On Oblivious Amplification of Coin-Tossing Protocols*

Nir Bitansky† Nathan Geier‡

November 13, 2019

Abstract

We consider the problem of amplifying two-party coin-tossing protocols: given a protocol where it is possible to bias the common output by at most ρ, we aim to obtain a new protocol where the output can be biased by at most $\rho^\star < \rho$. We rule out the existence of a natural type of amplifiers called *oblivious amplifiers* for every $\rho^\star < \rho$. Such amplifiers ignore the way that the underlying ρ-bias protocol works and can only invoke an oracle that provides ρ-bias bits.

We provide two proofs of this impossibility. The first is by a reduction to the impossibility of deterministic randomness extraction from Santha-Vazirani sources. The second is a direct proof that is more general and also rules outs certain types of asymmetric amplification. In addition, it gives yet another proof for the Santha-Vazirani impossibility.

*This work was supported in part by ISF grant 18/484, and by Len Blavatnik and the Blavatnik Family Foundation.

†Tel Aviv University, nirbitan@tau.ac.il. Member of the Check Point Institute of Information Security. Supported also by the Alon Young Faculty Fellowship.

‡Tel Aviv University, nathangeier@mail.tau.ac.il.

1 Introduction

Hardness amplification is a foundational problem in cryptography: given a cryptographic primitive that is *weakly secure* in some sense, we would like to make it *strongly secure*. Two famous examples of such amplification procedures are Yao's amplification of weak one-way functions into strong ones and Yao's Xor lemma for amplifying unpredictability [Yao82]. Other examples of primitives with known amplification procedures include public-key encryption [DNR04, LT13], oblivious transfer [DKS99, Wul08], commitments [DKS99, HR08] and cryptographic arguments [Hai13].

Coin-Tossing Protocols. We consider amplification of coin-tossing protocols [Blu81]. Such protocols allow two parties to jointly toss an unbiased coin, such that even if one party is malicious and diverges from the protocol, it cannot significantly bias the outcome. Concretely, a *ρ-bias coin-tossing protocol* is such that a cheating party cannot force the common outcome to be any specific bit b with probability greater than $1/2 + \rho$.

It is common knowledge that coin-tossing protocols against unbounded (or even PSPACE) adversaries cannot have bias $\rho < 1/2$. Accordingly, the common definition addresses efficient adversaries. Here coin-tossing protocols with negligible bias have been long known assuming one-way functions [Blu81, HILL99, Nao91]. In fact, a long line of work shows that one-way functions are, in fact, necessary provided that the bias is at most $\rho < 1/2 - \Omega(1)$ [IL89, MPS10, HO14, BHT18]. Whether one-way functions are necessary for any non-trivial bias $\rho < 1/2 - 1/\mathrm{poly}(n)$ (or even for some $\rho = 1/2 - o(1)$) remains an open problem.

Amplifying Coin Tossing. A coin-tossing amplifier should take a coin-tossing protocol π with (non-trivial) bias $\rho < 1/2$ and transform it into a new coin-tossing protocol $\pi^\star$ with smaller bias $\rho^\star < \rho$. One specific way for obtaining coin-tossing amplifiers is to first derive from the protocol π a one-way function f_π, and then construct optimal coin tossing from f_π. Indeed, following the known results mentioned above this would work for any

$$\mathrm{negl}(n) < \rho^\star < \rho < 1/2 - \Omega(1) \ .$$

In this work, we ask whether there exist "more direct" amplification procedures, which we call *oblivious amplifiers*. Such amplifiers completely ignore the way that the underlying protocol π works, they only obtain oracle access to the result — the adversary can adaptively bias the resulting bit of each oracle invocation as long as the bias is bounded by ρ. The amplifier is required to satisfy an information theoretic guarantee: *unbounded* attackers cannot bias the common output of the new protocol π^* by more than $\rho^* < \rho$. Addressing unbounded attackers is a natural choice as we wish to avoid computational assumptions (certainly, one-way functions, which would trivialize the problem).

We find the model of oblivious amplifiers quite natural and similar to other settings, such as Yao's Xor lemma [Yao82], where amplifiers satisfy an information theoretic guarantee. In particular, constructing such amplifiers seems like a natural route toward fully understanding the complexity of coin tossing. In particular, the existence of such efficient amplifiers for

$$\rho^\star \leq 1/2 - \Omega(1) < 1/2 - 1/\mathrm{poly}(n) \leq \rho$$

would completely resolve the question, showing that any non-trivial coin tossing is equivalent to one-way functions.

1.1 Results

We show that oblivious coin-tossing amplifiers do not exist.

Theorem 1.1 (Informal)**.** *There do not exist oblivious coin-tossing amplifiers for any $\rho^* < \rho$.*

Our theorem can further be extended to rule out oblivious amplification of *weak coin tossing* [BHT18] that essentially requires that one side cannot bias toward 0 and the other cannot bias toward 1. (See Remark 2.2.)

A More General Lower Bound. We give two proofs of the above theorem. The first is by a simple reduction to the impossibility of *deterministic randomness extraction from Santha-Vazirani sources* [SV86]. We also give a direct proof that provides a more general lower bound. In particular, we quantify the tradeoff between improving the potential bias caused by specific party A toward a specific bit b, at the account of making it worst for other party B and bit $1 - b$. This also rules out oblivious amplification for asymmetric notions of coin tossing where we allow different bounds on the bias for the two parties. The tradeoff is explained in the technical overview below and expressed in Figure 2.

The alternative proof also gives yet another proof of the Santha-Vazirani impossibility for deterministic extraction (in addition to several existing proofs [SV86, RVW04, ACM+17]).

1.2 Technical Overview

We now give a brief overview of our proofs.

Modeling Protocols as Trees. We model any possible amplifier protocol $\pi^\star$ as a full binary tree. The correspondence is natural: the root corresponds to the beginning of the protocol before any message is sent. Every inner node in the tree is either

- Controlled by one of the two parties A or B, meaning that it is this party's turn to send a message, without loss of generality, a single bit.
- Representing an oracle call to the underlying protocol π resulting in a common bit.

Whenever a message is sent or an oracle call is made, we move in accordance to the left or right child, until reaching a leaf labeled by the common outcome of the protocol. Here the execution ends.

Each node is associated with some (honest) distribution on the next bit to be sent or produced by the oracle. The adversary can gain control over the nodes representing one of the parties and arbitrarily fix their distributions. For any oracle node, the adversary can fix an arbitrary distribution provided that it has bias at most ρ. Note that every node corresponds to some partial execution of the protocol, represented by the path from the root to this node, and the adversarial response is adaptively fixed according to this path. (See illustration in Figure 1.)

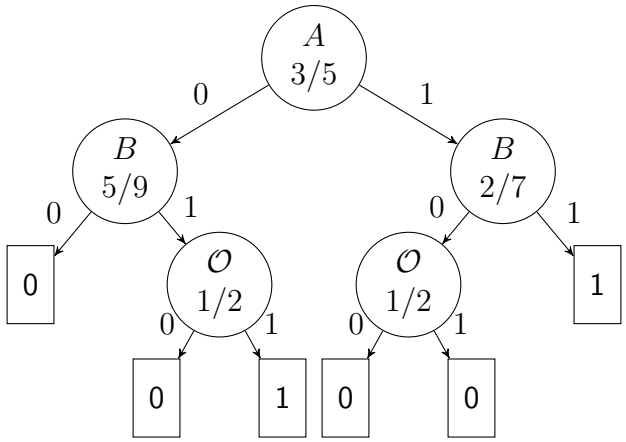

Figure 1: An example of a protocol represented by a tree. Each node specifies which party should send a bit as well as the distribution of this bit (when the party is honest). Alternatively, the parties may invoke the oracle and obtain a joint bit. The adversary may take control of one of the parties and arbitrarily replace its distributions; it can also replace the oracle call distributions provided that they remain ρ-biased.

The above model does not explicitly capture protocols where honest parties may store private coins through the interaction. However, we can transform any private-coin protocol to a protocol in the above model by considering parties that do not explicitly keep private randomness; instead, in every node they resample their private randomness conditioned on the execution transcript so far and answer according to that string. The transformation has no effect on the distribution of transcript and in particular on the outcome bit. However, the transformation may make the new protocol inefficient. This is not an issue as our lower bound will in fact hold for inefficient protocols as well.

A Reduction to Deterministic Santha-Vazirani Extractors. In their seminal work, Santha and Vazirani considered the problem of extracting a statistically uniform bit from a sequence of biased random bits $\mathbf{X} = X_1, \ldots, X_n$ such that X_i has bounded bias ρ conditioned on any fixing of $X_1, \ldots, X_{i-1}$. Such random sources $\mathbf{X}$ are called Santha-Vazirani Sources. They proved that there exist no deterministic extractors for such sources, namely, for any deterministic Boolean function E, there exists a Santha-Vazirani source such that $E(\mathbf{X})$ is far from uniform. In fact, they proved that $E(\mathbf{X})$ is ρ-biased.

Indeed, this bares similarity to the setting of oblivious coin-tossing amplification where the ρ-biased input bits naturally correspond to oracle calls to π. The gap between the two models is that a coin-tossing oblivious amplifier further allows interaction between two *randomized* parties. To bridge this gap, we prove that any oblivious amplifier can be turned into a non-interactive oblivious amplifier.

To convey the basic idea, imagine that we have an amplifier tree where the root belongs to party A (namely, it sends the first message). We argue that at least one of its left or right subtrees also has bias at most $\rho^\star$, and thus we can reduce one round of interaction. In fact, *both* subtrees must be such that an adversary controlling A can bias the outcome by at most $\rho^\star$, or else an adversary controlling A in the original protocol could have simply always

chosen the worse subtree in order to bias the outcome by more than $\rho^\star$. We then observe that at least one of the subtrees must be such that an adversary controlling B cannot bias by more than $\rho^\star$. Indeed, the bias caused by an adversary controlling B in the original protocol is just a convex combination of the its bias in the two corresponding subtrees.

The Second Proof and a More General Lower Bound. Our second proof of the theorem pours more light on the tradeoff between how much each party can bias the protocol towards a specific bit. We first give a correspondence between protocols and points on the plane. For a protocol π, let x_π be the maximum probability a malicious A^* (controlling A) can force output 0 and y_π be the maximum probability a malicious B^* can force output 1. We assign the protocol π coordinates (x_π, y_π). Considering the tree representing the protocol π, the point in the plane corresponding to the protocol π is determined by the points corresponding to its children themselves as protocols, as well as the operation at its root. (This is roughly done following a similar argument to the one used above, where we either take the maximum of two subtrees, or a convex combination thereof.)

Our basic Theorem 1.1 and its proof show that all points corresponding to protocols lie outside the axis-aligned square given by $x = 0$, $y = 0$, $x = 0.5 + \rho$, $y = 0.5 + \rho$. Our more general analysis, shows a more accurate picture — all points lie above a certain piecewise linear f (See Figure 2). In a bit more detail, we prove by induction on the depth of the tree that the point corresponding to every protocol lies above the function f, by showing closure of points above f to each of the operations that may appear at the root, and also that protocol leaves (the base case) lie above f. When considering a non-leaf protocol, we get from the induction that the sub-protocols rooted at its children lie above f, and since the set of points above f is closed under the operation at the protocols' root, the point corresponding to the protocol itself also lies above f.

This more general result shows that we cannot hope to improve the x_π-value of the oracle without strictly hurting its y_π-value, and furthermore gives some lower bound $g(\Delta_x)$, for every $0 \leq \Delta_x \leq x_\pi$, on how much y_π must increase in order to reduce x_π by Δ_x.

Organization

In Section 2, we define the relevant notion of oblivious amplifiers and their correspondence to trees. In Section 3, we prove the main impossibility result by a reduction to the impossibility of deterministic Santha-Vazirani extraction. In Section 4, we give our alternative (direct) proof leading to a more general lower bound.

2 Definitions

Throughout the paper, we denote by $Ber(p)$ the Bernoulli distribution with parameter p, namely, the value 1 gets probability p and 0 gets $1 - p$.

Definition 2.1 (Common-output two-party protocol with oracle access to ρ-biased bits)**.** We model a common-output two-party protocol with oracle access to ρ-biased bits using full binary trees such that:

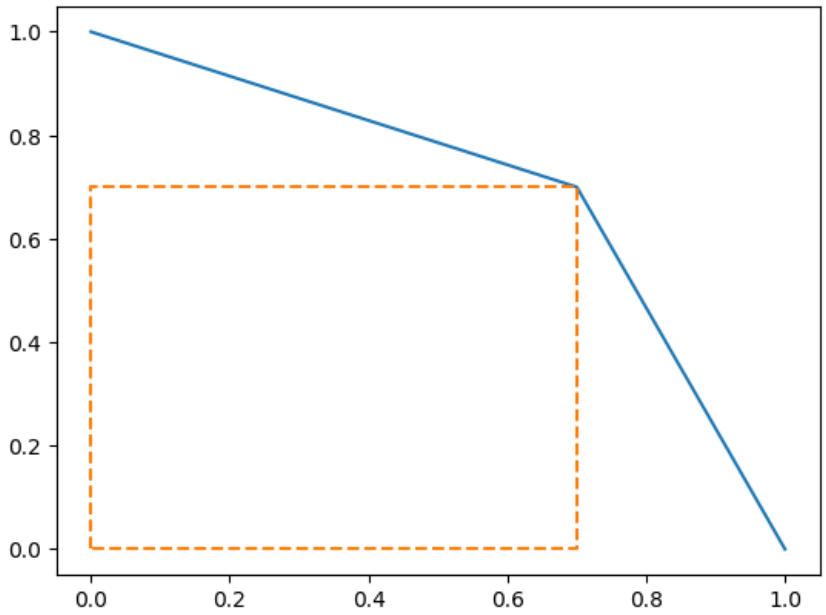

Figure 2: For the sake of this illustration, we set $\rho = 0.2$. The X-axis represents the ability of a malicious A^* to bias the common output toward 0, and the Y-axis represents the ability of a malicious B^* to bias the common output toward 1. While our basic Theorem 1.1 argues that all protocols lie outside the dotted square, our more general theorem shows that they in fact lie above the piece-wise linear function represented by the continuous line.

- Every leaf is labeled with either "0" or "1", which represent the common output of the protocol upon reaching that leaf.
- Every inner node v is labeled with "(P_v, q_v)" where
  1. $P_v \in \{A, B, \mathcal{O}\}$ represents whether it is A's turn to speak, B's turn to speak, or an oracle call.
  2. In an honest execution, the bit sent after reaching that node should be distributed according to $Ber(q_v)$. If $P_v = \mathcal{O}$ then q_v must be 0.5, otherwise q_v may be an arbitrary probability.

Execution. An execution of the protocol follows a path along the tree, starting at the root and going left/right in accordance to the bits sent, when 0 means left and 1 means right, until reaching a leaf which determines the common output of that execution.

Adversarial behavior. A malicious party can ignore q_v when it is their turn to speak and send an arbitrarily distributed bit instead. Also, whenever the oracle is called, the malicious party may change the output distribution of that call from $Ber(0.5)$ to $Ber(q_v)$ for any q_v such that $|q_v - 0.5| \leq \rho$. Note that q_v may depend on all the bits sent up to that call

Definition 2.2. Given a protocol π, A^* an adversarial behavior of A and $b \in \{0, 1\}$, we denote by $\pi(A^*, b)$ the probability of π's output being b when executed with A^* and an

honest B. Next, define $Opt(\pi, A^*, b) := \max_{A^*} \pi(A^*, b)$, where the maximum is taken over all adversarial behaviors A^* of A. $\pi(B^*, b)$ and $Opt(\pi, B^*, b)$ are defined symmetrically for B.

Definition 2.3 ((α, β)-oblivious coin-tossing amplifier)**.** A common-output two-party protocol π with oracle access to α-biased bits is an (α, β)-oblivious coin-tossing amplifier if

$$\max_{P^*, b} Opt(\pi, P^*, b) \leq 0.5 + \beta \ ,$$

where $P^* \in \{A^*, B^*\}$ and $b \in \{0, 1\}$. In addition, we require that the expected output of π over an honest execution is 0.5.

Remark 2.1. Actually, for the lower bound, we do not use the fact that the expected output of π over an honest execution is 0.5. We note that bounded biases already makes coin-tossing protocols non-trivial.

Remark 2.2. We may also consider the notion of (α, β)-oblivious weak coin-tossing amplifiers, which aim to produce a β-bias weak coin tossing (CT) protocol instead of standard β-CT. In the weak CT setting, we know in advance that A^* aims to bias the result toward 0 while B^* aims to bias the result toward 1. We only need to bound their ability to bias the output in their chosen direction. The difference is in the requirement

$$\max_{P^*, b} Opt(\pi, P^*, b) \leq 0.5 + \beta \ ,$$

which is replaced with

$$\max\{Opt(\pi, A^*, 0), Opt(\pi, B^*, 1)\} \leq 0.5 + \beta \ .$$

3 Impossibility of Oblivious Amplifiers

In this section, we state and prove our main result regarding the impossibility of non-trivial oblivious coin-tossing amplifiers.

Theorem 3.1 (main)**.** *For every common-output two-party protocol π with oracle access to ρ-biased bits, either $Opt(\pi, A^*, 0) \geq 0.5 + \rho$ or $Opt(\pi, B^*, 1) \geq 0.5 + \rho$. In particular, there do not exist (α, β)-oblivious coin-tossing amplifier, for any $\beta < \alpha$.*

Toward proving the theorem, we first analyze in Section 3.1 the best possible (unbounded) attacks. Then in Section 3.2, we give the reduction to the impossibility of deterministic extraction from Santha-Vazirani sources.

3.1 The Optimal Attacks

Let π be a common-output two-party protocol with oracle access to ρ-biased bits, and denote its root by r. Also, denote by π_0 and π_1 the protocols rooted at the left and right children of r. The best strategy of a malicious A^* to make the protocol π output 0 is to bias the first bit b as much as possible towards $\arg\max_{z \in \{0,1\}} Opt(\pi_z, A^*, 0)$, and after b is sent continue to recursively apply the best strategy at the resulting child. The best strategy of a malicious B^* to make the protocol π output 1 is symmetrical. Thus, we have that:

- If r is labeled with "A, p", then A can completely bias the bit while B cannot change its distribution at all, so

$$Opt(\pi, A^*, 0) = \max_{z\in\{0,1\}} Opt(\pi_z, A^*, 0)$$
$$Opt(\pi, B^*, 1) = \mathbb{E}_{z\sim Ber(p)}\left[Opt(\pi_z, B^*, 1)\right]$$

- If r is labeled with "B, p", then B can completely bias the bit while A cannot change its distribution at all, so

$$Opt(\pi, A^*, 0) = \mathbb{E}_{z\sim Ber(p)}\left[Opt(\pi_z, A^*, 0)\right]$$
$$Opt(\pi, B^*, 1) = \max_{z\in\{0,1\}} Opt(\pi_z, B^*, 1)$$

- If r is labeled with "$\mathcal{O}$", then both A and B can bias the bit by at most ρ, so

$$Opt(\pi, A^*, 0) = (0.5-\rho)\cdot \min_{z\in\{0,1\}} Opt_z(\pi, A^*, 0) + (0.5+\rho)\cdot \max_{z\in\{0,1\}} Opt(\pi_z, A^*, 0)$$
$$Opt(\pi, B^*, 1) = (0.5-\rho)\cdot \min_{z\in\{0,1\}} Opt(\pi_z, B^*, 1) + (0.5+\rho)\cdot \max_{z\in\{0,1\}} Opt(\pi_z, B^*, 1)$$

- If r is a leaf labeled with "0", then

$$Opt(\pi, A^*, 0) = 1$$
$$Opt(\pi, B^*, 1) = 0$$

- If r is a leaf labeled with "1", then

$$Opt(\pi, A^*, 0) = 0$$
$$Opt(\pi, B^*, 1) = 1$$

3.2 Reduction to Deterministic Santha-Vazirani Extraction

The following theorem states that allowing interaction does not help in creating more secure protocols, in the sense that it does not allow us to generate a better protocol, in either $Opt(\pi, A^*, 0)$ or $Opt(\pi, B^*, 1)$, than protocols already existing without interaction.

Theorem 3.2. *For every common-output two-party protocol π with oracle access to ρ-biased bits, there exists a common-output two-party protocol π' with oracle access to ρ-biased bits, with no inner nodes labeled by either "A, p" or "B, p", and such that $Opt(\pi', A^*, 0) \leq Opt(\pi, A^*, 0)$ and $Opt(\pi', B^*, 1) \leq Opt(\pi, B^*, 1)$*

In other words, allowing interaction does not help in producing protocols where it is harder for A^* to cheat towards 0 or for B^* to cheat towards 1.

Proof. We show how given any protocol π with "A, p"/"B, p" turns, we can strictly reduce its number of "A, p"/"B, p" turns without increasing its cheating probabilities $Opt(\pi, A^*, 0)$ and $Opt(\pi, B^*, 1)$. In essence, we show that for every inner node labeled with "A, p"/"B, p" there exists a child such that replacing the subtree rooted at that node with the subtree rooted at its child results a protocol that is only harder to cheat in, for both A^* towards 0 and B^* towards 1. In more detail: First, we pick some arbitrary inner node of π labeled with either "A, p" or "B, p". Denote this node by N, and its left and right children by N_0 and N_1, respectively. If N is labeled by "A, p" we choose $z := \arg\min_{b\in\{0,1\}} Opt(N_b, B^*, 1)$, otherwise choose $z := \arg\min_{b\in\{0,1\}} Opt(N_b, A^*, 0)$. We take π and replace in it the subtree rooted at N with the subtree rooted at N_z. Essentially, we fixed the bit sent at node N to always be z, instead of letting A/B choose it. We make two observations:

1. Both $Opt(N_z, A^*, 0) \leq Opt(N, A^*, 0)$ and $Opt(N_z, B^*, 1) \leq Opt(N, B^*, 1)$.
   To see this, assume w.l.o.g that N is labeled with "A, p" (symmetrical argument for "B, p"). We have that $Opt(N, A^*, 0) = \max_{b\in\{0,1\}} Opt(N_b, A^*, 0)$ and in particular $Opt(N, A^*, 0) \geq Opt(N_z, A^*, 0)$. Also, since we have that

   $$Opt(N, B^*, 1) = \mathbb{E}_{b\sim Ber(p)}\left[Opt(N_b, B^*, 1)\right]$$

   and we chose $z := \arg\min_{b\in\{0,1\}} Opt(N_b, B^*, 1)$ in case N is labeled with "A, p", then $Opt(N, B^*, 1) \geq Opt(N_z, B^*, 1)$.
   In other words, at N_z it is both harder for A^* to cheat towards 0 and for B^* to cheat towards 1.

2. Let π and T be protocols, N be some node of π, and π' be the protocol resulted by replacing in π the subtree rooted at N with T. If $Opt(T, A^*, 0) \leq Opt(N, A^*, 0)$ then $Opt(\pi', A^*, 0) \leq Opt(\pi, A^*, 0)$ and similarly if $Opt(T, B^*, 1) \leq Opt(N, B^*, 1)$ then $Opt(\pi', B^*, 1) \leq Opt(\pi, B^*, 1)$. The reason for this is that the value $Opt(R, A^*, 0)$ of an inner node R, for all three operations ("A, p","B, p","$\mathcal{O}$"), is a monotone function of the values $Opt(R_0, A^*, 0), Opt(R_1, A^*, 0)$ of its children. By not increasing the $(A^*, 0)$-value of some node, you cannot increase the $(A^*, 0)$-value of its father, which in turn will not increase the $(A^*, 0)$-value of its father and so on.
   In other words, by taking a protocol and replacing the sub-protocol run at some partial transcript with another sub-protocol where it is not easier for A^*/B^* to cheat towards $0/1$, we result a protocol where it is not easier for A^*/B^* to cheat towards $0/1$.

Combining these observations, we reach the conclusion that by taking a protocol and repeatedly getting rid of its "A, p"/"B, p" nodes by replacing the subtree rooted at them with the subtree rooted at a correctly chosen child of theirs, we can get rid of all the "A, p"/"B, p" turns without increasing the cheating probabilities of A^* towards 0 and of B^* towards 1. □

Definition 3.1 (SV Source)**.** For $0 \leq \rho \leq 0.5$, a source $X = X_1, \ldots, X_n$ of length n (A random variable taking values in $\{0,1\}^n$) is a Santha-Vazirani (SV) source with bias ρ if for every $i \in [n]$ and every $x_1, \ldots, x_{i-1} \in \{0,1\}$, the bias of X_i conditioned on $X_1 = x_1, \ldots, X_{i-1} = x_{i-1}$ is at most ρ. That is,

$$\left|\mathbb{E}\left[X_i | X_1 = x_1, \ldots, X_{i-1} = x_{i-1}\right] - 0.5\right| \leq \rho \ .$$

Theorem 3.3 ([SV86], [RVW04]). *For every function $f : \{0,1\}^n \to \{0,1\}$ and every $0 \leq \rho \leq 0.5$ there exists an SV source $X = X_1, \ldots, X_n$ of length n and bias ρ such that $f(X)$ has bias at least ρ.*

We remark that this version of the statement is taken from Reingold, Vadhan and Wigderson [RVW04], and that differently from them we refer to the bias as the distance from 0.5 and not double that amount.

The following theorem implies that there are no non-trivial oblivious coin-tossing amplifiers comprised solely of oracle calls.

Theorem 3.4. *For every common-output two-party protocol π with oracle access to ρ-biased bits and no inner nodes labeled by either "A, p" or "B, p", either $Opt(\pi, A^*, 0) \geq 0.5 + \rho$ or $Opt(\pi, B^*, 1) \geq 0.5 + \rho$*

In fact, in a protocol with no "A, p"/"B, p" turns there is no longer a difference between A and B so either $Opt(\pi, A^*, 0) = Opt(\pi, B^*, 0) \geq 0.5 + \rho$ or $Opt(\pi, A^*, 1) = Opt(\pi, B^*, 1) \geq 0.5 + \rho$

Proof. Let n be an upper bound on the number of oracle calls made by π in any execution. We can think w.l.o.g of π as always making exactly n oracle calls, with results denoted by $o_1, \ldots, o_n$, and then outputting $f(o_1, \ldots, o_n)$ for some function f. The impossibility result of deterministic extraction from Santha-Vazirani sources, Theorem 3.3, guarantees that there exists an SV source $X = X_1, \ldots, X_n$ of length n and bias ρ such that $f(X)$ has bias at least ρ. Therefore, if the malicious party can make sure the oracle calls' output distribution is X, they can succeed in biasing the output of π by at least ρ. (If it is towards 0 then $Opt(\pi, A^*, 0) \geq 0.5 + \rho$ and if it is towards 1 then $Opt(\pi, B^*, 1) \geq 0.5 + \rho$). The malicious party can make sure the oracle calls' output distribution is X by doing it bit-by-bit: After $0 \leq i \leq n-1$ calls with results $x_1, \ldots, x_i$ they ask the next oracle call to be distributed according to $X_{i+1} | X_1 = x_1, \ldots, X_i = x_i$, and their request from the ρ-biased bit oracle is legal by the definition of SV sources with bias ρ. □

Combining Theorem 3.2 and Theorem 3.4, Theorem 3.1 follows.

4 A More General Lower Bound

In this section, we give a direct proof by induction which gives some insight on the possible relations between $Opt(\pi, A^*, 0)$ and $Opt(\pi, B^*, 1)$ and the trade-offs we can get between them.

We correspond every protocol with a point on the plane using $Opt(\pi, A^*, 0)$ as the x-coordinate and $Opt(\pi, B^*, 1)$ as the y-coordinate. Our main theorem translates into showing that all protocol points lie outside or on the axis-aligned square with boundaries $x = 0.5 + \rho, x = 0$ and $y = 0.5 + \rho, y = 0$, but still inside of the first (i.e., $(+,+)$) quadrant (the second part is obvious from the definition). The area outside the square (but still inside the first quadrant) is not closed under the oracle-call operation (defined later), and thus the naïve induction attempt fails. Instead, we will strengthen the induction hypothesis into showing

that all points lie above $f := \min\left(1 - \frac{0.5-\rho}{0.5+\rho} \cdot x, \frac{0.5+\rho}{0.5-\rho} - \frac{0.5+\rho}{0.5-\rho} \cdot x\right)$, which is both closed under all operations (now the induction works) and lies above the square (so it implies what we want, and more). See Figure 2 for a graphic representation of the square and f.

Theorem 4.1. *For every common-output two-party protocol π with oracle access to ρ-biased bits, we have that* $Opt(\pi, B^*, 1) \geq \min(1 - \frac{0.5-\rho}{0.5+\rho} \cdot Opt(\pi, A^*, 0),\ \frac{0.5+\rho}{0.5-\rho} - \frac{0.5+\rho}{0.5-\rho} \cdot Opt(\pi, A^*, 0))$

Equivalently, if we take ℓ_1 to be the line between $(0, 1)$ and $(0.5 + \rho, 0.5 + \rho)$

$$\text{so } \ell_1 \text{ is } y = 1 - \frac{0.5 - \rho}{0.5 + \rho} \cdot x$$

and ℓ_2 to be the line between $(0.5 + \rho, 0.5 + \rho)$ and $(1, 0)$

$$\text{so } \ell_2 \text{ is } y = \frac{0.5 + \rho}{0.5 - \rho} - \frac{0.5 + \rho}{0.5 - \rho} \cdot x$$

then the point $(Opt(\pi, A^*, 0), Opt(\pi, B^*, 1))$ lies above (not strictly) at least one of ℓ_1, ℓ_2.

Corollary 4.1. *Theorem 3.1 follows*

Proof. Since both ℓ_1 and ℓ_2 are strictly decreasing (negative slope) and pass through the point $(0.5 + \rho, 0.5 + \rho)$, then $x < 0.5+\rho$ implies $\ell_1(x), \ell_2(x) > 0.5+\rho$. Thus $Opt(\pi, A^*, 0) < 0.5+\rho$ implies $Opt(\pi, B^*, 1) \geq \min\left(\ell_1(Opt(\pi, A^*, 0)), \ell_2(Opt(\pi, A^*, 0))\right) > 0.5 + \rho$ □

Proof of Theorem 4.1. Proof by induction on the depth.
Leaves labeled with "0" lie on ℓ_2 and leaves labeled with "1" lie on ℓ_1.
If π is not a leaf, denote its root by r, by π_0 and π_1 the protocols rooted at the left and right children of r, and let $f := \min(\ell_1, \ell_2)$. Also, denote

$$\begin{aligned} x_0, y_0 &= Opt(\pi_0, A^*, 0), Opt(\pi_0, B^*, 1) \\ x_1, y_1 &= Opt(\pi_1, A^*, 0), Opt(\pi_1, B^*, 1) \\ x', y' &= Opt(\pi, A^*, 0), Opt(\pi, B^*, 1) \end{aligned}$$

We assume (induction hypothesis) that $y_0 \geq f(x_0)$, $y_1 \geq f(x_1)$ and want to show that $y' \geq f(x')$. There are three types of operations to consider:

- If r is labeled with "A, p"

$$\begin{pmatrix} x' \\ y' \end{pmatrix} = \begin{pmatrix} \max(x_0, x_1) \\ (1-p) \cdot y_0 + p \cdot y_1 \end{pmatrix}$$

- If r is labeled with "B, p"

$$\begin{pmatrix} x' \\ y' \end{pmatrix} = \begin{pmatrix} (1-p) \cdot x_0 + p \cdot x_1 \\ \max(y_0, y_1) \end{pmatrix}$$

- If r is labeled with "$\mathcal{O}$"

$$\begin{pmatrix} x' \\ y' \end{pmatrix} = \begin{pmatrix} (0.5-\rho) \cdot \min(x_0, x_1) + (0.5+\rho) \cdot \max(x_0, x_1) \\ (0.5-\rho) \cdot \min(y_0, y_1) + (0.5+\rho) \cdot \max(y_0, y_1) \end{pmatrix}$$

We may assume without loss of generality that the points $(x_0, y_0), (x_1, y_1)$ lie on f and not strictly above it, namely, that $y_0 = f(x_0)$, $y_1 = f(x_1)$. This is true because y' is non-decreasing in y_0, y_1 in all three operations, so if $y' \geq f(x')$ for $y_0 = f(x_0)$, $y_1 = f(x_1)$, then we also have that $y' \geq f(x')$ for any $y_0 \geq f(x_0)$, $y_1 \geq f(x_1)$.
(Note that $f(x')$ is unaffected by changes to y_0, y_1)

Let $(x_0, y_0), (x_1, y_1)$ be any two points on f. Since ℓ_1 and ℓ_2 are decreasing, f is also decreasing. Let $i = \arg\max_{z \in \{0,1\}} x_z$, so we have that $x_i > x_{1-i}$ and $y_i < y_{1-i}$.

- The first operation gives us $(x', y') = (x_i, (1-p) \cdot y_0 + p \cdot y_1)$, and since

  $$y' = (1-p) \cdot y_0 + p \cdot y_1 > y_i = \min_{z \in \{0,1\}} y_z$$

  for every $0 < p < 1$, we have $y' > y_i = f(x_i) = f(x')$.

- The second operation gives us $(x', y') = ((1-p) \cdot x_0 + p \cdot x_1, y_{1-i})$, and since

  $$x' = (1-p) \cdot x_0 + p \cdot x_1 > x_{1-i} = \min_{z \in \{0,1\}} x_z$$

  for every $0 < p < 1$, we have $f(x') < f(x_{1-i}) = y_{1-i} = y'$.

- The third operation gives us

  $$\begin{pmatrix} x' \\ y' \end{pmatrix} = \begin{pmatrix} (0.5-\rho) \cdot x_{1-i} + (0.5+\rho) \cdot x_i \\ (0.5-\rho) \cdot y_i + (0.5+\rho) \cdot y_{1-i} \end{pmatrix}$$

  We separate into two cases:

  1. Both (x_0, y_0) and (x_1, y_1) lie on either ℓ_1 or ℓ_2. Denote this common line by ℓ. Notice that every convex combination of $(x_0, y_0), (x_1, y_1)$ also lies on ℓ, and in particular

     $$\begin{pmatrix} x_{0.5+\rho} \\ y_{0.5+\rho} \end{pmatrix} = (0.5-\rho) \cdot \begin{pmatrix} x_0 \\ y_0 \end{pmatrix} + (0.5+\rho) \begin{pmatrix} x_1 \\ y_1 \end{pmatrix} = \begin{pmatrix} (0.5-\rho) \cdot x_0 + (0.5+\rho) \cdot x_1 \\ (0.5-\rho) \cdot y_0 + (0.5+\rho) \cdot y_1 \end{pmatrix}$$

     lies on ℓ. Since $x_{1-i} = \min(x_0, x_1)$ and $x_i = \max(x_0, x_1)$, we have that

     $$x' = (0.5-\rho) \cdot x_{1-i} + (0.5+\rho) \cdot x_i \geq (0.5-\rho) \cdot x_0 + (0.5+\rho) \cdot x_1 = x_{0.5+\rho}$$

     Since $y_i = \min(y_0, y_1)$ and $y_{1-i} = \max(y_0, y_1)$, we have that

     $$y' = (0.5-\rho) \cdot y_i + (0.5+\rho) \cdot y_{1-i} \geq (0.5-\rho) \cdot y_0 + (0.5+\rho) \cdot y_1 = y_{0.5+\rho}$$

     Overall, we conclude that $f(x') \leq \ell(x') \leq \ell(x_{0.5+\rho}) = y_{0.5+\rho} \leq y'$.

  2. One of $(x_0, y_0), (x_1, y_1)$ lies on ℓ_1 and the other lies on ℓ_2. The lines ℓ_1, ℓ_2 intersect at $(0.5+\rho, 0.5+\rho)$ and the slope of ℓ_2 is steeper, so we can conclude that

     $$f(x) = \min(\ell_1, \ell_2) = \begin{cases} \ell_1(x) & x \leq 0.5+\rho \\ \ell_2(x) & x > 0.5+\rho \end{cases}$$

and that $x_{1-i} < 0.5 + \rho < x_i$. We have that

$$\begin{aligned}
&y' = (0.5-\rho)\cdot y_i + (0.5+\rho)\cdot y_{1-i} = (0.5-\rho)\cdot \ell_2(x_i) + (0.5+\rho)\cdot \ell_1(x_{1-i}) = \\
&= (0.5-\rho)\cdot\left(\frac{0.5+\rho}{0.5-\rho} - \frac{0.5+\rho}{0.5-\rho}\cdot x_i\right) + (0.5+\rho)\cdot\left(1 - \frac{0.5-\rho}{0.5+\rho}\cdot x_{1-i}\right) = \\
&= 0.5+\rho-(0.5+\rho)\cdot x_i + 0.5 + \rho - (0.5-\rho)\cdot x_{1-i} = \\
&= 1 + 2\rho - ((0.5+\rho)\cdot x_i + (0.5-\rho)\cdot x_{1-i}) = 1 + 2\rho - x'
\end{aligned}$$

Therefore, the point (x', y') lies on the line $y = 1 + 2\rho - x$ which we will denote by ℓ. Notice that ℓ also passes through $(0.5+\rho, 0.5+\rho)$, and that the slope of ℓ is steeper than that of ℓ_1 but more moderate than that of ℓ_2, and thus $\ell_1 \leq \ell$ for $x \leq 0.5+\rho$, and $\ell_2 \leq \ell$ for $x \geq 0.5+\rho$, and overall $f(x) \leq \ell(x)$. This implies that $f(x') \leq \ell(x') = y'$.

□

References

[ACM+17] Per Austrin, Kai-Min Chung, Mohammad Mahmoody, Rafael Pass, and Karn Seth. On the impossibility of cryptography with tamperable randomness. *Algorithmica*, 79(4):1052–1101, 2017.

[BHT18] Itay Berman, Iftach Haitner, and Aris Tentes. Coin flipping of *Any* constant bias implies one-way functions. *J. ACM*, 65(3):14:1–14:95, 2018.

[Blu81] Manuel Blum. Coin flipping by telephone. In Allen Gersho, editor, *Advances in Cryptology: A Report on CRYPTO 81, CRYPTO 81, IEEE Workshop on Communications Security, Santa Barbara, California, USA, August 24-26, 1981.*, pages 11–15. U. C. Santa Barbara, Dept. of Elec. and Computer Eng., ECE Report No 82-04, 1981.

[DKS99] Ivan Damgård, Joe Kilian, and Louis Salvail. On the (im)possibility of basing oblivious transfer and bit commitment on weakened security assumptions. In Jacques Stern, editor, *Advances in Cryptology - EUROCRYPT '99, International Conference on the Theory and Application of Cryptographic Techniques, Prague, Czech Republic, May 2-6, 1999, Proceeding*, volume 1592 of *Lecture Notes in Computer Science*, pages 56–73. Springer, 1999.

[DNR04] Cynthia Dwork, Moni Naor, and Omer Reingold. Immunizing encryption schemes from decryption errors. In Christian Cachin and Jan Camenisch, editors, *Advances in Cryptology - EUROCRYPT 2004, International Conference on the Theory and Applications of Cryptographic Techniques, Interlaken, Switzerland, May 2-6, 2004, Proceedings*, volume 3027 of *Lecture Notes in Computer Science*, pages 342–360. Springer, 2004.

[Hai13] Iftach Haitner. A parallel repetition theorem for any interactive argument. *SIAM J. Comput.*, 42(6):2487–2501, 2013.

[HILL99] Johan Håstad, Russell Impagliazzo, Leonid A. Levin, and Michael Luby. A pseudorandom generator from any one-way function. *SIAM J. Comput.*, 28(4):1364–1396, 1999.

[HO14] Iftach Haitner and Eran Omri. Coin flipping with constant bias implies one-way functions. *SIAM J. Comput.*, 43(2):389–409, 2014.

[HR08] Shai Halevi and Tal Rabin. Degradation and amplification of computational hardness. In Ran Canetti, editor, *Theory of Cryptography, Fifth Theory of Cryptography Conference, TCC 2008, New York, USA, March 19-21, 2008.*, volume 4948 of *Lecture Notes in Computer Science*, pages 626–643. Springer, 2008.

[IL89] Russell Impagliazzo and Michael Luby. One-way functions are essential for complexity based cryptography (extended abstract). In *30th Annual Symposium on Foundations of Computer Science, Research Triangle Park, North Carolina, USA, 30 October - 1 November 1989*, pages 230–235. IEEE Computer Society, 1989.

[LT13] Huijia Lin and Stefano Tessaro. Amplification of chosen-ciphertext security. In Thomas Johansson and Phong Q. Nguyen, editors, *Advances in Cryptology - EUROCRYPT 2013, 32nd Annual International Conference on the Theory and Applications of Cryptographic Techniques, Athens, Greece, May 26-30, 2013. Proceedings*, volume 7881 of *Lecture Notes in Computer Science*, pages 503–519. Springer, 2013.

[MPS10] Hemanta K. Maji, Manoj Prabhakaran, and Amit Sahai. On the computational complexity of coin flipping. In *51th Annual IEEE Symposium on Foundations of Computer Science, FOCS 2010, October 23-26, 2010, Las Vegas, Nevada, USA*, pages 613–622. IEEE Computer Society, 2010.

[Nao91] Moni Naor. Bit commitment using pseudorandomness. *J. Cryptology*, 4(2):151–158, 1991.

[RVW04] Omer Reingold, Salil Vadhan, and Avi Wigderson. A note on extracting randomness from santha-vazirani sources. *Unpublished manuscript*, 2004.

[SV86] Miklos Santha and Umesh V. Vazirani. Generating quasi-random sequences from semi-random sources. *J. Comput. Syst. Sci.*, 33(1):75–87, 1986.

[Wul08] Jürg Wullschleger. *Oblivious-transfer amplification.* PhD thesis, Universität Zürich, 2008.

[Yao82] Andrew Chi-Chih Yao. Theory and applications of trapdoor functions (extended abstract). In *23rd Annual Symposium on Foundations of Computer Science, Chicago, Illinois, USA, 3-5 November 1982*, pages 80–91. IEEE Computer Society, 1982.