The Randomness Complexity of Parallel Repetition

Kai-Min Chung* Rafael Pass†

September 1, 2011

Abstract

Consider a m-round interactive protocol with soundness error $1/2$. How much extra randomness is required to decrease the soundness error to δ through parallel repetition? Previous work, initiated by Bellare, Goldreich and Goldwasser, shows that for *public-coin* interactive protocols with *statistical soundness*, $m \cdot O(\log(1/\delta))$ bits of extra randomness suffices. In this work, we initiate a more general study of the above question.

- We establish the first derandomized parallel repetition theorem for public-coin interactive protocols with *computational soundness* (a.k.a. arguments). The parameters of our result essentially matches the earlier works in the information-theoretic setting.
- We show that obtaining even a sub-linear dependency on the number of rounds m (i.e., $o(m) \cdot \log(1/\delta)$) is impossible in the information-theoretic, and requires the existence of one-way functions in the computational setting.
- We show that non-trivial derandomized parallel repetition for private-coin protocols is impossible in the information-theoretic setting and requires the existence of one-way functions in the computational setting.

These results are tight in the sense that parallel repetition theorems in the computational setting can trivially be derandomized using pseudorandom generators, which are implied by the existence of one-way functions.

Keywords: interactive protocols, derandomization, parallel repetition, soundness amplification, randomness extractors

*Chung is supperted by a Simons Foundation Fellowship. Department of Computer Science, Cornell University, Upson Hall 4108, Ithaca, NY 14850, USA. http://www.cs.cornell.edu/~chung/. chung@cs.cornell.edu.

†Pass is supported in part by a Alfred P. Sloan Fellowship, Microsoft New Faculty Fellowship, NSF CAREER Award CCF-0746990, AFOSR YIP Award FA9550-10-1-0093, and DARPA and AFRL under contract FA8750-11-2-0211. The views and conclusions contained in this document are those of the authors and should not be interpreted as representing the official policies, either expressed or implied, of the Defense Advanced Research Projects Agency or the US government. Department of Computer Science, Cornell University, Upson Hall, Ithaca, NY 14850, USA. rafael@cs.cornell.edu.

1 Introduction

In an *interactive protocol*, two parties, called the prover P and the verifier V, receive some common inputs and perhaps some private inputs, toss some random coins, and interact with each other following some prescribed protocol. The prover attempts to convince the verifier V than a certain input x is in a language L. The soundness property of an interactive proof requires that when $x \notin L$, V will only accept with bounded (error) probability, even when he interacts with a certain class of adversarial cheating provers P^*. Such an upper bound on the error probability of V is called the *soundness error* of the protocol.

Two versions of the soundness property, *statistical soundness* and *computational soundness*, are commonly studied. *Statistical soundness* requires the upper bound on V's error probability (to accept incorrectly) to hold against computationally unbounded adversarial provers, whereas *computational soundness* only requires the soundness to hold against probabilistic polynomial time (PPT). Computational soundness is a weaker requirement than statistical soundness. However, in many settings, requiring only computational soundness allows us to improve the efficiency (e.g., in round complexity or communication complexity). When statistical soundness holds, the protocol is referred to as an interactive *proof*, whereas if only computational soundness holds, the protocol is referred to as an interactive *arguments*.

Ideally, we would like the soundness error to be negligible. But, in many settings, our starting point is a protocol with somewhat large soundness error. For example, to design an interactive proof for a language L, it may be easier to first design a protocol with soundness error $1/2$. This leads to the question of *soundness amplification*: Is there a way to decrease the soundness error of a given protocol?

A natural approach to soundness amplification is by *parallel repetition*, i.e., many instances of the protocols are executed in parallel, and the verifier accepts iff all instances accept. We denote by $\Pi^k = (P^k, V^k)$ the k-fold parallel repetition of a protocol $\Pi = (P, V)$. It is known that parallel repetition decrease soundness error at an optimal rate (i.e., from $1/2$ to $1/2^k$ under k-fold repetition) for interactive proofs [BM88, Gol01]. For the arguments case, optimal parallel repetition theorems are known for three-message private-coin protocols [BIN97, CHS05] and for public-coin protocols (i.e., protocols where the verifier does not keep secret) [PV07, HPWP10, CL10].

Note, however, that the parallelized verifier V^k uses k times more randomness than the original verifier V. We may also consider a *derandomized* parallel verifier V^k_G who generates V^k's coins by applying the function G to a "short" random seed; we usually refer to the function G as a *derandomizer*. A natural question that arises is thus:

> *Given an m-round protocol with soundness error, say $1/2$, how much extra randomness is required to decrease the soundness error to δ through parallel repetition?*

This question was first addressed by Bellare, Goldreich, and Goldwasser [BGG93] in 1990. They established that for public-coin interactive proofs with verifier sending t-bits messages in each round, $m \cdot (t + O(\log(1/\delta))$ bits of randomness suffice. Their construction is based on the notion of an *averaging sampler* introduced by Bellare and Rompel [BR94]. Relying on more recent constructions of averaging samplers [Zuc97, RVW01], the extra randomness required can be reduced to $m \cdot O(\log(m/\delta))$.

In this work, we initiate a more general study of the above question. More precisely, we focus on 1) extending the above treatment to both computationally sound protocols and private

coin protocols, and 2) investigating the randomness complexity required to perform soundness amplification through parallel repetition in all of these settings.

1.1 Our Results

We establish the first derandomized parallel repetition theorem for public-coin interactive protocols with *computational soundness*. The parameters of our result essentially matches the earlier works in the information-theoretic setting.

Theorem 1 (informal) *For every m, δ, there exists a polynomial k and a polynomial-time computable derandomizer G, such that for any m-round public-coin argument (P, V) with soundness error $1/2$, the protocol (P^k, V_G^k) has soundness error $\delta + \mathsf{ngl}$ and uses only $m \cdot O(\log(m/\delta))$-bits of extra randomness (compared to the original verifier V).*

We mention that the framework of Bellare *et. al.* [BGG93] does not apply to the computational setting. Rather, we develop a new framework for establishing the above theorem. Our framework applies to both the computational and the information-theoretic settings (i.e., for both proofs and arguments); incidentally, our analysis actually slightly improves the concrete parameters also of earlier works in the information-theoretic setting. Our main technique for proving Theorem 1 is establishing a connection between certain types of, so-called *invertible*, randomness extractors for high entropy sources, and derandomized parallel repetition; roughly speaking, we say that a strong randomness extractor is invertible if we can, given any r, y, efficiently sample a uniform x such that $\mathrm{Ext}(x, r) = y$. The extra randomness required in the statement of Theorem 1 corresponds to m times the "entropy-loss" of the invertible extractor used; the parameters in Theorem 1 are then obtained by relying on extractors due to Reingold *et. al.* [RVW01] and Guruswami *et. al.* [GUV09].

We mention that in the computational setting, the existence of a cryptographic pseudorandom generator (PRG) [BM84, Yao82], which in turn is implied by one-way functions [HILL99] can trivially be used to derandomize interactive protocols. Thus assuming the existence of one-way functions, derandomized parallel repetition doesn't require any additional randomness. The obvious advantage of Theorem 1 is that the derandomization is *unconditional*. A more subtle advantage of our approach is that the resulting protocol remains sound, even if the verifier, at each round, reveals all its random coin tosses. This has two benefits: By slightly changing the prover strategy (to expand the verifier messages using our derandomizer G) we can get a derandomized protocol that still is public coin and only increases the verifier communication complexity by an *additive* term of $m \cdot O(\log(m/\delta))$, whereas the PRG solution is private-coin, and increases the verifier communication complexity by a *multiplicative* factor of $k = \log(1/\delta)$.

We next show that obtaining even a sub-linear dependency on the number of rounds m is impossible in the information-theoretic, and requires the existence of one-way functions in the computational setting.

Theorem 2 (informal) *Consider some $m > 0, \delta > 0$. There does not exist a derandomizer G and a polynomial k such that for every m-round public-coin proof (P, V) with soundness error $1/2$, the protocol (P^k, V_G^k) has soundness error δ and uses only $m \cdot \log(1/\delta) - O(1)$-bits of extra randomness. For the case of arguments, the existence of such a derandomizer instead implies the existence of one-way functions.*

Our lower bound is actually a bit stronger than stated: we present a specific protocol for which every derandomizer must use at least $m \cdot \log(1/\delta) - O(1)$-bits of extra randomness. Additionally, note that we cannot hope to get an unconditional lower-bound in the computational setting, since, as mentioned, assuming the existence of a PRG, parallel repetition theorems can be derandomized without any additional randomness. However, if we require that the derandomizer protocol remains secure even if the verifier reveals its random coins at each round (which is the case for the derandomized protocols from our upper-bound), then the above lower bound holds unconditionally also in the computational setting.

We finally show that non-trivial derandomized parallel repetition for private-coin protocols is impossible in the information-theoretic setting, and requires proving the existence of one-way functions in the computational setting.

Theorem 3 (informal) *Consider some $m \geq 3, \delta > 0$. There does not exist a derandomizer G and a polynomial k such that for every m-round proof (P, V) with soundness error $1/2$, the protocol (P^k, V_G^k) has soundness error δ and uses $o(t \cdot \log(1/\delta))$ bits of randomness, where t is the random complexity of V. For the case of arguments, the existence of such a derandomizer instead implies the existence of one-way functions.*

As before, we here present a specific protocol for which every derandomizer must use at least $O(t \cdot \log(1/\delta))$ bits of randomness. Again, this results is "tight" as in the computational setting, pseudorandom generators can be used to trivially derandomize parallel repetition theorems also for private-coin interactive protocols.

Future Work Our results establish essentially tight upper and lower bounds on the randomness/soundness trade-off for soundness amplification through parallel repetition. We have not focused on minimizing the number of parallel repetitions needed for such randomness efficient soundness amplification; in other words, we have not focused on minimizing prover communication complexity. In our framework the number of parallel repetitions is 2 to the power of the seed length of the extractor we use. So one method for decreasing the number of parallel repetitions (and thus improving prover communication comlexity) would be to improve the seed length of known (invertible) extractors for high entropy sources, without hurting the entropy loss. We leave open the question of determining the trade-off between randomness, soundness and the number of parallel repetitions.

We have only focused on methods for derandomizing the *original* verifier strategy; that is, we consider a derandomizer that generates random coins, and then runs parallel instances of the original verifier. A less restrictive approach would be to allow the derandomizer to arbitrarily change the parallel verifier strategy (i.e., both how its messages are generated, and its acceptance rule) subject to preserving the completeness condition with respect to the original prover strategy. Our lower bounds do not extend to this setting. We leave open the question of whether more randomness efficient parallel repetition can be performed in this model.

Finally, in this paper we have only focused on establishing direct product theorems, as opposed to "Chernoff-type" theorems. Nonetheless, although we haven't checked the details, it seems that our techniques directly extend also to the Chernoff bound setting by plugging them into the framework of [HPWP10]; we leave the details for future work.

1.2 Organization of the Paper

Section 2 presents some notations and preliminaries on interactive protocols. We define the notion of derandomizers in Section 3. In Section 4, we present our main derandomized parallel repetition theorem (formal version of Theorem 1) and our new framework for proving the theorem; the formal proof is deferred to Section 6. We introduce preliminaries on randomness extractors and state the extractors we use in Section 5. Finally, we present our lower bounds in Section 7 and 8.

2 Preliminaries

We use $\mathbb{N}$ to denote the natural numbers $\{0, 1, \dots\}$, $[n]$ to denote the set $\{1, \dots, n\}$, and $|x|$ to denote the length of a string $x \in \{0,1\}^*$. By ngl, we mean a function negligible in n (i.e., $1/n^{\omega(1)}$). All log's are base 2 unless otherwise specified. For random variables X, Y, we use $\mathsf{P}_X(x)$ to denote $\Pr[X = x]$ and use $\mathsf{P}_{X|Y}(x|y)$ to denote $\Pr[X = x|Y = y]$.

2.1 Interactive Protocols

An **interactive protocol** Π is a pair of interactive Turing machines, (P, V), where V is probabilistic polynomial time (PPT). P is called the prover, while V is called the verifier. $\langle P, V\rangle(z)$ denotes the random variable (over the randomness of P and V) representing V's output at the end of the interaction on common input z. We often omit the input z and simply write $\langle P, V\rangle$. We count one round as two message exchanges from one party to the other and back, so a m-round protocol consists of $2m$ messages. Π is **public-coin** if the verifier's messages are simply independent uniformly random coins.

We are interested in the trade-off between the randomness complexity and the soundness property of protocols. The **randomness complexity** of a protocol Π is simply the number of random coins tossed by the verifier. A protocol Π for a language L has **statistical** (resp., **computational**) **soundness error** ε, if for every $z \notin L$, for every unbounded (resp., PPT) adversarial prover P^*,

$$\Pr[\langle P^*, V\rangle(z) = 1] \leq \varepsilon(|z|).$$

Π is referred to as an **interactive proof** (resp., **interactive argument**) for L if Π has bounded statistical (reps., computational) soundness error.

Let $\Pi^k = (P^k, V^k)$ denote the **k-fold parallel repetition** of Π, where k independent copies of Π are executed in parallel and at the end of interaction, $V^k = (V_1, \dots, V_k)$ accepts iff all sub-verifiers V_i's accept.

3 Definition of Derandomizers

In this section, we introduce our framework for studying the randomness complexity of parallel repetition by formalizing the notion of a *derandomizer*.

Recall that in a parallel repetition Π^k of an interactive protocol Π, k independent copies of the protocols are executed in parallel, and the parallel verifier V^k accepts iff all sub-verifiers V_i's accept. A *derandomizer* is simply an efficiently computable function G that on input a short seed U_s of s bits randomness, generates a random tape for the parallel verifier V^k, i.e., $G : \{0,1\}^s \to \{0,1\}^{k \cdot t}$, where t is the randomness complexity of Π. In other words, G derandomizes the random tape of

V^k. This induces a derandomized parallel protocol Π_G where the derandomized verifier V_G uses the derandomized random tape generated by G to interact with P^k.

Definition 4 (Derandomizer for Interactive Protocols) *Let Π be an interactive proof/argument with soundness error ε and randomness complexity t. A **derandomizer** for Π is simply an efficiently computable function $G : \{0,1\}^s \to \{0,1\}^{k \cdot t}$. G induces a derandomized k-fold parallel repetition $\Pi_G = (P^k, V_G)$ of Π, where the derandomized parallel verifier V_G first generates $G(U_s)$ using a uniform seed U_s, and then emulates V^k with coins $G(U_s)$.*

*We say that G is a $(\varepsilon \mapsto \delta)$-**statistically-sound derandomizer** (resp., **computationally-sound derandomizer**) **for** Π if Π_G has statistical (resp., computational) soundness error at most δ.*

Note that our definition of a derandomizer is general in the sense that we allow the derandomizer to depend on the protocol arbitrarily. Allowing a general definition of a derandomizer makes our lower bound results stronger.

On the other hand, the derandomizer G for public-coin protocols we construct in Theorem 6 works for *all* public-coin protocols of some fixed round complexity and message length. Additionally, as we shall see, there exists an efficient algorithm that given just the rounds complexity m and message length t, outputs such a protocol "oblivious" derandomizer G. Let us proceed to formalizing the notion of an oblivious derandomizer.

Definition 5 *Let $\mathcal{C}$ be a class of interactive protocols. G is an $(\varepsilon \mapsto \delta)$-**oblivous statistically-sound** (resp., **computationally-sound**) **derandomizer for** $\mathcal{C}$ if for every protocol $\Pi \in \mathcal{C}$ (for some language L) with statistical (resp., computational) soundness error at most ε, the derandomized parallel protocol Π_G has statistical (resp., computational) soundness error at most δ.*

4 A Derandomized Parallel Repetition Theorem

In this section, we state a formal version of Theorem 1, and give a detailed proof overview. We shall prove Theorem 6 formally in Section 6, after presenting some necessary preliminary on randomness extractors in Section 5.

For simplicity of exposition, we consider m-round public-coin protocols where the verifier sends a t-bit random message at each round. We refer to such protocols as (m,t)-**public-coin protocols**. We focus on constructing efficient and oblivious derandomizers for the class of (m,t)-public-coin protocols.

Theorem 6 *For every polynomially bounded $m, t : \mathbb{N} \to \mathbb{N}$, every $\varepsilon, \delta : \mathbb{N} \to (0,1)$, there exists a $((1-\varepsilon) \mapsto \delta)$-oblivious statistically-sound derandomizer (resp., $((1-\varepsilon) \mapsto \delta + \mathsf{ngl})$-oblivious computationally-sound derandomizer) $G : \{0,1\}^s \to \{0,1\}^{k \cdot m \cdot t}$ for (m,t)-public-coin interactive proofs (resp., arguments) with randomness complexity*

$$s = m \cdot t + m \cdot O(\log(1/\delta)) + O(\log(m/\varepsilon)),$$

and number of repetition $k = \mathrm{poly}(m, 1/\varepsilon, \log(1/\delta))$. Furthermore, for the case of interactive proofs, the constant in the $O(\log(1/\delta))$ term for the randomness complexity can be set to $(1+\gamma)$ for an arbitrarily small constant γ.

Our approach to derandomize parallel repetition of interactive protocols can be viewed as derandomizing the *analysis* of a parallel repetition theorem of Håstad, Pass, Pietrzak, and Wikström [HPWP10]. Therefore, we start by a high level overview of their result and their proof.

Håstad *et. al.* proved an efficient parallel repetition theorem for public-coin interactive arguments, stating that parallel repetition decreases the soundness error at an exponential rate. They proved the theorem by an efficient black-box reduction. Namely, suppose there exists an adversary P^{k*} for the parallel protocol Π^k that breaks the soundness with probability δ, then there exists an adversary P^* for the original protocol Π such that, given oracle access to P^{k*}, P^* can break the soundness with a much higher probability $\varepsilon \gg \delta$, which, in the contrapositive form, shows that the soundness error goes down from ε to δ under parallel repetition.

A general framework for such reductions is for the single-instance adversary P^* to interact with the verifier V by simulating the interaction between P^{k*} and V^k, where the external verifier V is embedded in some coordinate V_i of V^k, and P^* simulates P^{k*} and the remaining $k-1$ sub-verifiers (denoted by V_{-i}) of V^k internally and forwards P^{k*}'s messages at coordinate i to V. The task of P^* is to decide which coordinate to embed V, and to choose $k-1$ messages of V_{-i} at each round.

Håstad *et. al.* showed that the following **rejection sampling** strategy P^*_{rej} works. P^*_{rej} simply selects a uniformly random coordinate $i \in [k]$ to embed V. At each round j, upon receiving the external verifier V's message $x_{j,i}$, P^* repeatedly samples a random continuation of (P^{k*}, V^k) until he finds an **accepting continuation**, i.e., V^k accepts at the end of interaction. Then P^* selects the corresponding messages in the accepting continuation as the messages of V_{-i} at round j, and forward the corresponding response of P^{k*} to V. If P^* fails to find an accepting continuation, then P^* simply aborts.

To show that the rejection sampling strategy works, we consider a mental experiment, where the external verifier is also aware of the simulated interaction (P^{k*}, V^k), and also uses the rejection sampling strategy to selects his message $x_{j,i}$ at each round j. Namely, the verifier also repeatedly samples a random continuation of (P^{k*}, V^k) until a accepting continuation is found, and forwards the corresponding message in the accepting continuation to P^*_{rej}. We refer to this mental experiment as the **ideal experiment** (P^*_{rej}, V^*_{rej}), in contrast to the **real experiment** (P^*_{rej}, V).

Now, a key observation is that, both parties performing rejection sampling strategy is equivalent to them jointly sampling an accepting interaction. Therefore, in the ideal experiment, the verifier accepts with probability 1 at the end of interaction. The crux of the analysis is to show that the real experiment and the ideal experiment are *statistically close*, using a sampling lemma by Raz [Raz98].

Specifically, let E denote the event of accepting interaction. Recall that $\Pr[E] \geq \delta$. Consider the distribution of V^k's first message $\vec{X}_1 = (X_{1,1}, \ldots, X_{1,k})$. Since Π is public-coin, $\vec{X}_1$ is simply a uniform distribution. In the ideal experiment, P^*_{rej} and V^*_{rej} jointly select the first message from the conditional distribution $\vec{X}_1|_E$. In the real experiment, V selects $X_{1,i}$ uniformly without conditioning, and then P^*_{rej} selects the messages of V_{-i} according to distribution $X_{1,-i}|_{E,X_{1,i}}$. Recall that the coordinate i is uniformly random, the statistical distance of the first message between the two experiments is

$$\frac{1}{k}\sum_{i=1}^{k} \mathbf{SD}(X_{1,i}|_E, X_{1,i}),$$

which can be upper bounded by the following Raz's Lemma.

Lemma 7 (Raz's Lemma [Raz98]) *Let $X_1, \ldots, X_k$ be* independent *random variables on a finite domain U. Let E be an event over $\vec{X} = (X_1, \ldots, X_k)$. We have*

$$\frac{1}{k} \cdot \sum_{i=1}^{k} \mathbf{SD}(X_i|_E, X_i) \leq \sqrt{\frac{1}{k} \cdot \log \frac{1}{\Pr[E]}}.$$

Applying the Raz's Lemma to every round together with a hybrid argument, one can show that the statistical distance between the ideal and the real experiments is at most $m \cdot \sqrt{(\log(1/\delta))/k}$, and hence[1]

$$\Pr[(P^*, V) = 1] \geq 1 - m \cdot \sqrt{\frac{\log(1/\delta)}{k}}.$$

It turns out that to derandomize the parallel repetition theorem, it suffices to derandomize the Raz's Lemma in the sense of identifying derandomized distribution $\vec{X} = (X_1, \ldots, X_k)$ such that the conclusion of the lemma remains true. Note that the lemma is applied to the special case where X_i's are uniform. As observed by Shaltiel [Sha10], in this special case, the Raz's Lemma can be derandomized using strong randomness extractors. Recall that $\mathrm{Ext} : \{0,1\}^n \times [k] \to \{0,1\}^t$ is a strong $(n-\Delta, \varepsilon)$-randomness extractor if for every sources X with min-entropy $\mathrm{H}_\infty(X) \geq n-\Delta$, the distribution $(I, \mathrm{Ext}(X, I))$ is ε-close to (I, U_t) in statistical distance, where I is uniformly random seed over $[k]$. Note that

$$\mathbf{SD}((I, \mathrm{Ext}(X, I)), (I, U_t)) = \frac{1}{k} \cdot \sum_{i=1}^{k} \mathbf{SD}(\mathrm{Ext}(X, i), U_t).$$

Therefore, consider the distribution $(X_1, \ldots, X_k) \triangleq (\mathrm{Ext}(U_n, 1), \ldots, \mathrm{Ext}(U_n, k))$ and an event E over $(X_1, \ldots, X_k)$ with $\Pr[E] \geq 2^{-\Delta}$. Let $X = U_n|_E$, and note that $\mathrm{H}_\infty(X) \geq n - \log(1/\Pr[E]) \geq n - \Delta$. By the property of the extractor,

$$\mathbf{SD}((I, \mathrm{Ext}(X, I)), (I, U_t)) = \frac{1}{k} \cdot \sum_{i=1}^{k} \mathbf{SD}(X_i|_E, U_t) \leq \varepsilon,$$

which is the desired conclusion we want from the Raz's Lemma. Therefore, the parallel verifier V^k can be derandomized by replacing the independent messages with the outputs of a strong randomness extractor. Namely, at each round, the derandomized verifier V_G samples $X_j \equiv U_n$ and sends $(\mathrm{Ext}(X_j, 1), \ldots, \mathrm{Ext}(X_j, k))$ to P^k.

Note that the randomness extractor we need is only required to extract randomness from sources with high min-entropy. On the other hand, we want to minimize the entropy loss $n - \Delta - t$ (corresponds to the extra randomness used) and the seed length (corresponds to the number of repetition). Randomness extractors for high min-entropy sources with very good parameters has been constructed by Reingold, Vadhan, and Wigderson [RVW01].

However, there are two additional issues that we need to address. First, recall that to finish proof of parallel repetition theorem, we need to apply the Raz's Lemma to each round together with a hybrid argument. Except for the first round, there is already some partial interaction h that is determined before the j-th message $\vec{X}_j$ is chosen. To handle this issue, Håstad *et. al.* instead used the following generalized Raz's Lemma (formalized by Holenstein [Hol09]).

[1]The analysis presented here is slightly oversimplified and omits some technical details. Nevertheless, those technical details are irrelevant for the purpose of derandomization and are ignored from the informal discussion here.

Lemma 8 (Generalized Raz's Lemma [Raz98]) *Let $H, X_1, \ldots, X_k$ be independent random variables such that $X_i \equiv U_t$ are uniform*[2] *for every $i \in [k]$. Let E be an event over $(H, X_1, \ldots, X_k)$ with $\Pr[E] \geq \delta$. Then*

$$\frac{1}{k}\sum_{i=1}^{k} \mathbf{SD}((H, X_i)|_E, (H|_E, U_t)) \leq \sqrt{\frac{\log(1/\delta)}{k}}.$$

We observe that, the generalized Raz's Lemma can be derandomized using an average-case version of randomness extractor, introduced by Dodis, Ostrovsky, Reyzin, and Smith [DORS08].[3] Informally, an extractor $\mathrm{Ext} : \{0,1\}^n \times [k] \to \{0,1\}^t$ is a strong average-case $(n-\Delta, \varepsilon)$-randomness extractor if for every sources X with "average conditional min-entropy" $\mathrm{H}_\infty(X|H) \geq n - \Delta$ conditioned on some distribution H, the distribution $(I, H, \mathrm{Ext}(X, I))$ is ε-close to (I, H, U_t). Namely, Ext can extract t bits of randomness from X even when X only has sufficient average conditional min-entropy.

Now, let H and U_n be independent random variables and let $(X_1, \ldots, X_k) = (\mathrm{Ext}(U_n, 1), \ldots, \mathrm{Ext}(U_n, k))$. Let E be an event over (H, U_n) with $\Pr[E] \geq 2^{-\Delta}$, and let $(\tilde{H}, X) = (H, U_n)|_E$. It can be shown that $\mathrm{H}_\infty(X|\tilde{H}) \geq n - \Delta$, and hence, the property of average-case extractor implies

$$\begin{aligned}
&\mathbf{SD}((I, \tilde{H}, \mathrm{Ext}(X, I)), (I, \tilde{H}, U_t)) \\
&= \frac{1}{k} \cdot \sum_{i=1}^{k} \mathbf{SD}((\tilde{H}, \mathrm{Ext}(X, i)), (\tilde{H}, U_t)) \\
&= \frac{1}{k} \cdot \sum_{i=1}^{k} \mathbf{SD}((H, X_i)|_E, (H|_E, U_t)) \\
&\leq \varepsilon,
\end{aligned}$$

which is the desired conclusion we want from the generalized Raz's Lemma. Therefore, the analysis of Håstad *et. al.* can go through if we derandomize the parallel verifier using average-case extractor.

In fact, as proved by [DORS08], every ordinary randomness extractor is also an average-case extractor with a small loss in parameter. Furthermore, Vadhan [Vad11a] observed that such a parameter loss is actually not necessary. Therefore, the requirement of average-case extractor is not an extra requirement.

The second issue is about the efficiency of the rejection sampling strategy. Note that proving parallel repetition theorem for interactive arguments requires efficient reductions. Recall that upon receiving the external verifier V's message $x_{j,i}$, P^*_{rej} needs to sample a random continuation of (P^{k*}, V_G) in order to find an accepting continuation. This requires P^*_{rej} to generates the remaining $k-1$ subverifiers' message, conditioned on the i-th verifier's message is $x_{j,i}$.

Recall that V_G generates $x_{j,i}$ according to the distribution $\mathrm{Ext}(U_n, i)$. To ensure that the rejection sampling strategy can be implemented efficiently, we require the extractor to satisfy the following *invertible* property: for every output y and seed i, one can efficiently sample a random

[2] As in the basic Raz's Lemma, the generalized Raz's Lemma holds without requiring that X_i being uniform. We state the lemma for uniform X_i's since we only apply the lemma for this case, and it makes the connection to extractors more explicit.

[3] Dodis *et. al.* defined the notion for standard (non-strong) randomness extractor. We require the strong version definition which can be defined readily.

input x such that $\mathrm{Ext}(x,i)=y$. Fortunately, while not every randomness extractor satisfies the reconstructiblity property, we observe that the high min-entropy extractor constructed in Proposition 5 of Reingold *et. al.* [RVW01] is invertible and achieves very good parameters.

To summarize, we show that parallel repetition of public-coin protocols can be derandomized using randomness extractors that are strong, average-case, and invertible.

5 Invertible Randomness Extractors

We start with the definition of standard seeded randomness extractor.

Definition 9 (Min-entropy) *Let X be a finite distribution. The **min-entropy** of X is*

$$\mathrm{H}_\infty(X) = -\log \max_{x\in\mathrm{supp}(X)} \mathsf{P}_X(x).$$

Definition 10 (Strong Randomness Extractors) *A function $\mathrm{Ext}:\{0,1\}^n\times\{0,1\}^d\to\{0,1\}^m$ is a **strong** (k,ε)**-extractor** if for every source X over $\{0,1\}^n$ with $\mathrm{H}_\infty(X)\geq k$, the distribution $(U_d,\mathrm{Ext}(X,U_d))$ is ε-close to the uniform distribution (U_d,U_m).*

As in the above definition, when we discuss the extractors, we follow the convention in the literature, i.e., n is the length of the source X, k is the min-entropy of X, d is the seed length, m is the output length, and ε is the error parameter. Furthermore, $\Delta \triangleq n-k$ is the **entropy deficiency** of X, and $\Lambda \triangleq k-m$ (resp., $\Lambda \triangleq k+d-m$) is the **entropy loss** of a strong (resp., non-strong) extractor. An extractor Ext is **explicit** if Ext can be computed in polynomial time.

We need explicit strong randomness extractors for *high* min-entropy source with *short* seed length and *small* entropy loss (and some additional properties we discuss later). Specifically, we think of the entropy deficiency Δ as independent of n, and we require the seed length to be linear in $\log\Delta$ and $\log(1/\varepsilon)$ (so that in our application, the number of repetition is $\mathrm{poly}(\Delta/\varepsilon)$).

As mentioned, we need more general "average-case" extractors, which are able to extract randomness from sources with only sufficient "(average) conditional min-entropy". The following notions are introduced in [DORS08].

Definition 11 (Conditional Min-entropy) *Let (H,X) be a finite distribution. The **(average) conditional min-entropy** of $(X|H)$ is*

$$\mathrm{H}_\infty(X|H) = -\log\left(\mathop{\mathrm{E}}_{h\leftarrow H}\left[2^{-\mathrm{H}_\infty(X|_{H=h})}\right]\right) = -\log\left(\sum_h \mathsf{P}_H(h)\cdot\max_x \mathsf{P}_{X|H}(x|h)\right).$$

Definition 12 (Average-case Extractor) *A function $\mathrm{Ext}:\{0,1\}^n\times\{0,1\}^d\to\{0,1\}^m$ is a **strong average-case** (k,ε)**-extractor** if for every joint distribution (H,X) over $\{0,1\}^*\times\{0,1\}^n$ with $\mathrm{H}_\infty(X|H)\geq k$, the distribution $(U_d,H,\mathrm{Ext}(X,U_d))$ is ε-close to the distribution (U_d,H,U_m).*

Although average-case extractors seem more general, Dodis *et. al.* [DORS08] showed that, if $\mathrm{H}_\infty(X|W)\geq k+\log(1/\varepsilon')$, then a (k,ε)-extractor is still able to extract the randomness from X, at the price of increasing the error by ε'. Furthermore, Vadhan [Vad11b] observed that such a $\log(1/\varepsilon')$ loss in parameter is not necessary, as stated in the following lemma.

Lemma 13 ([Vad11b]) *If* $\mathrm{Ext} : \{0,1\}^n \times \{0,1\}^d \to \{0,1\}^m$ *is a strong* (k,ε)*-randomness extractor, then* Ext *is a strong average-case* $(k, 3\varepsilon)$ *extractor.*

In addition, we need the extractor to have the following invertible property.

Definition 14 (Invertibility) *An extractor* $\mathrm{Ext} : \{0,1\}^n \times \{0,1\}^d \to \{0,1\}^m$ *is* ***invertible*** *if there exists an efficient algorithm such that on input* $y \in \{0,1\}^m$ *and* $r \in \{0,1\}^d$*, outputs a uniformly random* $x \in \{0,1\}^n$ *such that* $\mathrm{Ext}(x,r) = y$*.*

We use high-min-entropy extractors from Reingold, Vadhan, Wigderson [RVW01]. Their main extractor construction, when plugged-in the best known explicit strong randomness extractor of Guruswami, Umans, and Vadhan [GUV09] (for general source instead of high-min-entropy ones), yields the following randomness extractor.

Lemma 15 ([RVW01, GUV09]) *Let* $\gamma > 0$ *be an arbitrarily small constant. For every* Δ, ε *and for every sufficiently large* $n \geq (1+\gamma)\cdot\Delta + \Omega(\log(1/\varepsilon))$*, there exists an explicit strong*[4] $(n-\Delta,\varepsilon)$*-randomness extractor* $\mathrm{Ext} : \{0,1\}^n \times \{0,1\}^d \to \{0,1\}^m$ *with seed length* $d = O(\log\Delta + \log(1/\varepsilon))$ *and entropy loss* $\gamma\cdot\Delta + O(\log(1/\varepsilon))$*.*

Note that the above extractor achieves very good seed length and entropy loss, in compared to the information theoretic limit of $\log\Delta + 2\log(1/\varepsilon) - O(1)$ on the seed length and $2\log(1/\varepsilon) - O(1)$ on the entropy loss.

However, one issue with the this extractor is that it seems not invertible. Fortunately, we observed that a more basic version of their construction (specifically, construction stated in Proposition 6.5 of [RVW01]) does yield invertible randomness extractors with slightly worse entropy loss stated as follows.

Lemma 16 ([RVW01, GUV09]) *For every* Δ, ε *and for every sufficiently large* $n \geq \Omega(\Delta + \log(1/\varepsilon))$*, there exists an explicit strong* $(n-\Delta,\varepsilon)$*-randomness extractor* $\mathrm{Ext} : \{0,1\}^n \times \{0,1\}^d \to \{0,1\}^m$ *with seed length* $d = O(\log\Delta + \log(1/\varepsilon))$ *and entropy loss* $O(\Delta + \log(1/\varepsilon))$*.*

We will show that this extractor is invertible in Section 5.1.

5.1 Reconstructibility of the Extractor in Lemma 16

In this section, we show that the extractor constructed based on Proposition 6.5 of Reingold *et. al.* [RVW01] (in particular, the one in Lemma 16) is invertible. We state Proposition 6.5 of [RVW01] as follows.

Lemma 17 ([RVW01], Proposition 6.5) *If* $\mathrm{Ext}_2 : \{0,1\}^{n_2} \times \{0,1\}^{d_2} \to \{0,1\}^{d_1}$ *is an explicit strong* $(n_2 - \Delta - \log(1/\varepsilon), \varepsilon)$*-extractor for* $d_1 = \Delta + 2\log(1/\varepsilon) + 2$ *with entropy loss* Λ_2*, then for any* $n = n_1 + n_2$*, there exists an explicit strong* $(n-\Delta, 3\varepsilon)$*-extractor* $\mathrm{RVW} : \{0,1\}^n \times \{0,1\}^{d_2} \to \{0,1\}^{n_1}$ *with entropy loss* $\Lambda_2 + \Delta + 3\log(1/\varepsilon) + O(1)$*.*

The RVW extractor is obtained by composing Ext_2 with a Goldreich-Wigderson extractor, stated below.

[4] [RVW01] stated their result for non-strong extractors, but as they mentioned, the strong version result follows readily by using strong extractors in their composition.

Theorem 18 (GW extractor [GW94], Theorem 5.3 of [RVW01]) *For any $\varepsilon > 0$ and $0 < \Delta < n$, there exists an explicit $(n - \Delta, \varepsilon)$-randomness extractor*

$$\mathrm{GW} : \{0,1\}^n \times \{0,1\}^d \to \{0,1\}^n,$$

with the seed length $d = \Delta + 2\log(1/\varepsilon) + 2$.

The GW extractor is obtained by using d truly random bits to do a random walk on an expander graph G of size 2^n. Specifically, the source x specifies a vertex in G and the seed r specifies a sequence of edges. The output $y = \mathrm{GW}(x, r) \in G$ is the final vertex of the walk starting from x and using edges r. One key property of the GW extractor is that given y and r, the source x is uniquely determined and can be computed efficiently by taking reverse walk from y.

Given the GW extractor, the RVW extractor is defined as follows.

$$\mathrm{RVW}(x_1 \circ x_2, r) = \mathrm{GW}(x_1, \mathrm{Ext}_2(x_2, r)).$$

Namely, we apply Ext_2 to the second part of the source x_2 to extractor a short randomness, and use it as a seed to extract randomness from the first part of the source x_1 using the GW extractor.

We proceed to argue that the RVW extractor is invertible. We need to show that given an output $y \in \{0,1\}^{n_1}$ and a seed $r \in \{0,1\}^{d_2}$, we can efficiently generate a uniform $x \in \{0,1\}^n$ such that $\mathrm{RVW}(x, r) = y$. The main observation is that, for every $x_2 \in \{0,1\}^{n_2}$, there exists a unique $x_1 \in \{0,1\}^{n_1}$ such that $\mathrm{RVW}(x_1 \circ x_2, r) = y$, since x_2 and r determines $z = \mathrm{Ext}_2(x_2, r)$, and then y and z determines x_1 by the property of the GW extractor. Furthermore, this x_1 can be efficiently computed given x_2, y, r. Therefore, to generate a uniform $x \in \{0,1\}^n$ such that $\mathrm{RVW}(x, r) = y$, we can simply sample a uniformly random x_2 and compute the corresponding x_1.

The extractor in Lemma 16 is constructed by instantiating Ext_2 using the best known explicit strong randomness extractor of Guruswami, Umans, and Vadhan [GUV09], and hence it is invertible.

5.2 Derandomized Generalized Raz's Lemma

As mentioned, one of our main observation is that the generalized Raz's Lemma can be derandomized using an average-case randomness extractor. In this section, we formalize and prove our derandomized generalized Raz's Lemma as follows. We will use this lemma to prove our derandomized parallel repetition theorem in the next section.

Lemma 19 (Derandomized Generalized Raz's Lemma) *Let $\mathrm{Ext} : \{0,1\}^\ell \times \{0,1\}^d \to \{0,1\}^t$ be a strong average-case $(\ell - \Delta, \varepsilon)$-randomness extractor. Let $\delta = 2^{-\Delta}$. Let (H, X) be a joint distribution over $\{0,1\}^* \times \{0,1\}^\ell$ such that H and X are independent, and $X \equiv U_\ell$ is uniform. Let E be an event over (H, X) with $\Pr[E] \geq \delta$. Then,*

$$\frac{1}{D} \sum_{i \in \{0,1\}^d} \mathbf{SD}((H, \mathrm{Ext}(X, i))|_E, (H|_E, U_t)) \leq \varepsilon.$$

Proof. Let $(\tilde{H}, \tilde{X}) \triangleq (H, X)|_E$. We claim that

$$\mathrm{H}_\infty(\tilde{X}|\tilde{H}) \geq n - \Delta,$$

and use the claim to prove the lemma first. Since $\mathrm{H}_\infty(\tilde{X}|\tilde{H}) \geq n - \Delta$, and Ext is a strong average-case $(n-\Delta, \varepsilon)$-randomness extractor, we know that $(U_d, \tilde{H}, \mathrm{Ext}(\tilde{X}, U_d))$ is ε-close to $(U_d, \tilde{H}, U_m)$. The lemma follows by observing that

$$\begin{aligned}
&\mathbf{SD}((U_d, \tilde{H}, \mathrm{Ext}(\tilde{X}, U_d)), (U_d, \tilde{H}, U_m)) \\
&= \frac{1}{D} \sum_{r \in \{0,1\}^d} \mathbf{SD}((\tilde{H}, \mathrm{Ext}(\tilde{X}, r)), (\tilde{H}, U_m)) \\
&= \frac{1}{D} \sum_{r \in \{0,1\}^d} \mathbf{SD}((H, \mathrm{Ext}(X, r))|_E, (H|_E, U_m))
\end{aligned}$$

It remains to show the following claim.

Claim 20 *Let (H, X) be independent with $X \equiv U_\ell$. Let E be an event over (H, X) with $\Pr[E] \geq 2^{-\Delta}$. Let $(\tilde{H}, \tilde{X}) \triangleq (H, X)|_E$. Then $\mathrm{H}_\infty(\tilde{X}|\tilde{H}) \geq n - \Delta$.*

Proof of claim: By definition, we need to show that

$$\sum_h \mathsf{P}_{\tilde{H}}(h) \cdot \max_x \mathsf{P}_{\tilde{X}|\tilde{H}}(x|h) \leq 2^{-(n-\Delta)}.$$

Fix any $h \in \mathrm{supp}(\tilde{H})$. For every $x \in \{0,1\}^n$, we have

$$\begin{aligned}
\mathsf{P}_{\tilde{X}|\tilde{H}}(x|h) &= \Pr[X = x | (H = h) \wedge E] \\
&= \frac{\Pr[(X = x) \wedge E | H = h]}{\Pr[E | H = h]} \\
&\leq \frac{\Pr[X = x | H = h]}{\Pr[E | H = h]} \\
&= \frac{2^{-n}}{\Pr[E | H = h]}
\end{aligned}$$

Hence,

$$\sum_h \mathsf{P}_{\tilde{H}}(h) \cdot \max_x \mathsf{P}_{\tilde{X}|\tilde{H}}(x|h) \leq \sum_h \Pr[H = h | E] \cdot \frac{2^{-n}}{\Pr[E | H = h]}.$$

By Bayes' rule,

$$\Pr[H = h | E] = \frac{\Pr[H = h] \cdot \Pr[E | H = h]}{\Pr[E]},$$

so

$$\sum_h \mathsf{P}_{\tilde{H}}(h) \cdot \max_x \mathsf{P}_{\tilde{X}|\tilde{H}}(x|h) \leq \sum_h \frac{\Pr[H = h] \cdot 2^{-n}}{\Pr[E]} \leq 2^{-(n-\Delta)}.$$

□

This completes the proof of the lemma. ■

6 Proof of Our Main Theorem

In this section, we prove Theorem 6, our derandomized parallel repetition theorems for public-coin interactive proofs and arguments. As outlined in Section 4, we derandomize the parallel verifier using randomness extractors, and we prove Theorem 6 via an efficient black-box reduction a la Håstad *et. al.* [HPWP10]. In Section 6.1, we present a formal description of our construction and prove our main lemma (see Lemma 21). Then we prove Theorem 6 using the lemma in Section 6.2

6.1 Our Construction and Main Reduction Lemma

Let $\mathrm{Ext} : \{0,1\}^\ell \times \{0,1\}^d \to \{0,1\}^t$ be an extractor, and Π be a (m,t)-public-coin protocol. Let $D = 2^d$ and we identify $\{0,1\}^d$ with $[D]$. We define a D-fold derandomized parallel repetition $\Pi_{\mathrm{Ext}} = (P^D, V_{\mathrm{Ext}})$ of Π, where the derandomized verifier V_{Ext} is defined in Figure 1.

Let $\mathrm{Ext} : \{0,1\}^\ell \times \{0,1\}^d \to \{0,1\}^t$ be an extractor. Define a derandomized verifier V_{Ext}:

- At each round $j \in [m]$, V_{Ext} samples a uniformly random $x_j \leftarrow U_\ell$, computes

$$(y_{j,1}, \ldots, y_{j,D}) = (\mathrm{Ext}(x_j, 1), \ldots, \mathrm{Ext}(x_j, D)),$$

  and sends $\vec{y_j} = (y_{j,1}, \ldots, y_{j,D})$ to P^D.
- At the end of interaction, V_{Ext} accepts iff all D subverifiers accept.

Figure 1: Formal description of the derandomized verifier V_{Ext} in Π_{Ext}.

Note that this implicitly defines a derandomizer $G : \{0,1\}^{m\cdot\ell} \to \{0,1\}^{m\cdot D\cdot t}$. We note that since V_{Ext}'s messages in different rounds are independent, V_{Ext} can simply send $x_j \in \{0,1\}^\ell$ to P^D, who can compute $\vec{y_j}$ by himself. This makes Π_{Ext} remain public-coin and reduces the verifier communication complexity.

We shall show that if Ext is a good randomness extractor, then G is a good oblivious derandomizer. We prove this by a black-box reduction, showing that there is a prover strategy P^* for Π such that given oracle access to an adversary P^{D*} that convinces V_{Ext} with a certain probability, P^* can convince V with a much higher probability. Formally, we prove the following lemma.

Lemma 21 *Let* $\mathrm{Ext} : \{0,1\}^\ell \times \{0,1\}^d \to \{0,1\}^t$ *be a strong, average-case* $(\ell - \Delta, \varepsilon)$*-randomness extractor. Let* $\delta = 2^{-\Delta}$*. Let* Π *be a* (m,t)*-public-coin protocol and* Π_{Ext} *the corresponding derandomized parallel protocol. There exists a prover strategy* P^* *such that for every* $n \in \mathbb{N}$ *and common input* $z \in \{0,1\}^n$*, and every parallel prover strategy* P^{D*}*, the following holds.*

1. $\Pr[\langle P^{D*}, V_{\mathrm{Ext}}\rangle(z) = 1] \geq \delta$

$$\Rightarrow \Pr\left[\left\langle P^{*(P^{D*})}, V\right\rangle(z) = 1\right] \geq 1 - 2\cdot m\cdot\varepsilon.$$

2. *If in addition,* Ext *is invertible, then* $P^{*(\cdot)}$ *runs in time* $\mathrm{poly}(n, \varepsilon^{-1}, \delta^{-1})$ *given oracle access to* P^{D*}*.*

We proceed to prove the lemma. We first note that we can assume without loss of generality that the parallel prover P^{D*} is deterministic, since we can fix P^{D*}'s coins without hurting the success probability of P^{D*} too much. We also note that in this case, the interaction (P^{D*}, V_{Ext}) can be described by V_{Ext}'s coins $(x_1, \ldots, x_m)$.

Definition of P^*. We consider a reduction prover P^*_{rej} who interacts with V by simulating the interaction between P^{D*} and V_{Ext} and uses a rejection sampling strategy of [HPWP10]. More precisely, P^*_{rej} selects a uniformly random coordinate $i \in [D]$ to embed V in V_{Ext}. At each round $j \in [D]$, upon receiving the external verifier V's message, P^* interprets it as V_i's message $y_{j,i}$, and repeatedly samples a random continuation of (P^{D*}, V_{Ext}) until he finds an **accepting continuation**, i.e., V_{Ext} accepts at the end of interaction. Note that sampling a random continuation amounts to sampling a uniformly random x_j conditioned on $\text{Ext}(x_j, i) = y_{j,i}$, and sampling uniformly random $x_{j+1}, \ldots, x_m$. Once an accepting continuation is found, P^*_{rej} chooses the corresponding x_j to simulate V_{Ext} at this round, and forward the i-th coordinate of P^{D*}'s message to V. If P^*_{rej} fails to find an accepting continuation in $M \triangleq 1/\delta\varepsilon$ trials, then P^*_{rej} simply aborts. ■

Note that when Ext is invertible, P^*_{rej} can sample a random continuation efficiently, and hence P^*_{rej} runs in time $\text{poly}(n, \varepsilon^{-1}, \delta^{-1})$, as asserted in Lemma 21. The proof of the first item is very similar to the proof of parallel repetition theorem for public-coin protocol in [HPWP10], with the difference that application of the generalized Raz's Lemma is replaced by a derandomized version.

To lower bound the success probability of P^*_{rej}, we refer to the interaction (P^*_{rej}, V) as a **real experiment**, and compare it with an **ideal experiment**, defined as follows.

Ideal Experiment $(\tilde{P}^*_{rej}, \tilde{V}^*_{rej})$. The ideal experiment $(\tilde{P}^*_{rej}, \tilde{V}^*_{rej})$ also simulate the interaction of (P^{D*}, V_{Ext}) as (P^*_{rej}, V), but with the following two differences. First, at each round j the verifier $\tilde{V}^*_{rej}$, instead of choosing the message $y_{j,i}$ uniformly at random, chooses $y_{j,i}$ using rejection sampling as well. Namely, $\tilde{V}^*_{rej}$ repeatedly simulates a random continuation of (P^{D*}, V_{Ext}) until an accepting continuation is found, and choose the corresponding $y_{j,i}$. Second, in the rejection sampling, instead of using bounded number of samples, they sample unbounded number of times until an accepting continuation is found. ■

Let us look at the ideal experiment closely. Let E denote the event of accepting interaction, i.e. $\langle P^{D*}, V_{\text{Ext}} \rangle = 1$. Note that performing rejection sampling is equivalent to selecting a random next message conditioned on E, and the interaction $(\tilde{P}^*_{rej}, \tilde{V}^*_{rej})$ is equivalent to choosing a uniformly random accepting interaction of (P^{D*}, V_{Ext}). Recall that the interaction (P^{D*}, V_{Ext}) can be described by V_{Ext}'s randomness $(X_1, \ldots, X_m)$. The outcome of $(\tilde{P}^*_{rej}, \tilde{V}^*_{rej})$ is simply $(X_1, \ldots, X_m)|_E$. Note that $\tilde{V}^*_{rej}$ accepts iff V_i of V_{Ext} accept, we have $\Pr[\langle \tilde{P}^*_{rej}, \tilde{V}^*_{rej} \rangle = 1] = 1$.

We next argue that the ideal and the real experiments are statistically close, which would give the desired lower bound on the success probability of P^*. Recall that there are the following two differences of the real experiment from the ideal one.

1. At each round j, V chooses a uniformly random $y_{j,i}$ instead of conditioning on E.
2. P^*_{rej} may abort when he fails to find an accepting continuation in M samples.

We will bound the statistical distance incurred by these differences round by round, and combine it using the following hybrid argument. Consider the following hybrid experiments H_j, where in the

first j rounds of the interaction, both parties choose messages according to the ideal experiment $(\tilde{P}^*_{rej}, \tilde{V}^*_{rej})$, and for the remaining rounds, they choose messages according to the real experiment (P^*_{rej}, V). Clearly, H_0 is the real experiment, and H_m is the ideal one. We will argue that the statistical distance between hybrids H_{j-1} and H_j is at most 2ε for every $j \in [m]$, and hence the statistical distance between the ideal and real experiments is at most $2m\varepsilon$.

Now, the only difference between H_{j-1} and H_j is at the j-th round, where the differences are precisely the above two items. We handle them separately, by further considering an intermediate hybrid H'_j, which is the same as H_j, except that at the j-th round, the prover chooses his message according to P^*_{rej} instead of $\tilde{P}^*_{rej}$, i.e., he aborts when he fails to find an accepting continuation in M samples.

We first bound the statistical distance between H_{j-1} and H'_j, where the only difference is Item (1) at the j-th round. We upper bound it by ε using the derandomized generalized Raz's Lemma (see Lemma 19 in Section 5.2) as follows.

Let $H = (X_1, \ldots, X_{j-1})$. Note that the first $j-1$ rounds interaction of both H_{j-1} and H'_j is simply $H|_E$, independent of which coordinate $i \in [D]$ played by the verifier. Hence, we can think of the coordinate $i \in [D]$ is chosen uniformly random at the beginning of the j-th round. At the j-th round, the verifier in H_{j-1} simply chooses a random $y_{j,i} \leftarrow U_t$, and the verifer in H'_j chooses $y_{j,i}$ according to $\mathrm{Ext}(X_j, i)|_{H,E}$. The statistical difference is exactly

$$\frac{1}{D} \sum_{i \in \{0,1\}^d} \mathbf{SD}((H, \mathrm{Ext}(X_j, i))|_E, (H|_E, U_t)),$$

which is upper bounded by ε by Lemma 19.

We next bound the statistical distance between H'_j and H_j, where the only difference is Item (2) at the j-th round. Note that this amounts to bound the aborting probability of P^*_{rej} at the j-th round with the history chosen according to the ideal experiment $(X_1, \ldots, X_{j-1}, Y_{j,i})|_E$. This step is exactly the same as [HPWP10], which we repeat as follows. By Lemma 2 of [HPWP10], when the prover uses rejection sampling to find an accepting continuation, the expected number of samples needed is $1/\Pr[E]$ (averaging over the history). Hence, by a Markov inequality, the probability that P^*_{rej} aborts is at most $(1/\Pr[E])/M \leq \varepsilon$. Therefore, the statistical distance between H'_j and H'_j is at most ε. We refer the reader to [HPWP10] for further details about computing the expectation.

To summarize, the above hybrid argument shows that the statistical distance between the ideal and the real experiments is at most $2m\varepsilon$. Since $\Pr[\langle \tilde{P}^*_{rej}, \tilde{V}^*_{rej} \rangle = 1] = 1$, it follows that

$$\Pr[\langle P^*_{rej}, V \rangle = 1] \geq 1 - 2 \cdot m \cdot \varepsilon,$$

which completes the proof of Lemma 21. ■

6.2 Proof of Theorem 6

In this section we prove Theorem 6 using Lemma 21 and randomness extractors from Lemma 16 and 15. The reason for using two extractors is that in the case of interactive proofs, the reduction P^* does not need to be efficient, and hence we can use any strong average-case extractor. In contrast, for interactive arguments, we need to use a invertible extractor to ensure that the reduction is efficient.

Proof. (of Theorem 6) We derandomize (m, t)-public-coin interactive proofs using the construction in previous section with the extractor from Lemma 15. More precisely, given parameters

$m, t, \varepsilon, \delta$, and γ, we set $\Delta = \log(1/\delta)$ and the error parameter ε_0 of the extractor by $\varepsilon_0 = \varepsilon/(8m)$. By Lemma 15, there exists a strong $(\ell - \Delta, \varepsilon_0)$-randomness extractor $\mathrm{Ext} : \{0,1\}^\ell \times \{0,1\}^d \to \{0,1\}^t$ with $d = O(\log \Delta + \log(1/\varepsilon_0))$ and $\ell = t + (1+\gamma)\Delta + O(\log(1/\varepsilon_0))$. By Lemma 13, Ext is also a strong average-case $(\ell - \Delta, 3\varepsilon_0)$-randomness extractor. Let $k = 2^d$. Let $G : \{0,1\}^{m \cdot \ell} \to \{0,1\}^{k \cdot m \cdot t}$ be the oblivious derandomizer constructed using this Ext in our construction in the previous section. Note that

$$s = m \cdot \ell = m \cdot (t + (1+\gamma)\Delta + O(\log(m/\varepsilon))), \text{ and } k = \mathrm{poly}(m, \log(1/\varepsilon), \log(1/\delta)),$$

as stated in the theorem.

We proceed to show that G is an $((1-\varepsilon) \mapsto \delta)$-oblivious derandomizer for (m,t)-public-coin interactive proofs. Let Π be a (m,t)-public-coin interactive proofs for a language L with soundness error at most ε, and let Π_G be the induced derandomized parallel repetition of Π. If Π_G has soundness error greater than δ, then there exists some P^{k*} and infinitely many inputs $z \notin L$ such that

$$\Pr[\left\langle P^{k*}, V_G \right\rangle (z) = 1] \geq \delta.$$

By Lemma 21, there exists an adversary P^* such that on these inputs,

$$\Pr[\langle P^*, V \rangle (z) = 1] \geq 1 - 2 \cdot m \cdot (3\varepsilon_0) > 1 - \varepsilon,$$

contradicting to the fact that Π has soundness error at most $1 - \varepsilon$. Therefore, Π_G has soundness error at most δ, which completes the proof of the theorem.

The proof for the case of interactive arguments follows exactly the same line as the above proof, except that we use a invertible extractor from Lemma 16 and add a negligible slackness to ensure that the reduction is efficient. We omit the detail to avoid repetitive arguments. ■

7 Lower Bounds for Public-coin Protocols

In this section, we prove the following formal version of Theorem 2, where we present a specific protocol for which every statistcally-sound derandomizer that reduces soundness error from $1/2$ to δ must use at least $m \cdot \log(1/\delta) - O(1)$-bits of extra randomness, and for which the existence of more randomness efficient computationally-sound derandomizer implies the existence of one-way functions.

Theorem 22 *For every polynomially bounded $m : \mathbb{N} \to \mathbb{N}$, there exists a m-round public-coin interactive proof Π (for the empty language L) with (statistical) soundness error at most $1/2$ such that the following holds. For every parameter $\delta : \mathbb{N} \to [0,1]$, there does not exist a derandomizer G for Π that decreases the soundness error to δ and uses only*

$$m \cdot (\log(1/\delta) - 3)$$

extra bits of randomness.

Additionally, the existence of a computationally-sound derandomizer that decreases the soundness error to $\delta - \mathsf{ngl}$ and uses only $m \cdot (\log(1/\delta) - 3)$ extra bits of randomness instead implies the existence of one-way functions.

The protocol Π asserted in Theorem 22 is very simple – at each round, the prover speaks first with the goal of guessing the verifier's next message. More precisely, at each round i, P sends y_i to V, who sends back a random message $x_i \in \{0,1\}^t$ to P for some message length t that we choose later, and at the end of interaction, V accepts iff at least one guess $y_i = x_i$. We refer to such protocols as (m,t)-**Guess-Next-Message Protocol**. A formal description of Π can be found in Figure 2.

Let m, t be parameters. Define (m,t)-**Guess-Next-Message Protocol** $\Pi = (P, V)$:

On a common input $z \in \{0,1\}^n$

- For each round $i = 1, \ldots, m$,
  - P sends a random message $y_i \in \{0,1\}^t$ to V.
  - V sends a random message $x_i \in \{0,1\}^t$ to P.
- V accepts iff there exists some $x_i = y_i$.

Figure 2: Formal description of the (m,t)-Guess-Next-Message protocol Π.

Clearly, no matter what the prover's strategy is, he can guess x_i correctly with probability 2^{-t} at each round. Hence, the soundness error of Π is $\varepsilon \triangleq 1 - (1-2^{-t})^m$. We choose the message length $t \triangleq \lceil \log m \rceil + 1$ so that

$$1/4 \leq \varepsilon \leq 1/2.$$

To prove Theorem 22, we shall show that for every derandomizer $G : \{0,1\}^s \to \{0,1\}^{k \cdot (m \cdot t)}$ for Π, the induced derandomized parallel protocol $\Pi_G = (P^k, V_G)$ has soundness error at least

$$\min\left\{m \cdot 2^{-(s/m)-1}, 1/2\right\}$$

by constructing an adversary P^{k*}. Note that this implies that for every derandomizer G that uses less than $m \cdot (\log(1/\delta) - 3)$ bits of extra randomness (i.e., $s \leq m \cdot t + m \cdot (\log(1/\delta) - 3)$, the soundness error of Π_G is greater than δ (recall that we set $t = \lceil \log m \rceil + 1$). This means that there does not exists a derandomizer G for Π that uses only $m \cdot (\log(1/\delta) - 3)$ extra bits of randomness such that Π_G has soundness error δ, which completes the proof of Theorem 22 for the case of statistical soundness. For the case of computational soundness, we further show that the adversary P^{k*} can be approximated efficiently with inverse polynomially small error if one-way functions do not exist.

We proceed to lower bound the soundness error of $\Pi_G = (P^k, V_G)$ by constructing P^{k*}. Note that in the interaction of Π_G, the verifier's messages $\vec{x} = (x_1, \ldots, x_m) \in \{0,1\}^{k \cdot t}$ are drawn from distribution $\vec{X} = (X_1, \ldots, X_m) \triangleq G(U_s)$. For V_G to accept, it suffices for P^{k*} to guess some X_i correctly. A natural adversary is to let P^{k*} guess, at each round i, an optimal $y_i \in \{0,1\}^{k \cdot t}$ based on the verifier's previous messages $x_{<i} = (x_1, \ldots, x_{i-1})$. Namely, P^{k*} guesses y_i that maximizes the probability $\Pr[X_i = y_1 | X_{<i} = x_{<i}]$.

Indeed, it is not hard to see that P^{k*} succeeds with good probability by the following simple analysis: Define random variables $\alpha_i \triangleq \max_{y_i} \Pr[X_i = y_1 | X_{<i} = x_{<i}]$. Clearly, P^{k*} can win with probability α_i at each round i. Noting that $\vec{x} \leftarrow \vec{X} = G(U_s)$, we have $\prod_{i=1}^m \alpha_i \geq 2^{-s}$ holds with

probability 1. It follows that $\Pr[\exists \alpha_i \geq 2^{-s/m}] = 1$. Therefore, P^{k*} can succeed with probability at least $2^{-s/m}$.[5]

One can preform a more careful analysis to show that, in fact, $\Pr[\langle P^{k*}, V_G \rangle = 1] \geq m \cdot 2^{-s/m-1}$. However, approximating P^{k*} efficiently under the assumption that one-way functions do not exist, while doable, is somewhat complicated.

We instead lower bound the soundness error of Π_G by a less careful adversary $\tilde{P}^{k*}$, who simply samples a random $y_i \leftarrow X_i|_{X_{<i}=x_{<i}}$ as his guess of x_i. We show in the following lemma that $\tilde{P}^{k*}$ can already succeed with desired probability.

Lemma 23 *The success probability of $\tilde{P}^{k*}$ is at least*

$$e^{-m \cdot 2^{-s/m}} \geq \min\left\{m \cdot 2^{-(s/m)-1}, 1/2\right\}.$$

Proof. We instead upper bound the probability that V_G rejects. Let $p(\cdot)$ denote the probability mass function of $\vec{X}$. For example, $p(x_1) = \Pr[X_1 = x_1]$ and $p(x_i|x_{<i}) = \Pr[X_i = x_i | X_{<i} = x_{<i}]$. We can express the probability of the complement event as follows.

$$\begin{aligned}
&\Pr[\langle P^*, V_G \rangle = 0] \\
&\leq \Pr[\forall i \in [m], X_i \neq Y_i] \\
&= \sum_{x_1} \Pr[X_1 = x_1 \wedge Y_1 \neq x_1] \cdot \sum_{x_2} \Pr[X_2 = x_2 \wedge Y_2 \neq x_2 | X_1 = x_1] \cdots \sum_{x_k} \Pr[X_k = x_k \wedge Y_k \neq x_k | X_{<k} = x_{<k}] \\
&= \sum_{x_1} p(x_1) \cdot (1 - p(x_1)) \cdot \sum_{x_2} p(x_2|x_1) \cdot (1 - p(x_2|x_1)) \cdots \sum_{x_m} p(x_m|x_{<m}) \cdot (1 - p(x_m|x_{<m})) \\
&\leq \sum_{\vec{x}} p(\vec{x}) e^{-(p(x_1) + p(x_2|x_1) + \cdots + p(x_m|x_{<m}))} \\
&\leq \sum_{\vec{x}} p(\vec{x}) e^{-m \cdot (p(x_1) \cdot p(x_2|x_1) \ldots p(x_m|x_{<m}))^{1/m}} \\
&= \sum_{\vec{x}} p(\vec{x}) e^{-m \cdot p(\vec{x})^{1/m}}
\end{aligned}$$

where the first inequality uses $(1 - x) \leq e^{-x}$ and the second inequality uses the arithmetic-mean-geometric-mean inequality and monotonicity of the exponential function. Recall that $\vec{X} = G(U_s)$, so $p(\vec{x}) \geq 2^{-s}$ for every $\vec{x}$, and

$$\begin{aligned}
&\sum_{\vec{x}} p(\vec{x}) e^{-m \cdot p(\vec{x})^{1/m}} \\
&\quad \leq \sum_{\vec{x}} p(\vec{x}) e^{-m \cdot (2^{-s})^{1/m}} \\
&\quad = e^{-m \cdot 2^{-s/m}} \\
&\quad \leq \max\{1 - (m \cdot 2^{-s/m})/2, e^{-1}\},
\end{aligned}$$

[5]This simple analysis is sufficient to show that the randomness complexity of G need to depend on the round complexity m, but only for sufficiently small soundness error $\delta \leq 1/m$.

where the last inequality uses the fact that $e^{-x} \leq 1 - x/2$ for $0 \leq x \leq 1$. Therefore,

$$\Pr[\left\langle \mathbf{P}^{k*}, V_G \right\rangle = 1] \geq \min\{m \cdot 2^{-(s/m)-1}, 1/2\},$$

as desired. ■

The above lemma shows that the soundness error of Π_G is at least $\min\left\{m \cdot 2^{-(s/m)-1}, 1/2\right\}$, which as argued, implies Theorem 22 for the case of statistical soundness. For the case of computational soundness, recall that if one-way functions do not exist, then one can efficiently sample a random pre-image of any efficient function [IL89] with inverse polynomially small error. This implies that the above $\tilde{P}^{k*}$ can be approximated efficiently with inverse polynomially small error assuming that one-way functions do not exist, and completes the proof of the case of computational soundness.

8 Lower Bounds for Private-coin Protocols

In this section, we present our impossibility result for non-trivial derandomization of the parallel repetition of private-coin protocols. We exhibit a sepcific private-coin protocol such that parallel repetition of the protocol cannot be derandomized non-trivially in the following strong sense – decreasing the randomness complexity of the parallel verifier by one would increase the soundness error by a factor of two. Formally, we prove the following theorem.

Theorem 24 *For every polynomially bounded $t : \mathbb{N} \to \mathbb{N}$, there exists a 3-message private-coin interactive proof Π (for the empty language L) with randomness complexity t and (statistical) soundness error $1/2$ such that the following holds. For every polynomially bounded $s, k : \mathbb{N} \to \mathbb{N}$ and every efficient $G : \{0,1\}^s \to \{0,1\}^{k \cdot t}$, the corresponding derandomized parallel protocol Π_G has (statistical) soundness error at least*

$$\varepsilon \triangleq \frac{2^{k \cdot (t-1)}}{2^s + 2^{k \cdot (t-1)}} \geq \min\left\{2^{-(s-(t-1)\cdot k+1)}, 1/2\right\}.$$

Furthermore, if Π_G has computational soundness error less than $\varepsilon - \mathsf{ngl}$, then one-way functions exist.

Note that by padding dummy messages, Theorem 24 can be extended to m-message protocols with $m > 3$ readily. We mention that Theorem 3 can be derived from Theorem 24 readily as a simple corollary.

To prove Theorem 24, we consider the following simple "guess-with-hint" protocol Π: the prover's goal is to guess the verifier's coins, with the help of a hint from the verifier. More precisely, P first sends a hash function $h : \{0,1\}^t \to \{0,1\}^\ell$ to V, who sends back the hashed value $z = h(x) \in \{0,1\}^\ell$ of its coins $x \in \{0,1\}^t$ to P. Then based on the hint z, P sends a guess y to V, who accepts iff $x = y$. We refer to this protocol as (t, ℓ)-Guess-with-Hint protocol, and a formal description of the protocol can be found in Figure 3.

It is not hard to see that the soundness error of Π is exactly $2^{-(t-\ell)}$, since an adversarial prover can readily learn ℓ bits of information about the secret x from the hint z, and the secret x has only t bits of entropy. The protocol asserted in Theorem 24 is simply the $(t, t-1)$-Guess-with-Hint protocol. Nevertheless, we will present our analysis for a general (t, ℓ)-Guess-with-Hint protocol Π.

Let t, ℓ be parameters, and $\mathcal{H} = \{h : \{0,1\}^t \to \{0,1\}^\ell\}$ be a pair-wise independent hash function family. Define (t,ℓ)-**Guess-with-Hint Protocol** $\Pi = (P, V)$:

On a common input $z \in \{0,1\}^n$

- P picks a random $h \leftarrow \mathcal{H}$ and sends h to V.
- V sends $z = h(x) \in \{0,1\}^\ell$ to P, where $x \in \{0,1\}^t$ is the random coins of V.
- P sends $y \in \{0,1\}^t$ to V.

At the end, V accepts iff $x = y$.

Figure 3: Formal description of the (t,ℓ)-Guess-with-Hint protocol Π.

For the parallel repetition of Π, a key observation is that the k-fold parallel repetition Π^k is simply a $(k \cdot t, k \cdot \ell)$-Guess-with-Hint protocol, since the concatenation of pairwise independent hash functions is again a pairwise independent hash function.

Let us proceed to consider a derandomizer $G : \{0,1\}^s \to \{0,1\}^{k \cdot t}$ and the corresponding derandomized parallel protocol Π_G. Now, note that the prover can potentially learn $k \cdot \ell$ bits information from the hint but the secret $x = (x_1, \ldots, x_k)$ has only s bits of entropy, the soundness error of Π_G could be as large as $2^{-(s-k\cdot\ell)}$. This turns out to be essentially true. However, since the secrete $x \leftarrow G(U_s)$ is not uniform, the analysis becomes more involved.

In what follows, we prove two lemmas regarding the soundness property of Π and Π_G respectively, form which Theorem 24 follows immediately.

Lemma 25 *The (t,ℓ)-Guess-with-Hint protocol Π defined in Figure 3 has soundness error at most $2^{-(t-\ell)}$. Furthermore, if there exists a* surjective *function $h : \{0,1\}^t \to \{0,1\}^\ell$ in the pair-wise independent hash family $\mathcal{H}$ used in Π, then Π has soundness error $2^{-(t-\ell)}$.*

Lemma 26 *Let $G : \{0,1\}^s \to \{0,1\}^{k\cdot t}$ be a derandomizer, and Π_G be the induced derandomized parallel repetition of the (t,ℓ)-Guess-with-Hint protocol Π defined in Figure 3. Π_G has soundness error at least*

$$\varepsilon \triangleq \frac{2^{k\cdot\ell}}{2^s + 2^{k\cdot\ell}} \geq \min\left\{2^{-(s-\ell\cdot k+1)}, 1/2\right\}.$$

Furthermore, if Π_G has computational soundness less than $\varepsilon - \mathsf{ngl}$, then one-way functions exist.

Proof. (of Lemma 25) We first upper bound the soundness error of Π. Note that it suffices to show that for every *deterministic* adversary P^*,

$$\Pr[\langle P^*, V\rangle = 1] \leq 2^{-(t-\ell)}.$$

Let P^* be a deterministic adversary, and $h \in \mathcal{H}$ be the hash function chosen by P^*. Let X and $Z = h(X)$ denote the verifier's coins and the hashed value. Note that conditioned on $Z = z$, $X|_{Z=z}$ is simply a uniform distribution over $h^{-1}(z)$, and hence P^* can guess $x \leftarrow X|_{Z=z}$ correctly with

probability at most $1/|h^{-1}(z)|$. We have

$$\begin{aligned}
&\Pr[\langle P^*, V\rangle = 1] \\
&= \sum_{z\in\{0,1\}^\ell} \Pr[Z = z] \cdot \Pr[P^*(Z) = X | Z = z] \\
&\leq \sum_{z\in\{0,1\}^\ell} \frac{|h^{-1}(z)|}{2^t} \cdot \frac{1}{|h^{-1}(z)|} \\
&= \sum_{z\in\mathrm{supp}(Z)} \frac{1}{2^t} \\
&\leq 2^{-(t-\ell)}.
\end{aligned}$$

On the other hand, to lower bound the soundness error of Π, we construct an adversary P^* with success probability $2^{-(t-\ell)}$ as follows.

- In the first message, P^* sends a surjective hash function $h \in \mathcal{H}$ to V.
- Upon receiving the hashed value z from V, P^* sends an arbitrary pre-image $y \in h^{-1}(z)$ to V.

Note that as long as P^*'s guess is in the pre-image $h^{-1}(z)$, P^*'s guess would be correct with probability $1/|h^{-1}(z)|$. We have

$$\begin{aligned}
&\Pr[\langle P^*, V\rangle = 1] \\
&= \sum_{z\in\{0,1\}^\ell} \Pr[Z = z] \cdot \Pr[P^*(Z) = X | Z = z] \\
&= \sum_{z\in\{0,1\}^\ell} \frac{|h^{-1}(z)|}{2^t} \cdot \frac{1}{|h^{-1}(z)|} \\
&= \sum_{z\in\mathrm{supp}(Z)} \frac{1}{2^t} \\
&= 2^{-(t-\ell)},
\end{aligned}$$

where the last equality uses the fact that h is surjective. ■

Proof. (of Lemma 26) Recall that Π^k is equivalent to the $(k \cdot t, k \cdot \ell)$-Guess-with-Hint protocol, and V_G generates coins according to the distribution $X \triangleq G(U_s)$. To lower bound the soundness error of Π_G, consider the following prover strategy P^*, who simply uses a random h and guess a random $y \leftarrow X|_{h(X)=z}$ after seeing the hint z. Namely,

- In the first message, P^* chooses a random $h \leftarrow \mathcal{H}^k$ and sends h to V_G.
- Upon receiving $z \in \{0,1\}^{k\cdot\ell}$ from V_G, P^* samples $y \leftarrow X|_{h(X)=z}$ as his guess of x, and sends y to V_G.

We shall show that

$$\Pr[\langle P^*, V_G\rangle = 1] \geq \frac{2^{k\cdot\ell}}{2^s + 2^{k\cdot\ell}}. \tag{1}$$

By definition, we can express the success probability of P^* as follows.

$$\Pr[\langle P^*, V_G\rangle = 1] = \sum_h \Pr[H = h] \cdot \sum_z \Pr[h(X) = z] \cdot \Pr[X = Y | h(X) = z].$$

Recall that Y is drawn from distribution $X|_{h(X)=z}$, $\Pr[X = Y|h(X) = z]$ is simply the collision probability of $X|_{h(X)=z}$, which is at least $1/|h^{-1}(z)|$. Hence, we have

$$\Pr[\langle P^*, V_G\rangle = 1] \geq \sum_h \Pr[H = h] \cdot \sum_z \Pr[h(X) = z] \cdot \frac{1}{|h^{-1}(z)|} = \mathop{\mathrm{E}}_{H,X}\left[\frac{1}{|H^{-1}(H(X))|}\right].$$

It remains to lower bound the expectation. We shall show that for any $x \in \mathrm{supp}(X)$,

$$\mathop{\mathrm{E}}_{H}\left[\frac{1}{|H^{-1}(H(x))|}\right] \geq \frac{2^{k\cdot\ell}}{2^s + 2^{k\cdot\ell}}.$$

This clearly implies Eq. (1).

For every $x' \in \mathrm{supp}(X)$, define an indicator random variable $A_{x'}$ such that $A_{x'} = 1$ iff $H(x) = H(x')$. Note that $A_x = 1$ with probability 1, and for every $x' \neq x$, $\Pr[A_{x'} = 1] = 2^{-k\cdot\ell}$ by pairwise independence of H. We have

$$\mathrm{E}\left[|H^{-1}(H(X))|\right] = \sum_{x'} \mathrm{E}[A_{x'}] = 1 + \frac{\mathrm{supp}(X) - 1}{2^{k\cdot\ell}} \leq \frac{2^{k\cdot\ell} + 2^s}{2^{k\cdot\ell}},$$

where the last inequality follows by $\mathrm{supp}(X) \leq 2^s$. Finally, by Jensen's inequality,

$$\mathop{\mathrm{E}}_{H}\left[\frac{1}{|H^{-1}(H(x))|}\right] \geq \frac{1}{\mathrm{E}\left[|H^{-1}(H(X))|\right]} \geq \frac{2^{k\cdot\ell}}{2^s + 2^{k\cdot\ell}}.$$

This completes the proof of the first part of Lemma 26. For the furthermore part, recall that if one-way functions do not exist, then one can efficiently sample a random pre-image of any efficient function [IL89] with inverse polynomially small error. This implies that the above P^* can be implemented efficiently (with inverse polynomially small error) and hence Π_G has computational soundness error at least $\varepsilon - \mathsf{ngl}$. ■

Acknowledgments

We thank Salil Vadhan for very useful discussions, and in particular, for suggesting to use the notion of averaging-case randomness extractors to simplify our original proof. We also thank Feng-Hao Liu for the collaboration in the early stage of this research, and anonymous reviewers for useful suggestions.

References

[BGG93] M. Bellare, O. Goldreich, and S. Goldwasser. Randomness in interactive proofs. *Computational Complexity*, 3(4):319–354, 1993.

[BIN97] Mihir Bellare, Russell Impagliazzo, and Moni Naor. Does parallel repetition lower the error in computationally sound protocols? In *FOCS*, pages 374–383, 1997.

[BM84] Manuel Blum and Silvio Micali. How to generate cryptographically strong sequences of pseudo-random bits. *SIAM J. Comput.*, 13(4):850–864, 1984.

[BM88] László Babai and Shlomo Moran. Arthur-merlin games: A randomized proof system, and a hierarchy of complexity classes. *J. Comput. Syst. Sci.*, 36(2):254–276, 1988.

[BR94] M. Bellare and J. Rompel. Randomness-efficient oblivious sampling. In *Proceedings of the 35th Annual IEEE Symposium on Foundations of Computer Science*, pages 276–287, 1994.

[CHS05] Ran Canetti, Shai Halevi, and Michael Steiner. Hardness amplification of weakly verifiable puzzles. In *TCC*, pages 17–33, 2005.

[CL10] Kai-Min Chung and Feng-Hao Liu. Parallel repetition theorems for interactive arguments. In *TCC*, pages 19–36, 2010.

[DORS08] Yevgeniy Dodis, Rafail Ostrovsky, Leonid Reyzin, and Adam Smith. Fuzzy extractors: How to generate strong keys from biometrics and other noisy data. *SIAM J. Comput.*, 38(1):97–139, 2008.

[Gol01] Oded Goldreich. *The Foundations of Cryptography - Volume 1.* Cambridge University Press, 2001.

[GUV09] Venkatesan Guruswami, Christopher Umans, and Salil P. Vadhan. Unbalanced expanders and randomness extractors from parvaresh–vardy codes. *J. ACM*, 56(4), 2009.

[GW94] Oded Goldreich and Avi Wigderson. Tiny families of functions with random properties (preliminary version): a quality-size trade-off for hashing. In *STOC*, pages 574–584, 1994.

[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.

[Hol09] Thomas Holenstein. Parallel repetition: Simplification and the no-signaling case. *Theory of Computing*, 5(1):141–172, 2009.

[HPWP10] Johan Håstad, Rafael Pass, Douglas Wikström, and Krzysztof Pietrzak. An efficient parallel repetition theorem. In Daniele Micciancio, editor, *TCC*, volume 5978 of *Lecture Notes in Computer Science*, pages 1–18. Springer, 2010.

[IL89] Russell Impagliazzo and Michael Luby. One-way functions are essential for complexity based cryptography (extended abstract). In *FOCS*, pages 230–235, 1989.

[PV07] Rafael Pass and Muthuramakrishnan Venkitasubramaniam. An efficient parallel repetition theorem for arthur-merlin games. In *STOC*, pages 420–429, 2007.

[Raz98] Ran Raz. A parallel repetition theorem. *SIAM J. Comput.*, 27(3):763–803, 1998.

[RVW01] Omer Reingold, Salil P. Vadhan, and Avi Wigderson. Entropy waves, the zig-zag graph product, and new constant-degree expanders and extractors. *Electronic Colloquium on Computational Complexity (ECCC)*, 8(18), 2001.

[Sha10] Ronen Shaltiel. Derandomized parallel repetition theorems for free games. In *IEEE Conference on Computational Complexity*, pages 28–37, 2010.

[Vad11a] Salil Vadhan, 2011. Personal communication.

[Vad11b] Salil P. Vadhan. *Pseudorandomness*. Now Publishers, 2011.

[Yao82] Andrew C. Yao. Theory and applications of trapdoor functions. In *Proceedings of the 23rd Annual IEEE Symposium on Foundations of Computer Science*, pages 80–91, 1982.

[Zuc97] David Zuckerman. Randomness-optimal oblivious sampling. *Random Struct. Algorithms*, 11(4):345–367, 1997.