. We are using a *forecaster* for π, guaranteed by the following theorem (proof readily follows from Haitner et al. [16, Thm 3.8]).

Theorem 3.2 (Haitner et al. [16], existence of forecasters)**.** *Let Δ be a no-input, single-bit output two-party protocol. Then for any constant $\rho > 0$, there exists a* PPT *constant-output length algorithm F (forecaster) mapping transcripts of Δ into (the binary description of) pairs in $[0,1] \times [0,1]$ and an infinite set $\mathcal{I} \in \mathbb{N}$ such that the following holds: let $O^{\mathsf{A},\kappa}$, $O^{\mathsf{B},\kappa}$ and T^κ denote the parties' outputs and protocol transcript, respectively, in a random execution of $\Delta(1^\kappa)$. Let $m(\kappa) \in \mathrm{poly}$ be a bound on the number of coins used by F on transcripts in $\mathrm{supp}(T^\kappa)$, and let S^κ be a uniform string of length $m(\kappa)$. Then,*

[5] Definition 2.1 requires perfect uniformity: the common output in an honest execution is an unbiased bit. The proof given below, however, easily extends to any non-trivial uniformity condition, e.g., the common output equals 1 with probability 3/4.

- $(O^{\mathsf{A},\kappa}, T^\kappa, S^\kappa) \stackrel{\mathrm{C}}{\approx}_{\rho,\mathcal{I}} (U_{p^{\mathsf{A}}}, T^\kappa, S^\kappa)_{(p^{\mathsf{A}},\cdot)=\mathsf{F}(T^\kappa;S^\kappa)}$, *and*

- $(O^{\mathsf{B},\kappa}, T^\kappa, S^\kappa) \stackrel{\mathrm{C}}{\approx}_{\rho,\mathcal{I}} (U_{p^{\mathsf{B}}}, T^\kappa, S^\kappa)_{(\cdot,p^{\mathsf{B}})=\mathsf{F}(T^\kappa;S^\kappa)}$.

letting U_p be a Boolean random variable taking the value 1 with probability p.[6]

Since we require a forecaster for all (intermediate) backup values of π, we apply Theorem 3.2 with respect to the following variant of protocol π, which simply stops the execution at a random round.

Protocol 3.3 ($\widetilde{\pi} = \left(\widetilde{\mathsf{A}}, \widetilde{\mathsf{B}}\right)$).

Common input: security parameter 1^κ.

Description:

1. *$\widetilde{\mathsf{A}}$ samples $i \leftarrow [r]$ and sends it to $\widetilde{\mathsf{B}}$.*

2. *The parties interact in the first i rounds of a random execution of $\pi(1^\kappa)$, with $\widetilde{\mathsf{A}}$ and $\widetilde{\mathsf{B}}$ taking the role of A and B receptively.*

   Let z_i^{A} and z_i^{B} be the i^{th} backup values of A and B as computed by the parties in the above execution.

3. *$\widetilde{\mathsf{A}}$ outputs z_i^{A}, and $\widetilde{\mathsf{B}}$ outputs z_i^{B}.*

Let $\rho = 10^{-6} \cdot r^{-5/2}$. Let $\mathcal{I} \subseteq \mathbb{N}$ and a PPT F be the infinite set and PPT forecaster resulting by applying Theorem 3.2 with respect to protocol $\widetilde{\pi}$ and ρ, and let S^κ denote a long enough uniform string to be used by F on transcripts of $\widetilde{\pi}(1^\kappa)$. The following holds with respect to π.

Claim 3.4. *For $I \leftarrow [r]$, it holds that*

- $(Z_I^{\mathsf{A},\kappa}, M_{\le I}^\kappa, S^\kappa) \stackrel{\mathrm{C}}{\approx}_{\rho,\mathcal{I}} (U_{p^{\mathsf{A}}}, M_{\le I}^\kappa, S^\kappa)_{(p^{\mathsf{A}},\cdot)=\mathsf{F}(M_{\le I};S^\kappa)}$, *and*

- $(Z_I^{\mathsf{B},\kappa}, M_{\le I}^\kappa, S^\kappa) \stackrel{\mathrm{C}}{\approx}_{\rho,\mathcal{I}} (U_{p^{\mathsf{B}}}, M_{\le I}^\kappa, S^\kappa)_{(\cdot,p^{\mathsf{B}})=\mathsf{F}(M_{\le I};S^\kappa)}$,

letting $\mathsf{F}(m_{\le i}; r) = \mathsf{F}(i, m_{\le i}; r)$.

Proof. Immediate, by Theorem 3.2 and the definition of $\widetilde{\pi}$. □

We assume without loss of generality that the common output appears on the last message of π (otherwise, we can add a final message that contains this value, which does not hurt the security of π). Hence, without loss of generality it holds that $\mathsf{F}(m_{\le r}; \cdot) = (b, b)$, where b is the output bit as implied by $m_{\le r}$ (otherwise, we can change F to do so without hurting its forecasting quality).

For $\kappa \in \mathbb{N}$, we define the random variables $Y_0^\kappa, \ldots, Y_r^\kappa$, by

$$Y_i^\kappa = (Y_i^{\mathsf{A},\kappa}, Y_i^{\mathsf{B},\kappa}) = \mathsf{F}(M_{\le i}; S^\kappa) \tag{12}$$

[6]Haitner et al. [16] do not limit the output length of F. Nevertheless, by applying [16] with parameter $\rho/2$ and chopping each of the forecaster's outputs to the first $\lceil \log 1/\rho \rceil + 1$ (most significant) bits, yields the desired constant output length forecaster.

The expected outcome sequence. To attack the protocol, it is useful to evaluate at each round the expected outcome of the protocol conditioned on the forecasters' outputs so far. To alleviate notation, we assume that the value of κ is determined by $|S^\kappa|$.

Definition 3.5 (the expected outcome function)**.** *For $\kappa \in \mathbb{N}$, $i \in [r]$, $y_{\le i} \in \operatorname{supp}(Y^\kappa_{\le i})$ and $s \in \operatorname{Supp}(S^\kappa)$, let*

$$g(y_{\le i}, s) = \mathbf{E}\left[O^\kappa \mid Y^\kappa_{\le i} = y_{\le i}, S^\kappa = s\right].$$

Namely, $g(y_{\le i}, s)$ is the probability that the output of the protocol in a random execution is 1, given that $\mathsf{F}(T_{\le j}; s) = y_j$ for every $j \in (i)$ and $T_1, \ldots, T_r$ being the transcript of this execution.

Expected outcome sequence is approximable. The following claim, proven in Section 3.1, yields that the expected outcome sequence can be approximated efficiently.

Claim 3.6 (Expected outcome sequence is approximable)**.** *There exists* PPTM G *such that*

$$\Pr\left[\mathsf{G}(Y^\kappa_{\le i}, S^\kappa) \notin g(Y^\kappa_{\le i}, S^\kappa) \pm \rho\right] \le \rho,$$

for every $\kappa \in \mathbb{N}$ and $i \in [r]$.

Algorithm G approximates the value of g on input $(y_{\le i}, s) \in \operatorname{supp}(Y^\kappa_{\le i}, S^\kappa)$ by running multiple independent instances of protocol $\pi(1^\kappa)$ and keeping track of the number of times it encounters $y_{\le i}$ and the protocol outputs one. Standard approximation techniques yield that, unless $y_{\le i}$ is very unlikely, the output of G is close to $g(y_{\le i}, s)$. Claim 3.6 follows by carefully choosing the number of iterations for G and bounding the probability of encountering an unlikely $y_{\le i}$.

Forecasted backup values are close to expected outcome sequence. The following claim bounds the probability that the expected outcome sequence and the forecaster's outputs deviate by more than $1/8\sqrt{r}$. The proof is given in Section 3.2.

Claim 3.7 (Forecasted backup values are close to expected outcome sequence)**.** *Assuming π is $\frac{1}{6400\sqrt{r}}$-fair, then*

$$\Pr\left[\exists i \in [r] \text{ s.t. } \left|g(Y^\kappa_{\le i}, S^\kappa) - Y_i^{\mathsf{P},\kappa}\right| \ge 1/8\sqrt{r}\right] < 1/100$$

for both $\mathsf{P} \in \{\mathsf{A}, \mathsf{B}\}$ and large enough $\kappa \in \mathcal{I}$.

Loosely speaking, Claim 3.7 states that the expected output sequence and the forecaster's outputs are close for a fair protocol. If not, then either of the following attackers P^*_0, P^*_1 can bias the output of party P: for fixed randomness $s \in \operatorname{supp}(S^\kappa)$, attacker P^*_z computes $y_i = \mathsf{F}(m_{\le i}, s)$ for partial transcript $m_{\le i}$ at round $i \in [r]$, and aborts as soon as $(-1)^{1-z}(\mathsf{G}(y^\kappa_{\le i}, s) - y_i) \ge 1/8\sqrt{r} - \rho$. The desired bias is guaranteed by the accuracy of the forecaster (Claim 3.4), the accuracy of algorithm G (Claim 3.6) and the presumed frequency of occurrence of a suitable gap. The details of the proof are given in Section 3.2.

Expected outcome sequence has large gap. Similarly to [10], the success of our attack depends on the occurrence of large gaps in the expected outcome sequence. The latter is guaranteed by [10], [5], since the expected outcome sequence is a suitable martingale.

Claim 3.8 (Expected outcomes have large gap)**.** *For every $\kappa \in \mathbb{N}$, it holds that* $\Pr\left[\exists i \in [r] \colon \left|g(Y^{\kappa}_{\le i}, S^{\kappa}) - g(Y^{\kappa}_{\le i-1}, S^{\kappa})\right| \ge 1/4\sqrt{r}\right] > 1/20$.

Proof. Consider the sequence of random variables $G^{\kappa}_0, \ldots, G^{\kappa}_r$ defined by $G^{\kappa}_i = g(Y^{\kappa}_{\le i}, S^{\kappa})$. Observe that this is a Doob (and hence, strong) martingale sequence, with respect to the random variables $Z_0 = S^{\kappa}$ and $Z_i = Y^{\kappa}_i$ for $i \in [r]$, and the function $f(S^{\kappa}, Y^{\kappa}_{\le r}) = g(Y^{\kappa}_{\le r}, S^{\kappa}) = Y^{\kappa}_r[0]$ (i.e., the function that outputs the actual output of the protocol, as implied by Y^{κ}_r). Clearly, $G^{\kappa}_0 = 1/2$ and $G^{\kappa}_r \in \{0,1\}$ (recall that we assume that $\mathsf{F}(M_{\le r}; \cdot) = (b, b)$, where b is the output bit as implied by $M_{\le r}$). Thus, the proof follows by Theorem 2.4. □

Independence of attack decision. Claim 3.4 immediately yields that the expected values of Y_i and Z^{P}_i are close, for both $\mathsf{P} \in \{\mathsf{A}, \mathsf{B}\}$ and every $i \in [r]$. Assuming io-key-agreement does not exist, the following claim essentially states that Y_i and Z^{P}_i remain close in expectation, even if we condition on some event that depends on the other party's next message. This observation will allow us to show that, when a large gap in the expected outcome is observed by one of the parties, the (expected value of the) backup value of the other party still lags behind. The following claim captures the core of the novel idea in our attack, and its proof is the most technical aspect towards proving our main result.

Claim 3.9 (Independence of attack decision)**.** *Let C be a single-bit output* PPTM*. For $\kappa \in \mathbb{N}$ and $\mathsf{P} \in \{\mathsf{A}, \mathsf{B}\}$, let $E^{\mathsf{P},\kappa}_1, \ldots, E^{\mathsf{P},\kappa}_r$ be the sequence of random variables such that $E^{\mathsf{P},\kappa}_i$ is the indicator for the event*

$$\mathsf{P} \textit{ sends the } i^{\text{th}} \textit{ message in } \pi(1^{\kappa}) \wedge\ \mathsf{C}(Y^{\kappa}_{\le i}, S^{\kappa}) = 1.$$

Assume io-key-agreement protocols do not exist. Then, for any $\mathsf{P} \in \{\mathsf{A}, \mathsf{B}\}$ and infinite subset $\mathcal{I}' \subseteq \mathcal{I}$, there exists an infinite set $\mathcal{I}'' \subseteq \mathcal{I}'$ such that

$$\mathbf{E}\left[E^{\mathsf{P},\kappa}_{i+1} \cdot (Z^{\overline{\mathsf{P}},\kappa}_i - Y^{\overline{\mathsf{P}},\kappa}_i)\right] \in \pm 4r\rho$$

for every $\kappa \in \mathcal{I}''$ and $i \in (r-1)$, where $\overline{\mathsf{P}}$ denotes (the party in) $\{\mathsf{A}, \mathsf{B}\} \setminus \{\mathsf{P}\}$.

Since $\mathbf{E}\left[E^{\mathsf{P},\kappa}_{i+1} \cdot (Z^{\overline{\mathsf{P}},\kappa}_i - Y^{\overline{\mathsf{P}},\kappa}_i)\right] = \mathbf{E}\left[E^{\mathsf{P},\kappa}_{i+1} \cdot \mathbf{E}\left[Z^{\overline{\mathsf{P}},\kappa}_i - Y^{\mathsf{P},\kappa}_i \mid E^{\mathsf{P},\kappa}_{i+1} = 1\right]\right]$, Claim 3.9 yields that the expected values of Y_i and Z^{P}_i remain close, even when conditioning on a likely enough event over the next message of P.

The proof of Claim 3.9 is given in Section 3.3. In essence, we use the recent dichotomy of Haitner et al. [16] to assert that if io-key-agreement does not exist, then the values of $E^{\mathsf{P},\kappa}_{i+1}$ and $Z^{\overline{\mathsf{P}},\kappa}_i$ conditioned on $T_{\le i}$ (which determines the value of $Y^{\overline{\mathsf{P}},\kappa}_i$), are (computationally) close to be in a product distribution.

Putting everything together. Equipped with the above observations, we prove Theorem 3.1.

Proof of Theorem 3.1. Let π be an $\varepsilon = \frac{1}{25600\sqrt{r}}$-fair coin flipping protocol. By Claims 3.7 and 3.8, we can assume without loss of generality that there exists an infinite subset $\mathcal{I}' \subseteq \mathcal{I}$ such that

$$\Pr\left[\exists i \in [r] \colon \mathsf{A} \text{ sends } i^{\text{th}} \text{ message in } \pi(1^\kappa) \wedge g(Y^\kappa_{\leq i}, S^\kappa) - Y^{\mathsf{B},\kappa}_{i-1} \geq \frac{1}{8\sqrt{r}}\right] \geq \frac{1}{80} - \frac{1}{100} = \frac{1}{400} \quad (13)$$

We define the following PPT fail-stop attacker A^* taking the role of A in π. We will show below that assuming io-key-agreement do not exist, algorithm A^* succeeds in biasing the output of B towards zero by ε for all $\kappa \in \mathcal{I}''$, contradicting the presumed fairness of π.

In the following, let G be the PPTM guaranteed to exist by Claim 3.6.

Algorithm 3.10 (A^*)**.**

Input: security parameter 1^κ.

Description:

1. *Sample* $s \leftarrow S^\kappa$ *and start a random execution of* $\mathsf{A}(1^\kappa)$.
2. *Upon receiving the* $(i-1)$ *message* m_{i-1}, *do*
   (a) *Forward* m_{i-1} *to* A, *and let* m_i *be the next message sent by* A.
   (b) *Compute* $y_i = (y_i^{\mathsf{A}}, y_i^{\mathsf{B}}) = \mathsf{F}(m_{\leq i}, s)$.
   (c) *Compute* $\widetilde{g}_i = \mathsf{G}(y_{\leq i}, s)$.
   (d) *If* $\widetilde{g}_i \geq y^{\mathsf{B}}_{i-1} + 1/16\sqrt{r}$, *abort (without sending further messages).* *Otherwise, send* m_i *to* B *and proceed to the next round.*

. . . .

It is clear that A^* is a PPTM. We conclude the proof showing that assuming io-key-agreement do not exist, B's output when interacting with A^* is biased towards zero by at least ε.

The following random variables are defined with respect to a a random execution of $(\mathsf{A}^*, \mathsf{B})(1^\kappa)$. Let S^κ and $Y^\kappa = (Y^\kappa_1, \ldots, Y^\kappa_r)$ denote the values of s and $y_1, \ldots, y_r$ sampled by A^*. Let $Z^{\mathsf{B},\kappa} = (Z^{\mathsf{B},\kappa}_1, \ldots, Z^{\mathsf{B},\kappa}_r)$ denote the backup values computed by B. For $i \in [r]$, let E^κ_i be the event that A^* decides to abort in round i. Finally, let J^κ be the index i with $E^\kappa_i = 1$, setting it to $r+1$ if no such index exist. Below, if we do not quantify over κ, it means that the statement holds for any $\kappa \in \mathbb{N}$.

By Claim 3.6 and Equation (13),

$$\Pr\left[J^\kappa \neq r+1\right] > \frac{1}{400} - \rho \geq \frac{1}{800} \quad (14)$$

for every $\kappa \in \mathcal{I}'$. Where since the events E^κ_i and E^κ_j for $i \neq j$ are disjoint,

$$\begin{aligned}
\mathbf{E}\left[Z^{\mathsf{B},\kappa}_{J^\kappa-1} - Y^{\mathsf{B},\kappa}_{J^\kappa-1}\right] &= \mathbf{E}\left[\sum_{i=1}^{r+1} E^\kappa_i \cdot (Z^{\mathsf{B},\kappa}_{i-1} - Y^{\mathsf{B},\kappa}_{i-1})\right] \quad (15)\\
&= \sum_{i=1}^{r+1} \mathbf{E}\left[E^\kappa_i \cdot (Z^{\mathsf{B},\kappa}_{i-1} - Y^{\mathsf{B},\kappa}_{i-1})\right]\\
&= \sum_{i=1}^{r} \mathbf{E}\left[E^\kappa_i \cdot (Z^{\mathsf{B},\kappa}_{i-1} - Y^{\mathsf{B},\kappa}_{i-1})\right].
\end{aligned}$$

The last inequality holds since the protocol's output appears in the last message, by assumption, and thus without loss of generality $Z_r^{\mathsf{B},\kappa} = Y_r^{\mathsf{B},\kappa}$. Consider the single-bit output PPTM C defined as follows: on input $(y_{\leq i} = ((y_1^{\mathsf{A}}, y_1^{\mathsf{B}}), \ldots, (y_i^{\mathsf{A}}, y_i^{\mathsf{B}})), s)$, it outputs 1 if $\mathsf{G}(y_{\leq i}, s) - y_{i-1}^{\mathsf{B}} \geq 1/16\sqrt{r}$, and $\mathsf{G}(y_{\leq j}, s) - y_{j-1}^{\mathsf{B}} < 1/16\sqrt{r}$ for all $j < i$; otherwise, it outputs zero. The event that A sends the i^{th} message in $\pi(1^\kappa)$ and $\mathsf{C}(Y_{\leq i}^\kappa, S^\kappa) = 1$ and the event E_i^κ are identically distributed, for any fixing of $(Y^\kappa, S^\kappa, Z^{\mathsf{B},\kappa})$. Thus, assuming io-key-agreement protocols do not exist, Claim 3.9 yields that that there exists an infinite set $\mathcal{I}'' \subset \mathcal{I}'$ such that

$$\mathbf{E}\left[E_{i+1}^\kappa \cdot (Z_i^{\mathsf{B},\kappa} - Y_i^{\mathsf{B},\kappa})\right] \in \pm 4r\rho \tag{16}$$

for every $\kappa \in \mathcal{I}''$ and $i \in [r-1]$. Putting together Equations (15) and (16), we conclude that

$$\mathbf{E}\left[Z_{J^\kappa-1}^{\mathsf{B},\kappa} - Y_{J^\kappa-1}^{\mathsf{B},\kappa}\right] \in \pm 4r^2\rho \tag{17}$$

for every $\kappa \in \mathcal{I}''$.

Recall that our goal is to show that $\mathbf{E}\left[Z_{J^\kappa-1}^{\mathsf{B},\kappa}\right]$ is significantly smaller than $1/2$. We do it by showing that it is significantly smaller than $\mathbf{E}\left[g(Y_{\leq J^\kappa}^\kappa, S^\kappa)\right]$ which equals $1/2$, as we show next. Letting $Y_{r+1}^\kappa = Y_r^\kappa$ and $g(y_1, \ldots, y_{r+1}, s) = g(y_1, \ldots, y_r, s)$, we compute

$$\begin{aligned}
\mathbf{E}\left[g(Y_{\leq J^\kappa}^\kappa, S^\kappa)\right] &= \mathbf{E}_{j \leftarrow J^\kappa}\left[\mathbf{E}_{(y,s \leftarrow (Y^\kappa, S^\kappa) | J^\kappa = j}\left[O^\kappa \mid (Y_{\leq i}^\kappa, S^\kappa) = (y_{\leq i}, s)\right]\right] \\
&= \mathbf{E}_{j \leftarrow J^\kappa}\left[\mathbf{E}_{(y,s) \leftarrow (Y^\kappa, S^\kappa) | J^\kappa = j}\left[O^\kappa \mid (Y_{\leq i}^\kappa, S^\kappa, J^\kappa) = (y_{\leq i}, s, j)\right]\right] \\
&= \mathbf{E}\left[O^\kappa\right] \\
&= 1/2.
\end{aligned} \tag{18}$$

Finally, let G_i be the value of $\mathsf{G}(Y_{\leq i}, S^\kappa)$ computed by A^* in the execution of $(\mathsf{A}^*, \mathsf{B})(1^\kappa)$ considered above, letting $G_{r+1} = g(Y_{\leq r+1}^\kappa, S^\kappa)$. Claim 3.6 yields that

$$\mathbf{E}\left[g(Y_{\leq J^\kappa}^\kappa, S^\kappa) - G_{J^\kappa}\right] \leq 2r\rho \tag{19}$$

Putting all the above observations together, we conclude that, for every $\kappa \in \mathcal{I}''$,

$$\begin{aligned}
&\mathbf{E}\left[Z_{J^\kappa-1}^{\mathsf{B},\kappa}\right] \\
&= \mathbf{E}\left[g(Y_{\leq J^\kappa}^\kappa, S^\kappa)\right] - \mathbf{E}\left[G_{J^\kappa} - Y_{J^\kappa-1}^{\mathsf{B},\kappa}\right] + \mathbf{E}\left[Z_{J^\kappa-1}^{\mathsf{B},\kappa} - Y_{J^\kappa-1}^{\mathsf{B},\kappa}\right] - \mathbf{E}\left[g(Y_{\leq J^\kappa}^\kappa, S^\kappa) - G_{J^\kappa}\right] \\
&\leq \frac{1}{2} - \mathbf{E}\left[G_{J^\kappa} - Y_{J^\kappa-1}^{\mathsf{B},\kappa} \mid J^\kappa \neq r+1\right] \cdot \Pr\left[J^\kappa \neq r+1\right] + 4r^2\rho + 2r\rho \\
&\leq \frac{1}{2} - (1/16\sqrt{r}) \cdot (1/800) + 4r^2\rho + 2r\rho \\
&< \frac{1}{2} - \frac{1}{25600\sqrt{r}}.
\end{aligned}$$

The first inequality holds by Equations (17) to (19). The second inequality holds by the definition of J^κ and Equation (14). The last inequality holds by our choice of ρ.

□

3.1 Approximating the Expected Outcome Sequence

In this section we prove Claim 3.6, restated below.

Claim 3.11 (Claim 3.6, restated). *There exists* PPTM G *such that*

$$\Pr\left[\mathsf{G}(Y^{\kappa}_{\leq i}, S^{\kappa}) \notin g(Y^{\kappa}_{\leq i}, S^{\kappa}) \pm \rho\right] \leq \rho,$$

for every $\kappa \in \mathbb{N}$ *and* $i \in [r]$.

The proof of Claim 3.11 is straightforward. Since there are only constant number of rounds and F has constant output length, when fixing the randomness of F, the domain of G has constant size. Hence, the value of of g can be approximated well via sampling. Details below.

Let c be a bound on the number of possible outputs of F (recall that F has constant output length). We are using the following implementation for G.

In the following, let $\overline{\mathsf{F}}((m_1, \ldots, m_i); s) = (\mathsf{F}(m_1; s), \ldots, (\mathsf{F}(m_i; s))$ (i.e., $\overline{\mathsf{F}}(M_{\leq i}; S^{\kappa}) = Y_{\leq i}$).

Algorithm 3.12 (G).

Parameters: $v = \left\lceil \frac{1}{2} \cdot \left(\frac{2c^r}{\rho}\right)^4 \cdot \ln\left(\frac{8}{\rho}\right) \right\rceil$.

Input: $y_{\leq i} \in \operatorname{supp}(Y^{\kappa}_{\leq i})$ *and* $s \in \operatorname{Supp}(S^{\kappa})$.

Description:

1. *Sample* v *transcripts* $\left\{m^j, \mathsf{out}^j\right\}_{j \in [v]}$ *by taking the (full) transcripts and outputs of* v *independent executions of* $\pi(1^{\kappa})$.
2. *For every* $j \in [v]$ *let* $y^j_i = \overline{\mathsf{F}}(m^j_{\leq i}; s)$.
3. *Let* $q = \left|\left\{j \in [v] \colon y^j_{\leq i} = y_{\leq i}\right\}\right|$ *and* $p = \left|\left\{j \in [v] \colon y^j_{\leq i} = y_{\leq i} \wedge \mathsf{out}^j = 1\right\}\right|$.
4. *Set* $\widetilde{g} = p/q$. *(Set* $\widetilde{g} = 0$ *if* $q = p = 0$.*)*
5. *Output* $\widetilde{g}$.

Remark 3.13 (A more efficient approximator.). *The running time of algorithm* G *above is exponential in* r*. While this does not pose a problem for our purposes here, since* r *is constant, it might leave the impression that out approach cannot be extended to protocols with super-constant round complexity. So it is worth mentioning that the running time of* G *can be reduced to be polynomial in* r*, by using the augmented weak martingale paradigm of Beimel et al. [5]. Unfortunately, we currently cannot benefit from this improvement, since the result of [16] only guarantees indistinguishablity for constant* ρ*, which makes it useful only for attacking constant-round protocols.*

We prove Claim 3.11 by showing that the above algorithm approximates g well.

Proof of Claim 3.11 . To prove the quality of G in approximating g, it suffices to prove the claim for every every $\kappa \in \mathbb{N}$, $i \in [r]$ and fixed $s \in \operatorname{supp}(S^{\kappa})$. That is

$$\Pr\left[\left|g(\overline{\mathsf{F}}(M_{\leq i}, s), s) - \mathsf{G}(\overline{\mathsf{F}}(M_{\leq i}, s), s)\right| \geq \rho\right] \leq \rho, \tag{20}$$

where the probability is also taken over the random coins of G.

Fix $\kappa \in \mathbb{N}$ and omit it from the notation, and fix $i \in [r]$ and $s \in S^\kappa$. Let $\mathcal{D}_i = \left\{y_{\leq i} \colon \Pr\left[\overline{\mathsf{F}}(M_{\leq i}, s) = y_{\leq i}\right] \geq \rho/2c^r\right\}$. By Hoeffding's inequality [17], for every $y_{\leq i} \in \mathcal{D}$, it holds that

$$\begin{aligned}\Pr\left[|g(y_{\leq i}, s) - \mathsf{G}(y_{\leq i}, s)| \geq \rho\right] &\leq 4 \cdot \exp\left(-2 \cdot v \cdot (\rho/2c^r)^4\right) \\ &\leq 4 \cdot \exp\left(-\frac{v\rho^4}{8c^{4r}}\right) \\ &\leq \rho/2.\end{aligned} \tag{21}$$

It follows that

$$\begin{aligned}&\Pr\left[\left|g(\overline{\mathsf{F}}(M_{\leq i}, s), s) - \mathsf{G}(\overline{\mathsf{F}}(M_{\leq i}, s), s)\right| \geq \rho\right] \\ &\leq \Pr\left[(\overline{\mathsf{F}}(M_{\leq j}, s) \notin \mathcal{D}\right] + \rho/2 \\ &\leq \left|\mathrm{Supp}(\overline{\mathsf{F}}(M_{\leq j}, s))\right| \cdot \rho/2c^r + \rho/2 \\ &\leq c^r \cdot \rho/2c^r + \rho/2 = \rho.\end{aligned}$$

□

3.2 Forecasted Backup Values are Close to Expected Outcome Sequence

In this section, we prove Claim 3.7 (restated below).

Claim 3.14 (Claim 3.7, restated). *Assuming π is $\frac{1}{6400\sqrt{r}}$-fair, then*

$$\Pr\left[\exists i \in [r] \text{ s.t. } \left|g(Y^\kappa_{\leq i}, S^\kappa) - Y_i^{\mathsf{P},\kappa}\right| \geq 1/8\sqrt{r}\right] < 1/100$$

for both $\mathsf{P} \in \{\mathsf{A}, \mathsf{B}\}$ and large enough $\kappa \in \mathcal{I}$.

Proof. Assume the claim does not holds for $\mathsf{P} = \mathsf{B}$ and infinitely many security parameters $\mathcal{I}$ (the case $\mathsf{P} = \mathsf{A}$ is proven analogously). That is, for all $\kappa \in \mathcal{I}$ and without loss of generality, it holds that

$$\Pr\left[\exists i \in [r] \text{ s.t. } g(Y^\kappa_{\leq i}, S^\kappa) - Y_i^{\mathsf{B},\kappa} \geq \frac{1}{8\sqrt{r}}\right] \geq \frac{1}{200} \tag{22}$$

Consider the following PPT fail-stop attacker A^* taking the role of A in π to bias the output of B towards zeros.

Algorithm 3.15 (A^*).

Input: security parameter 1^κ.

Description:

1. *Samples $s \leftarrow S^\kappa$ and start a random execution of $\mathsf{A}(1^\kappa)$.*

2. *For $i = 1 \dots r$:*

   After sending (or receiving) the prescribed message m_i:

(a) Let $y_i = \mathsf{F}(m_{\leq i}; s)$ *and* $\mu_i = \mathsf{G}(y_{\leq i}, s) - y_i$*.*

(b) Abort if $\mu_i \geq \frac{1}{8\sqrt{r}} - \rho$ *(without sending further messages).*
Otherwise, proceed to the next round.

In the following, we fix a large enough $\kappa \in \mathcal{I}$ such that Equation (22) holds, and we omit it from the notation when the context is clear. We show that algorithm A^* biases the output of B towards zero by at least $1/(6400\sqrt{r})$.

We associate the following random variables with a random execution of $(\mathsf{A}^*, \mathsf{B})$. Let J denote the index where the adversary aborted, i.e., the smallest j such that $\mathsf{G}(Y_{\leq j}, S) - Y_j^{\mathsf{B}} \geq \frac{1}{8\sqrt{r}} - \rho$, or $J = r$ if no abort occurred. The following expectations are taken over $(Y_{\leq i}, S)$ and the random coins of G. We bound $\mathbf{E}\left[Z_J^{\mathsf{B}}\right]$, i.e. the expected output of the honest party.

$$\begin{aligned}
&\mathbf{E}\left[Z_J^{\mathsf{B}}\right] \\
&= \mathbf{E}\left[Z_J^{\mathsf{B}}\right] + \mathbf{E}\left[g(Y_{\leq J}, S)\right] - \mathbf{E}\left[g(Y_{\leq J}, S)\right] + \mathbf{E}\left[\mathsf{G}(Y_{\leq J}, S) - Y_J^{\mathsf{B}}\right] - \mathbf{E}\left[\mathsf{G}(Y_{\leq J}, S) - Y_J^{\mathsf{B}}\right] \\
&= \mathbf{E}\left[g(Y_{\leq J}, S)\right] - \mathbf{E}\left[\mathsf{G}(Y_{\leq J}, S) - Y_J^{\mathsf{B}}\right] + \mathbf{E}\left[\mathsf{G}(Y_{\leq J}, S) - g(Y_{\leq J}, S)\right] + \mathbf{E}\left[Z_J^{\mathsf{B}} - Y_J^{\mathsf{B}}\right] \\
&= \frac{1}{2} - \mathbf{E}\left[\mathsf{G}(Y_{\leq J}, S) - Y_J^{\mathsf{B}}\right] + \mathbf{E}\left[\mathsf{G}(Y_{\leq J}, S) - g(Y_{\leq J}, S)\right] + \mathbf{E}\left[Z_J^{\mathsf{B}} - Y_J^{\mathsf{B}}\right].
\end{aligned} \tag{23}$$

The last equation follows from $\mathbf{E}\left[g(Y_{\leq J}, S)\right] = \mathbf{E}\left[\mathsf{out}\right]$ and thus $\mathbf{E}\left[g(Y_{\leq J}, S)\right] = \frac{1}{2}$ (for a more detailed argument see Equation (18) and preceding text). We bound each of the terms above separately. First, observe that

$$\begin{aligned}
&\Pr\left[J \neq r\right] \\
&\geq \Pr\left[(\forall i \in [r]\colon\ |\mathsf{G}(Y_{\leq i}, S) - g(Y_{\leq i}, S)| \leq \rho) \wedge \left(\exists j \in [r]\colon g(Y_{\leq j}, S) - Y_j^{\mathsf{B}} \geq \frac{1}{8\sqrt{r}}\right)\right] \\
&\geq \Pr\left[\exists j \in [r]\colon g(Y_{\leq j}, S) - Y_j \geq \frac{1}{8\sqrt{r}}\right] - \Pr\left[\exists i \in [r]\colon\ |\mathsf{G}(Y_{\leq i}, S) - g(Y_{\leq i}, S)| > \rho\right] \\
&\geq \frac{1}{200} - \rho \\
&\geq \frac{1}{400}.
\end{aligned} \tag{24}$$

The penultimate inequality is by Equation (23) and Claim 3.6. It follows that

$$\begin{aligned}
\mathbf{E}\left[g(Y_{\leq J}, S) - Y_J^{\mathsf{B}}\right] &= \Pr\left[J \neq r\right] \cdot \mathbf{E}\left[g(Y_{\leq J}, S) - Y_J^{\mathsf{B}} \mid J \neq r\right] \\
&\geq \frac{1}{400} \cdot \left(\frac{1}{8\sqrt{r}} - \rho\right) - \mathbf{E}\left[\mathsf{G}(Y_{\leq J}, S) - g(Y_{\leq J}, S)\right] \\
&\geq \frac{1}{400} \cdot \frac{1}{8\sqrt{r}} - 3\rho.
\end{aligned} \tag{25}$$

The penultimate inequality is by Claim 3.6. Finally, since we were taking κ large enough, Claim 3.4 and a data-processing argument yields that

$$\mathbf{E}\left[Z_J^{\mathsf{B}} - Y_J^{\mathsf{B}}\right] \leq r\rho \tag{26}$$

We conclude that $\mathbf{E}\left[g(Y_{\leq J}, S) - Y_J^{\mathsf{B}}\right] \geq \frac{1}{400} \cdot \frac{1}{8\sqrt{r}} - (r+3)\rho > 1/(6400\sqrt{r})$, in contradiction to the assumed fairness of π.

□

3.3 Independence of Attack Decision

In this section, we prove Claim 3.9 (restated below).

Claim 3.16 (Claim 3.9, restated)**.** *Let* C *be a single-bit output* PPTM*. For* $\kappa \in \mathbb{N}$ *and* $\mathsf{P} \in \{\mathsf{A}, \mathsf{B}\}$*, let* $E_1^{\mathsf{P},\kappa}, \ldots, E_r^{\mathsf{P},\kappa}$ *be the sequence of random variables such that* $E_i^{\mathsf{P},\kappa}$ *is the indicator for the event*

$$\mathsf{P} \text{ sends the } i^{\text{th}} \text{ message in } \pi(1^\kappa) \wedge \; \mathsf{C}(Y_{\leq i}^\kappa, S^\kappa) = 1.$$

Assume io-key-agreement protocols do not exist. Then, for any $\mathsf{P} \in \{\mathsf{A}, \mathsf{B}\}$ *and infinite subset* $\mathcal{I}' \subseteq \mathcal{I}$*, there exists an infinite set* $\mathcal{I}'' \subseteq \mathcal{I}'$ *such that*

$$\mathbf{E}\left[E_{i+1}^{\mathsf{P},\kappa} \cdot (Z_i^{\overline{\mathsf{P}},\kappa} - Y_i^{\overline{\mathsf{P}},\kappa})\right] \in \pm 4r\rho$$

for every $\kappa \in \mathcal{I}''$ *and* $i \in (r-1)$*, where* $\overline{\mathsf{P}}$ *denotes (the party in)* $\{\mathsf{A}, \mathsf{B}\} \setminus \{\mathsf{P}\}$*.*

We prove for $\mathsf{P} = \mathsf{A}$. Consider the following variant of π in which the party playing A is outputting E_i^{A} and the party playing B is outputting its backup value.

Protocol 3.17 ($\widehat{\pi} = \left(\widehat{\mathsf{A}}, \widehat{\mathsf{B}}\right)$)**.**

Common input: security parameter 1^κ*.*

Description:

1. *Party* $\widehat{\mathsf{A}}$ *samples* $i \leftarrow [r]$ *and* $s \leftarrow S^\kappa$*, and sends them to* $\widehat{\mathsf{B}}$*.*
2. *The parties interact in the first* $i-1$ *rounds of a random execution of* $\pi(1^\kappa)$*, with* $\widehat{\mathsf{A}}$ *and* $\widehat{\mathsf{B}}$ *taking the role of* A *and* B *respectively.*

   Let $m_1, \ldots, m_{i-1}$ *be the messages, and let* z_{i-1}^{B} *be the* $(i-1)$ *backup output of* B *in the above execution.*
3. $\widehat{\mathsf{A}}$ *sets the value of* e_i^{A} *as follows:*

   If A *sends the* $i-1$ *message above, then it sets* $e_i^{\mathsf{A}} = 0$*.*

   Otherwise, it

   (a) *Continues the above execution of* π *to compute its next message* m_i*.*
   (b) *Computes* $y_i = \mathsf{F}(m_{\leq i}, s)$*.*
   (c) *Let* $e_i^{\mathsf{A}} = \mathsf{C}(y_{\leq i}, s)$*.*
4. $\widehat{\mathsf{A}}$ *outputs* e_i^{A} *and* B *outputs* z_{i-1}^{B}*.*

We apply the the following dichotomy result of Haitner et al. [16] on the above protocol.

Theorem 3.18 (Haitner et al. [16], Thm. 3.18, dichotomy of two-party protocols)**.** *Let Δ be an efficient single-bit output two-party protocol. Assume io-key-agreement protocol does not exits, then for any constant $\rho > 0$ and infinite subset $\mathcal{I} \subseteq \mathbb{N}$, there exists a* PPT *algorithm* Dcr *(decorelator) mapping transcripts of Δ into (the binary description of) pairs in $[0,1] \times [0,1]$ and an infinite set $\mathcal{I}' \in \mathbb{N}$, such that the following holds: let $O^{\mathsf{A},\kappa}$, $O^{\mathsf{B},\kappa}$ and T^κ denote the parties output and protocol transcript in a random execution of $\Delta(1^\kappa)$. Let $m(\kappa) \in \operatorname{poly}$ be a bound on the number of coins used by F on transcripts in $\operatorname{supp}(T^\kappa)$, and let S^κ be a uniform string of length $m(\kappa)$. Then*

$$(O^{\mathsf{A},\kappa}, O^{\mathsf{B},\kappa}, T^\kappa, S^\kappa) \overset{\mathrm{C}}{\approx}_{\rho,\mathcal{I}'} (U_{p^\mathsf{A}}, U_{p^\mathsf{A}}, T^\kappa, S^\kappa)_{(p^\mathsf{A},p^\mathsf{B})=\mathsf{Dcr}(T^\kappa;S^\kappa)}$$

letting U_p be a Boolean random variable taking the value 1 with probability p.

Proof of Claim 3.16. Assume io-key-agreement does not exits, and let $\mathcal{I}'' \subseteq \mathcal{I}'$ and a PPT Dcr be the infinite set and PPT decorrelator resulting by applying Theorem 3.18 with respect to protocol $\widehat{\pi}$ and ρ. Let $\widehat{S}^\kappa$ denote a long enough uniform string to be used by Dcr on transcripts of $\widehat{\pi}(1^\kappa)$. Then for $I \leftarrow (r-1)$, it holds that

$$(E^{\mathsf{A},\kappa}_{I+1}, Z^{\mathsf{B},\kappa}_I, M^\kappa_{\leq i}, S^\kappa, \widehat{S}^\kappa) \overset{\mathrm{C}}{\approx}_{\rho,\mathcal{I}''} (U_{p^\mathsf{A}}, U_{p^\mathsf{B}}, M^\kappa_{\leq I}, S^\kappa, \widehat{S}^\kappa)_{(p^\mathsf{A},p^\mathsf{B})=\mathsf{Dcr}(M_{\leq I},S^\kappa;\widehat{S}^\kappa)} \tag{27}$$

letting $\mathsf{Dcr}(m_{\leq i}, s; \widehat{s}) = \mathsf{Dcr}(i, s, m_{\leq i}; \widehat{s})$.

For $i \in [r]$, let $W^\kappa_i = (W^{\mathsf{A},\kappa}_i, W^{\mathsf{B},\kappa}_i) = \mathsf{Dcr}(M_{\leq i}, S^\kappa; \widehat{S}^\kappa)$. The proof of Claim 3.19 follows by the following three observations, proven below, that hold for large enough $\kappa \in \mathcal{I}''$.

Claim 3.19. $\mathbf{E}\left[E^{\mathsf{A},\kappa}_{I+1} \cdot Z^{\mathsf{B},\kappa}_I - W^{\mathsf{A},\kappa}_I \cdot W^{\mathsf{B},\kappa}_I\right] \in \pm\rho$.

Claim 3.20. $\mathbf{E}\left[W^{\mathsf{A},\kappa}_I \cdot Y^{\mathsf{B},\kappa}_I - E^{\mathsf{A},\kappa}_{I+1} \cdot Y^{\mathsf{B},\kappa}_I\right] \in \pm\rho$.

Claim 3.21. $\mathbf{E}\left[W^{\mathsf{A},\kappa}_I \cdot W^{\mathsf{B},\kappa}_I - W^{\mathsf{A},\kappa}_I \cdot Y^{\mathsf{B},\kappa}_I\right] \in \pm 2\rho$.

We conclude that $\mathbf{E}\left[E^{\mathsf{P},\kappa}_{I+1} \cdot Z^{\overline{\mathsf{P}},\kappa}_I - E^{\mathsf{P},\kappa}_{I+1} \cdot Y^{\overline{\mathsf{P}},\kappa}_I\right] \in \pm 4\rho$, and thus $\mathbf{E}\left[E^{\mathsf{P},\kappa}_{i+1} \cdot Z^{\overline{\mathsf{P}},\kappa}_i - E^{\mathsf{P},\kappa}_{i+1} \cdot Y^{\overline{\mathsf{P}},\kappa}_i\right] \in \pm 4r\rho$ for every $i \in (r-1)$. □

Proving Claim 3.19.

Proof of Claim 3.19. Consider algorithm D that on input $(z^\mathsf{A}, z^\mathsf{B}, \cdot)$, outputs $z^\mathsf{A} z^\mathsf{B}$. By definition,

1. $\Pr\left[\mathsf{D}(U_{W^{\mathsf{A},\kappa}_I}, U_{W^{\mathsf{B},\kappa}_I}, M^\kappa_{\leq I}, S^\kappa) = 1\right] = \mathbf{E}\left[U_{W^{\mathsf{A},\kappa}_I} \cdot U_{W^{\mathsf{B},\kappa}_I}\right] = \mathbf{E}\left[W^{\mathsf{A},\kappa}_I \cdot W^{\mathsf{B},\kappa}_I\right]$, and
2. $\Pr\left[\mathsf{D}(E^{\mathsf{A},\kappa}_{I+1}, Z^{\mathsf{B},\kappa}_I, M^\kappa_{\leq I}, S^\kappa) = 1\right] = \mathbf{E}\left[E^{\mathsf{A},\kappa}_{I+1} \cdot Z^{\mathsf{B},\kappa}_I\right]$.

Hence, the proof follows by Equation (27). □

Proving Claim 3.20.

Proof of Claim 3.20. Consider the algorithm D that on input $(z^{\mathsf{A}}, z^{\mathsf{B}}, (m_{\leq I}, s))$: (1) computes $(\cdot, y^{\mathsf{B}}) = \mathsf{F}(m_{\leq I}; s)$, (2) samples $u \leftarrow U_{y^{\mathsf{B}}}$, (3) outputs $z^{\mathsf{A}} \cdot u$. By definition,

1. $\Pr\left[\mathsf{D}(U_{W_I^{\mathsf{A},\kappa}}, U_{W_I^{\mathsf{B},\kappa}}, M_{\leq I}^{\kappa}, S^{\kappa}) = 1\right] = \mathbf{E}\left[U_{W_I^{\mathsf{A},\kappa}} \cdot U_{Y_I^{\mathsf{B},\kappa}}\right] = \mathbf{E}\left[W_I^{\mathsf{A},\kappa} \cdot Y_I^{\mathsf{B},\kappa}\right]$, and
2. $\Pr\left[\mathsf{D}(E_{I+1}^{\mathsf{A},\kappa}, Z_I^{\mathsf{B},\kappa}, M_{\leq I}^{\kappa}, S^{\kappa}) = 1\right] = \mathbf{E}\left[E_{I+1}^{\mathsf{A},\kappa} \cdot U_{Y_I^{\mathsf{B},\kappa}}\right] = \mathbf{E}\left[E_{I+1}^{\mathsf{A},\kappa} \cdot Y_I^{\mathsf{B},\kappa}\right]$.

Hence, also in this case the proof follows by Equation (27). ☐

Proving Claim 3.21.

Proof of Claim 3.21. Since $\left|W_I^{\mathsf{A},\kappa}\right| \leq 1$, it suffices to prove $\mathbf{E}\left[\left|W_I^{\mathsf{B},\kappa} - Y_I^{\mathsf{B},\kappa}\right|\right] \leq 2\rho$. We show that if $\mathbf{E}\left[\left|W_I^{\mathsf{B},\kappa} - Y_I^{\mathsf{B},\kappa}\right|\right] > 2\rho$, then there exists a distinguisher with advantage greater than ρ for either the real outputs of $\widehat{\pi}$ and the emulated outputs of Dcr, or, the real outputs of $\widetilde{\pi}$ and the emulated outputs of F, in contradiction with the assumed properties of Dcr and F.

Consider algorithm D that on input $(z^{\mathsf{A}}, z^{\mathsf{B}}, m_{\leq i}, s)$ acts as follows: (1) samples $\widehat{s} \leftarrow \widehat{S}^{\kappa}$, (2) computes $(\cdot, y^{\mathsf{B}}) = \mathsf{F}(m_{\leq i}; s)$ and $(\cdot, w^{\mathsf{B}}) = \mathsf{Dcr}(m_{\leq i}, s; \widehat{s})$, (3) outputs z^{B} if $w^{\mathsf{B}} \geq y^{\mathsf{B}}$, and $1 - z^{\mathsf{B}}$ otherwise. We compute the difference in probability that D outputs 1 given a sample from $\mathsf{Dcr}(M_{\leq I}^{\kappa})$ or a sample from $\mathsf{F}(M_{\leq I}^{\kappa})$ (we omit the superscript κ and subscript I below to reduce clutter)

$$
\begin{aligned}
&\Pr\left[\mathsf{D}(U_{W_I^{\mathsf{A},\kappa}}, U_{W_I^{\mathsf{B},\kappa}}, M_{\leq I}^{\kappa}, S^{\kappa}) = 1\right] - \Pr\left[\mathsf{D}(U_{Y_I^{\mathsf{A},\kappa}}, U_{Y_I^{\mathsf{B},\kappa}}, M_{\leq I}^{\kappa}, S^{\kappa}) = 1\right] \\
&= \mathbf{E}\left[U_{W^{\mathsf{B}}} \mid W^{\mathsf{B}} \geq Y^{\mathsf{B}}\right] \cdot \Pr\left[W^{\mathsf{B}} \geq Y^{\mathsf{B}}\right] + \mathbf{E}\left[1 - U_{W^{\mathsf{B}}} \mid W^{\mathsf{B}} < Y^{\mathsf{B}}\right] \cdot \Pr\left[W^{\mathsf{B}} < Y^{\mathsf{B}}\right] \\
&\quad - \mathbf{E}\left[U_{Y^{\mathsf{B}}} \mid W^{\mathsf{B}} \geq Y^{\mathsf{B}}\right] \cdot \Pr\left[W^{\mathsf{B}} \geq Y^{\mathsf{B}}\right] - \mathbf{E}\left[1 - U_{Y^{\mathsf{B}}} \mid W^{\mathsf{B}} < Y^{\mathsf{B}}\right] \cdot \Pr\left[W^{\mathsf{B}} < Y^{\mathsf{B}}\right] \\
&= \mathbf{E}\left[W^{\mathsf{B}} \mid W^{\mathsf{B}} \geq Y^{\mathsf{B}}\right] \cdot \Pr\left[W^{\mathsf{B}} \geq Y^{\mathsf{B}}\right] - \mathbf{E}\left[W^{\mathsf{B}} \mid W^{\mathsf{B}} < Y^{\mathsf{B}}\right] \Pr\left[W^{\mathsf{B}} < Y^{\mathsf{B}}\right] \\
&\quad - \mathbf{E}\left[Y^{\mathsf{B}} \mid W^{\mathsf{B}} \geq Y^{\mathsf{B}}\right] \cdot \Pr\left[W^{\mathsf{B}} \geq Y^{\mathsf{B}}\right] + \mathbf{E}\left[Y^{\mathsf{B}} \mid W^{\mathsf{B}} < Y^{\mathsf{B}}\right] \cdot \Pr\left[W^{\mathsf{B}} < Y^{\mathsf{B}}\right] \\
&= \mathbf{E}\left[W^{\mathsf{B}} - Y^{\mathsf{B}} \mid W^{\mathsf{B}} \geq Y^{\mathsf{B}}\right] \cdot \Pr\left[W^{\mathsf{B}} \geq Y^{\mathsf{B}}\right] + \mathbf{E}\left[-W^{\mathsf{B}} + Y^{\mathsf{B}} \mid W^{\mathsf{B}} < Y^{\mathsf{B}}\right] \Pr\left[W^{\mathsf{B}} < Y^{\mathsf{B}}\right] \\
&= \mathbf{E}\left[\left|W^{\mathsf{B}} - Y^{\mathsf{B}}\right|\right] \\
&> 2\rho.
\end{aligned}
$$

An averaging argument yields that either D is a distinguisher for $(U_{Y_I^{\mathsf{A},\kappa}}, U_{Y_I^{\mathsf{B},\kappa}}, M_{\leq I}^{\kappa}, S^{\kappa})$ and $(Z_I^{\mathsf{A},\kappa}, Z_I^{\mathsf{B},\kappa}, M_{\leq I}^{\kappa}, S^{\kappa})$ with advantage greater than ρ, in contradiction with Claim 3.4, or, D is a distinguisher for $(U_{W_I^{\mathsf{A},\kappa}}, U_{W_I^{\mathsf{B},\kappa}}, M_{\leq I}^{\kappa}, S^{\kappa})$ and $(E_I^{\mathsf{A},\kappa}, Z_I^{\mathsf{B},\kappa}, M_{\leq I}^{\kappa}, S^{\kappa})$ with advantage greater than ρ, in contradiction with Equation (27). ☐

References

[1] B. Alon and E. Omri. Almost-optimally fair multiparty coin-tossing with nearly three-quarters malicious. In *Proceedings of the 14th Theory of Cryptography Conference, TCC 2016-B, part I*, pages 307–335, 2016.

[2] B. Averbuch, M. Blum, B. Chor, S. Goldwasser, and S. Micali. How to implement Bracha's $O(\log n)$ Byzantine agreement algorithm, 1985. Unpublished manuscript.

[3] B. Awerbuch, M. Blum, B. Chor, S. Goldwasser, and S. Micali. How to implement Bracha's $O(\log n)$ byzantine agreement algorithm. Unpublished manuscript, 1985.

[4] A. Beimel, E. Omri, and I. Orlov. Protocols for multiparty coin toss with a dishonest majority. *Journal of Cryptology*, 28(3):551–600, 2015.

[5] A. Beimel, I. Haitner, N. Makriyannis, and E. Omri. Tighter bounds on multi-party coin flipping, via augmented weak martingales and differentially private sampling. Technical Report TR17-168, Electronic Colloquium on Computational Complexity, 2017.

[6] I. Berman, I. Haitner, and A. Tentes. Coin flipping of any constant bias implies one-way functions. *Journal of the ACM*, 65(3):14, 2018.

[7] M. Blum. How to exchange (secret) keys. *ACM Transactions on Computer Systems*, 1983.

[8] N. Buchbinder, I. Haitner, N. Levi, and E. Tsfadia. Fair coin flipping: Tighter analysis and the many-party case. In *Proceedings of the 28th Annual ACM-SIAM Symposium on Discrete Algorithms (SODA)*, pages 2580–2600, 2017.

[9] R. Cleve. Limits on the security of coin flips when half the processors are faulty. In *Proceedings of the 18th Annual ACM Symposium on Theory of Computing (STOC)*, pages 364–369, 1986.

[10] R. Cleve and R. Impagliazzo. Martingales, collective coin flipping and discrete control processes (extended abstract). http://citeseerx.ist.psu.edu/viewdoc/summary?doi=10.1.1.51.1797, 1993.

[11] D. Dachman-Soled, Y. Lindell, M. Mahmoody, and T. Malkin. On the black-box complexity of optimally-fair coin tossing. In *Proceedings of the 8th Theory of Cryptography Conference, TCC 2011*, volume 6597, pages 450–467, 2011.

[12] D. Dachman-Soled, M. Mahmoody, and T. Malkin. Can optimally-fair coin tossing be based on one-way functions? In Y. Lindell, editor, *Theory of Cryptography - 11th Theory of Cryptography Conference, TCC 2014*, volume 8349 of *Lecture Notes in Computer Science*, pages 217–239. Springer, 2014.

[13] I. Haitner and E. Omri. Coin flipping with constant bias implies one-way functions. *SIAM Journal on Computing*, 43(2):389–409, 2014.

[14] I. Haitner and E. Tsfadia. An almost-optimally fair three-party coin-flipping protocol. In *Proceedings of the 46th Annual ACM Symposium on Theory of Computing (STOC)*, pages 817–836, 2014.

[15] I. Haitner and E. Tsfadia. An almost-optimally fair three-party coin-flipping protocol. *SIAM J. Comput.*, 46(2):479–542, 2017.

[16] I. Haitner, K. Nissim, E. Omri, R. Shaltiel, and J. Silbak. Computational two-party correlation. Technical Report TR18-071, Electronic Colloquium on Computational Complexity, 2018.

[17] W. Hoeffding. Probability inequalities for sums of bounded random variables. *Journal of the American Statistical Association*, pages 13–30, 1963.

[18] R. Impagliazzo and M. Luby. One-way functions are essential for complexity based cryptography. In *Proceedings of the 30th Annual Symposium on Foundations of Computer Science (FOCS)*, pages 230–235, 1989.

[19] H. K. Maji, M. Prabhakaran, and A. Sahai. On the computational complexity of coin flipping. In *Proceedings of the 51st Annual Symposium on Foundations of Computer Science (FOCS)*, pages 613–622, 2010.

[20] T. Moran, M. Naor, and G. Segev. An optimally fair coin toss. In *Proceedings of the 6th Theory of Cryptography Conference, TCC 2009*, pages 1–18, 2009.

[21] T. Moran, M. Naor, and G. Segev. An optimally fair coin toss. *Journal of Cryptology*, 29(3): 491–513, 2016.

ECCC ISSN 1433-8092

https://eccc.weizmann.ac.il