Computational Complexity:

A Conceptual Perspective

Oded Goldreich

Department of Computer Science and Applied Mathematics
Weizmann Institute of Science, Rehovot, Israel.

April 12, 2008

7.2.1.2 Yao's XOR Lemma

Having obtained a mildly inapproximable predicate, we wish to obtain a strongly inapproximable one. The information theoretic context provides an appealing suggestion: Suppose that X is a Boolean random variable (representing the mild inapproximability of the aforementioned predicate) that equals 1 with probability ε. Then XORing the outcome of n/ε independent samples of X yields a bit that equals 1 with probability $0.5 \pm \exp(-\Omega(n))$. It is tempting to think that the same should happen in the computational setting. That is, if f is hard to approximate correctly with probability exceeding $1-\varepsilon$ then XORing the output of f on n/ε non-overlapping parts of the input should yield a predicate that is hard to approximate correctly with probability that is non-negligibly higher than $1/2$. The latter assertion turns out to be correct, but (even more than in Section 7.1.2) the proof of the computational phenomenon is considerably more complex than the analysis of the information theoretic analogue.

Theorem 7.34 (Yao's XOR Lemma): *There exist a universal constant $c>0$ such that the following holds. If, for some polynomials p_1 and p_2, the Boolean function f is $(p_1, 1/p_2)$-inapproximable, then the function $F(x_1,...,x_{t(n)}) = \oplus_{i=1}^{t(n)} f(x_i)$, where $t(n) = n \cdot p_2(n)$ and $x_1,...,x_{t(n)} \in \{0,1\}^n$, is p'-inapproximable for $p'(t(n)\cdot n) = p_1(n)^c/t(n)^{1/c}$. Furthermore, the claim holds also if the polynomials p_1 and p_2 are replaced by any integer functions.*

Combining Theorem 7.12 (and Exercise 7.13), and Theorem 7.13, we obtain a proof of Theorem 7.10. (Recall that an alternative proof is presented in §7.2.1.3.)

We note that proving Theorem 7.13 seems more difficult than proving Theorem 7.5 (i.e., the amplification of one-way functions), due to two issues. Firstly, unlike in Theorem 7.5, the computational problems are not in $\mathcal{PC}$ and thus we cannot efficiently recognize correct solutions to them. Secondly, unlike in Theorem 7.5, solutions to instances of the transformed problem do not correspond of the concatenation of solutions for the original instances, but are rather a function of the latter that losses almost all the information about the latter. The proof of Theorem 7.13 presented next deals with each of these two difficulties separately.

Several different proofs of Theorem 7.13 are known. As just stated, the proof that we present is conceptually appealing because it deal separately with two unrelated difficulties. Furthermore, this proof benefits most from the material already presented in Section 7.1. The proof proceeds in two steps:

1. First we prove that the corresponding "direct product" function $P(x_1,...,x_{t(n)}) = (f(x_1),...,f(x_{t(n)}))$ is difficult to compute in a strong average-case sense.
2. Next we establish the desired result by an application of Theorem 7.8.

Thus, given Theorem 7.8, our main focus is on the first step, which is of independent interest (and is thus generalized from Boolean functions to arbitrary ones).

Theorem 7.35 (The Direct Product Lemma): *Let p_1 and p_2 be two polynomials, and suppose that $f : \{0,1\}^* \to \{0,1\}^*$ is such that for every family of p_1-size* *circuits, $\{C_n\}_{n\in\mathbb{N}}$, and all sufficiently large $n \in \mathbb{N}$, it holds that $\Pr[C_n(U_n) \neq f(U_n)] > 1/p_2(n)$. Let $P(x_1, ..., x_{t(n)}) = (f(x_1), ..., f(x_{t(n)}))$, where $x_1, ..., x_{t(n)} \in \{0,1\}^n$ and $t(n) = n \cdot p_2(n)$. Then, for any $\varepsilon' : \mathbb{N} \to (0,1]$, setting p' such that $p'(t(n) \cdot n) = p_1(n)/\mathrm{poly}(t(n)/\varepsilon'(t(n) \cdot n))$, it holds that every family of p'-size circuits, $\{C'_m\}_{m\in\mathbb{N}}$, satisfies $\Pr[C'_m(U_m) = P(U_m)] < \varepsilon'(m)$. Furthermore, the claim holds also if the polynomials p_1 and p_2 are replaced by any integer functions.*

In particular, for an adequate constant $c > 0$, selecting $\varepsilon'(t(n) \cdot n) = p_1(n)^{-c}$, we obtain $p'(t(n) \cdot n) = p_1(n)^c/t(n)^{1/c}$, and so $\varepsilon'(m) \leq 1/p'(m)$.

Deriving Theorem 7.13 from Theorem 7.14. Theorem 7.13 follows from Theorem 7.14 by considering the function $P'(x_1, ..., x_{t(n)}, r) = b(f(x_1) \cdots f(x_{t(n)}), r)$, where f is a Boolean function, $r \in \{0,1\}^{t(n)}$, and $b(y, r)$ is the inner-product modulo 2 of the $t(n)$-bit long strings y and r. Note that, for the corresponding P, we have $P'(x_1, ..., x_{t(n)}, r) \equiv b(P(x_1, ..., x_{t(n)}), r)$, whereas $F(x_1, ..., x_{t(n)}) = P'(x_1, ..., x_{t(n)}, 1^{t(n)})$. Intuitively, the inapproximability of P' should follow from the strong average-case hardness of P (via Theorem 7.8), whereas it should be possible to reduce the approximation of P' to the approximation of F (and thus derive the desired inapproximability of F). Indeed, this intuition does not fail, but detailing the argument seems a bit cumbersome (and so we only provide the clues here). Assuming that f is $(p_1, 1/p_2)$-inapproximable, we first apply Theorem 7.14 (with $\varepsilon'(t(n) \cdot n) = p_1(n)^{-c}$) and then apply Theorem 7.8 (see Exercise 7.14), inferring that P' is p'-inapproximable for $p'(t(n) \cdot n) = p_1(n)^{\Omega(1)}/\mathrm{poly}(t(n))$. The less obvious part of the argument is reducing the approximation of P' to the approximation of F. The key observation is that

$$P'(x_1, ..., x_{t(n)}, r) = F(z_1, ..., z_{t(n)}) \oplus \bigoplus_{i:r_i=0} f(z_i) \tag{7.18}$$

where $z_i = x_i$ if $r_i = 1$ and is an arbitrary n-bit long string otherwise. Now, if somebody provides us with samples of the distribution $(U_n, f(U_n))$, then we can use these samples in the role of the pairs $(z_i, f(z_i))$ for the indices i that satisfy $r_i = 0$. Considering a best choice of such samples (i.e., one for which we obtain the best approximation of P'), we obtain a circuit that approximates P' (by using a circuit that approximates F and the said choices of samples). (The details are left for Exercise 7.17.) Theorem 7.13 follows.

Proving Theorem 7.14. Note that Theorem 7.14 is closely related to Theorem 7.5; see Exercise 7.20 for details. This suggests employing an analogous proof strategy; that is, converting circuits that violate the theorem's conclusion into circuits that violate the theorem's hypothesis. We note, however, that things were much simpler in the context of Theorem 7.5: there we could (efficiently) check whether or not a value contained in the output of the circuit that solves the direct-product problem constitutes a correct answer for the corresponding instance of the basic problem. Lacking such an ability in the current context, we shall have to use such values more carefully. Loosely speaking, we shall take a weighted majority vote among various answers, where the weights reflect our confidence in the correctness of the various answers.

We establish Theorem 7.14 by applying the following lemma that provides quantitative bounds on the feasibility of computing the direct product of two functions. In this lemma, $\{Y_m\}_{m\in\mathbb{N}}$ and $\{Z_m\}_{m\in\mathbb{N}}$ are independent probability ensembles such that $Y_m, Z_m \in \{0,1\}^m$, and $X_n = (Y_{\ell(n)}, Z_{n-\ell(n)})$ for some function $\ell:\mathbb{N}\to\mathbb{N}$. The lemma refers to the success probability of computing the direct product function $F:\{0,1\}^* \to \{0,1\}^*$ defined by $F(yz) = (F_1(y), F_2(z))$, where $|y| = \ell(|yz|)$, when given bounds on the success probability of computing F_1 and F_2 (separately). Needless to say, these probability bounds refer to circuits of certain sizes. (The slackness parameter ε represents a deviation from an idealized result in which the probability of correctly computing F is upper-bounded by the product of the probabilities of correctly computing F_1 and F_2, where tightening this slackness (i.e., decreasing ε) comes at the cost of decreasing the size of the circuit for which the success probability bound holds.) We stress that *the lemma is not symmetric with respect to the two functions*: *it guarantees a stronger* (and in fact lossless) *preservation of circuit sizes for one of the functions* (which is arbitrarily chosen to be F_1).

Lemma 7.36 (Direct Product, a quantitative two argument version): *For $\{Y_m\}$, $\{Z_m\}$, F_1, F_2, ℓ, $\{X_n\}$ and F as in the foregoing, let $\rho_1(\cdot)$ be an upper-bound on the success probability of $s_1(\cdot)$-size circuits in computing F_1 over $\{Y_m\}$. That is, for every such circuit family $\{C_m\}$*

$$\mathsf{Pr}[C_m(Y_m)=F_1(Y_m)] \leq \rho_1(m).$$

Likewise, suppose that $\rho_2(\cdot)$ is an upper-bound on the probability that $s_2(\cdot)$-size circuits compute F_2 over $\{Z_m\}$. Then, for every function $\varepsilon:\mathbb{N}\to\mathbb{R}$, the function ρ defined as

$$\rho(n) \stackrel{\text{def}}{=} \rho_1(\ell(n)) \cdot \rho_2(n-\ell(n)) + \varepsilon(n)$$

is an upper-bound on the probability that families of $s(\cdot)$-size circuits correctly compute F over $\{X_n\}$, where

$$s(n) \stackrel{\text{def}}{=} \min\left\{ s_1(\ell(n))\ ,\ \frac{s_2(n-\ell(n))}{\mathrm{poly}(n/\varepsilon(n))} \right\}.$$

Theorem 7.14 is derived from Lemma 7.15 by using a *careful induction*, which capitalizes on the highly quantitative form of Lemma 7.15 and in particular on the fact that no loss is incurred for one of the two functions that are used. We first detail this argument, and next establish Lemma 7.15 itself.

Deriving Theorem 7.14 from Lemma 7.15. We write $P(x_1, x_2, ..., x_{t(n)})$ as $P^{(t(n))}(x_1, x_2, ..., x_{t(n)})$, where $P^{(i)}(x_1, ..., x_i) = (f(x_1), ..., f(x_i))$ and $P^{(i)}(x_1, ..., x_i) \equiv (P^{(i-1)}(x_1, ..., x_{i-1}), f(x_i))$. For any function ε, *we shall prove by induction on i that circuits of size $s(n) = p_1(n)/\mathrm{poly}(t(n)/\varepsilon(n))$ cannot compute $P^{(i)}(U_{i\cdot n})$ with success probability greater than* $(1-(1/p_2(n))^i + (i-1)\cdot\varepsilon(n)$, where p_1 and p_2 are as in Theorem 7.14. Thus, no $s(n)$-size circuit can compute $P^{(t(n))}(U_{t(n)\cdot n})$ with success probability greater than $(1-(1/p_2(n))^{t(n)}+(t(n)-1)\cdot\varepsilon(n) = \exp(-n)+(t(n)-1)\cdot\varepsilon(n)$. Recalling that this is established for any function ε, Theorem 7.14 follows (by using $\varepsilon(n)=\varepsilon'(t(n)\cdot n)/t(n)$, and observing that the setting $s(n)=p'(t(n)\cdot n)$ satisfies $s(n)=p_1(n)/\mathrm{poly}(t(n)/\varepsilon(n))$).

Turning to the induction itself, we first note that its basis (i.e., $i=1$) is guaranteed by the theorem's hypothesis (i.e., the hypothesis of Theorem 7.14 regarding f). The induction step (i.e., from i to $i+1$) will be proved by using Lemma 7.15 with $F_1=P^{(i)}$ and $F_2=f$, along with the parameter setting $\rho_1^{(i)}(i\cdot n)=(1-(1/p_2(n))^i+(i-1)\cdot\varepsilon(n)$, $s_1^{(i)}(i\cdot n)=s(n)$, $\rho_2^{(i)}(n)=1-(1/p_2(n))$ and $s_2^{(i)}(n)=\mathrm{poly}(n/\varepsilon(n))\cdot s(n)=p_1(n)$. Details follow.

Note that the induction hypothesis (regarding $P^{(i)}$) implies that F_1 satisfies the hypothesis of Lemma 7.15 (w.r.t size $s_1^{(i)}$ and success rate $\rho_1^{(i)}$), whereas the theorem's hypothesis regarding f implies that F_2 satisfies the hypothesis of Lemma 7.15 (w.r.t size $s_2^{(i)}$ and success rate $\rho_2^{(i)}$). Thus, $F=P^{(i+1)}$ satisfies the lemma's conclusion with respect to circuits of size $\min(s_1^{(i)}(i\cdot n), s_2^{(i)}(n)/\mathrm{poly}(n/\varepsilon(n)))=s(n)$ and success rate $\rho_1^{(i)}(i\cdot n)\cdot\rho_2^{(i)}(n)+\varepsilon(n)$ which is upper-bounded by $(1-(1/p_2(n))^{i+1}+i\cdot\varepsilon(n)$. This completes the induction step.

We stress the fact that we used induction for a non-constant number of steps, and that this was enabled by the highly quantitative form of the inductive claim and the small loss incurred by the inductive step. Specifically, the size bound did not decrease during the induction (although we could afford a small additive loss in each step, but not a constant factor loss).[26] Likewise, the success rate suffered an additive increase of $\varepsilon(n)$ in each step, which was accommodated by the inductive claim. Thus, *assuming the correctness of Lemma 7.15, we have established Theorem 7.14.* □

Proof of Lemma 7.15: Proceeding (as usual) by the contrapositive, we consider a family of $s(\cdot)$-size circuits $\{C_n\}_{n\in\mathbb{N}}$ that violates the lemma's conclusion; that is, $\mathsf{Pr}[C_n(X_n)=F(X_n)]>\rho(n)$. We will show how to use such circuits in order to obtain either circuits that violate the lemma's hypothesis regarding F_1 or circuits that violate the lemma's hypothesis regarding F_2. Towards this end, it is instructive to write the success probability of C_n in a conditional form, while denoting the i^{th} output of $C_n(x)$ by $C_n(x)_i$ (i.e., $C_n(x)=(C_n(x)_1,C_n(x)_2)$):

$$\begin{aligned}
&\mathsf{Pr}[C_n(Y_{\ell(n)},Z_{n-\ell(n)})=F(Y_{\ell(n)},Z_{n-\ell(n)})]\\
&\quad=\ \mathsf{Pr}[C_n(Y_{\ell(n)},Z_{n-\ell(n)})_1=F_1(Y_{\ell(n)})]\\
&\qquad\cdot\mathsf{Pr}[C_n(Y_{\ell(n)},Z_{n-\ell(n)})_2=F_2(Z_{n-\ell(n)})\mid C_n(Y_{\ell(n)},Z_{n-\ell(n)})_1=F_1(Y_{\ell(n)})].
\end{aligned}$$

The basic idea is that if the first factor is greater than $\rho_1(\ell(n))$ then we immediately derive a circuit (i.e., $C_n'(y)=C_n(y,Z_{n-\ell(n)})_1$) contradicting the lemma's

[26] Note that if we had $s(n)=\min\{s_1(\ell(n)),s_2(n-\ell(n))\}/2$ (in Lemma 7.15) then the foregoing argument would yield that $P^{(i+1)}$ is hard for circuits of size $s_1^{(i+1)}$ such that $s_1^{(i+1)}((i+1)\cdot n)=\min\{s_1^{(i)}(i\cdot n),s_2(n)\}/2$, which would yield a meaningless result for $P=P^{(t(n))}$ (since $s_1^{(t(n))}(t(n)\cdot n)=\min\{p_1(n),p_2(n)\}/2^{t(n)}$, where $t(n)\gg n$ and $p_i(n)<2^n$).

hypothesis regarding F_1, whereas if the second factor is significantly greater than $\rho_2(n-\ell(n))$ then we can obtain a circuit contradicting the lemma's hypothesis regarding F_2. The treatment of the latter case is indeed not obvious. The idea is that a sufficiently large sample of $(Y_{\ell(n)}, F_1(Y_{\ell(n)}))$, which may be hard-wired into the circuit, allows using the conditional probability space (in such a circuit) towards an attempt to approximate F_2. That is, on input z, we select uniformly a string y satisfying $C_n(y,z)_1 = F_1(y)$ (from the aforementioned sample), and output $C_n(y,z)_2$. For a fixed z, sampling of the conditional space (i.e., y's satisfying $C_n(y,z)_1 = F_1(y)$) is possible provided that $\mathsf{Pr}[C_n(Y_{\ell(n)}, z)_1 = F_1(Y_{\ell(n)})]$ holds with noticeable probability. The last caveat motivates a separate treatment of z's having a noticeable value of $\mathsf{Pr}[C_n(Y_{\ell(n)}, z)_1 = F_1(Y_{\ell(n)})]$ and of the rest of z's (which are essentially ignored). Details follow.

Let us first simplify the notations by fixing a generic n and using the abbreviations $C = C_n$, $\varepsilon = \varepsilon(n)$, $\ell = \ell(n)$, $Y = Y_\ell$, and $Z = Y_{n-\ell}$. We call z good if $\mathsf{Pr}[C(Y,z)_1 = F_1(Y)] \geq \varepsilon/2$ and let G be the set of good z's. Next, rather than considering the event $C(Y,Z) = F(Y,Z)$, we consider the combined event $C(Y,Z) = F(Y,Z) \wedge Z \in G$, which occurs with almost the same probability (up to an additive error term of $\varepsilon/2$). This is the case because, for any $z \notin G$, it holds that

$$\mathsf{Pr}[C(Y,z) = F(Y,z)] \;\leq\; \mathsf{Pr}[C(Y,z)_1 = F_1(Y)] \;<\; \varepsilon/2$$

and thus z's that are not good do not contribute much to $\mathsf{Pr}[C(Y,Z) = F(Y,Z)]$; that is, $\mathsf{Pr}[C(Y,Z) = F(Y,Z) \wedge Z \in G]$ is lower-bounded by $\mathsf{Pr}[C(Y,Z) = F(Y,Z)] - \varepsilon/2$. Using $\mathsf{Pr}[C(Y,z) = F(Y,z)] > \rho(n) = \rho_1(\ell) \cdot \rho_2(n-\ell) + \varepsilon$, we have

$$\mathsf{Pr}[C(Y,Z) = F(Y,Z) \;\wedge\; Z \in G] > \rho_1(\ell) \cdot \rho_2(n-\ell) + \frac{\varepsilon}{2}. \tag{7.19}$$

We proceed according to the forgoing outline, first showing that if $\mathsf{Pr}[C(Y,Z)_1 = F_1(Y)] > \rho_1(\ell)$ then we immediately derive circuits violating the hypothesis concerning F_1. Actually, we prove something stronger (which we will actually need for the other case).

Claim 7.15.1: For every z, it holds that $\mathsf{Pr}[C(Y,z)_1 = F_1(Y)] \leq \rho_1(\ell)$.

Proof: Otherwise, using any $z \in \{0,1\}^{n-\ell}$ that satisfies $\mathsf{Pr}[C(Y,z)_1 = F_1(Y)] > \rho_1(\ell)$, we obtain a circuit $C'(y) \stackrel{\text{def}}{=} C(y,z)_1$ that contradicts the lemma's hypothesis concerning F_1. □

Using Claim 7.15.1, we show how to obtain a circuit that violates the lemma's hypothesis concerning F_2, and doing so we complete the proof of the lemma.

Claim 7.15.2: There exists a circuit C'' of size $s_2(n-\ell)$ such that

$$\begin{aligned} \mathsf{Pr}[C''(Z) = F_2(Z)] \;&\geq\; \frac{\mathsf{Pr}[C(Y,Z) = F(Y,Z) \;\wedge\; Z \in G]}{\rho_1(\ell)} - \frac{\varepsilon}{2} \\ &>\; \rho_2(n-\ell) \end{aligned}$$

Proof: The second inequality is due to Eq. (7.11), and thus we focus on establishing the first inequality. We construct the circuit C'' as suggested in the foregoing outline. Specifically, we take a poly(n/ε)-large sample, denoted S, from the distribution $(Y, F_1(Y))$ and let $C''(z) \stackrel{\text{def}}{=} C(y,z)_2$, where (y,v) is a uniformly selected among the elements of S for which $C(y,z)_1 = v$ holds. Details follow.

Let m be a sufficiently large number that is upper-bounded by a polynomial in n/ε, and consider a random sequence of m pairs, generated by taking m independent samples from the distribution $(Y, F_1(Y))$. We stress that we do not assume here that such a sample, denoted S, can be produced by an efficient (uniform) algorithm (but, jumping ahead, we remark that such a sequence can be fixed non-uniformly). For each $z \in G \subseteq \{0,1\}^{n-\ell}$, we denote by S_z the set of pairs $(y,v) \in S$ for which $C(y,z)_1 = v$. Note that S_z is a random sample of the residual probability space defined by $(Y, F_1(Y))$ conditioned on $C(Y,z)_1 = F_1(Y)$. Also, with overwhelmingly high probability, $|S_z| = \Omega(n/\varepsilon^2)$, because $z \in G$ implies $\Pr[C(Y,z)_1 = F_1(Y)] \geq \varepsilon/2$ and $m = \Omega(n/\varepsilon^3)$.[27] Thus, for each $z \in G$, with overwhelming probability (taken over the choices of S), the sample S_z provides a good approximation to the conditional probability space.[28] In particular, with probability greater than $1 - 2^{-n}$, it holds that

$$\frac{|\{(y,v) \in S_z : C(y,z)_2 = F_2(z)\}|}{|S_z|} \geq \Pr[C(Y,z)_2 = F_2(z) \,|\, C(Y,z)_1 = F_1(Y)] - \frac{\varepsilon}{2}. \tag{7.20}$$

Thus, with positive probability, Eq. (7.12) holds for all $z \in G \subseteq \{0,1\}^{n-\ell}$. The circuit C'' computing F_2 is now defined as follows. The circuit will contain a set $S = \{(y_i, v_i) : i = 1, ..., m\}$ (i.e., S is "hard-wired" into the circuit C'') such that the following two conditions hold:

1. For every $i \in [m]$ it holds that $v_i = F_1(y_i)$.
2. For each good z the set $S_z = \{(y,v) \in S : C(y,z)_1 = v\}$ satisfies Eq. (7.12). (In particular, S_z is not empty for any good z.)

On input z, the circuit C'' first determines the set S_z, by running C for m times and checking, for each $i = 1, ..., m$, whether or not $C(y_i, z) = v_i$. In case S_z is empty, the circuit returns an arbitrary value. Otherwise, the circuit selects uniformly a pair $(y,v) \in S_z$ and outputs $C(y,z)_2$. (The latter random choice can be eliminated by an averaging argument; see Exercise 7.16.) Using the definition of C'' and Eq. (7.12), we have:

$$\begin{aligned}
\Pr[C''(Z) = F_2(Z)] &\geq \sum_{z \in G} \Pr[Z = z] \cdot \Pr[C''(z) = F_2(z)] \\
&= \sum_{z \in G} \Pr[Z = z] \cdot \frac{|\{(y,v) \in S_z : C(y,z)_2 = F_2(z)\}|}{|S_z|}
\end{aligned}$$

[27] Note that the expected size of S_z is $m \cdot \varepsilon/2 = \Omega(n/\varepsilon^2)$. Using Chernoff Bound, we get $\Pr_S[|S_z| < m\varepsilon/4] = \exp(-\Omega(n/\varepsilon^2)) < 2^{-n}$.

[28] For $T_z = \{y : C(y,z)_1 = F_1(y)\}$, we are interested in a sample S' of T_z such that $|\{y \in S' : C(y,z)_2 = F_2(z)\}|/|S'|$ approximates $\Pr[C(Y,z)_2 = F_2(z) \,|\, Y \in T_z]$ up-to an additive term of $\varepsilon/2$. Using Chernoff Bound again, we note that a random $S' \subset T_z$ of size $\Omega(n/\varepsilon^2)$ provides such an approximation with probability greater than $1 - 2^{-n}$.

$$\geq \sum_{z\in G} \mathsf{Pr}[Z=z] \cdot \left(\mathsf{Pr}[C(Y,z)_2=F_2(z) \mid C(Y,z)_1=F_1(Y)] - \frac{\varepsilon}{2}\right)$$
$$= \sum_{z\in G} \mathsf{Pr}[Z=z] \cdot \left(\frac{\mathsf{Pr}[C(Y,z)_2=F_2(z) \wedge C(Y,z)_1=F_1(Y)]}{\mathsf{Pr}[C(Y,z)_1=F_1(Y)]} - \frac{\varepsilon}{2}\right)$$

Next, using Claim 7.15.1, we have:

$$\begin{aligned}\mathsf{Pr}[C''(Z)=F_2(Z)] &\geq \left(\sum_{z\in G} \mathsf{Pr}[Z=z] \cdot \frac{\mathsf{Pr}[C(Y,z)=F(Y,z)]}{\rho_1(\ell)}\right) - \frac{\varepsilon}{2} \\ &= \frac{\mathsf{Pr}[C(Y,Z)=F(Y,Z) \wedge Z\in G]}{\rho_1(\ell)} - \frac{\varepsilon}{2}\end{aligned}$$

Finally, using Eq. (7.11), the claim follows. □

This completes the proof of the lemma. ■

Comments. Firstly, we wish to call attention to the care with which an inductive argument needs to be carried out in the computational setting, especially when a non-constant number of inductive steps is concerned. Indeed, our inductive proof of Theorem 7.14 involves invoking a quantitative lemma (i.e., Lemma 7.15) that allows to keep track of the relevant quantities (e.g., success probability and circuit size) throughout the induction process. Secondly, we mention that Lemma 7.15 (as well as Theorem 7.14) has a uniform complexity version that assumes that one can efficiently sample the distribution $(Y_{\ell(n)}, F_1(Y_{\ell(n)}))$ (resp., $(U_n, f(U_n))$). For details see [102]. Indeed, a good lesson from the proof of Lemma 7.15 is that non-uniform circuits can "effectively sample" any distribution. Lastly, we mention that Theorem 7.5 (the amplification of one-way functions) and Theorem 7.13 (Yao's XOR Lemma) also have (tight) quantitative versions (see, e.g., [91, Sec. 2.3.2] and [102, Sec. 3], respectively).