Randomness Condensers for Efficiently Samplable, Seed-Dependent Sources

Yevgeniy Dodis* Thomas Ristenpart† Salil Vadhan‡

January 5, 2012

Abstract

We initiate a study of randomness condensers for sources that are efficiently samplable but may depend on the seed of the condenser. That is, we seek functions $\mathsf{Cond} : \{0,1\}^n \times \{0,1\}^d \to \{0,1\}^m$ such that if we choose a random seed $S \leftarrow \{0,1\}^d$, and a source $X = \mathcal{A}(S)$ is generated by a randomized circuit $\mathcal{A}$ of size t such that X has min-entropy at least k given S, then $\mathsf{Cond}(X; S)$ should have min-entropy at least some k' given S. The distinction from the standard notion of randomness condensers is that the source X may be correlated with the seed S (but is restricted to be efficiently samplable). Randomness *extractors* of this type (corresponding to the special case where $k' = m$) have been implicitly studied in the past (by Trevisan and Vadhan, FOCS '00).

We show that:

- Unlike extractors, we can have randomness condensers for samplable, seed-dependent sources whose computational complexity is smaller than the size t of the adversarial sampling algorithm $\mathcal{A}$. Indeed, we show that sufficiently strong collision-resistant hash functions are seed-dependent condensers that produce outputs with min-entropy $k' = m - \mathcal{O}(\log t)$, i.e. logarithmic *entropy deficiency*.
- Randomness condensers suffice for key derivation in many cryptographic applications: when an adversary has negligible success probability (or negligible "squared advantage" [3]) for a uniformly random key, we can use instead a key generated by a condenser whose output has logarithmic entropy deficiency.
- Randomness condensers for seed-dependent samplable sources that are robust to side information generated by the sampling algorithm imply soundness of the Fiat-Shamir Heuristic when applied to any constant-round, public-coin interactive proof system(and thus imply that such proof systems cannot be zero knowledge). In fact, this only requires condensers for "leaky sources" — ones that are uniform prior to conditioning on the adversary's side information — and we show that such condensers are also *necessary* for soundness of the Fiat–Shamir Heuristic.

*Department of Computer Science, New York University. Partially supported by NSF Grants CNS-1065134, CNS-1065288, CNS-1017471, CNS-0831299 and Google Faculty Award. Email: `dodis@cs.nyu.edu`.

†Department of Computer Sciences, University of Wisconsin–Madison. Email: `rist@cs.wisc.edu`. Partially supported by NSF Grant CNS-1065134.

‡School of Engineering and Applied Sciences and Center for Research on Computation and Society, Harvard University. Supported by NSF grant CCF-1116616. Work done in part while visiting Microsoft Research SVC and Stanford University. Email: `salil@seas.harvard.edu`.

1 Introduction

Randomness extractors — functions that convert sources of biased and/or correlated bits into almost uniformly distributed bits — have a wide variety of applications in cryptography and other parts of theoretical computer science. However, to extract randomness from rich models of sources, e.g. sources for which we only have a lower bound on their min-entropy (or even sources where each bit is mildly unpredictable given the previous ones), deterministic functions cannot be randomness extractors [33]. Thus the general definition of randomness extractor by Nisan and Zuckerman [30] allows the extractor to be probabilistic — the extractor is given a uniformly random *seed* that it can use as a catalyst for extraction.

The need for a seed, however, is a problem in some applications of randomness extractors. First, if the reason for extraction is lack of access to high-quality random bits, then we may not have any way to generate the seed.[1] (In algorithmic applications of randomness extractors, it is often possible to try all possible seeds, and combine the results obtained for each extractor output. But this does not work in most cryptographic applications. Even one bad seed can compromise one's secrets, and thus eliminate security.) Second, even if we can generate a uniformly random seed, it is crucial that the weak random source from which we extract is independent from the seed. This means that it is problematic to generate the seed once and for all (perhaps using an expensive source of randomness) in hope that it can be used for all future randomness extractions. If there is any chance that the future weak sources can be influenced by the seed, then the extractor guarantees will be lost. For example, if the seed is stored in some hardware random number generator (RNG) that extracts from physical sources of randomness within the computer (e.g. timing of various events), these sources may be affected by the internal computations of the RNG itself and thus we have correlations between the seed and the sources.

Such considerations and others have motivated a revival in the study of *deterministic extractors* over the past decade, i.e. extractors that do not require a seed. Since deterministic extraction is impossible for general weak sources of randomness, this body of work has sought to identify the richest classes of sources for which deterministic extraction is possible, and construct explicit extractors for those sources. Most of the studied models of such "extractable sources" (e.g. bit-fixing sources [10], discrete control sources [29] or multiple independent sources [9]) implicitly or explicitly require independence between different portions of the source. To avoid this, Trevisan and Vadhan [37] suggested studying the class of *samplable sources*, sources generated by efficient algorithms, e.g. polynomial-sized circuits. They showed that for every t, there exist (non-explicit) deterministic extractors for sources generated by circuits of size t, provided that the min-entropy of the source is $\omega(\log t)$. Moreover, this result is based on a probabilistic argument, and can be viewed as giving an explicit *seeded* extractor that works for *seed-dependent* sources in the following sense. We generate once and for all a random seed S for the extractor, then an adversary $\mathcal{A}$ of size t generates a source $X = \mathcal{A}(S)$ (using additional randomness) with the property that X has enough min-entropy given S, and our extractor $\mathsf{Ext}(X; S)$ produces an output that is statistically close to uniform given S. (We remark that [37] also gave an explicit and seedless extractor for samplable sources having min-entropy rate close to 1 based on some strong complexity assumptions, and subsequent works have given explicit and seedless extractors for sources sampled by weaker models of computation, such as small-space algorithms [27, 28, 26] and constant-depth circuits [38].)

A deficiency of the above extractors is that their computational complexity is $\mathrm{poly}(t)$ — larger than the complexity of the adversary generating the source. As observed in [37], this is inherent. If the adversary has more resources than the extractor, then it can randomly generate inputs on which the first few bits of the extractor's output is constant (and this will be a high min-entropy source). More precisely, if the adversary's running time is larger than the extractor's by a factor of t, it can fix roughly

[1] Actually, using *2-source extractors* [9, 13], the seed can also be weakly random, but it still needs to be independent from the source.

$\log t$ bits of the output (and generate a source on n bits of min-entropy approximately $n - \log t$).

The starting point for our paper is the observation that the above attack is not so bad. If the adversary can only reduce the min-entropy of the extractor's output by a logarithmic number of bits, we have still achieved something very nontrivial and useful. Indeed, we will have what is called a *randomness condenser* [31, 32] — which takes an n-bit source with at least some k bits of min-entropy and outputs an m-bit source with at least some k' bits of min-entropy. Randomness condensers are nontrivial when the output entropy *deficiency* $m - k'$ is smaller than the input entropy deficiency $n - k$ (otherwise we could condense just by truncating the source). They have been extensively studied in the literature as a building block towards constructing randomness extractors (starting with [32], and continuing in some of the latest extractors [23]), as well as bipartite expander graphs [36, 8].

Here we note that condensers are useful in their own right. If the entropy deficiency of the output is at most β, then any event that occurs with probability p under a uniformly random string can occur under the condenser's output with probability at most $p' = 2^{\beta} \cdot p$. For example, if p is negligible and β is logarithmic, then p' is also negligible.

Motivated by the above, we initiate a study of condensers for samplable sources.

Defining seed-dependent randomness condensers. We define a condenser for seed-dependent samplable sources to be a function $\mathsf{Cond} : \{0,1\}^n \times \{0,1\}^d \to \{0,1\}^m$ with the following property. If $S \leftarrow U_d$, and $X = \mathcal{A}(S)$ is a source with (min-)entropy at least k given S, generated by a randomized circuit $\mathcal{A}$ of size at most t, then we require that $\mathsf{Cond}(X; S)$ should be (close to) a source with min-entropy at least k' given S. We provide a number of variants of this definition, using different measures of conditional entropy, and also consider the case that $\mathcal{A}$ generates side information along with X (to be discussed more below).

Condensers from CR hashing. We show that sufficiently strong collision-resistant hash functions provide good seed-dependent condensers for samplable sources. Here the seed is simply a description of a hash function h from the family, and $\mathsf{Cond}(x; h) = h(x)$. We show that if efficient algorithms can find collisions in the hash functions with probability at most $2^{\beta}/2^m$, then the condenser output will have min-entropy $k' \approx m - \beta$ given the seed (for sources of min-entropy larger than m). Note that a birthday attack will find collisions with probability $O(t^2/2^m)$ in time t. If time t algorithms cannot do much better, e.g. the probability of finding collisions is at most $\mathrm{poly}(t)/2^m$, then we can achieve entropy deficiency $\beta = O(\log t)$, within a constant factor of the lower bound mentioned above.

Condensers and key derivation. We formalize the applicability of seed-dependent condensers to key derivation. Specifically, we consider using the output of a condenser as a key in a cryptographic application, and show that for "unpredictability" applications (where an adversary can win in a security game with at most negligible probability), security is preserved if the output entropy deficiency β is small enough (e.g. logarithmic). For indistinguishability applications, we follow [3] and show that security is preserved if the "squared advantage" is negligible, which can be achieved for a number of applications. These results provide the first formal evidence that when seed-dependent sources arise in practice [24] security is not immediately compromised.

Condensers and Fiat–Shamir. We investigate seed-dependent condensers for adversaries $\mathcal{A}(S)$ that generate some *side information* Z in addition to X (with the requirement that X has min-entropy at least k given S and Z), analogously to the notion of average-case extractors introduced by [14]. We observe that the most natural generalization of our condenser definition to this setting, namely requiring that $\mathsf{Cond}(X; S)$ has min-entropy at least k' given S and Z, is impossible to achieve: the adversary $\mathcal{A}(S)$ can simply compute $Z = \mathsf{Cond}(X; S)$ as its side information. However, it seems plausible to have good condensers if we provide the side information also as input to the condenser. While this may not be feasible in some applications (because we do not know the side information), we show that condensers satisfying this definition can be used to obtain a sound implementation of the Fiat–Shamir Heuristic for all constant-round, public-coin interactive proof systems (ones with *statistical* soundness), and hence show that such protocols cannot be zero knowledge (by connections established by Dwork et al. [17]). This novel connection between the Fiat–Shamir Heuristic and randomness condensing is obtained by observing a close relation between seed-dependent condensers for samplable sources tolerating side information and some conjectures of Barak, Lindell, and Vadhan [4] (made in the study of zero knowledge and Fiat–Shamir). In fact, this connection only requires condensers for "leaky sources" — ones that are uniform prior to conditioning on the adversary's side information — and we show that such condensers are also *necessary* for soundness of the Fiat–Shamir Heuristic. It remains an intriguing open problem to give a construction of condensers for leaky sources based on some more well-studied complexity assumptions.

2 Definitions and Preliminaries

NOTATION. In a slight abuse of notation, we use upper-case letters such as X, Y to denote both variables taking on values in some set and random variables that define the distribution from which the individual values are sampled. We likewise overload the $\leftarrow$ operator, so that for variables X, Y the notation $X \leftarrow Y$ denotes assigning X the value of a variable Y and that X is a random variable with distribution equal to that of Y. For variable X and set S (e.g., $\{0,1\}^m$ for some number m), the notation $X \leftarrow S$ denotes both assigning X a value uniformly chosen from S and letting X be a uniform random variable over S. We denote by U_m the uniform distribution over m bits, i.e. $U_m \leftarrow \{0,1\}^m$. A randomized circuit $\mathcal{A}$ implicitly takes an additional input, called the randomness, that is a variable $R \leftarrow \{0,1\}^r$ for some appropriate number r. For variables X, Y and algorithm $\mathcal{A}$, we write $X \leftarrow \mathcal{A}(Y)$ to denote running $\mathcal{A}$ (with fresh randomness) on input Y and assigning the result to X. Likewise, X is the random variable describing the distribution of $\mathcal{A}(Y)$.

ENTROPY AND STATISTICAL DISTANCE. We start by defining the relevant notions of entropy that we use, which are min-entropy, collision (also known as Renyi) entropy and Shannon entropy. The *Shannon entropy* and *min-entropy* of a random variable X are defined as $\mathbf{H}_1(X) \stackrel{\text{def}}{=} \mathbb{E}_{x\leftarrow X}[-\log \Pr[X = x]]$ and $\mathbf{H}_\infty(X) \stackrel{\text{def}}{=} -\log(\max_x \Pr[X = x])$. We also define *average (aka conditional) Shannon entropy* and *average min-entropy* of a random variable X conditioned on another random variable Z by

$$\begin{aligned}
\mathbf{H}_1(X|Z) &\stackrel{\text{def}}{=} \mathbb{E}_{(x,z)\leftarrow(X,Z)}\left[-\log \Pr[X = x|Z = z]\right] \\
\mathbf{H}_\infty(X|Z) &\stackrel{\text{def}}{=} -\log\left(\mathbb{E}_{z\leftarrow Z}\left[\ \max_x \Pr[X = x|Z = z]\ \right]\right)
\end{aligned}$$

respectively, where $\mathbb{E}_{z\leftarrow Z}$ denotes the expected value over $z \leftarrow Z$.

The *collision probability* of a random variable X is defined as $\mathbf{Col}(X) \stackrel{\text{def}}{=} \sum_x \Pr[X = x]^2$, and the *collision entropy* of X is $\mathbf{H}_2(X) = \log(1/\mathbf{Col}(X))$. It is easy to see that for any X, $\mathbf{H}_\infty(X) \le \mathbf{H}_2(X) \le \mathbf{H}_1(X)$ and $\mathbf{H}_2(X) \le 2\mathbf{H}_\infty(X)$. We can also define *average* collision probability and collision entropy of a random variable X conditioned on another random variable Z by $\mathbf{Col}(X|Z) = \mathbb{E}_{z\leftarrow Z}[\mathbf{Col}(X|Z = z)]$ and $\mathbf{H}_2(X|Z) = \log(1/\mathbf{Col}(X|Z))$. Once again, $\mathbf{H}_\infty(X|Z) \le \mathbf{H}_2(X|Z) \le \mathbf{H}_1(X|Z)$ and $\mathbf{H}_2(X|Z) \le 2\mathbf{H}_\infty(X|Z)$.

We denote with $\mathsf{dist}_D(X, Y)$ the advantage of a function D in distinguishing the random variables X, Y: $\mathsf{dist}_D(X, Y) \stackrel{\text{def}}{=} |\ \Pr[D(X) = 1] - \Pr[D(Y) = 1]\ |$. The *statistical distance* between two random variables X, Y is defined by

$$\mathsf{SD}(X, Y) \stackrel{\text{def}}{=} \frac{1}{2}\sum_x |\Pr[X = x] - \Pr[Y = x]| \;=\; \max_D \ \mathsf{dist}_D(X, Y)$$

We say that X and Y are ε-*close* if $\mathsf{SD}(X, Y) \le \varepsilon$. We also note that any tuple (X, Z) is ε-close to (X', Z) such that $\mathbf{H}_\infty(X'|Z) \ge \mathbf{H}_2(X|Z) - \log(1/\varepsilon)$, which is often much better than bounding $\mathbf{H}_\infty(X|Z) \ge \frac{1}{2} \cdot \mathbf{H}_2(X|Z)$.

Extractors and Condensers. Recall, a randomness extractor [30] (from now on simply extractor) takes a source X of high entropy and outputs an almost uniform random string. To achieve such an ambitious task, an extractor requires an additional random seed S. Below we further generalize the notion of extractor to two notions of entropy we defined (min- and collision),[2] as well as to the setting when the source X is *efficiently samplable* by some circuit $\mathcal{A}$ of some bounded size t. For our purposes it is useful to think of $\mathcal{A}$ as a *probabilistic adversary* trying to produce the "worst" possible (high-entropy) source X to "fool" our extractor.

Definition 2.1 (Extractor) *Let $c \in \{2, \infty\}$. An efficient function $\mathsf{Ext} : \{0,1\}^n \times \{0,1\}^d \to \{0,1\}^m$ is a* $([\mathbf{H}_c \geq k], \varepsilon, t)$-extractor *if for all probabilistic adversaries $\mathcal{A}$ of size at most t who sample a distribution $X \leftarrow \mathcal{A}$ of entropy $\mathbf{H}_c(X) \geq k$, the joint distribution $(S, \mathsf{Ext}(X; S))$ is ε-close to (S, U_m), where the seed $S \leftarrow \{0,1\}^d$ is chosen independently of X.*

When c is clear from the context, we say Ext is a (k, ε, t)-extractor. We omit the reference to t when $t = \infty$.

Notice, an extractor outputs an m-bit string statistically close to uniform (even given the seed), which is a string with "full entropy". Condensers, defined below, relax this requirement to outputting a (value close to a) string of "high", but not necessarily "full", entropy. For full generality, we allow the notion of "output entropy" $\mathbf{H}_{c'}$ to be different from the notion of "input entropy" $\mathbf{H}_c$.

Definition 2.2 (Condenser) *Let $c, c' \in \{1, 2, \infty\}$. An efficient function $\mathsf{Cond} : \{0,1\}^n \times \{0,1\}^d \to \{0,1\}^m$ is a* $([\mathbf{H}_c \geq k] \to_\varepsilon [\mathbf{H}_{c'} \geq k'], t)$-condenser *if for all probabilistic adversaries $\mathcal{A}$ of size at most t who sample a distribution $X \leftarrow \mathcal{A}$ of entropy $\mathbf{H}_c(X) \geq k$, the joint distribution $(S, \mathsf{Cond}(X; S))$ is ε-close to some (S, R), where $\mathbf{H}_{c'}(R|S) \geq k'$ and the seed $S \leftarrow \{0,1\}^d$ is chosen independently of X.*

The quantity $\beta \stackrel{def}{=} m - k'$ is called the entropy deficit *of the condenser. When $c = c'$ is clear from the context, we say that Cond is a $(k \to_\varepsilon k', t)$-condenser. We omit the reference to ε and/or t when $\varepsilon = 0$ and/or $t = \infty$, respectively.*

Clearly, an extractor is a special case of a condenser whose entropy deficit is $\beta = 0$, which means that $R \leftarrow \{0,1\}^m$ and $m = k'$.

3 Seed-Dependent Condensers

We now generalize the notion of a condenser to the *seed-dependent* setting, in which the adversarial sampler $\mathcal{A}$ of size t can depend on the seed S. As we will see, seed-dependent condensers are useful for important applications such as cryptographic key derivation.

Definition 3.1 (Seed-Dependent Condenser) *Let $c, c' \in \{1, 2, \infty\}$. An efficient function $\mathsf{Cond} : \{0,1\}^n \times \{0,1\}^d \to \{0,1\}^m$ is a* seed-dependent $([\mathbf{H}_c \geq k] \to_\varepsilon [\mathbf{H}_{c'} \geq k'], t)$-condenser *if for all probabilistic adversaries $\mathcal{A}$ of size at most t who take a random seed $S \leftarrow \{0,1\}^d$ and output (using more coins) a sample $X \leftarrow \mathcal{A}(S)$ of entropy $\mathbf{H}_c(X|S) \geq k$, the joint distribution $(S, \mathsf{Cond}(X; S))$ is ε-close to some (S, R), where $\mathbf{H}_{c'}(R|S) \geq k'$.*

The quantity $\beta \stackrel{def}{=} m - k'$ is called the entropy deficit *of the condenser. When $c = c'$ is clear from the context, we say that Cond is a seed-dependent $(k \to_\varepsilon k', t)$-condenser. We omit the reference to ε and/or t when $\varepsilon = 0$ and/or $t = \infty$, respectively.*

A notion for traditional condensers arises by replacing $\mathcal{A}$ in the definition above with an unbounded circuit that does not take the seed S as input. Unlike with traditional condensers, seed-dependent

[2]It is easy to see that no meaningful parameters can be achieved for Shannon entropy.

condensers require that $\mathcal{A}$ be efficient. Otherwise, an inefficient $\mathcal{A}$ can, by repeatedly evaluating the condenser using the seed S, always find a high entropy distribution of inputs that map to a low entropy output distribution. Second, while a seed-dependent extractor can be defined as a special case of the definition above corresponding to $k' = m$, Proposition 3.1 below implies that it is impossible to build a (non-trivial) seed-dependent extractor.

The following lemma will be useful in several of our later results.

Lemma 3.1 *Let $c \in \{1, 2, \infty\}$. Then,*

- **"Output $(\infty \to 2 \to 1)$":** *If $c' \geq c''$ and Cond is a seed-dependent $(([\mathbf{H}_c \geq k] \to_\varepsilon [\mathbf{H}_{c'} \geq k']), t)$-condenser, then Cond is also a seed-dependent $(([\mathbf{H}_c \geq k] \to_\varepsilon [\mathbf{H}_{c''} \geq k']), t)$-condenser.*
- **"Output $(2 \to \infty)$":** *For any $\gamma > 0$, if Cond is seed-dependent $(([\mathbf{H}_c \geq k] \to_\varepsilon [\mathbf{H}_2 \geq k']), t)$-condenser, then Cond is also a seed-dependent $(([\mathbf{H}_c \geq k] \to_{\varepsilon+\gamma} [\mathbf{H}_\infty \geq k' - \log(1/\gamma)]), t)$-condenser and also a seed-dependent $(([\mathbf{H}_c \geq k] \to_\varepsilon [\mathbf{H}_\infty \geq k'/2]), t)$-condenser.*
- **"Input $(1 \to 2 \to \infty)$":** *If $c' \leq c''$ and Cond is seed-dependent $(([\mathbf{H}_{c'} \geq k] \to_\varepsilon [\mathbf{H}_c \geq k']), t)$-condenser, then Cond is also a seed-dependent $(([\mathbf{H}_{c''} \geq k] \to_\varepsilon [\mathbf{H}_c \geq k']), t)$-condenser.*

Proof. The first part follows from $\mathbf{H}_1(R|S) \geq \mathbf{H}_2(R|S) \geq \mathbf{H}_\infty(R|S)$. The second part follows from the fact that (R, S) is γ-close to some (R', S) where $\mathbf{H}_\infty(R'|S) \geq \mathbf{H}_2(R|S) - \log(1/\gamma)$ and from $\mathbf{H}_2(R|S) \leq 2\mathbf{H}_\infty(R|S)$. The third part follows from $\mathbf{H}_1(X|S) \geq \mathbf{H}_2(X|S) \geq \mathbf{H}_\infty(X|S)$ □

Thus, it is somewhat preferable (but also the hardest) to build a seed-dependent $([\mathbf{H}_2 \geq k] \to_\varepsilon [\mathbf{H}_\infty \geq k'])$ condenser, since it implies $([\mathbf{H}_c \geq k] \to_\varepsilon [\mathbf{H}_{c'} \geq k'])$-condenser for any $c, c' \in \{2, \infty\}$. In contrast, it is preferable to base a security of a given application on a $([\mathbf{H}_\infty \geq k] \to_\varepsilon [\mathbf{H}_2 \geq k'])$-condenser, since such condensers are likely to have slightly better parameters k and k'.

The following negative result shows that the output entropy deficiency $\beta = m - k'$ must be at least roughly $\log t$ to work for samplers computable in time t, if the condenser is computable in time significantly less than t. In particular, we cannot hope for a seed-dependent *extractor* (i.e. $\beta = 0$) that is computable in time significantly less than t, generalizing an observation of Trevisan and Vadhan [37] about deterministic extractors for samplable sources.

Proposition 3.1 *Let $\mathsf{Cond} : \{0,1\}^n \times \{0,1\}^d \to \{0,1\}^m$ be computable by a circuit of size t', and let $\beta \in [0, m]$, $\varepsilon, \delta \in (0, 1/2)$. Then for Cond to be a $(([\mathbf{H}_\infty \geq n - \alpha] \to_\varepsilon [\mathbf{H}_1 \geq m - \beta]), t)$-condenser for $\alpha = \lceil (\beta + 1)/(1 - \varepsilon - \delta) \rceil$, it must be that $\alpha \geq \log t - \log t' - O(\log(1/\delta))$ or $\alpha \geq m$.*

Note that as $\varepsilon, \delta \to 0$, the ratio between α and β approaches 1. Thus, the proposition says that if we want to decrease the entropy deficiency by any significant factor, we must settle for output entropy deficiency $\beta \approx \alpha$ that is at least roughly $\log t$.

Proof. Without loss of generality, assume that the output length of Cond is $m = \alpha$. (Otherwise, we can restrict to an α-length prefix of the output of Cond; doing so does not change the entropy *deficiency* of the output of Cond.) For a source string $x \in \{0,1\}^n$ and a seed $s \in \{0,1\}^d$, let $p(x, s)$ be the fraction of source strings x' that produce the same condenser output as x on seed s. That is,

$$p(x, s) = \frac{\#\{x' : \mathsf{Cond}(x', s) = \mathsf{Cond}(x, s)\}}{2^n}.$$

Note that for every seed s, we have

$$\begin{aligned}
\mathbb{E}_{X\leftarrow\{0,1\}^n}\left[\frac{1}{p(X,s)}\right] &= \sum_{x\in\{0,1\}^n}\frac{1}{\#\{x' : \mathsf{Cond}(x',s)=\mathsf{Cond}(x,s)\}} \\
&= \sum_{r\in\mathsf{Cond}(\{0,1\}^n,s)}\ \sum_{x\in\{0,1\}^n:\mathsf{Cond}(x,s)=r}\frac{1}{\#\{x' : \mathsf{Cond}(x',s)=r\}} \\
&= |\mathsf{Cond}(\{0,1\}^n,s)| \\
&\le 2^\alpha.
\end{aligned}$$

Thus, by averaging, there exists an fixed $x^* \in \{0,1\}^n$ such that

$$\mathbb{E}_{S\leftarrow\{0,1\}^d}\left[\frac{1}{p(x^*,S)}\right] \le 2^\alpha.$$

Now our adversarial sampler $\mathcal{A}(s)$ behaves as follows: it selects $\ell = \lceil 2\cdot 2^\alpha\cdot(1/\delta)\cdot\ln(2/\delta)\rceil$ random inputs $x_1,\ldots,x_\ell \leftarrow \{0,1\}^n$, and outputs the first value x_i such that $\mathsf{Cond}(x_i,s)=\mathsf{Cond}(x^*,s)$. If no such value occurs, then $\mathcal{A}(s)$ outputs a uniformly random string $x\leftarrow\{0,1\}^n$.

By inspection, $\mathcal{A}$ is computable by a circuit of size $O(\ell\cdot t')$, which is smaller than t for $\alpha = \log t - \log t' - O(\log(1/\delta))$.

We now analyze the min-entropy of the source generated by $\mathcal{A}$. Notice that for every seed s, the output distribution of $\mathcal{A}(s)$ has min-entropy at least $\log(p(x^*,s)\cdot 2^n)$. Thus,

$$2^{-\mathbf{H}_\infty(\mathcal{A}(S)|S)} = \mathbb{E}_{s\leftarrow S}\left[2^{-\mathbf{H}_\infty(\mathcal{A}(s))}\right] \le \mathbb{E}_{s\leftarrow S}\left[\frac{1}{p(x^*,s)\cdot 2^n}\right] \le \frac{2^\alpha}{2^n},$$

so we have $\mathbf{H}_\infty(\mathcal{A}(S)|S) \ge n-\alpha$, as desired.

Now we look at the output Shannon entropy of the condenser when fed this source. First, by Markov's inequality, we have $1/p(x^*,s) \le 2^\alpha/(\delta/2)$ for all but a $\delta/2$ fraction of seeds s. For each such seed s, the probability that $\mathcal{A}$ fails to find a value x_i such that $\mathsf{Cond}(x_i,s)=\mathsf{Cond}(x^*,s)$ is at most $(1-p(x^*,s))^\ell < \delta/2$. Thus, with probability greater than $1-\delta$, $\mathsf{Cond}(\mathcal{A}(S),S)=\mathsf{Cond}(x^*,S)$. And for any source (R,S) that is ε-close to $(\mathsf{Cond}(\mathcal{A}(S)),S)$, we have $R=\mathsf{Cond}(x^*,S)$ with probability greater than $1-\delta-\epsilon$. This implies that the entropy of R given S is less than $(\delta+\epsilon)\cdot\alpha+1 \le \alpha-\beta = m-\beta$. (The extra one bit of entropy is for the indicator of the event that $R=\mathsf{Cond}(x^*,S)$.) $\square$

Handling Side Information. One can naturally generalize the notion of (regular) extractors and condensers to handle some *side information* Z about the source X, yielding the notion of *average-case* extractors/condensers [14]. Formally, the adversarial sampler $\mathcal{A}$ produces a pair (X,Z) such that $\mathbf{H}_c(X|Z)\ge k$, and one requires that the joint distribution $(Z,S,\mathsf{Ext}(X;S))$ (resp. $(Z,S,\mathsf{Cond}(X;S))$) is ε-close to (Z,S,U_m) (resp. (Z,S,R) where $\mathbf{H}_{c'}(R|(S,Z))\ge k'$).

However, things become a bit trickier in the seed-dependent case that we introduce in this work. Naturally, the sampler $\mathcal{A}$ now takes the seed S to produce the pair (X,Z). Unfortunately, this means that $\mathcal{A}$ can now run the condenser $\mathsf{Cond}(X;S)$ and simply record all or part of this output in the side information Z. This still leaves the entropy of X high enough (say, if k is noticeably larger than m), but now the output entropy k' drops to 0. Thus, to make a meaningful but *satisfiable* definition in the case of side information, we will relax the syntax of the condenser Cond to also take the side information Z as part of its input. While less convenient for some applications, now the previous attack no longer applies, since the sampler $\mathcal{A}(S)$ has to choose Z before $R=\mathsf{Cond}((X,Z);S)$ is derived, making it much harder to "correlate" R and Z.

Definition 3.2 (Average-Case Seed-Dependent Condenser) *Let $c,c'\in\{1,2,\infty\}$. An efficient function* $\mathsf{Cond} : \{0,1\}^n\times\{0,1\}^*\times\{0,1\}^d\to\{0,1\}^m$ *is an* average-case seed-dependent $([\mathbf{H}_c\ge k]\to_\varepsilon$

$[\mathbf{H}_{c'} \geq k'], t)$-condenser *if for all probabilistic adversaries* $\mathcal{A}$ *of size at most* t *who take a random seed* $S \leftarrow \{0,1\}^d$ *and output (using more coins) a tuple* $(X, Z) \leftarrow \mathcal{A}(S)$ *satisfying* $\mathbf{H}_c(X|(S,Z)) \geq k$*, the joint distribution* $(Z, S, \mathsf{Cond}((X,Z); S))$ *is* ε*-close to* (Z, S, R)*, where* $\mathbf{H}_{c'}(R|(S,Z)) \geq k'$*.*

The quantity $\beta \stackrel{def}{=} m - k'$ *is called the* entropy deficit *of the condenser. When* $c = c'$ *is clear from the context, we say that* Cond *is an average-case seed-dependent* $(k \to_\varepsilon k', t)$*-condenser. We omit the reference to* ε *and/or* t *when* $\varepsilon = 0$ *and/or* $t = \infty$*, respectively.*

We notice that Lemma 3.1 clearly extends to the average-case setting. Also, when Z is empty, this still generalizes the "worst-case" seed-dependent condenser from Definition 3.1. However, the introduction of side information makes the notion of seed-dependent condenser very non-trivial to satisfy even when the source X is perfectly uniform, but some side information $Z = f(X)$ is "leaked" to the attacker. Indeed, we show in Section 6 that this special case of average-case condensers (see Definition 6.1) is exactly what is needed to instantiate the Fiat-Shamir heuristic.

Finally, an equivalent way to think about average-case condensers is to interpret the output (X, Z) of the sampler as a single (variable-length) source X', so that the condenser is simply applied to X', but a subset of (known) *physical bits* Z of X' is leaked to the attacker/distinguisher. This setting could be viewed as a variant of *exposure-resilient cryptography* [7, 16], where we do not assume that the initial distribution X' of the secret is uniform (but rather is efficiently samplable and has some entropy conditioned on the leaked bits), and where the set of bits that are leaked is known (namely it is the suffix Z) rather than adversarial.

4 Condensers from Collision Resistance

In this section we show that a sufficiently strong collision-resistant hash function (CRHF) gives a good seed-dependent (but *not average-case*) $([\mathbf{H}_2 \geq k] \to_0 [\mathbf{H}_2 \geq k'])$ condenser, which also implies non-trivial bounds for other input/output entropy settings when $c, c' \in \{2, \infty\}$, by Lemma 3.1.

Definition 4.1 *A family of hash function* $\mathcal{H} = \{h : \{0,1\}^* \to \{0,1\}^m\}$ *is* (t, δ)-collision-resistant *if for any (non-uniform) attacker* $\mathcal{B}$ *of size at most* t*,*

$$\Pr[H(X_1) = H(X_2) \wedge X_1 \neq X_2] \leq \delta$$

where $H \leftarrow \mathcal{H}$ *and* $(X_1, X_2) \leftarrow \mathcal{B}(H)$*.*

Theorem 4.1 *Fix any* $\beta > 0$*. If* $\mathcal{H}$ *is a* $(2t, 2^{\beta-1}/2^m)$*-collision-resistant hash function family, then* $\mathsf{Cond}(X; H) \stackrel{\mathsf{def}}{=} H(x)$ *for* $H \leftarrow \mathcal{H}$ *is a seed-dependent* $(([\mathbf{H}_2 \geq m - \beta + 1] \to [\mathbf{H}_2 \geq m - \beta]), t)$*-condenser with entropy deficit* β *and no error.*

In particular, it is also a seed-dependent $(([\mathbf{H}_\infty \geq m - \beta + 1] \to [\mathbf{H}_2 \geq m - \beta]), t)$*-condenser and* $(([\mathbf{H}_\infty \geq m - \beta + 1] \to_\varepsilon [\mathbf{H}_\infty \geq m - \beta + \log \varepsilon]), t)$*-condenser.*

Proof. Consider any (t, k)-bounded sampler $\mathcal{A}$ with $k \geq m - \beta + 1$. Define a collision-finding adversary $\mathcal{B}(H)$ to work by running $X_1 \leftarrow \mathcal{A}(H)$ and $X_2 \leftarrow \mathcal{A}(H)$, and then outputting (X_1, X_2). Clearly, $\mathcal{B}$ runs in time at most $2t$. From the security of $\mathcal{H}$, and letting $H \leftarrow \mathcal{H}$ and $(X_1, X_2) \leftarrow \mathcal{B}(H)$, we have that

$$\Pr[H(X_1) = H(X_2) \ \wedge \ X_1 \neq X_2] \leq \frac{2^{\beta-1}}{2^m} \tag{1}$$

Hence, combining Equation (1) with the fact that $\mathbf{Col}(X \mid H) \leq 2^{\beta-1}/2^m$ for $X \leftarrow \mathcal{A}(H)$, we get

$$\begin{aligned} \mathbf{Col}(H(X) \mid H) &\leq \mathbf{Col}(X \mid H) + \Pr[H(X_1) = H(X_2) \ \wedge \ X_1 \neq X_2] \\ &\leq \frac{2^{\beta-1}}{2^m} + \frac{2^{\beta-1}}{2^m} = \frac{2^\beta}{2^m} \end{aligned}$$

The implications to other entropy norms are immediate from Lemma 3.1. □

Parameters. To obtain good entropy deficit β as a function on the sampler's complexity t, we need to understand the best possible $(2t, \delta)$-collision-resistant security of $\mathcal{H}$. Clearly, a birthday attack (essentially) implies that $\delta = \Omega(t^2/2^m)$, since the attacker can pick t random points, evaluate h on them, and hope for some collision. Conversely, this bound is tight in the random oracle model, and state-of-the-art hash functions more or less assume that the "birthday attack" is the only possible attack on a good CRHF design. For example, birthday attacks are currently the best known attacks on many popular hash functions, such as SHA-256, SHA-512, and the new SHA-3 functions, as well as discrete-log based CRHFs over many elliptic curve groups (c.f., [35]). Thus, under such (strong but reasonable) assumptions, all the above popular hash functions achieve $\delta = O(t^2/2^m)$, which means that we can set $2^{\beta-1} = O(t^2)$ resulting in $\beta = 2\log t + O(1)$. More generally, if the best collision-finding attack has success probability $\delta = \mathrm{poly}(t)/2^m$, then $\beta = O(\log t)$.

Corollary 4.1 *Assuming the existence of $(t, \frac{O(t^2)}{2^m})$-collision-resistant hash functions, there exists a seed-dependent $(([\mathbf{H}_2 \geq m - \beta + 1] \to [\mathbf{H}_2 \geq m - \beta]), t)$-condenser with entropy deficit $\beta = 2\log t + O(1)$ and no error.*

In particular, it is also a seed-dependent $(([\mathbf{H}_\infty \geq m - \beta + 1] \to [\mathbf{H}_2 \geq m - \beta]), t)$-condenser with entropy deficit $\beta = 2\log t + O(1)$ and no error, and $(([\mathbf{H}_\infty \geq m - \beta + 1] \to_\varepsilon [\mathbf{H}_\infty \geq m - \beta - \log(1/\varepsilon)]), t)$-condenser with entropy deficit $\beta' = (2\log t + \log(1/\varepsilon) + O(1))$ and error ε.

Average-Case Setting? Unfortunately, the proof of Theorem 4.1 does not extend to average-case seed-dependent condensers. The problem is that when estimating the value $\mathbf{Col}(H(X,Z)|(H,Z))$, one already needs to sample two sources X_1 and X_2 corresponding to the *same side information* Z, which seems to be hard. A bit more formally, a natural attempt to define a collision-finding adversary $\mathcal{B}$ would be to first let $\mathcal{B}(H)$ run $\mathcal{A}(H)$ to produce a tuple (X_1, Z_1), and then run $\mathcal{A}(H)$ several more times to try to produce a second tuple (X_2, Z_2) with the hope that $Z_2 = Z_1$. But this will not be guaranteed to be efficient unless Z is very short (e.g., just a few bits). In some sense, the difficulty of handling side information might be expected, since we show that average-case seed-dependent condensers are enough to instantiate the random oracle in the Fiat-Shamir heuristic (see Section 6), which is a long-standing open problem.

5 Application to Key Derivation

Consider any cryptographic primitive P (e.g., digital signatures, encryption, etc.), which uses randomness $R \in \{0,1\}^m$ to derive its secret (and, public, if needed) key(s). Without loss of generality, we can assume that R itself is the secret key. In the "ideal" setting, $R \leftarrow \{0,1\}^m$ is chosen uniformly at random, and the attacker $\mathcal{B}$ against P obtains no knowledge about the choice of R, except for what is revealed by P. In practice, however, R is not perfectly uniform. For example, it may be the output of a system random number generator (RNG) that attempts to extract uniform bits from a source of entropy. To guarantee security for the widest range of settings, we ask for the key-derivation to be secure even against seed-dependent[3], adversarially-manipulated sources. However, Proposition 3.1 shows that, at least in general, no extractors exist that work for such a strong adversarial model. We therefore turn to seed-dependent condensers, showing that these yield strong positive results about the security of key-derivation.

Towards this, we model the "real" seed-dependent setting as follows. Let $S \leftarrow \{0,1\}^d$ be a random seed that is chosen and $X \leftarrow \mathcal{A}(S)$ is sampled by an adversarial sampler $\mathcal{A}$. Finally, the cryptographic primitive P uses $R \leftarrow \mathsf{Cond}(X; S)$ as the key. While the above model is the one of greatest most direct

[3]For example the Linux RNG folds back into its entropy pool prior outputs [24].

practical interest, we will actually consider the more general case of *average-case* condensing, in which an attacker $\mathcal{B}$ against P obtains part of the input to the condenser, the side-information Z. The resulting real/ideal settings for deriving the key for P are formalized by the procedures $\mathsf{Real}(\mathcal{A})$ and $\mathsf{Ideal}(\mathcal{A})$:

$\mathsf{Real}(\mathcal{A})$:	$\mathsf{Ideal}(\mathcal{A})$:
$S \leftarrow \{0,1\}^d$	$S \leftarrow \{0,1\}^d$
$(X, Z) \leftarrow \mathcal{A}(S)$	$(X, Z) \leftarrow \mathcal{A}(S)$
$R \leftarrow \mathsf{Cond}((X, Z); S)$	$R \leftarrow \{0,1\}^m$
Return (R, S, Z)	Return (R, S, Z)

The two procedures are parameterized by a sampler $\mathcal{A}$ that on input the seed S outputs a pair (X, Z). We assume that the sampler $\mathcal{A}$ has size at most t and produces a source X of (conditional) min-entropy $\mathbf{H}_\infty(X|(S, Z)) \geq k$, for some parameters t and k. We call such samplers (t, k)-*bounded*. Sometimes, to emphasize the dependence on the sampler complexity t and source min-entropy k, we will refer to the above two settings as the (t, k)-*real* and (t, k)-*ideal* models, respectively.

The side information Z naturally models information about the random source X that may be leaked to an adversary via a side channel. However, in most or all practical scenarios, our assumption that the value of Z is known and available to the condenser is unrealistic. Thus, we will also state our results for the analogous models *without side information*, meaning we omit Z in both the real and ideal models.

Defining Real/Ideal Security. We assume that the security of the cryptographic primitive P is defined via an interactive game between a probabilistic attacker $\mathcal{B}(s, z)$ and a probabilistic challenger $\mathcal{C}(r)$. Here one should think of s and z as particular values of the seed and the side information, respectively, and r as a particular value used by the challenger in the key generation algorithm of P. We note that $\mathcal{C}$ only uses the secret key r and does not directly depend on s and z. In particular, in the ideal model, the values s and z are not really useful to the actual attacker $\mathcal{B}$, since the key r used by the challenger $\mathcal{C}$ is chosen completely independently from these values. Still, we include them for consistency.

At the end of the game, $\mathcal{C}(r)$ outputs a bit b, where $b = 1$ indicates that the attacker "won the game". Since $\mathcal{C}$ is fixed by the definition of P (e.g., $\mathcal{C}$ runs the unforgeability game for signature or the semantic security game for encryption, etc.), we denote by $\mathcal{D}_{\mathcal{B}}(r, s, z)$ the (abstract) distinguisher which simulates the entire game between $\mathcal{B}(s, z)$ and $\mathcal{C}(r)$ and outputs the bit b. We also let

$$\mathbf{Adv}_{\mathcal{B}}(r, s, z) \stackrel{\text{def}}{=} \Pr[\mathcal{D}_{\mathcal{B}}(r, s, z) = 1] - c$$

be the advantage of $\mathcal{B}(s, z)$ to win the game against $\mathcal{C}(r)$, where $c = 0$ for unpredictability applications (one-way functions, signatures, etc.) and $c = 1/2$ for indistinguishability applications (encryption, pseudorandom functions, etc.). Thus, $\mathbf{Adv}_{\mathcal{B}}(\cdot) \in [0, 1]$ for unpredictability applications and $\mathbf{Adv}_{\mathcal{B}}(\cdot) \in [-\frac{1}{2}, \frac{1}{2}]$ for indistinguishability applications. When $\mathcal{B}$ is clear from the context, we simply write $\mathbf{Adv}(r, s, z)$.

In the following security definition for P, we will use the letter T to denote the maximum allowable resources of $\mathcal{B}$, which include all the efficiency measures we might care about in the corresponding application, such as the circuit size, number of oracle queries, etc. We say that such a $\mathcal{B}$ is T-*limited*.

Definition 5.1 *Given a sampler $\mathcal{A}$ and an attacker $\mathcal{B}$, we define their* ideal advantage

$$\mathbf{\Delta}(\mathcal{A}, \mathcal{B}) \stackrel{\text{def}}{=} |\ \mathbb{E}[\mathbf{Adv}_{\mathcal{B}}(\mathsf{Ideal}(\mathcal{A}))]\ | \,.$$

We say that P is (T, δ)-secure *in the (t, k)-ideal model if for any (t, k)-bounded sampler $\mathcal{A}$ and any T-limited attacker $\mathcal{B}$, $\mathbf{\Delta}(\mathcal{A}, \mathcal{B}) \leq \delta$. Similarly, given $\mathcal{A}$ and $\mathcal{B}$, we define their* real advantage

$$\widetilde{\mathbf{\Delta}}(\mathcal{A}, \mathcal{B}) \stackrel{\text{def}}{=} |\ \mathbb{E}[\mathbf{Adv}_{\mathcal{B}}(\mathsf{Real}(\mathcal{A}))]\ | \,.$$

We say that P is (T', δ')-secure *in the (t, k)-real model if for any (t, k)-bounded sampler $\mathcal{A}$ and any T'-limited attacker $\mathcal{B}$, $\widetilde{\mathbf{\Delta}}(\mathcal{A}, \mathcal{B}) \leq \delta'$.*

5.1 Simple Bound for Unpredictability Applications

As our first attempt, we would like to argue that if P is (T, δ)-secure in the ideal setting, then P is also (T', δ')-secure in the real setting, where T' is not much lower than T, and, more importantly, δ' is not much larger than δ. With traditional extractors, this is done by arguing that the derived real key R is (statistically) ε-close to U_m, even conditioned on S and Z. This means that $\delta' \leq \delta + \varepsilon$. Unfortunately, in the seed-dependent settings it is impossible to achieve statistical extraction, as shown by Proposition 3.1. In this section, we observe that is not strictly necessary to argue statistical extraction: if the original ideal security δ is low enough, a good enough condenser (achievable even in the seed-dependent setting) might result in "real" security δ' not much larger than the "ideal" security δ. At least, we show that this intuition is true for unpredictability applications (where, recall, $\mathbf{Adv}(\cdot) \geq 0$) in the following lemma.

Lemma 5.1 *Assume P is some unpredictability application which is (T, δ)-secure in the (t, k)-ideal model, and* Cond *is an average-case seed-dependent $(([\mathbf{H}_\infty \geq k] \to_\varepsilon [\mathbf{H}_\infty \geq k']), t)$-condenser with entropy deficit $\beta = m - k'$. Then P is (T, δ')-secure in the (t, k)-real model, where*

$$\delta' \leq \varepsilon + \delta \cdot 2^\beta. \tag{2}$$

If instead Cond *is an (non-average-case) seed-dependent $(([\mathbf{H}_\infty \geq k] \to_\varepsilon [\mathbf{H}_\infty \geq k']), t)$-condenser, then P is (T, δ')-secure in the (t, k)-real model without side information.*

Proof. Fix any (t, k)-bounded sampler $\mathcal{A}$ and any T-limited attacker $\mathcal{B}$. Recall, the random variable $(\mathsf{Cond}((X, Z); S), S, Z)$ is ε-close to a random variable (R, S, Z), where $\mathbf{H}_\infty(R|(S, Z)) \geq k'$. Thus, to prove the bound in Equation (2), it is enough to prove that $\delta' \leq \delta \cdot 2^\beta$ assuming that the value R in the real model satisfies $\mathbf{H}_\infty(R|(S, Z)) \geq k'$.

For brevity, let $y = (s, z)$, and q_y denote the probability that $Y = y$ (which is the same, for every $y = (s, z)$, in both the real and the ideal models). Also, for a given $r \in \{0, 1\}^m$, let $p_{r|y}$ be the probability of outputting $R = r$ in the real model conditioned on $Y = y$. By definition, $p_{r|y} \leq 2^{-\mathbf{H}_\infty(R|Y=y)}$ and $\mathbb{E}_y[2^{-\mathbf{H}_\infty(R|Y=y)}] = 2^{-\mathbf{H}_\infty(R|Y)} \leq 2^{-k'}$ by the condenser property. Then, using non-negativity of $\mathbf{Adv}_\mathcal{B}$, we have

$$\begin{aligned}
\widetilde{\mathbf{\Delta}}(\mathcal{A}, \mathcal{B}) &= \sum_{y,r} q_y \cdot p_{r|y} \cdot \mathbf{Adv}_\mathcal{B}(r, y) \leq \sum_y q_y \cdot 2^{-\mathbf{H}_\infty(R|Y=y)} \cdot \left(\sum_r \mathbf{Adv}_\mathcal{B}(r, y)\right) \\
&\leq \sum_y q_y \cdot 2^{-\mathbf{H}_\infty(R|Y=y)} \cdot (2^m \cdot \delta) = \delta \cdot 2^{m - \mathbf{H}_\infty(R|Y)} \leq \delta \cdot 2^{m-k'} \\
&= \delta \cdot 2^\beta
\end{aligned}$$

where the second inequality follows from the fact that, for any y, $(\sum_r \frac{1}{2^m} \cdot \mathbf{Adv}_\mathcal{B}(r, y) \leq \delta)$, which, in turn, is true since $\mathcal{B}(\cdot, y)$ defines a valid T-limited ideal model attacker $\mathcal{B}'$, who replaces his actual "advice" $y' = (s', z')$ with the fixed "advice" y without affecting its success probability, since the ideal key $R \leftarrow \{0, 1\}^m$ is independent from y and y'. $\square$

Parameters. In essence, Lemma 5.1 states that the security δ degrades exponentially with the entropy deficit β of our seed-dependent condenser. Recall that $\beta = O(\log t)$ is the best we can hope for (by Proposition 3.1); this would give a meaningful security guarantee $\delta' \approx \delta \cdot \text{poly}(t)$, as long as $\delta \ll 1/\text{poly}(t)$.

For example, for the non-average-case setting, we can combine the bound in Lemma 5.1 with the construction from Corollary 4.1 to show that a $\mathcal{O}(t^2)/2^m$-collision-resistant hash function suffices for real model security:

Corollary 5.1 *Assuming the existence of $(t, \frac{O(t^2)}{2^m})$-collision-resistant hash functions, if P is (T,δ)-secure in the $(t, m-2\log t+O(1))$-ideal model with no side information, then using a collision-resistant function as a condenser makes P to be (T,δ')-secure in the $(t, m-2\log t+O(1))$-real model with no side information, where*

$$\delta' \leq O(t\cdot\sqrt{\delta})\ . \tag{3}$$

Proof. For any $\varepsilon > 0$, Corollary 4.1 gives us a $(([\mathbf{H}_\infty \geq m-\beta'] \to_\varepsilon [\mathbf{H}_\infty \geq m-\beta'-\log(1/\varepsilon)]), t)$-condenser with error ε, where $\beta' = 2\log t + O(1)$. This gives entropy deficit $\beta = \beta' + \log(1/\varepsilon)$ satisfying $2^\beta = O(t^2/\varepsilon)$. Using Lemma 5.1, we get $\delta' \leq \varepsilon + O(\delta t^2/\varepsilon)$. Setting, $\varepsilon = t\cdot\sqrt{\delta}$ gives the desired bound $\delta' \leq O(t\cdot\sqrt{\delta})$. □

5.2 General Bound through Squared Advantage

The bound of Lemma 5.1 only holds for unpredictability applications, and also requires seed-dependent condensers guaranteeing the *min-entropy* of the extracted key R. In this section we show a more general bound which also holds for indistinguishability applications, has better dependence on the entropy deficit of the condenser, and needs a slightly weaker type of seed-dependent condenser for *collision* entropy. However, the small price we pay for such improvements is that we can no longer directly relate the real-security δ' of our application to its ideal security δ. Rather, we use the notion of the *squared advantage* $\mathbf{\Delta}_2(\mathcal{A},\mathcal{B})$, and will relate $\widetilde{\mathbf{\Delta}}(\mathcal{A},\mathcal{B})$ to $\mathbf{\Delta}_2(\mathcal{A},\mathcal{B})$, which will in turn relate δ' to the "square-security" σ which we define below. This notion of squared advantage/security was implicitly introduced by Barak et al. [3] in the "seed-independent" setting (to improve the entropy loss of the Leftover Hash Lemma), who also showed that for many important applications the value σ is not "too much worse" than δ (as discussed more below).

Definition 5.2 *Given a sampler $\mathcal{A}$ and an attacker $\mathcal{B}$, we define their* (ideal) square advantage

$$\mathbf{\Delta}_2(\mathcal{A},\mathcal{B}) \stackrel{\mathsf{def}}{=} \mathbb{E}[\mathbf{Adv}_{\mathcal{B}}(\mathsf{Ideal}(\mathcal{A}))^2]\ .$$

We say that P is (T,σ)-square-secure *in the (t,k)-ideal model if for any (t,k)-bounded sampler $\mathcal{A}$ and any T-limited attacker $\mathcal{B}$, $\mathbf{\Delta}_2(\mathcal{A},\mathcal{B}) \leq \sigma$.*

We can now state our improved bound, and then compare it to our previous bound from Lemma 5.1.

Lemma 5.2 *Assume P any application which is (T,σ)-square-secure in the (t,k)-ideal model, and* Cond *is an average-case seed-dependent $(([\mathbf{H}_\infty \geq k] \to_\varepsilon [\mathbf{H}_2 \geq k']), t)$-condenser with entropy deficit $\beta = m-k'$. Then P is (T,δ')-secure in the (t,k)-real model, where*

$$\delta' \leq \varepsilon + \sqrt{\sigma\cdot 2^\beta}. \tag{4}$$

If instead Cond *is an (non-average-case) seed-dependent $(([\mathbf{H}_\infty \geq k] \to_\varepsilon [\mathbf{H}_\infty \geq k']), t)$-condenser, then P is (T,δ')-secure in the (t,k)-real model without side information.*

Proof. Fix any (t,k)-bounded sampler $\mathcal{A}$ and any T-limited attacker $\mathcal{B}$. Recall, the random variable $(\mathsf{Cond}((X,Z);S), S, Z)$ is ε-close to a random variable (R,S,Z), where $\mathbf{H}_2(R|(S,Z)) \geq k'$. Thus, to prove the bound in Equation (2), it is enough to prove that $\delta' \leq \sqrt{\sigma\cdot 2^\beta}$ assuming that the value R in the real model satisfies $\mathbf{H}_2(R|(S,Z)) \geq k'$.

For brevity, let $y = (s,z)$, and q_y denote the probability that $Y = y$ (which is the same, for every $y = (s,z)$, in both the real and the ideal models). Also, for a given $r \in \{0,1\}^m$, let $p_{r|y}$ be the probability of outputting $R = r$ in the real model conditioned on $Y = y$. Recall the Cauchy-Schwartz inequality $|\sum a_i b_i| \leq \sqrt{(\sum a_i^2)\cdot(\sum b_i^2)}$. We have

$$
\begin{aligned}
\widetilde{\mathbf{\Delta}}(\mathcal{A},\mathcal{B}) &= \left|\sum_{y,r} q_y \cdot p_{r|y} \cdot \mathbf{Adv}_{\mathcal{B}}(r,y)\right| = \left| \sum_{y,r} \left(\sqrt{\frac{q_y}{2^m}} \cdot \mathbf{Adv}_{B}(r,y)\right) \cdot \left(\sqrt{2^m \cdot q_y} \cdot p_{r|y}\right) \right| \\
&\le \sqrt{\left(\sum_{y,r} \frac{q_y}{2^m} \cdot \mathbf{Adv}_{\mathcal{B}}(r,y)^2\right) \cdot \left(2^m \sum_{y,r} q_y \cdot p_{r|y}^2\right)} \\
&= \sqrt{\mathbf{\Delta}_2(\mathcal{A},\mathcal{B}) \cdot 2^m \cdot \mathbf{Col}(R|Y)} \le \sqrt{\sigma \cdot 2^m \cdot 2^{-k'}} \\
&\le \sqrt{\sigma \cdot 2^{\beta}}
\end{aligned}
$$

□

Using Corollary 4.1, we obtain a nearly optimal security degradation in the real model with no side information:

Corollary 5.2 *Assuming the existence of* $(t, \frac{O(t^2)}{2^m})$*-collision-resistant hash functions, if* P *is* (T,σ)*-square-secure in the* $(t, m - 2\log t + O(1))$*-ideal model with no side information, then using a collision-resistant function as a condenser makes* P *to be* (T,δ')*-secure in the* $(t, m - 2\log t + O(1))$*-real model with no side information, where*

$$\delta' \le O(t \cdot \sqrt{\sigma}) \; . \tag{5}$$

Discussion. We start by making a few observations about square-security σ (used in Equation (4) and Equation (5)) and its relation to the regular security δ. These observations were all implicitly made by [3], but here we make them explicit. First, we always have $\sigma \ge \delta^2$, which means that the term $\sqrt{\sigma}$ in Equation (4) is greater than or equal to the term δ is in Equation (2). Thus, we are indeed slightly worse off in this aspect. On the other hand, in case we do have $\sigma \approx \delta^2$, the bound in Equation (4) becomes much better than the one in Equation (2) in several respects: not only is it more general and applies to indistinguishability applications, but it has much better dependence on the entropy deficit ($\sqrt{2^\beta}$ vs 2^β) and uses a better "output entropy norm" ($\mathbf{H}_2$ rather than $\mathbf{H}_\infty$, which is better by Lemma 3.1).

In fact, while we cannot generically prove that $\sigma \approx \delta^2$, for unpredictability applications it is easy to see that $\sigma \le \delta$, since $\mathbf{Adv}_{\mathcal{B}}(\cdot)^2 \le \mathbf{Adv}_{\mathcal{B}}(\cdot)$. More surprisingly, [3] showed that for prominent indistinguishability applications, including chosen plaintext secure encryption and weak pseudorandom functions, one can also show that $(2T,\delta)$-security in the ideal model implies (T,δ)-*square*-security. Thus, ignoring a factor of 2 in the resources on $\mathcal{B}$, we effectively again have $\sigma \le \delta$ for these *indistinguishability* applications. To summarize, for a very large class of applications we still get relatively good bounds $\delta' \le \varepsilon + \sqrt{\delta \cdot 2^\beta}$ in Equation (4) and $\delta' = O(t \cdot \sqrt{\delta})$ in Equation (5) (the latter assuming a very strong collision-resistant hash function). Notice, for unpredictability applications the later bound matches the bound in Equation (3). However, now the same bound also holds for a large class of indistinguishability applications, such as chosen plaintext encryption.

6 Side-Information and Fiat-Shamir

One of the earliest and most influential applications of the Random Oracle Model in cryptography (predating its formalization by Bellare and Rogaway [5]) was to analyze the Fiat-Shamir Heuristic [18]. In the Fiat–Shamir Heuristic, a hash function is used to eliminate interaction in constant-round public-coin protocols, replacing the verifier's random challenges with hashes of the transcript so far. If the hash function is modeled as a random oracle, then this heuristic is known to preserve soundness of the underlying protocol (up to a factor polynomial in the number of queries made by the adversary to the random oracle). However, there are no natural examples of protocols for which the Fiat–Shamir

Heuristic has been proven sound when the hash function is implemented by an efficiently computable family of functions.

The original motivation for the Fiat–Shamir Heuristic was as a method to convert identification schemes into digital signature schemes, and the method gave rise to many efficient digital signature schemes in practice [18, 34, 22] (albeit with only a proof in the Random Oracle Model). Another compelling motivation for understanding the soundness of the Fiat–Shamir Heuristic is its close connection to the zero-knowledge property of the underlying protocols, as pointed out by Dwork, Naor, Reingold, and Stockmeyer [17]. Dwork et al. showed that the soundness of the Fiat–Shamir Heuristic on a given protocol is essentially equivalent to that protocol *not* being (auxiliary-input) zero knowledge unless the underlying language is in BPP.[4] There are many constant-round public-coin protocols whose zero knowledge status is a long-standing open problem (e.g. ones obtained by starting some underlying basic zero-knowledge protocol and applying parallel repetition to make the soundness error negligible). While these protocols cannot be *black-box* zero knowledge (for nontrivial languages) [19], they may still be non-black-box (auxiliary-input) zero knowledge.

Indeed, Barak [2] constructed a constant-round, public-coin (non-black-box) zero-knowledge argument system for NP (assuming the existence of collision-resistant hash functions), thereby yielding a natural protocol on which the Fiat–Shamir heuristic is unsound (for any efficiently computable family of hash functions). Goldwasser and Kalai [20] extended Barak's techniques to construct 3-message public-coin identification schemes on which the Fiat–Shamir Heuristic is unsound. In both of these counterexamples to the Fiat–Shamir Heuristic, the initial interactive protocol is only *computationally* sound, and the results seem to use this in an essential way.

Thus, Barak, Lindell, and Vadhan [4] conjectured that there *is* a sound implementation of the Fiat–Shamir Heuristic for any *statistically sound* interactive proof of language membership (and thus that there can be no constant-round public-coin zero-knowledge *proof* system with negligible soundness for a language outside BPP). Indeed, they provided a plausible property for a family of hash functions that suffices for it to provide a sound implementation of Fiat–Shamir on proof systems. While they conjectured that such hash families exist, it remains open to construct one based on a standard complexity assumption.

The significance of statistical soundness for reducing interaction was further highlighted by the recent work of Kalai and Raz [25], who showed that a method proposed by Aiello et al. [1] (based on Private Information Retrieval) can be used to convert (statistically sound) interactive proofs into 2-message argument systems. However, this construction does not subsume Fiat–Shamir, because the 2-message argument system it produces is private coin (so the verifier's first message cannot be published as a CRS and shared by all verifiers, as needed for the application to digital signatures) and it does not have the connection to zero knowledge mentioned above.

Here we show that condensers for seed-dependent samplable sources that can handle side information (i.e. *average-case* condensers) imply hash functions for which the Fiat–Shamir Heuristic is sound for proof systems. In fact, we only require condensers for the case that the initial source X is uniform and the adversary's side-information Z consists of a bounded-length "leakage" $f(X, S)$ on the source and seed, for an efficiently computable leakage function f. We also show a partial converse — some form of such condensers are also *necessary* for the Fiat–Shamir heuristic to be sound for all proof systems.

Our results are inspired by a similarity between the definition of condensers for samplable sources and the aforementioned conjectures of Barak et al. [4]. While the existence of such condensers and hash functions remains an open problem, the connection between randomness condensing and the Fiat–Shamir Heuristic, along with our construction of condensers without side information (Theorem 4.1),

[4]The forward direction is shown as follows: if there is an efficiently computable family of hash functions for which the Fiat–Shamir heuristic is sound, then it is infeasible to simulate a verifier that has a random hash function from the family as auxiliary input, and obtains its challenges by applying the hash function to the transcript so far. Indeed, an efficient simulator would constitute a prover strategy that generates accepting proofs for the Fiat-Shamir-collapsed protocol, which would only be possible for inputs in the language.

seem to yield a clearer picture of what is needed for the Fiat–Shamir Heuristic to work. (In particular, we find the definition of a seed-dependent average-case condenser more natural than the conjectures in [4].)

6.1 Condensers for Leaky Sources

We begin by defining the restricted form of average-case condensers that we relate to the Fiat–Shamir heuristic:

Definition 6.1 (Condensers for Leaky Sources) *Let $c' \in \{1, 2, \infty\}$. An efficient function* $\mathsf{Cond} : \{0,1\}^n \times \{0,1\}^\alpha \to \{0,1\}^m$ *is an* $(\varepsilon, [\mathbf{H}_{c'} \geq k'], t)$-condenser for leaky sources *if for all probabilistic adversaries $\mathcal{A}$ of size at most t who take a random source $X \leftarrow \{0,1\}^n$ and output a string $Z := \mathcal{A}(X)$ of length α, the joint distribution $(Z, \mathsf{Cond}(X, Z))$ is ε-close to (Z, R), where $\mathbf{H}_{c'}(R|Z) \geq k'$.*

When $\varepsilon = 0$, we will refer to Cond *as an* $([\mathbf{H}_{c'} \geq k'], t)$-condenser for leaky sources. *The quantity $\beta \stackrel{def}{=} m - k'$ is called the* entropy deficit *of the condenser.*

Thus, instead of allowing an arbitrary efficiently samplable source X that has high entropy given the adversary's side information Z, we restrict to $X \leftarrow \{0,1\}^n$ and Z of bounded length α. For natural measures of conditional entropy, this implies that $\mathbf{H}(X|Z) \geq n - \alpha$, so an average-case condenser for entropy $k = n - \alpha$ according to Definition 3.2 is also condenser for leaky sources according to Definition 6.1. Note that in the case of leaky sources, we do not provide the condenser with a seed; that is because any seed can be viewed as part of the uniformly random source X. Indeed, average-case condensers with seeds imply seedless condensers for leaky sources:

Proposition 6.1 *If* $\mathsf{Cond} : \{0,1\}^n \times \{0,1\}^* \times \{0,1\}^d \to \{0,1\}^m$ *is an* average-case seed-dependent $([\mathbf{H}_c \geq k] \to_\varepsilon [\mathbf{H}_{c'} \geq k'], t)$-condenser *and we define* $\mathsf{Cond}' : \{0,1\}^{n+d} \times \{0,1\}^{n-k} \to \{0,1\}^m$ *by* $\mathsf{Cond}'((x, s), z) = \mathsf{Cond}(x, z, s)$*, then* Cond' *is an $(\varepsilon, [\mathbf{H}_{c'} \geq k'], t - O(n))$-condenser for leaky sources.*

Proof. Given an adversary $\mathcal{A}'$ for Cond' running, define an adversary $\mathcal{A}$ for Cond as follows: on input seed S, $\mathcal{A}$ generates a uniformly random $X \leftarrow \{0,1\}^n$, sets $Z = \mathcal{A}'(X, S)$, and outputs (x, z). If $\mathcal{A}'$ runs in time $t' = t - O(n)$, then $\mathcal{A}$ runs in time at most t. Starting with $X \leftarrow \{0,1\}^n$, we have

$$\mathbf{H}_c(X|(S, Z)) \geq \mathbf{H}_c(X|S) - \alpha = n - (n - k) = k \ ,$$

where the first inequality follows from the fact that conditioning on a string of length α (namely Z) reduces entropy by at most α (for all of our measures $\mathbf{H}_\infty$, $\mathbf{H}_1$, and $\mathbf{H}_2$).

Thus, by the condenser property of Cond, $(Z, S, \mathsf{Cond}((X, Z); S)) = (Z, S, \mathsf{Cond}'((X, S), Z))$ is ε-close to (Z, S, R), where

$$\mathbf{H}_{c'}(R|Z) \geq \mathbf{H}_{c'}(R|(S, Z)) \geq k' \ ,$$

where the first inequality holds because conditioning can only reduce entropy. (Again, this holds for $\mathbf{H}_\infty$, $\mathbf{H}_1$, and $\mathbf{H}_2$.) □

We can also observe the following analog of Lemma 3.1 for condensers for leaky sources, whose proof is immediate from the relations between different entropy measures.

Lemma 6.1

- **"Output** $(\infty \to 2 \to 1)$**":** *If $c' \geq c''$ and* Cond *is an $(\varepsilon, [\mathbf{H}_{c'} \geq k'], t)$-condenser for leaky sources, then* Cond *is also an $(\varepsilon, [\mathbf{H}_{c''} \geq k'], t)$-condenser for leaky sources.*
- **"Output** $(2 \to \infty)$**":** *If* Cond *is an $(\varepsilon, [\mathbf{H}_2 \geq k'], t)$-condenser for leaky sources, then*
  (a) for any $\gamma > 0$, Cond *is also an $(\varepsilon + \gamma, [\mathbf{H}_\infty \geq k' - \log(1/\gamma)], t)$-condenser for leaky sources;*
  (b) Cond *is also an $(\varepsilon, [\mathbf{H}_\infty \geq k'/2], t)$-condenser for leaky sources.*

6.2 Condensers and Fiat-Shamir Heuristics

Now we define the Fiat–Shamir heuristic more precisely. Let (P, V) be a public-coin interactive protocol, where the parties receive no inputs (except a security parameter κ), there are $2r+1$ messages exchanged starting with P. We denote the lengths of P's messages by $\ell = \ell(\kappa)$ and the lengths of V's messages by $m = m(\kappa)$.

Definition 6.2 *For a language $L = L(\kappa) \subseteq \{0,1\}^\ell$, we say that (P, V) is a (t, ε)-*sound interactive argument *for L iff there is no prover strategy P^* of circuit size at most t that convinces V to accept on a transcript whose first message is not in L with probability greater than ε.*

*We say that (P, V) is an ε-*sound interactive proof *for L iff it is an (∞, ε) interactive argument for L (i.e. it holds for computationally unbounded prover strategies P^*).*

Ordinarily, interactive proofs are formulated with the input x (whose membership in L is being determined) being provided separately as a common input to P and V. However, incorporating x into the first message of the protocol is notationally more convenient for us.

Fiat and Shamir [18] suggested a way to remove the interaction from protocols as above, by replacing the verifier's messages with hashes of the transcript:

Definition 6.3 *For an interactive protocol (P, V) as above, $\alpha = r \cdot \ell + (r-1) \cdot m$, and a family of hash functions $\mathcal{H} = \mathcal{H}(\kappa) = \{h : \{0,1\}^\alpha \to \{0,1\}^m\}$, the* Fiat-Shamir collapse of (P, V) using $\mathcal{H}$ *is the 2-message public-coin protocol (P', V') defined as follows:*

(1) *V' sends P' a random hash function $H \leftarrow \mathcal{H}$,*

(2) *P' sends V' a tuple $(M_1, M_2, \ldots, M_{r+1}) \in (\{0,1\}^\ell)^{r+1}$,*

(3) *V' accepts iff V accepts on the transcript $(M_1, R_1, M_2, R_2, \ldots, M_r, R_r, M_{r+1})$ where $R_i \stackrel{\mathsf{def}}{=} H(M_1, R_1, \ldots, M_{i-1}, R$ for each $i \in [r]$.*

We say that the Fiat-Shamir heuristic using $\mathcal{H}$ is (t, ε')-sound on (P, V) *iff (P', V') is a (t, ε')-sound interactive argument for the language $L' = \{(M_1, \ldots, M_{r+1}) : M_1 \in L\}$.*

Now we prove that we can use condensers for leaky sources to construct hash functions for which the Fiat–Shamir heuristic is secure:

Theorem 6.1 *Let (P, V) be an interactive protocol as above, and let $\alpha = r \cdot \ell + (r-1) \cdot m$. Given $\mathsf{Cond} : \{0,1\}^n \times \{0,1\}^\alpha \to \{0,1\}^m$, define $\mathcal{H} = \{h_x : \{0,1\}^\alpha \to \{0,1\}^m\}_{x \in \{0,1\}^n}$ by $h_x(z) = \mathsf{Cond}(x, z)$.*

Then if (P, V) is an ε_1-sound interactive proof for some language L and Cond is an $(\varepsilon_2, [\mathbf{H}_\infty \geq m - \beta], t)$-condenser for leaky sources, then the Fiat-Shamir heuristic is (t', ε')-sound on (P, V), for $t' = t - (r-1) \cdot t_{\mathsf{Cond}} - O(n)$ and

$$\varepsilon' = 2^{r\beta} \cdot \varepsilon_1 + \frac{2^{r\beta} - 1}{2^\beta - 1} \cdot \varepsilon_2 \leq 2^{r\beta} \cdot (\varepsilon_1 + \varepsilon_2). \tag{6}$$

For intuition about the parameters, consider the standard, polynomial-time asymptotic setting. Here all length parameters of the proof system (ℓ, m) are some fixed polynomial in the security parameter κ, and we are interested in protocols whose soundness error ε_1 is negligible, i.e. $\varepsilon_1 = \kappa^{-\omega(1)}$. We focus on constant-round proof systems, so $r = O(1)$. We take the length $n = \mathrm{poly}(\kappa)$ of the condenser source to be significantly larger than $m + \alpha = r \cdot (\ell + m)$. This means that the condenser should work for sources with entropy at least $k = n - \alpha$, which is significantly larger than m. By analogy with Theorem 4.1, we can hope for the output to have min-entropy deficiency $\beta = O(\log t)$, which is $O(\log \kappa)$ for any polynomial $t = t(\kappa)$, possibly with some negligible statistical difference $\varepsilon_2 = \kappa^{-\omega(1)}$. Thus the new soundness error satisfies

$$\varepsilon' \leq 2^{r\beta} \cdot (\varepsilon_1 + \varepsilon_2) = 2^{O(\log \kappa)} \cdot (\kappa^{-\omega(1)} + \kappa^{-\omega(1)}) = \kappa^{-\omega(1)},$$

which is still negligible.

For intuition about the proof, consider a cheating prover strategy, that given the description X of a random hash function from the family, tries to construct a transcript $(M_1, R_1, \ldots, M_r, R_r, M_{r+1})$ such that $M_1 \notin L$, the original verifier accepts, and each R_i is the hash of the prefix preceding it, i.e.

$$R_i = h_X(M_1, R_1, \ldots, M_i) = \mathsf{Cond}(X, (M_1, R_1, \ldots, M_i)).$$

Viewing $Z_i = (M_1, R_1, \ldots, M_i)$ as the adversary's side information (which is of length at most $r \cdot \ell + (r-1) \cdot m$), the condenser property says that R_i is ε_2-close to having min-entropy deficiency at most β given the prefix $M_1, R_1, \ldots, M_i$. Compared to R_i being uniform and independent of the prefix, this should increase the soundness error by an additive ε_2 and a multiplicative 2^β. Incurring this blow up for each of the rounds i yields the bound in the theorem. We now proceed with the formal proof.

Proof. Consider any size t' prover strategy P^* that takes a description X of a hash function from $\mathcal{H}$ and outputs a sequence of messages $(M_1, \ldots, M_{r+1}) \in \{0,1\}^{\ell \cdot r}$. For $i = 0, \ldots, r$, let p_i be the maximum acceptance probability over all unbounded prover strategies P^{**} in the following hybrid protocol:

(1) Choose $X \leftarrow \{0,1\}^n$ uniformly at random.
(2) Let $(M_1, \ldots, M_i)$ be the length i prefix of $P^*(X)$.
(3) Inductively define $R_1, \ldots, R_i$ by $R_j = h_X(M_1, R_1, \ldots, M_{j-1}, R_{j-1}, M_j)$.
(4) For $j = i+1$ to $r+1$, set $M_j = P^{**}(M_1, R_1, \ldots, M_{j-1}, R_{j-1})$ and choose $R_j \leftarrow \{0,1\}^m$ uniformly at random.
(5) Accept iff $V(M_1, R_1, \ldots, M_r, R_r, M_{r+1})$ accepts.

By soundness of (P, V), we have $p_0 \leq \varepsilon_1$. We will show below that for each $i = 1, \ldots, r$, $p_i \leq 2^\beta \cdot p_{i-1} + \varepsilon_2$. This implies that

$$p_r \leq 2^{r\beta} \cdot \varepsilon_1 + \frac{2^{r\beta} - 1}{2^\beta - 1} \cdot \varepsilon_2.$$

The proof is completed by noting that the acceptance probability of P^* in the Fiat-Shamir collapse of (P, V) is at most p_r (since the unbounded P^{**} can only do better than P^* in selecting the last message).

We now proceed to show $p_i \leq 2^\beta \cdot p_{i-1} + \varepsilon_2$. Let $A_i(x)$ be the size $t' + r \cdot t_{\mathsf{Cond}} \leq t$ circuit that given $X \in \{0,1\}^n$, computes the length i prefix $(M_1, \ldots, M_i)$ of $P^*(X)$, inductively computes $R_1, \ldots, R_{i-1}$ by $R_j \leftarrow h_X(M_1, R_1, \ldots, M_{j-1}, R_{j-1}, M_j)$, and outputs $Z_i \leftarrow (M_1, R_1, \ldots, M_{i-1}, R_{i-1}, M_i)$.

Note Z_i is of length at most $r \cdot \ell + (r-1) \cdot m$. By the condenser property we have that $(Z_i, R_i) = (Z_i, h_X(Z_i)) = (Z_i, \mathsf{Cond}(X, Z_i))$ is ε_2-close to (Z_i, R_i'), where $\mathcal{H}_\infty(R_i'|Z_i) \geq m - \beta$.

Note that $(Z_i, R_i) = (M_1, R_1, \ldots, M_i, R_i)$ are generated exactly as in the i'th hybrid protocol. Let $p(Z_i, R_i)$ denote the maximum acceptance probability given this partial transcript. Then we have

$$\begin{aligned} p_i &= \mathbb{E}[p(Z_i, R_i)] \\ &\leq \mathbb{E}[p(Z_i, R_i')] + \varepsilon_2 \\ &\leq 2^\beta \cdot \mathbb{E}[p(Z_i, U_m)] + \varepsilon_2 \\ &\leq 2^\beta \cdot p_{i-1} + \varepsilon_2 \ , \end{aligned}$$

where the second-to-last inequality uses the fact that $\mathbf{H}_\infty(R_i'|Z_i) \geq m - \beta$, and the last inequality comes from noting that the only difference in the experiments defining $p(Z_i, U_m)$ and p_{i-1} are that in the latter P^{**} chooses M_i optimally (whereas in the former it is chosen by P^*). ☐

Many interactive proofs of interest have only three messages (i.e. $r = 1$ above) and have optimal soundness $\varepsilon_1 = 1/2^m$, meaning that for every initial prover message not in L, there is at most 1 verifier challenge that can lead to an accepting transcript. Examples include parallel repetitions of Blum's Hamiltonicity protocol [6], the Goldwasser-Micali-Rackoff Quadratic Residuosity Protocol (to which Fiat-Shamir was originally applied) [21], and any Σ protocol [12]. Setting $r = 1$ and $\varepsilon_1 = 1/2^m$, we see that the resulting soundness error is $\varepsilon' = 2^\beta/2^m + \varepsilon_2$, which is small even for entropy deficiency β that is quite close to m, i.e. the output entropy of the condenser need only be $k' = m - \beta = \log(1/\varepsilon_3)$ to achieve soundness error $\varepsilon_2 + \varepsilon_3$:

Corollary 6.1 *Let* Cond*,* $\mathcal{H}$*, and* (P, V) *be as in Theorem 6.1. Suppose further that* (P, V) *has 3 messages (i.e.* $r = 1$*), and has soundness* $\varepsilon_1 = 1/2^m$*, where* m *is the length of the verifier's challenge.*

Then if Cond *is a* $(\varepsilon_2, [\mathbf{H}_\infty \geq \log(1/\varepsilon_3)], t)$*-condenser for leaky sources computable in time* t_{Cond}*, then the Fiat-Shamir heuristic is* (t', ε')*-sound on* (P, V)*, for* $t' = t - O(n)$ *and* $\varepsilon' = \varepsilon_2 + \varepsilon_3$*.*

If, instead, Cond *is an* $(\varepsilon_2, [\mathbf{H}_2 \geq \log(1/\varepsilon_3)], t)$*-condenser for leaky sources computable in time* t_{Cond}*, it follows from Lemma 6.1 that the Fiat-Shamir heuristic is* (t', ε')*-sound on* (P, V)*, for* $t' = t - O(n)$ *and* $\varepsilon' = \varepsilon_2 + \sqrt{\varepsilon_3}$*.*

Theorem 6.1 and Corollary 6.1 are stated using average min-entropy (or collision entropy) as the entropy measure for the output of the condenser. We now discuss what happens for the more challenging case of Shannon entropy.

6.3 Condensers for Shannon Entropy

If the condenser output is only guaranteed to have high Shannon entropy, we can only deduce that the Fiat–Shamir Heuristic has soundness error bounded by a constant. This is still quite nontrivial, and indeed the soundness error can be made negligible without adding interaction by repeating the heuristic with several independent hash functions. This case (obtaining constant error using condensers for Shannon entropy) actually follows from the results in [4] and the connection between condensers for leaky sources and the conjectures in [4].

Let $\mathsf{Cond} : \{0,1\}^n \times \{0,1\}^\alpha \to \{0,1\}^m$ and $\mathcal{H} = \{h_x : \{0,1\}^\alpha \to \{0,1\}^m\}_{x \in \{0,1\}^n}$ correspond to each other as in Theorem 6.1, i.e. $h_X(Z) = \mathsf{Cond}(X, Z)$. If Cond is an $(\varepsilon, [\mathbf{H}_1 \geq k'], t)$-condenser for leaky sources, then for every size t adversary $\mathcal{A} : \{0,1\}^n \to \{0,1\}^\alpha$, and $X \leftarrow \{0,1\}^n$, we have

$$\mathbf{H}_1(h_X(A(X))|A(X)) = \mathbf{H}_1(\mathsf{Cond}(X, A(X))|A(X)) \geq k' ,$$

where in the latter inequality we use the condenser property. Barak et al. [4] referred to hash families $\mathcal{H} = \{h_x\}$ with this property as "ensuring conditional entropy k'," and conjectured that such hash families exist with $k' = m - O(\log \kappa)$ (where κ is the security parameter). By the above, hash families that ensure conditional entropy k' are equivalent to condensers for leaky sources whose output has conditional Shannon entropy at least k' (but we view the condenser formulation as more natural).

Barak et al. proved that such hash families with entropy deficit $\beta = O(\log \kappa)$ imply that the Fiat–Shamir Heuristic has constant soundness when applied to constant-round public-coin interactive proof systems with negligible soundness (and consequently no such interactive proof system can be zero knowledge, unless the underlying language is in BPP).[5]

For 3-message protocols with optimal soundness, as in Corollary 6.1, they showed that we only need output entropy $1/\mathrm{poly}(\kappa)$ to deduce that the Fiat–Shamir Heuristic has soundness error at most $1 - 1/\mathrm{poly}(\kappa)$, which again is enough to deduce that no 3-message public-coin proof system with optimal soundness can be zero knowledge (except for languages in BPP). This result can be deduced from Corollary 6.1, so we restate and prove it here:

Corollary 6.2 ([4]) *Let* Cond*,* $\mathcal{H}$*, and* (P, V) *be as in Corollary 6.1. If* Cond *is a* $([\mathbf{H}_1 \geq \mathbf{H}(\delta) + \delta \cdot m], t)$*-condenser for leaky sources, it follows that the Fiat-Shamir Heuristic is* $(t', 1 - \delta)$*-sound on* (P, V)*, for* $t' = t - O(n)$*.*

[5] Actually, for $r > 1$, their result uses an independently generated hash function for each round of the protocol. Since the Fiat–Shamir Heuristic was originally formulated in the $r = 1$ case, either extension to $r > 1$ is reasonable.

In particular, if $\ell, m, n, 1/\delta \le \mathrm{poly}(\kappa)$, Cond and V are computable in time $\mathrm{poly}(\kappa)$, $t = \kappa^{\omega(1)}$, and (P, V) has completeness error $\kappa^{-\omega(1)}$, then (P, V) cannot be zero knowledge unless $L = L(\kappa)$ is in BPP *(decidable by probabilistic algorithms running in time* $\mathrm{poly}(\kappa)$*).*

Proof. By Corollary 6.1, it suffices to show that Cond is an $(\varepsilon_2, [\mathbf{H}_\infty \ge \log(1/\varepsilon_3)], t)$-condenser for leaky sources with $\varepsilon_2 = 0$ and and $\varepsilon_3 = 1 - \delta$. This follows from the fact that for any two random variables (Y, Z) such that Y takes values in $\{0,1\}^m$ and $\mathbf{H}_1(Y|Z) \ge \mathbf{H}(\delta) + \delta \cdot m$, we also have $\mathbf{H}_\infty(Y|Z) \ge \log(1/(1-\delta))$. Indeed, suppose that $\mathbf{H}_\infty(Y|Z) < \log(1/(1-\delta))$. This means that there is a function g such that $\Pr[g(Z) = Y] > 1 - \delta$. Then Fano's Inequality (see [11]) implies that $\mathbf{H}_1(Y|Z) < \mathbf{H}(\delta) + \delta \cdot m$.

The "in particular" follows from the known connections between the soundness of the Fiat–Shamir Heuristic and impossibility of zero knowledge. See [17, 4]. □

Finally, the weakest conjecture of Barak et al. [4] only requires the hash functions to ensure nonzero entropy. Restated in the language of condensers, it says:

Conjecture 6.1 ([4]) *Let κ be a security parameter. For every $\alpha, m \le \mathrm{poly}(\kappa)$ such that $m = \omega(\log \kappa)$, there is an $n = \mathrm{poly}(\kappa)$ and a $\mathrm{poly}(\kappa)$-time computable $\mathsf{Cond} : \{0,1\}^n \times \{0,1\}^\alpha \to \{0,1\}^m$ such that Cond is a $([\mathbf{H}_1 > 0], k^{\omega(1)})$-condenser for leaky sources.*

Theorem 6.2 ([4]) *Assuming Conjecture 6.1, only languages in* coAM *can have 3-round public-coin zero-knowledge proofs with optimal soundness and perfect completeness.*

In particular, under Conjecture 6.1, the $\omega(\log k)$-fold parallel repetition of Blum's Hamiltonicity Protocol for NP [6] cannot be zero knowledge unless NP $\subseteq$ coAM and the Polynomial-time Hierarchy collapses. An interesting feature of Conjecture 6.1 is that it has an equivalent reformulation that refers only to *worst-case* complexity. (See [4] for details.)

6.4 Condensers for Leaky Sources are Necessary

Now we prove a partial converse to Corollary 6.2 and Theorem 6.2, and show that condensers for leaky sources are also *necessary* for the soundness of the Fiat–Shamir Heuristic.

Theorem 6.3 *Let $\mathsf{Cond} : \{0,1\}^n \times \{0,1\}^\alpha \to \{0,1\}^m$ and $\mathcal{H} = \{h_x : \{0,1\}^\alpha \to \{0,1\}^m\}_{x \in \{0,1\}^n}$ correspond to each other as in Theorem 6.1, i.e. $h_x(z) = \mathsf{Cond}(x, z)$.*

If Cond is not an $([\mathbf{H}_1 > 0], t)$-condenser for leaky sources, then there is a 3-round public-coin interactive proof (P, V) with optimal soundness where the first prover message is of length $\ell = \alpha$, the verifier challenge is of length m, the verifier's acceptance predicate is computable by a circuit of size t, and the Fiat–Shamir Heuristic is not (t, ε')-sound on (P, V) for any $\varepsilon' < 1$.

Proof. Let $\mathcal{A}$ be a size t adversary such that for $X \leftarrow \{0,1\}^n$, we have $\mathbf{H}_1(\mathsf{Cond}(X, \mathcal{A}(X))|\mathcal{A}(X)) = 0$. That is, $\mathsf{Cond}(X, \mathcal{A}(X))$ is uniquely determined by $\mathcal{A}(X)$. Fixing the coins of $\mathcal{A}$ preserves this property, so we may assume that $\mathcal{A}$ is deterministic without loss of generality.

We construct the protocol (P, V) as follows:

(1) P sends a string $Z \in \{0,1\}^\alpha$ to V.

(2) V sends a uniformly random $R \leftarrow \{0,1\}^m$ to P.

(3) P sends a string $X \in \{0,1\}^n$ to V.

(4) V accepts iff $Z = \mathcal{A}(X)$ and $R = \mathsf{Cond}(X, Z)$.

This protocol has optimal soundness (for the empty language) because $\mathcal{A}(X)$ uniquely determines $\mathsf{Cond}(X, \mathcal{A}(X))$, so for every Z, there is a unique R such that $\mathcal{A}(X) = Z$ and $\mathsf{Cond}(X, \mathcal{A}(X)) = R$ for some X. However, the Fiat–Shamir collapse (P', V') of (P, V) using $\mathcal{H}$ can be broken by the following strategy P^*: given a description X of a hash function h_X from $\mathcal{H}$, P^* sets $Z = \mathcal{A}(X)$, and sends (Z, X) to V'. V' will compute the verifier's challenge as $R = h_X(z) = \mathsf{Cond}(X, Z)$ and thus will always accept. $\square$

Theorem 6.3, together with Theorem 6.1 and the results of [4], shows that the existence of condensers for leaky sources is intimately related to the soundness of the Fiat–Shamir Heuristic for proof systems, and thus it is of great interest to relate such condensers to more well-understood complexity assumptions. At the same time, we note that the proof of Theorem 6.3 gives a protocol (P, V) where the verifier's acceptance predicate and the length of the third message both depend on the hash family $\mathcal{H}$ used to implement Fiat–Shamir. Thus, even if condensers for leaky sources do not exist, it still leaves open the possibility that for every protocol (P, V), there is a family of hash functions $\mathcal{H}$ (of complexity and/or description length larger than (P, V)) such that the Fiat–Shamir Heuristic using $\mathcal{H}$ is sound on (P, V).

References

[1] Aiello, W., Bhatt, S.N., Ostrovsky, R., Rajagopalan, S.: Fast verification of any remote procedure call: Short witness-indistinguishable one-round proofs for np. In: Montanari, U., Rolim, J.D.P., Welzl, E. (eds.) Automata Languages and Programming: 27th International Colloquim (ICALP 2000). LNCS, vol. 1853, pp. 463–474. Springer-Verlag (Jul 9–15 2000)

[2] Barak, B.: How to go beyond the black-box simulation barrier. In: 42nd Annual Symposium on Foundations of Computer Science. pp. 106–115. IEEE, Las Vegas, Nevada (Oct 2001), preliminary full version available on `http://www.wisdom.weizmann.ac.il/~boaz`

[3] Barak, B., Dodis, Y., Krawczyk, H., Pereira, O., Pietrzak, K., Standaert, F.X., Yu, Y.: Leftover hash lemma, revisited. In: CRYPTO11. pp. 1–20 (2011)

[4] Barak, B., Lindell, Y., Vadhan, S.: Lower bounds for non-black-box zero knowledge. Journal of Computer and System Sciences 72(2), 321–391 (March 2006), special Issue on FOCS '03

[5] Bellare, M., Rogaway, P.: Random oracles are practical: A paradigm for designing efficient protocols. In: Denning, D., Pyle, R., Ganesan, R., Sandhu, R., Ashby, V. (eds.) First ACM Conference on Computer and Communication Security. pp. 62–73. ACM (Nov 3–5 1993)

[6] Blum, M.: Coin flipping by telephone. In: Proc. 1982 IEEE COMPCON, High Technology in the Information Age. pp. 133–137 (1982)

[7] Canetti, R., Dodis, Y., Halevi, S., Kushilevitz, E., Sahai, A.: Exposure-resilient functions and all-or-nothing transforms. In: Preneel, B. (ed.) Advances in Cryptology—EUROCRYPT 2000. LNCS, vol. 1807, pp. 453–469. Springer-Verlag (2000)

[8] Capalbo, M., Reingold, O., Vadhan, S., Wigderson, A.: Randomness conductors and constant-degree lossless expanders. In: 34th Annual ACM Symposium on Theory of Computing (STOC '02). pp. 659–668. ACM, Montréal, CA (May 2002), joint session with *CCC '02*.

[9] Chor, B., Goldreich, O.: Unbiased bits from sources of weak randomness and probabilistic communication complexity. SIAM Journal on Computing 17(2), 230–261 (1988)

[10] Chor, B., Goldreich, O., Håstad, J., Friedman, J., Rudich, S., Smolensky, R.: The bit extraction problem or t-resilient functions. In: Proceedings of the 26th IEEE Symposium on Foundation of Computer Science. pp. 396–407 (1985)

[11] Cover, T., Thomas, J.: Elements of information theory. Wiley Series in Telecommunications and Signal Processing, Wiley-Interscience (2006), http://books.google.com/books?id=EuhBluW31hsC

[12] Cramer, R., Damgård, I., Schoenmakers, B.: Proofs of partial knowledge and simplified design of witness hiding protocols. In: Advances in Cryptology – CRYPTO 1994. pp. 174–187 (1994)

[13] Dodis, Y., Elbaz, A., Oliveira, R., Raz, R.: Improved randomness extraction from two independent sources. In: APPROX-RANDOM. pp. 334–344 (2005)

[14] Dodis, Y., Ostrovsky, R., Reyzin, L., Smith, A.: Fuzzy extractors: How to generate strong keys from biometrics and other noisy data. SIAM Journal on Computing 38(1), 97–139 (2008)

[15] Dodis, Y., Ristenpart, T., Vadhan, S.: Randomness condensers for efficiently samplable, seed-dependent sources, full version of this paper. Available from authors' websites.

[16] Dodis, Y., Sahai, A., Smith, A.: On perfect and adaptive security in exposure-resilient cryptography. In: Pfitzmann, B. (ed.) Advances in Cryptology—EUROCRYPT 2001. LNCS, vol. 2045, pp. 301–324. Springer-Verlag (2001)

[17] Dwork, C., Naor, M., Reingold, O., Stockmeyer, L.: Magic functions. In: 40th Annual Symposium on Foundations of Computer Science. pp. 523–534. IEEE, New York (Oct 1999)

[18] Fiat, A., Shamir, A.: How to prove yourself: Practical solutions to identification and signature problems. In: Odlyzko, A.M. (ed.) Advances in Cryptology—CRYPTO '86. LNCS, vol. 263, pp. 186–194. Springer-Verlag, 1987 (11–15 Aug 1986)

[19] Goldreich, O., Krawczyk, H.: On the composition of zero-knowledge proof systems. SIAM Journal on Computing 25(1), 169–192 (Feb 1996)

[20] Goldwasser, S., Kalai, Y.T.: On the (in)security of the Fiat-Shamir paradigm. In: 44th Annual Symposium on Foundations of Computer Science. pp. 102–113. IEEE, Cambridge, Massachusetts (Oct 2003)

[21] Goldwasser, S., Micali, S., Rackoff, C.: The knowledge complexity of interactive proof systems. SIAM Journal on Computing 18, 186–208 (1989)

[22] Guillou, L.C., Quisquater, J.J.: A "paradoxical" indentity-based signature scheme resulting from zero-knowledge. In: Goldwasser, S. (ed.) Advances in Cryptology—CRYPTO '88. LNCS, vol. 403, pp. 216–231. Springer-Verlag, 1990 (21–25 Aug 1988)

[23] Guruswami, V., Umans, C., Vadhan, S.P.: Unbalanced expanders and randomness extractors from parvaresh–vardy codes. J. ACM 56(4) (2009)

[24] Gutterman, Z., Pinkas, B., Reinman, T.: Analysis of the linux random number generator. In: 27th IEEE Symposium on Security and Privacy. pp. 371–385. IEEE Computer Society (2006)

[25] Kalai, Y.T., Raz, R.: Probabilistically checkable arguments. In: Halevi, S. (ed.) Advances in Cryptology - CRYPTO 2009. LNCS, vol. 5677, pp. 143–159. Springer-Verlag (2009)

[26] Kamp, J., Rao, A., Vadhan, S.P., Zuckerman, D.: Deterministic extractors for small-space sources. In: Kleinberg, J.M. (ed.) STOC. pp. 691–700. ACM, Seattle, WA, USA (21–23 May 2006)

[27] Koenig, R., Maurer, U.: Extracting randomness from generalized symbol-fixing and Markov sources. In: Proceedings of 2004 IEEE International Symposium on Information Theory. p. 232 (Jun 2004)

[28] Koenig, R., Maurer, U.: Generalized strong extractors and deterministic privacy amplification. In: Smart, N. (ed.) Cryptography and Coding 2005. LNCS, vol. 3796, pp. 322–339. Springer-Verlag (Dec 2005)

[29] Lichtenstein, D., Linial, N., Saks, M.: Some extremal problems arising from discrete control processes. Combinatorica 9(3), 269–287 (1989)

[30] Nisan, N., Zuckerman, D.: Randomness is linear in space. Journal of Computer and System Sciences 52(1), 43–53 (1996)

[31] Raz, R., Reingold, O.: On recycling the randomness of states in space bounded computation. In: Annual ACM Symposium on Theory of Computing (Atlanta, GA, 1999), pp. 159–168 (electronic). ACM, New York (1999), `http://dx.doi.org/10.1145/301250.301294`

[32] Reingold, O., Shaltiel, R., Wigderson, A.: Extracting randomness via repeated condensing. SIAM Journal on Computing 35(5), 1185–1209 (electronic) (2006), `http://dx.doi.org/10.1137/S0097539703431032`

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

[34] Schnorr, C.P.: Efficient signature generation by smart cards. Journal of Cryptology 4(3), 161–174 (1991)

[35] Shamir, A., Tauman, Y.: Improved online/offline signature schemes. In: Kilian, J. (ed.) Advances in Cryptology—CRYPTO 2001. LNCS, vol. 2139, pp. 355–367. Springer-Verlag (2001)

[36] Ta-Shma, A., Umans, C., Zuckerman, D.: Lossless condensers, unbalanced expanders, and extractors. Combinatorica 27(2), 213–240 (2007), `http://dx.doi.org/10.1007/s00493-007-0053-2`

[37] Trevisan, L., Vadhan, S.: Extracting randomness from samplable distributions. In: 41st Annual Symposium on Foundations of Computer Science. pp. 32–42. IEEE, Redondo Beach, California (Nov 2000)

[38] Viola, E.: Extractors for circuit sources. In: IEEE Symposium on Foundations of Computer Science (FOCS) (2011)