Tight Computational Indistinguishability Bound of Product Distributions

Nathan Geier
Tel Aviv University
nathangeier@mail.tau.ac.il

Abstract

Assume that X_0, X_1 (respectively Y_0, Y_1) are d_X (respectively d_Y) indistinguishable for circuits of a given size. It is well known that the product distributions X_0Y_0, X_1Y_1 are $d_X + d_Y$ indistinguishable for slightly smaller circuits. However, in probability theory where unbounded adversaries are considered through statistical distance, it is folklore knowledge that in fact X_0Y_0 and X_1Y_1 are $d_X + d_Y - d_X \cdot d_Y$ indistinguishable, and also that this bound is tight.

We formulate and prove the computational analog of this tight bound. Our proof is entirely different from the proof in the statistical case, which is non-constructive. As a corollary, we show that if X and Y are d indistinguishable, then k independent copies of X and k independent copies of Y are almost $1 - (1-d)^k$ indistinguishable for smaller circuits, as against $d \cdot k$ using the looser bound. Our bounds are useful in settings where only weak (i.e. non-negligible) indistinguishability is guaranteed. We demonstrate this in the context of cryptography, showing that our bounds yield simple analysis for amplification of weak oblivious transfer protocols.

1 Introduction

Computational indistinguishability is a basic concept in computational complexity and cryptography. One of the most basic bounds in this context, which is easy to see using a simple hybrid argument, is that for distributions X_0, X_1 of distance d_X, and Y_0, Y_1 of distance d_Y, with d_{XY} denoting the distance between X_0Y_0, X_1Y_1, we have that

$$d_{XY} \leq d_X + d_Y,$$

which holds both statistically and in the computational setting holds for slightly smaller circuits. However, in probability theory where statistical distance, or equivalently, indistinguishability against unbounded attackers is considered, it is folklore knowledge [Kon12, Lemma 2.2] that a better, tight bound holds:

$$d_{XY} \leq d_X + d_Y - d_X \cdot d_Y.$$

It is tight in the sense that for every choice of d_X, d_Y, there exist distributions X_0, X_1 with distance d_X and distributions Y_0, Y_1 with distance d_Y, such that $d_{XY} = d_X + d_Y - d_X \cdot d_Y$. The proof of this bound uses coupling [Hol12], and is thus inherently non-constructive. We provide a proof of the tight bound in the computational setting, both uniform and non-uniform, with an additive loss of ε which can be made as small as we want, by paying in increasing the running time or circuit size with relation to $1/\varepsilon$. To be more specific, for the non-uniform case, we (roughly) show that

Theorem 1.1 (Informal)**.** *Let X_0, X_1 be d_X indistinguishable for size s_X circuits. (Respectively Y_0, Y_1, d_Y, s_Y.) Then, for every $k \in \mathbb{N}$, we have that (X_0, Y_0) and (X_1, Y_1) are $(d_X + d_Y - d_X \cdot d_Y + \varepsilon_k)$ indistinguishable for size s_k circuits, where*

$$\varepsilon_k \leq (d_Y)^k, \qquad s_k \approx \min\{s_Y, s_X/k\}.$$

Corollary 1.1 (Informal)**.** *Let D, Q be distributions that are d indistinguishable for size s circuits. Then, for every $m \in \mathbb{N}$ and ε, we have that $D^{\otimes m}, Q^{\otimes m}$ are $(1 - (1-d)^m + \varepsilon)$ indistinguishable for size $s_{m,\varepsilon}$ circuits, where*

$$s_{m,\varepsilon} \approx s(1-d)^m / \log(1/\varepsilon).$$

And we also show similar results in the uniform setting. First we prove the isolated non-uniform analog, which we later show how to generalize to the uniform computation model. Then we show how to arrive at the corollary, that if the computational distance between X and Y is at most d, then the computational distance between the k-product of X and the k-product of Y is upper bounded by almost $1 - (1-d)^k$ for smaller circuits, as against $d \cdot k$ resulted by the looser well known bound, which in particular may be larger than 1. The proof of the corollary essentially follows by (carefully) applying the bound of the isolated case again and again. It should be noted that the difference between the bounds is especially interesting when k is not very small compared to $1/d$. For example, if $d = 0.5$, $k = 3$, the tight bound is 0.875 while the looser bound of $1.5 \geq 1$ is trivial. Finally, we show how these bounds may be used for amplification of weak oblivious transfer protocols [DKS99, Wul07], in the computational setting, providing an alternative simple analysis to the fact that the information theoretic amplification process also works computationally. Some of the techniques and statement formulations presented in this paper were inspired by Levin's proof of the XOR Lemma [Lev87], and its presentation in [GNW95].

2 Definitions

For a distribution D, denote by $D^{\otimes k}$ the distribution of k independent copies of D. For distributions X_0, X_1 over Ω, a distinguisher is a boolean $A : \Omega \to \{0,1\}$, and we let $\mathrm{adv}^+_A(X_0, X_1) := \mathbb{E}[A(X_1) - A(X_0)]$. (The expectation is also over A if it is not deterministic.) We say that distributions X_0, X_1 are d indistinguishable for size s circuits if for any such circuit C, we have that $\mathrm{adv}^+_C(X_0, X_1) \le d$. For distributions X, Y we will denote by (X, Y) the product distribution, given by two independent samples from X and Y. We denote by $B(p)$ the Bernoulli distribution with parameter p, and more generally by $B^\ell(p)$ the distribution that is equal to 1^ℓ with probability p and otherwise 0^ℓ. For a string s, we denote by $s[i]$ the i'th bit of s. We will denote by $[m]$ the set $\{1, \ldots, m\}$. We denote by $X_{1/2}$ the distribution given by $b \leftarrow \{0,1\}, x \leftarrow X_b$. An ensemble of distributions $X = \{X_n\}$ is efficiently samplable if there exists a uniform PPT sampler that given 1^n outputs a sample from X_n.

2.1 Notation

When the same distribution is used multiple times in a single expression, e.g. $(f(D), g(D))$ for D, it should be interpreted that a single value $d \leftarrow D$ is sampled and given to both f and g, rather than two independent samples.

3 The Non-Uniform Bounds and Tightness

Let us start with the non-uniform version as it is more simple and clean. The uniform version will be a generalization of the ideas presented below. Roughly speaking, we show that given a distinguisher C for $(X_0, Y_0), (X_1, Y_1)$, if $C(x, \cdot)$ is not a good enough distinguisher between Y_0, Y_1 for all values of x, then we can build an amplifier for X_0, X_1 distinguishers. We then use this amplifier to turn the trivial distinguisher that always outputs 1 into a good enough distinguisher.

Theorem 3.1. *Let X_0, X_1 be distributions over ℓ_X bits that are d_X indistinguishable for size s_X circuits. (Respectively $Y_0, Y_1, \ell_Y, d_Y, s_Y$.) Then, for every $k \in \mathbb{N}$, we have that (X_0, Y_0) and (X_1, Y_1) are $(d_X + d_Y - d_X \cdot d_Y + \varepsilon_k)$ indistinguishable for size s_k circuits, where*

$$\varepsilon_k := \frac{(d_Y)^k \cdot d_X (1 - d_Y)}{1 - (d_Y)^k} \le (d_Y)^k, \qquad s_k := \min\left\{ s_Y - \ell_X, \frac{s_X - 1}{k} - 5\ell_Y - 1 \right\}.$$

Remark 3.1. We note that our starting point, $k = 1$, matches the simple hybrid argument bound of $d_X + d_Y$ since $\varepsilon_1 = d_X \cdot d_Y$, and as k grows larger our bound gets closer and closer to the tight bound of $d_X + d_Y - d_X \cdot d_Y$, while the circuits bound grows smaller. Also note that the bound is asymmetric with respect to the circuit size bounds. This asymmetry is important for preserving a similar circuit size when applying the isolated case over and over again. See a similar argument in [GNW95, Section 3].

Proof. Assume toward contradiction that for some circuit C of size s_k, we have that

$$\mathrm{adv}^+_C((X_0, Y_0), (X_1, Y_1)) > (d_X + d_Y - d_X \cdot d_Y + \varepsilon_k).$$

For every fixed x, it must be that $C(x,\cdot)$ is able to distinguish between Y_0 and Y_1 by at most d_Y, otherwise we get a contradiction as the size of this circuit is $s_k + \ell_X \leq s_Y$. Then, for every candidate distinguisher A between X_0 and X_1, we have that

$$\begin{aligned}
&\mathrm{adv}_C^+\left(\left(X_1, Y_{A(X_1)}\right), (X_1, Y_1)\right) \leq d_Y \cdot \Pr\left[A(X_1) = 0\right] \\
&\mathrm{adv}_C^+\left((X_0, Y_0), \left(X_0, Y_{A(X_0)}\right)\right) \leq d_Y \cdot \Pr\left[A(X_0) = 1\right]
\end{aligned}$$

where $x, y \leftarrow X_1, Y_{A(X_1)}$ is resulted by $x \leftarrow X_1$, $b \leftarrow A(x)$, $y \leftarrow Y_b$. This holds because

$$\begin{aligned}
&\mathrm{adv}_C^+\left(\left(X_1, Y_{A(X_1)}\right), (X_1, Y_1)\right) = \mathbb{E}\left[C(X_1, Y_1) - C(X_1, Y_{A(X_1)})\right] = \\
&= \mathbb{E}\left[C(X_1, Y_1) - C(X_1, Y_0) | A(X_1) = 0\right] \cdot \Pr\left[A(X_1) = 0\right] + \\
&+ \mathbb{E}\left[C(X_1, Y_1) - C(X_1, Y_1) | A(X_1) = 1\right] \cdot \Pr\left[A(X_1) = 1\right] = \\
&= \mathbb{E}_{x \leftarrow X_1 | A(X_1) = 0}\left[C(x, Y_1) - C(x, Y_0)\right] \cdot \Pr\left[A(X_1) = 0\right] = \\
&= \mathbb{E}_{x \leftarrow X_1 | A(X_1) = 0}\left[\mathrm{adv}_{C(x,\cdot)}^+\left(Y_0, Y_1\right)\right] \cdot \Pr\left[A(X_1) = 0\right] \leq d_Y \cdot \Pr\left[A(X_1) = 0\right]
\end{aligned}$$

and using a symmetric argument for the second inequality. Using that (in general)

$$\sum_{i \in [n]} \mathrm{adv}_C^+(D_i, D_{i+1}) = \mathrm{adv}_C^+(D_1, D_{n+1})$$

we conclude that

$$\begin{aligned}
&\mathrm{adv}_C^+\left((X_0, Y_0), (X_1, Y_1)\right) = \mathrm{adv}_C^+\left((X_0, Y_0), \left(X_0, Y_{A(X_0)}\right)\right) + \\
&+ \mathrm{adv}_C^+\left(\left(X_0, Y_{A(X_0)}\right), \left(X_1, Y_{A(X_1)}\right)\right) + \mathrm{adv}_C^+\left(\left(X_1, Y_{A(X_1)}\right), (X_1, Y_1)\right)
\end{aligned}$$

and thus

$$\begin{aligned}
&\mathrm{adv}_C^+\left(\left(X_0, Y_{A(X_0)}\right), \left(X_1, Y_{A(X_1)}\right)\right) = \mathrm{adv}_C^+\left((X_0, Y_0), (X_1, Y_1)\right) - \\
&- \mathrm{adv}_C^+\left(\left(X_1, Y_{A(X_1)}\right), (X_1, Y_1)\right) - \mathrm{adv}_C^+\left((X_0, Y_0), \left(X_0, Y_{A(X_0)}\right)\right) > \\
&> (d_X + d_Y - d_X \cdot d_Y + \varepsilon_k) - (d_Y \cdot \Pr\left[A(X_1) = 0\right]) - (d_Y \cdot \Pr\left[A(X_0) = 1\right]) = \\
&= (d_X - d_X \cdot d_Y + \varepsilon_k) + d_Y(1 - \Pr\left[A(X_1) = 0\right] - \Pr\left[A(X_0) = 1\right]) = \\
&= (d_X - d_X \cdot d_Y + \varepsilon_k) + d_Y(\Pr\left[A(X_1) = 1\right] - \Pr\left[A(X_0) = 1\right]) = \\
&= (d_X - d_X \cdot d_Y + \varepsilon_k) + d_Y(\mathbb{E}\left[A(X_1)\right] - \mathbb{E}\left[A(X_0)\right]) = \\
&= (d_X - d_X \cdot d_Y + \varepsilon_k) + d_Y \cdot \mathrm{adv}_A^+\left(X_0, X_1\right).
\end{aligned}$$

In other words, we can build a new distinguisher A' for X_0, X_1 by applying A to our input x, sampling $y \leftarrow Y_{A(x)}$ and feeding (x, y) to C, and have that

$$\mathrm{adv}_{A'}^+\left(X_0, X_1\right) > (d_X - d_X \cdot d_Y + \varepsilon_k) + d_Y \cdot \mathrm{adv}_A^+\left(X_0, X_1\right).$$

If we start from A_0 being the trivial distinguisher that always outputs 1 and keep repeating this process for k steps, we get that

$$\begin{aligned}
&\text{adv}^+_{A_k}(X_0, X_1) > (d_X - d_X \cdot d_Y + \varepsilon_k) + d_Y \cdot \text{adv}^+_{A_{k-1}}(X_0, X_1) > \\
&> (d_X - d_X \cdot d_Y + \varepsilon_k) + d_Y \cdot (d_X - d_X \cdot d_Y + \varepsilon_k) + (d_Y)^2 \cdot \text{adv}^+_{A_{k-2}}(X_0, X_1) > \\
&> \cdots > (d_X - d_X \cdot d_Y + \varepsilon_k) \sum_{i=0}^{k-1} (d_Y)^i + (d_Y)^k \cdot \text{adv}^+_{A_0}(X_0, X_1) = \\
&= (d_X - d_X \cdot d_Y + \varepsilon_k) \sum_{i=0}^{k-1} (d_Y)^i = \frac{(d_X - d_X \cdot d_Y + \varepsilon_k)\left(1 - (d_Y)^k\right)}{1 - d_Y} = \\
&= \frac{\left(d_X(1-d_Y) + \frac{(d_Y)^k \cdot d_X (1-d_Y)}{1-(d_Y)^k}\right)\left(1-(d_Y)^k\right)}{1-d_Y} = \left(d_X + \frac{(d_Y)^k \cdot d_X}{1-(d_Y)^k}\right)\left(1-(d_Y)^k\right) = \\
&= d_X\left(1-(d_Y)^k\right) + (d_Y)^k \cdot d_X = d_X.
\end{aligned}$$

And so, we have concluded that A_k distinguishes X_0 from X_1 with advantage better than d_X. Next, for the circuit size, in order to implement A_k we start by applying A_{k-1}, sample $y_0 \leftarrow Y_0, y_1 \leftarrow Y_1$, use a multiplexer to choose $y \leftarrow y_b$ where b is the output gate of A_{k-1}, and finally use the circuit C. Instead of sampling y_0, y_1, we can simply use non-uniformity to hard-code the best samples, at the cost of $2\ell_Y$ gates. Implementing the multiplexer can be done using $3\ell_Y + 1$ gates, with one gate computing $\neg b$ and for every $i \in [\ell_Y]$ another 3 gates to compute $y[i] = (y_0[i] \wedge \neg b) \vee (y_1[i] \wedge b)$. Overall, we conclude that $\text{size}(A_k) = \text{size}(A_{k-1}) + 5\ell_Y + 1 + s_k$ and therefore

$$\text{size}(A_k) = \text{size}(A_0) + k \cdot (5\ell_Y + 1 + s_k) \leq 1 + k \cdot \left(5\ell_Y + 1 + \left(\frac{s_X - 1}{k} - 5\ell_Y - 1\right)\right) = s_X$$

which is a contradiction to our assumption that d_X is an upper bound on the advantage of size s_X circuits distinguishing X_0 from X_1. □

3.1 The N-Fold Case

Corollary 3.1. *Let D, Q be distributions over ℓ bits that are d indistinguishable for size s circuits. Then, for every $m \in \mathbb{N}$ and ε, we have that $D^{\otimes m}, Q^{\otimes m}$ are $(1 - (1-d)^m + \varepsilon)$ indistinguishable for size $s_{m,\varepsilon}$ circuits, where*

$$s_{m,\varepsilon} = \frac{s-1}{k_{m,\varepsilon}} - 5m\ell - 1, \qquad k_{m,\varepsilon} = \left\lceil \frac{\log(d\varepsilon)}{\log(1-(1-d)^m + \varepsilon)} \right\rceil \leq \left\lceil \frac{\log(1/d\varepsilon)}{(1-d)^m - \varepsilon} \right\rceil.$$

Proof. If $\varepsilon \geq (1-d)^m$ the statement is trivially true. Otherwise, we start from D, Q and use Theorem 3.1 to repeatedly add copies of D, Q for $m-1$ times, using $k_{m,\varepsilon}$ set at the statement, where each time the added copy of D, Q is treated as X_0, X_1 and $D^{\otimes i}, Q^{\otimes i}$ are treated as Y_0, Y_1. Let d_i denote the bound on the advantage of i copies, then we have that $d_1 = d$ and $d_i \leq d_{i-1} + d - d_{i-1} \cdot d + (d_{i-1})^{k_{m,\varepsilon}}$. We can see by induction that $d_i \leq 1 - (1-d)^i + \varepsilon$ for

$i \in [m]$ as

$$\begin{aligned}
d_i &\leq d_{i-1} + d - d_{i-1} \cdot d + (d_{i-1})^{k_{m,\varepsilon}} = (1-d)d_{i-1} + d + (d_{i-1})^{k_{m,\varepsilon}} \leq \\
&\leq (1-d)\left(1-(1-d)^{i-1}+\varepsilon\right) + d + \left(1-(1-d)^{i-1}+\varepsilon\right)^{k_{m,\varepsilon}} = \\
&= 1 - d - (1-d)^i + (1-d)\varepsilon + d + \left(1-(1-d)^{i-1}+\varepsilon\right)^{k_{m,\varepsilon}} = \\
&= 1 - (1-d)^i + (1-d)\varepsilon + \left(1-(1-d)^{i-1}+\varepsilon\right)^{k_{m,\varepsilon}} \leq \\
&\leq 1 - (1-d)^i + (1-d)\varepsilon + (1-(1-d)^m+\varepsilon)^{k_{m,\varepsilon}} \leq 1-(1-d)^i + \varepsilon
\end{aligned}$$

where in the last inequality we used the choice of $k_{m,\varepsilon}$. For the circuit size, we can easily see by induction on i that $s_{i,\varepsilon} \geq (s-1)/k_{m,\varepsilon} - 5i\ell - 1$, as we have that $s_{1,\varepsilon} = s$ and

$$\begin{aligned}
s_{i,\varepsilon} &\geq \min\left\{s_{(i-1),\varepsilon} - \ell, \frac{s-1}{k_{m,\varepsilon}} - 5(i-1)\ell - 1\right\} \geq \\
&\geq \min\left\{\frac{s-1}{k_{m,\varepsilon}} - 5(i-1)\ell - 1 - \ell, \frac{s-1}{k_{m,\varepsilon}} - 5(i-1)\ell - 1\right\} \geq \frac{s-1}{k_{m,\varepsilon}} - 5i\ell - 1.
\end{aligned}$$

□

3.2 Tightness

This is somewhat folklore knowledge, that we explicitly state for the sake of completeness. We show that for every choice of $d_X, d_Y, s_X, s_Y, \ell_X, \ell_Y$ there exist two pairs of distributions X_0, X_1 and Y_0, Y_1, such that X_0, X_1 are over ℓ_X bits and cannot be distinguished with advantage better than d_X by size s_X circuits (resp. for Y_0, Y_1 with ℓ_Y, d_Y, s_Y), yet (X_0, Y_0) and (X_1, X_1) can be distinguished with advantage $d_X + d_Y - d_X \cdot d_Y$ using a size 1 circuit. For the n-fold case, we show that for every choice of d, s, ℓ there exist distributions X, Y over ℓ bits with distance at most d against s-sized circuits, such that $X^{\otimes k}, Y^{\otimes k}$ can be distinguished with advantage $1-(1-d)^k$ using a circuit of size $2k-1$. We will use statistical distance in these examples, noting that the statistical distance between distributions is equal to the maximal advantage of unbounded adversaries distinguishing between them, and that the statistical distance from a constant variable is equal to the probability to differ from it.

For the isolated case, we let $X_0 \equiv 0^{\ell_X}$, $X_1 \coloneqq B^{\ell_X}(d_X)$, $Y_0 \equiv 0^{\ell_Y}$, $Y_1 \coloneqq B^{\ell_Y}(d_Y)$. We have that size s_X circuits can distinguish between X_0, X_1 with advantage at most d_X (resp. for Y_0, Y_1 with s_Y, d_Y) as this is the statistical distance between them. Also, it is easy to verify that the simple size 1 circuit which given (x, y) computes $x[1] \vee y[1]$ distinguishes between (X_0, Y_0) and (X_1, Y_1) with advantage $1-(1-d_X)(1-d_Y) = d_X + d_Y - d_X \cdot d_Y$.

For the n-fold case, let $X \equiv 0^\ell$, $Y \coloneqq B^\ell(d)$, then size s circuits can distinguish X from Y with advantage at most d. Yet the circuit of size $2k-1$ which given $(z_1, \ldots, z_k)$ computes $\vee_i z_i[1]$ (using a full binary tree of OR gates) distinguishes between $X^{\otimes k}$ and $Y^{\otimes k}$ with advantage $1-(1-d)^k$.

4 The Uniform Variant

We used non-uniformity two times in the proof of Theorem 3.1. The second time, which is easier to deal with, is in the circuit size analysis where we hard-coded the best samples of

y_0, y_1 to each iteration of A_i. Instead, in the uniform version, we will use uniform samplers of Y_0, Y_1. The first use of non-uniformity was when we assumed that $C(x, \cdot)$ is at most a d_Y-distinguisher between Y_0 and Y_1, for every fixed x. More specifically, we used this to get that

$$\mathrm{adv}_C^+\left(\left(X_1, Y_{A(X_1)}\right), (X_1, Y_1)\right) \leq d_Y \cdot \Pr\left[A(X_1) = 0\right].$$

For the uniform case, we will relax this condition to x not being easy to hard-code, in the following sense:

$$\Pr_{x \leftarrow X_{1/2}}\left[\mathrm{adv}_{C(x,\cdot)}^+(Y_0, Y_1) > d_Y + \varepsilon_k\right] \leq \varepsilon_k$$

where $X_{1/2}$ is given by $b \leftarrow \{0,1\}, x \leftarrow X_b$. If this condition doesn't hold then we can efficiently compute a good x, except for negligible probability, assuming that efficient uniform samplers for X_0, X_1, Y_0, Y_1 exist. Otherwise, we will see that

$$\mathrm{adv}_C^+\left(\left(X_1, Y_{A(X_1)}\right), (X_1, Y_1)\right) \leq d_Y \cdot \Pr\left[A(X_1) = 0\right] + 3\varepsilon_k$$

and so almost the same argument from the non-uniform case works, except that now we lose another small additive term. Let us state and prove this more formally:

Lemma 4.1. *Let $X_0 = \{X_{0,n}\}, X_1 = \{X_{1,n}\}, Y_0 = \{Y_{0,n}\}, Y_1 = \{Y_{1,n}\}$ be ensembles of efficiently samplable distributions, and $d_X(n), d_Y(n)$ be efficiently computable functions between 0 and 1. Then, for every $k \in \mathbb{N}$ and time $t(n)$ Turing machine M distinguishing (X_0, Y_0) from (X_1, Y_1) infinitely often with advantage at least $(d_X + d_Y - d_X \cdot d_Y + 7\varepsilon_k)$ for*

$$\varepsilon_k := \frac{(d_Y)^k \cdot d_X (1 - d_Y)}{1 - (d_Y)^k} \leq (d_Y)^k,$$

we have that either M efficiently yields a distinguisher for Y_0, Y_1 through a hard-coding of x, in the sense that for infinitely many n's

$$\Pr_{x \leftarrow X_{1/2}}\left[\mathrm{adv}_{M(1^n,x,\cdot)}^+(Y_0, Y_1) > d_Y + \varepsilon_k\right] > \varepsilon_k,$$

or there exists a time $t \cdot \mathrm{poly}(nk)$ infinitely often distinguisher between X_0, X_1 with advantage at least d_X.

Proof. For the sake of notational ease, we will drop the asymptotic notation and replace $M(1^n)$ with C. Assume that for all but finitely many n's,

$$\Pr_{x \leftarrow X_{1/2}}\left[\mathrm{adv}_{C(x,\cdot)}^+(Y_0, Y_1) > d_Y + \varepsilon_k\right] \leq \varepsilon_k.$$

Then, for every candidate distinguisher A between X_0 and X_1, for all but finitely many n's, we have that

$$\mathrm{adv}_C^+\left(\left(X_1, Y_{A(X_1)}\right), (X_1, Y_1)\right) \leq d_Y \cdot \Pr\left[A(X_1) = 0\right] + 3\varepsilon_k$$
$$\mathrm{adv}_C^+\left((X_0, Y_0), \left(X_0, Y_{A(X_0)}\right)\right) \leq d_Y \cdot \Pr\left[A(X_0) = 1\right] + 3\varepsilon_k$$

where $x, y \leftarrow X_1, Y_{A(X_1)}$ is resulted by $x \leftarrow X_1$, $b \leftarrow A(x)$, $y \leftarrow Y_b$. To see this, we first note that

$$\begin{aligned}\varepsilon_k &\geq \Pr_{x \leftarrow X_{1/2}}\left[\mathrm{adv}^+_{C(x,\cdot)}\left(Y_0, Y_1\right) > d_Y + \varepsilon_k\right] \geq \\ &\geq \frac{1}{2}\Pr\left[A(X_1) = 0\right]\Pr_{x \leftarrow X_1 | A(X_1)=0}\left[\mathrm{adv}^+_{C(x,\cdot)}\left(Y_0, Y_1\right) > d_Y + \varepsilon_k\right]\end{aligned}$$

which implies that

$$\mathbb{E}_{x \leftarrow X_1 | A(X_1)=0}\left[\mathrm{adv}^+_{C(x,\cdot)}\left(Y_0, Y_1\right)\right] \leq d_Y + \varepsilon_k + \frac{2\varepsilon_k}{\Pr\left[A(X_1) = 0\right]} \leq d_Y + \frac{3\varepsilon_k}{\Pr\left[A(X_1) = 0\right]}.$$

Plugging it into the last inequality in the following, we get

$$\begin{aligned}&\mathrm{adv}^+_C\left(\left(X_1, Y_{A(X_1)}\right), \left(X_1, Y_1\right)\right) = \mathbb{E}\left[C(X_1, Y_1) - C(X_1, Y_{A(X_1)})\right] = \\ &= \mathbb{E}\left[C(X_1, Y_1) - C(X_1, Y_0) | A(X_1) = 0\right] \cdot \Pr\left[A(X_1) = 0\right] + \\ &+ \mathbb{E}\left[C(X_1, Y_1) - C(X_1, Y_1) | A(X_1) = 1\right] \cdot \Pr\left[A(X_1) = 1\right] = \\ &= \mathbb{E}_{x \leftarrow X_1 | A(X_1)=0}\left[C(x, Y_1) - C(x, Y_0)\right] \cdot \Pr\left[A(X_1) = 0\right] = \\ &= \mathbb{E}_{x \leftarrow X_1 | A(X_1)=0}\left[\mathrm{adv}^+_{C(x,\cdot)}\left(Y_0, Y_1\right)\right] \cdot \Pr\left[A(X_1) = 0\right] \leq d_Y \cdot \Pr\left[A(X_1) = 0\right] + 3\varepsilon_k\end{aligned}$$

and use a symmetric argument for the second upper bound. Using that (in general)

$$\sum_{i \in [n]} \mathrm{adv}^+_C(D_i, D_{i+1}) = \mathrm{adv}^+_C(D_1, D_{n+1})$$

we conclude that

$$\begin{aligned}&\mathrm{adv}^+_C\left(\left(X_0, Y_0\right), \left(X_1, Y_1\right)\right) = \mathrm{adv}^+_C\left(\left(X_0, Y_0\right), \left(X_0, Y_{A(X_0)}\right)\right) + \\ &+ \mathrm{adv}^+_C\left(\left(X_0, Y_{A(X_0)}\right), \left(X_1, Y_{A(X_1)}\right)\right) + \mathrm{adv}^+_C\left(\left(X_1, Y_{A(X_1)}\right), \left(X_1, Y_1\right)\right)\end{aligned}$$

and thus

$$\begin{aligned}&\mathrm{adv}^+_C\left(\left(X_0, Y_{A(X_0)}\right), \left(X_1, Y_{A(X_1)}\right)\right) = \mathrm{adv}^+_C\left(\left(X_0, Y_0\right), \left(X_1, Y_1\right)\right) - \\ &- \mathrm{adv}^+_C\left(\left(X_1, Y_{A(X_1)}\right), \left(X_1, Y_1\right)\right) - \mathrm{adv}^+_C\left(\left(X_0, Y_0\right), \left(X_0, Y_{A(X_0)}\right)\right) > \\ &> \left(d_X + d_Y - d_X \cdot d_Y + 7\varepsilon_k\right) - \left(d_Y \cdot \Pr\left[A(X_1) = 0\right] + 3\varepsilon_k\right) - \left(d_Y \cdot \Pr\left[A(X_0) = 1\right] + 3\varepsilon_k\right) = \\ &= \left(d_X - d_X \cdot d_Y + \varepsilon_k\right) + d_Y\left(1 - \Pr\left[A(X_1) = 0\right] - \Pr\left[A(X_0) = 1\right]\right) = \\ &= \left(d_X - d_X \cdot d_Y + \varepsilon_k\right) + d_Y\left(\Pr\left[A(X_1) = 1\right] - \Pr\left[A(X_0) = 1\right]\right) = \\ &= \left(d_X - d_X \cdot d_Y + \varepsilon_k\right) + d_Y\left(\mathbb{E}\left[A(X_1)\right] - \mathbb{E}\left[A(X_0)\right]\right) = \\ &= \left(d_X - d_X \cdot d_Y + \varepsilon_k\right) + d_Y \cdot \mathrm{adv}^+_A\left(X_0, X_1\right).\end{aligned}$$

In other words, we can build a new distinguisher A' for X_0, X_1 by applying A to our input x, sampling $y \leftarrow Y_{A(x)}$ and feeding (x, y) to C, and have that

$$\mathrm{adv}^+_{A'}\left(X_0, X_1\right) > \left(d_X - d_X \cdot d_Y + \varepsilon_k\right) + d_Y \cdot \mathrm{adv}^+_A\left(X_0, X_1\right).$$

If we start from A_0 being the trivial distinguisher that always outputs 1 and keep repeating this process for k steps, we get that

$$\begin{aligned}
&\mathrm{adv}^+_{A_k}(X_0, X_1) > (d_X - d_X \cdot d_Y + \varepsilon_k) + d_Y \cdot \mathrm{adv}^+_{A_{k-1}}(X_0, X_1) > \\
&> (d_X - d_X \cdot d_Y + \varepsilon_k) + d_Y \cdot (d_X - d_X \cdot d_Y + \varepsilon_k) + (d_Y)^2 \cdot \mathrm{adv}^+_{A_{k-2}}(X_0, X_1) > \\
&> \cdots > (d_X - d_X \cdot d_Y + \varepsilon_k) \sum_{i=0}^{k-1} (d_Y)^i + (d_Y)^k \cdot \mathrm{adv}^+_{A_0}(X_0, X_1) = \\
&= (d_X - d_X \cdot d_Y + \varepsilon_k) \sum_{i=0}^{k-1} (d_Y)^i = \frac{(d_X - d_X \cdot d_Y + \varepsilon_k)\left(1 - (d_Y)^k\right)}{1 - d_Y} = \\
&= \frac{\left(d_X(1-d_Y) + \frac{(d_Y)^k \cdot d_X (1-d_Y)}{1-(d_Y)^k}\right)\left(1 - (d_Y)^k\right)}{1 - d_Y} = \left(d_X + \frac{(d_Y)^k \cdot d_X}{1 - (d_Y)^k}\right)\left(1 - (d_Y)^k\right) = \\
&= d_X\left(1 - (d_Y)^k\right) + (d_Y)^k \cdot d_X = d_X.
\end{aligned}$$

And so, we have concluded that A_k distinguishes X_0 from X_1 with advantage better than d_X. In order to implement A_k we need to run C, sample Y_0, Y_1 and use a multiplexer, for k times, so we conclude that $\mathrm{time}(A_k) = t \cdot \mathrm{poly}(n, k)$. □

Remark 4.1. In particular, we can use this lemma to show that if X_0, X_1 are d_X ind. and Y_0, Y_1 are d_Y ind. then (X_0, Y_0) and (X_1, Y_1) are $d_X + d_Y - d_X \cdot d_Y + 7\varepsilon_k$ ind. for Turing machines with running time of

$$t = \min\{t_X/\mathrm{poly}(n, k), t_Y/\mathrm{poly}(n, 1/\varepsilon_k)\},$$

which may be good enough for a constant number of uses, but does not work well beyond that, as every use costs us a division of the time bound by a polynomial. This is why we cannot prove the n-fold case immediately by repeatedly applying Lemma 4.1. The key idea is that we do not need to keep resampling and testing over and over again, but instead, once we find a good enough x in the i'th coordinate, we fix it for the rest of the process, or if the hard-coding of the i'th coordinate does not succeed, the above lemma states we can distinguish there.

Theorem 4.1. *Let $X = \{X_n\}, Y = \{Y_n\}$ be ensembles of efficiently samplable distributions that are $d(n)$ indistinguishable for time $t(n)$ Turing machines. Then, for every $m = m(n)$, we have that $X^{\otimes m}$ and $Y^{\otimes m}$ are $(1 - (1-d)^m + 7m\varepsilon)$ indistinguishable for time $t_{m,\varepsilon}$ Turing machines, where*

$$t_{m,\varepsilon} = t/\mathrm{poly}(n, m, k_{m,\varepsilon}, 1/\varepsilon), \quad k_{m,\varepsilon} = \left\lceil \frac{\log(\varepsilon)}{\log(1 - (1-d)^m + 7m\varepsilon)} \right\rceil \leq \left\lceil \frac{\log(1/\varepsilon)}{(1-d)^m - 7m\varepsilon} \right\rceil.$$

Proof. For $i = 0, 1, \ldots, m-1$, we try to hard-code the $m-i$'th coordinate using $\mathrm{poly}(n, 1/\varepsilon)$ samples, and getting a distinguisher for $X^{\otimes m-i}, Y^{\otimes m-i}$ with advantage of at least $1 - (1 - d)^{m-i} + 7(m-i)\varepsilon$ except for negligible probability (the probability that the estimate was good but not truthful to the expectation) until for some i we fail to find a good value to hard-code (if we reached $i = m-1$ and succeeded then we are done). Once we fail, we apply the isolated case of Lemma 4.1, which essentially states that if the hard-coding of X, Y into such circuit failed, then one can build a distinguisher for them, and we are done.

Let us be more explicit about how we sample and hard-code the $m-i$'th coordinate: We are given (except for negligible probability) good samples for the coordinates in $m-i+1, \ldots, m$ and hard-code them into A, getting a $1-(1-d)^{m-i}+7(m-i)\varepsilon$ distinguisher for $X^{\otimes m-i}, Y^{\otimes m-i}$, which we view as the product of $X^{\otimes m-i-1}, Y^{\otimes m-i-1}$ with X, Y. We first note that our choice of k guarantees that $\varepsilon_k \leq \varepsilon$ for all $1-(1-d)^{m-i}+7(m-i)\varepsilon$. We start by trying to work under the "hard-coding" assumption that

$$\Pr_{z \leftarrow X/Y}\left[\mathrm{adv}^{+}_{A(z,\cdot)}\left(X^{\otimes m-i-1}, Y^{\otimes m-i-1}\right) > 1-(1-d)^{m-i-1}+7(m-i-1)\varepsilon+\varepsilon\right] > \varepsilon$$

and generate a distinguisher for $X^{\otimes m-i-1}, Y^{\otimes m-i-1}$ as follows: Keep sampling $z \leftarrow X/Y$ and estimating $\mathrm{adv}^{+}_{A(z,\cdot)}(X^{\otimes m-i-1}, Y^{\otimes m-i-1})$ using r samples from $X^{\otimes m-i-1}/Y^{\otimes m-i-1}$, until we succeed in finding z with an estimate of at least $1-(1-d)^{m-i-1}+7(m-i-1)\varepsilon+0.5\varepsilon$, then fix this good z in this coordinate and move forward, or stop after q tries if no such z has been found. Using Hoeffding's inequality, for every z, the probability that the estimate's error is greater than $\varepsilon/2$ is at most $2e^{-r\cdot(\varepsilon/2)^2/2}$. If all estimates were $\varepsilon/2$ accurate and a good z has been drawn, the process succeeds in finding a z with advantage of at least $1-(1-d)^{m-i-1}+7(m-i-1)\varepsilon$ and we can move on, so our probability to fail at that, under the above assumption, is at most

$$q \cdot 2e^{-r\cdot\varepsilon^2/32}+(1-\varepsilon)^q \leq 2e^{\log(q/2)-r\cdot\varepsilon^2/32}+e^{-q\cdot\varepsilon} \leq \mathrm{neg}(n)$$

by choosing, say,

$$q = n/\varepsilon = \mathrm{poly}(n, 1/\varepsilon), \quad r = 64n/\varepsilon^3 > (\log(q/2)+n)\cdot 32/\varepsilon^2 = \mathrm{poly}(n, 1/\varepsilon).$$

Hence paying with a time complexity of $t_{m,\varepsilon}\cdot \mathrm{poly}(n, 1/\varepsilon)$ for every coordinate.

If we could not find a good z, we use Lemma 4.1: If we can distinguish $X^{\otimes m-i}, Y^{\otimes m-i}$ with advantage

$$\begin{aligned}
&(1-d)\left(1-(1-d)^{m-i-1}+7(m-i-1)\varepsilon\right)+d+7\varepsilon = \\
&= 1-(1-d)^{m-i}+(1-d)7(m-i-1)\varepsilon+7\varepsilon \leq \\
&\leq 1-(1-d)^{m-i}+7(m-i)\varepsilon \leq \mathrm{adv}^{+}_{A}\left(X^{\otimes m-i}, Y^{\otimes m-i}\right)
\end{aligned}$$

and the assumption about finding a good z to hard-code for $X^{\otimes m-i-1}, Y^{\otimes m-i-1}$ does not hold, then we can build a d-distinguisher for X, Y in time $t_{m,\varepsilon}\cdot \mathrm{poly}(n, k)$. The probability that at some point in the process we failed to hard-code a good z at the $m-i$'th coordinate even though the assumption held is $m(n)\cdot \mathrm{neg}(n) = \mathrm{neg}(n)$. $\square$

We remark this proof is easily generalized to the case where not all pairs in the product are identical, that is, for $\bigotimes X_i$ and $\bigotimes Y_i$, with a distance bound of $(1-\prod_i(1-d_i)+7m\varepsilon)$.

5 Applications

As an application, we consider the amplification of weak oblivious transfer protocols. We briefly explain how our bounds, paired with Yao's XOR lemma, yield a natural generalization in the computational setting of the amplification process presented in [DKS99, Subsection 4.3]. We stress that it is already known that the same amplification process also works computationally [Wul07], yet we find the following approach more straightforward. For the sake of simplicity, we consider the amplification of error-less (p, q)-weak semi-honest 1-2 OT: The receiver with bit c is trying to learn b_c, where b_0, b_1 is the database of the sender. We say the protocol is (p, q) weak if the view of the sender when $c = 0$ is p-indistinguishable from its view when $c = 1$ (equivalently, c is at most p-correlated to the view of the sender), and the view of the receiver when $b_{\overline{c}} = 0$ is q-indistinguishable from its view when $b_{\overline{c}} = 1$. We have an operation called S-Reduce that amplifies indistinguishability against the sender but worsens indistinguishability against the receiver, and an operation called R-Reduce that amplifies indistinguishability against the receiver but worsens indistinguishability against the sender. Our goal is to use them repeatedly one after the other in order to amplify both parameters. It is already shown in [DKS99, Lemma 4] exactly how this is done, so our focus will be on showing that almost the same analysis of the S-Reduce and R-Reduce operations also holds computationally. Let us start with the security of the receiver: In S-Reduce where $c = \bigoplus_i c_i$ we can use Yao's XOR Lemma to show that p is reduced to $p^k + \varepsilon$, and in R-Reduce where $c_i = c$ we use our bound to show that p is increased to at most $1 - (1 - p)^k + \varepsilon$. For the sender, roughly speaking: In S-Reduce, we have a product of k execution pairs that are q-indistinguishable each (whether "the other bit" is 0 or 1), and we can use our bound to get that they are $1 - (1 - q)^k + \varepsilon$ indistinguishable. In R-Reduce, "the other bit" is equal to the XOR of k bits that are at most q-correlated to the execution, independently, so we use the XOR lemma to conclude that "the other bit" is at most $q^k + \varepsilon$ correlated to the execution. We conclude that essentially, the same analysis from the information theoretic setting works, up to an additive ε for each use. Let $p(n)$ be a bound on the total number of calls to the original protocol in the information-theoretic transformation, then all advantages throughout the process are $1/p(n)$-bounded away from 1, otherwise we wouldn't be able to reduce them to negligible. If we choose $\varepsilon' = \varepsilon/p(n)$ then for every advantage d through the process we have $d + \varepsilon' \leq \varepsilon + (1 - \varepsilon)d$, so we can imagine, for a simple analysis, as if every call to either S-Reduce or R-Reduce incurs a chance of ε at failing and revealing everything, and otherwise works exactly like the information-theoretic world. Since the number of calls is polynomial, the total probability of failing is at most $\text{poly}(n) \cdot \varepsilon$ and we can make it as (polynomially) small as we want. There is one issue however - the running time. In the information-theoretic process we make $\log(n)$ calls to S-Reduce and R-Reduce, and each such call, when using Yao's XOR Lemma or the bounds in this paper, decreases the bound on the running time by a division in a polynomial. Therefore, we need the assumption that our weak OT is secure against $n^{O(\log n)}$ adversaries. We believe this stronger requirement may be removed by a more careful analysis (perhaps we again waste too much time on resampling and testing unnecessary values, see Remark 4.1), but without a proof.

References

[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.

[GNW95] Oded Goldreich, Noam Nisan, and Avi Wigderson. On yao's xor-lemma. *Electron. Colloquium Comput. Complex.*, 2(50), 1995.

[Hol12] F. Hollander. Probability theory : The coupling method. 2012.

[Kon12] Aryeh Kontorovich. Obtaining measure concentration from markov contraction. *Markov Processes and Related Fields*, 18(4):613–638, 2012.

[Lev87] Leonid A. Levin. One-way functions and pseudorandom generators. *Comb.*, 7(4):357–363, 1987.

[Wul07] Jürg Wullschleger. Oblivious-transfer amplification. In Moni Naor, editor, *Advances in Cryptology - EUROCRYPT 2007, 26th Annual International Conference on the Theory and Applications of Cryptographic Techniques, Barcelona, Spain, May 20-24, 2007, Proceedings*, volume 4515 of *Lecture Notes in Computer Science*, pages 555–572. Springer, 2007.