Bounded Indistinguishability and the Complexity of Recovering Secrets*

Andrej Bogdanov† Yuval Ishai‡ Emanuele Viola§ Christopher Williamson¶

June 6, 2016

Abstract

Motivated by cryptographic applications, we study the notion of *bounded indistinguishability*, a natural relaxation of the well studied notion of bounded independence.

We say that two distributions μ and ν over Σ^n are *k-wise indistinguishable* if their projections to any k symbols are identical. We say that a function $f\colon \Sigma^n \to \{0,1\}$ is *ϵ-fooled by k-wise indistinguishability* if f cannot distinguish with advantage ϵ between any two k-wise indistinguishable distributions μ and ν over Σ^n.

We are interested in characterizing the class of functions that are fooled by k-wise indistinguishability. While the case of k-wise independence (corresponding to one of the distributions being uniform) is fairly well understood, the more general case remained unexplored.

When $\Sigma = \{0,1\}$, we observe that whether f is fooled is closely related to its approximate degree. For larger alphabets Σ, we obtain several positive and negative results. Our results imply the first efficient secret sharing schemes with a high secrecy threshold in which the secret can be reconstructed in AC^0. More concretely, we show that for every $0 < \sigma < \rho \leq 1$ it is possible to share a secret among n parties so that any set of fewer than σn parties can learn nothing about the secret, any set of at least ρn parties can reconstruct the secret, and where both the sharing and the reconstruction are done by constant-depth circuits of size $\mathrm{poly}(n)$. We present additional cryptographic applications of our results to low-complexity secret sharing, visual secret sharing, leakage-resilient cryptography, and eliminating "selective failure" attacks.

1 Introduction

For a finite alphabet Σ, a distribution μ over Σ^n is *k-wise independent* if its projection to every k coordinates is uniform. There is a large body of work studying bounded independence, namely, the conditions under which a given function $f : \Sigma^n \to \{0,1\}$ cannot distinguish between any distribution on n bits that is k-wise independent and the uniform distribution with advantage ϵ, for various choices of ϵ and k. Classes of functions that are fooled by bounded independence include combinatorial rectangles [24], small-depth circuits [36, 8, 45, 9, 51], and sign polynomials [20, 21], to name a few.

*An extended abstract of this paper appears in the Proceedings of Crypto 2016.

†Chinese University of Hong Kong. Email: andrejb@cse.cuhk.edu.hk.

‡Technion and UCLA. Email: yuvali@cs.technion.ac.il

§Northeastern University. Email: viola@ccs.neu.edu.

¶Chinese University of Hong Kong. Email: chris@cse.cuhk.edu.hk.

In this work we consider a relaxation of bounded independence that we call *bounded indistinguishability*. Two distributions μ and ν over Σ^n are *k-wise indistinguishable* if for all subsets $S \subseteq [n]$ of size k, the projections $\mu|_S$ and $\nu|_S$ of μ and ν to the coordinates in S are identical. For instance, if μ (resp., ν) is uniform over n-bit strings whose parity is 0 (resp., 1), then μ and ν are both $(n-1)$-wise independent and hence are also $(n-1)$-wise indistinguishable. However, if we let $\mu' = \mu \circ \mu$ (i.e., a concatenation of two identical copies of μ) and similarly $\nu' = \nu \circ \nu$, then μ' and ν' are still $(n-1)$-wise indistinguishable but are not even 2-independent.

Bounded indistinguishability arises naturally in cryptographic applications that involve secret sharing or secure multiparty computation. We will be interested in the complexity of distinguishing between two k-wise indistinguishable distributions.

Definition 1.1. For $\epsilon \in (0,1)$, we say that a function $f\colon \Sigma^n \to \{0,1\}$ is *ϵ-fooled by k-wise indistinguishability* if for any two k-wise indistinguishable distributions μ and ν over Σ^n,

$$|\Pr[f(\mu) = 1] - \Pr[f(\nu) = 1]| \leq \epsilon.$$

Our goal is to understand which functions f are fooled by k-wise indistinguishability. For instance, polylogarithmic independence fools all AC^0 circuits [9]. Is this also the case for polylogarithmic indistinguishability?

We start by observing that over the binary alphabet $\Sigma = \{0,1\}$, whether f is fooled by k-wise indistinguishability is closely related to the *approximate degree* of f, a notion introduced in the seminal work of Nisan and Szegedy [40]. This connection is central to our work so we formalize it next. The ϵ-approximate degree of a function $f : \{0,1\}^n \to \{0,1\}$ is defined to be the smallest degree of a real-valued polynomial $p : \{0,1\}^n \to \mathbb{R}$ such that $|f(x) - p(x)| \leq \epsilon$ for every $x \in \{0,1\}^n$.

Theorem 1.2. *For every n, k, $\epsilon \in (0,1)$, and $f\colon \{0,1\}^n \to \{0,1\}$, the following are equivalent:*

1. *f is not ϵ-fooled by k-wise indistinguishability.*
2. *The $\epsilon/2$-approximate degree of f is bigger than k.*

Proof. It follows from linear programming duality (see for example §3 in [48] or Theorem 1 in [11]) that 2. is equivalent to the following statement:

3. *There exists a function $g\colon \{0,1\}^n \to \mathbb{R}$ such that (i) $\sum_{x \in \{0,1\}^n} g(x)f(x) > \epsilon/2$, (ii) $\sum_x |g(x)| = 1$, and (iii) $\sum_x g(x) \prod_{i \in S} x_i = 0$ for every set $S \subseteq [n]$ of size at most k (including the empty set).*

We now show that 1. and 3. are equivalent. To see that 1. implies 3., we assume without loss of generality that $\Pr[f(\mu) = 1] - \Pr[f(\nu) = 1] > \epsilon$ and set $g(x) = \frac{1}{2C}(\mu(x) - \nu(x))$, where C is the statistical distance between μ and ν. The first two requirements for g are immediate. The third requirement follows from k-wise indistinguishability of μ and ν.

To see that 3. implies 1., set $\mu(x) = 2\max\{g(x), 0\}$ and $\nu(x) = 2\max\{-g(x), 0\}$. Since $\sum g(x) = 0$ and $\sum |g(x)| = 1$, we have $\sum \mu(x) = \sum \nu(x) = 1$ and so μ and ν are probability distributions. Condition (i) implies that $\Pr[f(\mu) = 1] - \Pr[f(\nu) = 1] > \epsilon$. Finally, by linearity we have that condition (iii) implies that μ and ν are indistinguishable by k-juntas so they are k-wise indistinguishable. □

As a corollary, we get a similar connection between being *non-trivially* fooled by bounded indistinguishability and *threshold degree*, a notion introduced in the classical work of Minsky and Papert [38]. Recall that the threshold degree of a function $f : \{0,1\}^n \to \{0,1\}$ is the smallest degree of a real-valued polynomial $p : \{0,1\}^n \to \mathbb{R}$ such that the sign of $p(x)$ corresponds to $f(x)$ for every $x \in \{0,1\}^n$.

Corollary 1.3. *For every n, k and $f\colon \{0,1\}^n \to \{0,1\}$, the following are equivalent:*

1. *There is a pair of k-wise indistinguishable distributions μ, ν that are perfectly distinguished by f, namely $|\Pr[f(\mu) = 1] - \Pr[f(\nu) = 1]| = 1$.*
2. *The threshold degree of f is bigger than k.*

Combining the above with known results on approximate degree, we conclude that bounded indistinguishability over $\Sigma = \{0,1\}$ behaves very differently from bounded independence. For example, $O(1)$-wise independence suffices to 1/3-fool the OR function on n bits, but $\Omega(\sqrt{n})$-wise indistinguishability is required, due to the corresponding lower bound on the approximate degree of OR [40]. This answers the aforementioned question of whether polylogarithmic indistinguishability fools AC^0 in the negative. A separation of $\Omega(n)$ is achieved by the Majority function: $O(1)$-wise independence suffices to 1/3-fool this function [20], but $\Omega(n)$-wise indistinguishability is required by Paturi's lower bound [43].

We turn to study the case of larger alphabets Σ. Here the equivalence with previously studied notions seems to break down. We restrict the attention to alphabets of the form $\Sigma = \{0,1\}^s$, viewing the function f as being computed by a circuit with sn input bits. This setting comes up naturally in cryptographic applications, as explained below. But first we remark that, over such larger alphabets, we construct "simple" functions f that are *not* fooled by k-wise indistinguishability for much larger values of k than what is known for $\Sigma = \{0,1\}$. For example, over $\Sigma = \{0,1\}^{\mathrm{poly}(n)}$ we show that $(n - n/\mathrm{poly}\log n)$-wise indistinguishability does not $(1 - 2^{-n})$-fool AC^0 (Theorem 1.4), and that $0.99n$-wise indistinguishability does not 0.99-fool DNF (Corollary 2.10). In contrast, over alphabet $\Sigma = \{0,1\}$ it is only known that $\tilde{\Omega}(n^{2/3})$-wise indistinguishability does not fool AC^0 (by work of Aaronson and Shi [2] and Theorem 1.2).

1.1 Secret sharing schemes

A secret sharing scheme allows a dealer to share a secret between n parties, so that any k parties learn nothing about the secret from their shares whereas any r parties can reconstruct the secret from their shares. Unlike the case of *threshold secret sharing*, where $r = k + 1$, we allow a bigger gap between r and k. Such secret sharing schemes are often referred to as *ramp schemes*.

We are interested in the computational complexity of sharing and (especially) reconstructing secrets. A simple secret sharing scheme for $k = n - 1$ and $r = n$ shares a bit s into n bits $s_1, \ldots, s_n$ that are random subject to the restriction that their parity is s. This scheme cannot be implemented by constant depth circuits (in the class AC^0) as reconstruction requires computing the parity of n bits. Other secret sharing schemes, such as Shamir's [47], employ linear functions over finite fields and suffer from the same limitation.

A pair of k-wise indistinguishable distributions (μ, ν), together with a function f that *can* tell the two distributions apart, can be viewed as a secret sharing scheme for a one-bit secret: Shares of 0 and 1 are samples of μ and ν, respectively, and f is the reconstruction algorithm. Applying

this connection together with techniques for sampling by constant-depth circuits, we obtain the following secret sharing scheme in the class AC^0.

Theorem 1.4 (Secret sharing in AC^0)**.** *Let d be a constant. For every n and δ there exist:*

- Sharing in AC^0: *circuits S_0, S_1 of constant depth and size* $\mathrm{poly}(n, \log 1/\delta)$ *that sample $(n - n/(\log n)^d)$-wise indistinguishable distributions μ, ν over Σ^n, $\Sigma = \{0,1\}^{\mathrm{poly}(n)}$,*
- Reconstruction in AC^0: *a circuit R of size* $\mathrm{poly}(n)$ *and depth $d + O(1)$ such that* $\Pr[R(\mu) = 0] \geq 1 - \delta$ *and* $\Pr[R(\nu) = 1] \geq 1 - \delta$.

Moreover, the circuits S_0, S_1, and R can be constructed deterministically in time polynomial in n and $\log 1/\delta$.

Theorem 1.4 gives an explicit construction, but requires that all n parties participate in reconstruction. If one does not insist on a fully explicit construction and settles for a probabilistic construction that fails with negligible probability, the secrecy-recovery gap can be moved to an arbitrary location: In Theorem 2.13 we obtain an AC^0 secret sharing scheme that provides secrecy against any σn parties and allows reconstruction by any ρn parties for any pair of constants $0 \leq \sigma < \rho \leq 1$ and sufficiently large n.

We obtain several other schemes with incomparable features. If we do not insist on sharing in AC^0 and only require that reconstruction be done in AC^0, then we can achieve similar results with *perfect reconstruction* ($\delta = 0$). This variant builds on Corollary 1.3 and known results on the threshold degree of DNF [38]. Alternatively, we can strengthen Theorem 1.4 by allowing an AC^0 sharing algorithm that indicates failure with probability δ, but otherwise supports perfect reconstruction. In Corollary 2.10, we improve the reconstruction function complexity to a polynomial-size DNF formula (with terms of size $O(\log n)$), at the cost of a small constant reconstruction error and a slightly worse secrecy threshold.

Finally, we complement the above positive results with some negative results, showing limitations of secret reconstruction by disjunctions of juntas (Theorem 2.18) or small decision trees (Theorem 2.20). In particular, the negative results imply that the positive result of Corollary 2.10 for DNF reconstruction does not hold if the secrecy threshold is much closer to n or if the DNF is restricted to have a polynomial-size decision tree.

Techniques. In §2 we rephrase known results on approximate degree in the language of secret sharing using the connection in Theorem 1.2. The resulting schemes have AC^0 reconstruction, but achieve somewhat poor secrecy ($k \leq n^{2/3}$) and do not come with algorithms for sampling the shares. In §2.1 we show that the distributions of the shares can be sampled in AC^0. Then, in §2.2 we give a reduction that trades alphabet size for secrecy, allowing us to derive our main positive results. This reduction makes use of unbalanced disperser graphs. Our negative results, presented in §2.4, are obtained by reducing the large alphabet to a binary alphabet using a suitable set system, and then using Fourier analysis for obtaining the negative result in the binary case.

Related work. The randomized encoding technique of Applebaum et al. [7] can transform any secret sharing scheme into one where the shares are sampled by circuits in which each output depends on a fixed number of random bits (i.e, in the class NC^0), but at the cost of further increasing the complexity of reconstruction. Druk and Ishai [22] and Cramer et al. [16] consider the

question of minimizing the circuit *size* of secret sharing. They construct near-threshold schemes (i.e., with $r = (1+\epsilon) \cdot k$) in which sharing and reconstruction can be performed by circuits of size $O(n)$; however, the depth of these circuits is logarithmic in n. The above results left open the existence of nontrivial secret sharing schemes in which reconstruction can be done by constant depth circuits or by other "simple" nonlinear functions, even when the computational complexity of sharing the secret is unbounded.

1.2 Visual cryptography

Naor and Shamir [39] initiated the study of "visual cryptography" — a method for sharing secrets which allows for a physical implementation using transparencies. It can be phrased as a secret sharing scheme with ℓ-bit shares, where reconstruction proceeds by first applying bitwise-OR to the shares and then applying an approximate threshold function (with constant fractional threshold gap). The bitwise-OR is implemented by physically stacking transparencies, and the approximate threshold function is implemented by visually distinguishing between ℓ-tuples of bits (pixels) that have a low Hamming weight and those that have a high Hamming weight. The ratio between the threshold gap and ℓ is referred to as the *contrast*.

It is known that the optimal contrast of such visual schemes vanishes exponentially with the secrecy parameter k [39, 34], assuming that one requires sharp threshold reconstruction by any subset of $r = k+1$ parties. The latter assumption has been made in all works on visual cryptography we are aware of.

In §2.3 we give a visual "ramp scheme" that allows a quadratic gap between the secrecy and reconstruction thresholds:

Theorem 1.5 (Visual secret sharing)**.** *For every n and r there exists a pair of distributions μ, ν over $\{0,1\}^n$ that are $\Omega(\sqrt{r})$-wise indistinguishable so that for every subset $S \subseteq [n]$ of size r,*

$$|\Pr[\mathrm{OR}(\mu|_S) = 1] - \Pr[\mathrm{OR}(\nu|_S) = 1]| \geq 0.2.$$

Moreover, μ and ν are samplable by explicit circuits S_0, S_1 of constant depth and size polynomial in n.

The benefits are a dramatic improvement in contrast, making it independent of k and visually noticeable even for large k, as well as shorter (1-bit) shares and simpler reconstruction. The latter two properties are also achieved by other probabilistic visual schemes from the literature [35, 15]. However, this is the first visual scheme whose (probabilistic) contrast does not vanish exponentially with k. To give a better sense of the achievable parameters, in Appendix A we give some specific parameter choices along with an image demonstrating the level of contrast we achieve.

1.3 Additional cryptographic applications

The above positive results for secret sharing rely on functions f that are *not* fooled by bounded indistinguishability. Such functions can be used to recover a secret from its shares. We observe that when f *is* fooled by bounded indistinguishability, this has positive consequences for leakage-resilient cryptography. Concretely, in every implementation of a cryptographic primitive that guarantees *local secrecy,* in the sense that different values of the underlying secrets induce k-wise indistinguishable distributions of the internal state, leaking the output of f on the internal state does not compromise the secrets.

Therefore *all* secret sharing schemes with a sufficiently high secrecy parameter k protect the secret against global leakage functions that output few bits, where each output bit has a low approximate degree (significantly smaller than k). More concretely:

Theorem 1.6. *There exists a universal constant C such that the following holds. Let μ, ν be k-wise indistinguishable distributions over $\{0,1\}^n$. Let $L : \{0,1\}^n \to \{0,1\}^t$ be a leakage function such that the $1/3$-approximate degree of each of its t outputs is at most d. Then the statistical distance between $L(\mu)$ and $L(\nu)$ is bounded by δ, provided that $k \geq Cdt(t + \log \frac{1}{\delta})$.*

This theorem can be applied to leakage functions whose outputs are computed by small decision trees or disjunctions of small juntas. It can also be applied to establish leakage resilience of protocols for secure multiparty computation and the related object of "private circuits." See §3.1 and §3.2 for more details and concrete applications.

Eliminating selective failure attacks. The above applications can be relevant to any $f : \Sigma^n \to \{0,1\}$ that is fooled by bounded indistinguishability. We show that the special case where $f = \mathrm{OR}$ can be useful for eliminating so-called "selective failure" attacks. A selective failure attack is an attack that makes a computation fail only if the input satisfies some predicate. Such attacks enable an adversary to tamper with the computation and learn a bit of information about the secret input even when the tampering is detected and the output is replaced by an indication of failure. Selective failure attacks arise in different areas of cryptography and are often difficult to protect against.

We propose the following natural methodology for protecting against such attacks. Suppose that the computation of $g(w)$ can be reduced to n sub-computations $g_1(w_1), \ldots, g_n(w_n)$, where each k of the w_i jointly hide w. The computation of g via this reduction fails if at least one of the sub-computations fails. Assume further that an adversary tampers with each sub-computation g_i by choosing an arbitrary function of $F_i(w_i)$ that determines whether this sub-computation fails. Then, a corollary of Theorem 1.6 (with $t = 1$ and $L = \mathrm{OR}$) is that if $k \gg \sqrt{n}$ (the approximate degree of OR), then no tampering strategy can significantly correlate the event of failure with w. In §3.3 we describe a simple concrete application of this methodology to eliminating selective failure attacks in error-detecting coding schemes.

Organization. In §2 we present our results on secret sharing. In §2.4 we prove our negative results and in §3 we give the details of the additional cryptographic applications described above. In Appendix D we discuss an approximate notion of bounded indistinguishability.

2 Secret sharing

In this section we prove our results on secret sharing. Our starting observation is that bounded indistinguishability is closely related to the complexity of secret sharing. Specifically, the distributions μ and ν over Σ^n capture the joint distributions of shares obtained by sharing the secrets 0 and 1, respectively. The k-wise indistinguishability of the distributions corresponds to the parties gaining no information from any k shares. However, if bounded indistinguishability does *not* fool some function $f : \Sigma^n \to \{0,1\}$ we can think of f as the reconstruction function that maps the shares back to the secret.

In this setting it is natural to think of the distinguishing advantage as being close to (and ideally equal to) one. We will be interested in the complexity of the function f as well as the complexity of sampling μ and ν.

A different connection between secret sharing and approximation theory is obtained in the visual cryptography literature [39] (see also [34] and the citations therein). However, it was confined to analyzing the so-called contrast of visual cryptography schemes.

We give next a formal definition of secret sharing for a one-bit secret.[1]

Definition 2.1. An (n, k, r) *bit secret sharing scheme* with alphabet Σ, reconstruction function $f : \Sigma^r \to \{0, 1\}$ and reconstruction advantage α is a pair of k-wise indistinguishable distributions μ and ν over Σ^n such that μ and ν are k-wise indistinguishable but for every set S of size r we have $\Pr[f(\mu|_S) = 1] - \Pr[f(\nu|_S) = 1] \geq \alpha$. Here $\mu|_S$ is the projection of μ to the symbols in S, and similarly for ν. The secret sharing scheme has *perfect reconstruction* if $\alpha = 1$. The scheme is *explicit* if f is explicit and there are explicit algorithms to sample μ and ν.

As mentioned earlier, the distributions μ and ν are the joint distributions of shares obtained by sharing the secret 0 and 1, respectively. We sometimes omit reference to the alphabet when $\Sigma = \{0, 1\}$ and omit r from the notation when $r = n$.

We note that Item 1. in Theorem 1.2 is equivalent to the assertion that there exists an (n, k) bit secret sharing scheme (with $r = n$ and one-bit shares) with reconstruction function f having reconstruction advantage ϵ. Item 1. in Corollary 1.3 is equivalent to the assertion that there exists a similar scheme with perfect reconstruction.

Theorem 1.2, combined with the body of works on approximate and threshold degree immediately gives the following consequences.

Corollary 2.2. *The following secret sharing schemes over $\Sigma = \{0, 1\}$ exist:*

1. *An $(n, \Omega(\sqrt{\delta n}))$ bit secret sharing scheme with reconstruction by OR with advantage $1 - \delta$, for any δ.*
2. *An $(n, \Omega(n))$ bit secret sharing scheme with reconstruction by majority with constant advantage.*
3. *An $(n, \Omega((n/\log n)^{2/3})$ bit secret sharing scheme with reconstruction by the element distinctness DNF and constant reconstruction advantage.*
4. *An $(n, \Omega(n^{1/3}))$ bit secret sharing scheme with perfect reconstruction by the DNF $AND_{n^{1/3}} \circ OR_{n^{2/3}}$.*
5. *An $(n, \Omega(\sqrt{n}))$ bit secret sharing scheme with perfect reconstruction by some AC^0 function.*

Proof. The schemes follows by Theorem 1.2 and the following works: 1. by Nisan and Szegedy [40] and refinements by Bun and Thaler [11] (Proposition 14); 2. by Paturi [43]; 3. by Aaronson and Shi [2]; 4. by Minsky and Papert [38]; and 5. by Sherstov [49]. □

These results show that for an interesting range of parameters, the reconstruction procedure of a secret sharing scheme can be implemented by simple functions, and in particular by constant depth circuits.

[1] Restricting the attention to a one-bit secret is without loss of generality; an ℓ-bit secret can be shared by invoking a scheme for a one-bit secret ℓ times in parallel.

Bounded independence versus bounded indistinguishability. In many secret sharing schemes (e.g., Shamir's scheme [47] over a field of characteristic 2), the distributions μ and ν are not only k-wise indistinguishable but also k-wise *independent*. Such distributions cannot be distinguished by AC^0 functions and sign polynomials of degree 2 unless k is at most polylogarithmic in n. In contrast, the above examples give examples of k-wise indistinguishable distributions that are distinguishable by such function even when k grows polynomially with n.

Remark 2.3. *Aaronson [1] considers a different relaxation of bounded independence that has a dramatic effect on distinguishability by* AC^0 *functions. He considers distributions where for any k bits the probability that those bits take any fixed value is within $\epsilon 2^{-k}$ of 2^{-k} and gives a family of depth 3 polynomial-size circuit that distinguishes such a distribution from a uniform one with constant advantage for any k and $\epsilon = k \cdot \mathrm{poly}\log(n)/n$.*

2.1 Sampling the shares in AC^0

In this section we show the existence of secret sharing schemes in which sharing the secret can be performed by constant-depth circuits, i.e., in the class AC^0, and reconstructing the secret can be done by a "simple" function. (As discussed in §1.1, the problem of minimizing the complexity of sharing alone is much simpler and can be solved via the techniques of [7].)

We start by showing how to sample distributions that are exponentially close to the k-wise indistinguishable distributions corresponding to the schemes we described. In Appendix C we give a refinement that gives distributions that are (exactly) k-wise indistinguishable, i.e., we achieve perfect secrecy.

Theorem 2.4. *For schemes 1. to 4. in Corollary 2.2 there exist pairs of circuit families of constant depth and size polynomial in n and $\log(1/\epsilon)$ that sample distributions within statistical distance ϵ of μ and ν, respectively.*

We leave the existence of efficient samplers for scheme 5. as an open question.

Note that we can achieve statistical distance $\epsilon = 2^{-n^c}$ for any constant c with circuits of size $\mathrm{poly}(n)$. The reason for this loss in statistical distance is that our distributions over the shares have probability masses that may not be powers of two, and so if we want to sample them using random bits we have to incur some slight error.

We now give the proof of this theorem. Our analysis relies on known explicit constructions of "dual polynomials," i.e., of the function g in Item 3. in Theorem 1.2. This area of research has been quite active since Špalek [50] gave the first explicit dual polynomial for OR.

Let Γ be a group of permutations acting on $[n]$. Then Γ also acts on $\{0,1\}^n$ by permuting the coordinates. The next claim is immediate.

Claim 2.5. Let Γ be a group of permutations on $[n]$. Assume $f(x) = f(\sigma x)$ for all $x \in \{0,1\}^n$ and all $\sigma \in \Gamma$. If (μ, ν) is an (n, k, r) bit secret sharing scheme with reconstruction function f and advantage α, then so is $(\overline{\mu}, \overline{\nu})$ where

$$\overline{\mu}(x) = \mathrm{E}_{\sigma\sim\Gamma}[\mu(\sigma x)] \quad \text{and} \quad \overline{\nu}(x) = \mathrm{E}_{\sigma\sim\Gamma}[\nu(\sigma x)].$$

In particular, if f is symmetric under permutation of its input coordinates, then the distributions μ and ν can be assumed to assign the same probability to all strings of the same Hamming weight. These $n+1$ probabilities can be found in polynomial time by solving a linear program.

Moreover, we argue that in such a case μ is AC^0-samplable; the same argument applies to ν. Let μ' be the distribution on Hamming weights induced by μ. To sample from μ, we first sample a weight $w \in \{0, \ldots, n\}$ from μ', then output a random permutation of the string $1^w 0^{n-w}$. Both of these steps can be implemented in AC^0; cf. [53].

Therefore secret sharing with reconstruction by OR and majority can both be implemented in AC^0.

A description of the bit sharing scheme for element distinctness can be extracted from the work of Bun and Thaler [12]. They first construct a bit secret sharing scheme for a partial function f whose inputs are strings of length N over an alphabet Σ of size $O(N)$. In the yes inputs of f all symbols are distinct, while in the no inputs all symbols occur exactly twice. Their distributions μ and ν are supported on strings where m/a symbols occur exactly a times and $(N-m)/b$ symbols occur exactly b times for various choices of m, a, b.

We can represent the input to f as a binary string $x_1 \ldots x_N \in (\{0,1\}^\Sigma)^N$, where x_i is an indicator vector for the i-th input symbol of f. Under this representation, f is a partial boolean function from $\{0,1\}^{|\Sigma| \cdot N}$ to $\{0,1\}$. By Claim 2.5 we may assume μ and ν are invariant over both permutations of the alphabet and permutations of the input positions. Now μ and ν can be sampled by first sampling (m, a, b) from the marginal distribution, then writing down an arbitrary string with the correct counts, and applying random permutations to both the alphabet and the input positions. All of these steps can be implemented in AC^0. The bit secret sharing scheme for OR is obtained by projecting the entries of μ and ν on random subsets of size n, which can also be implemented by sampling a random permutation.

An explicit description of the bit sharing scheme for the Minsky-Papert function can be extracted from the work of O'Donnell and Servedio [42] (Appendix A). They first sample an integer t of magnitude at most $n^{1/3}$ (even for μ, odd for ν) then choose an independent random string of Hamming weight $(t-i)^2$ in the i-th block. Both steps can be implemented in AC^0.

2.2 Trading alphabet size for secrecy

We now give a general method of composing secret sharing schemes. We will apply this method to improve the secrecy of the above schemes at the cost of an increase in alphabet size and a slight increase in depth of the reconstruction. Our construction makes use of disperser graphs.

Definition 2.6. A $n \times m$ bipartite graph G with left degree d is a (k, ϵ) disperser if any subset of $[n]$ of size k has at least $(1-\epsilon)n$ neighbors in $[m]$.

The loss in reconstruction efficiency is related to the degree d of the disperser. So we obtain the best results with Zuckerman's construction of dispersers with degree linear in $\log n/\epsilon$.

Theorem 2.7 (Theorem 1.9 of [54] with $\alpha = 1/2$)**.** *For every constant δ, and for every n and ϵ there is an explicit (n^δ, ϵ) disperser G with $d = O(\log n/\epsilon)$ and $m = \delta n/2$.*

We now show how to turn an (n, k) secret sharing scheme L over alphabet $\{0,1\}$ into a $(m, m - \epsilon m)$ secret sharing scheme R over alphabet $\{0,1\}^n$. The alphabet is actually $\{0,1\}^{d'}$ where d' is the maximum right-hand side degree of the disperser graph. It is possible to obtain d' close to the average degree nd/m, but in our settings this will always be $n^{\Omega(1)}$ and so for simplicity we do not optimize this parameter.

The parties of L and R are associated to the left and right vertices of the bipartite graph respectively. To share a bit in R, first sample shares for L and label each left vertex $v \in [n]$ by its

corresponding share $s(r) \in \{0,1\}$. Now for each of the d edges $e_1, \ldots, e_d$ incident to r, choose a bit $s(e_i)$ at random conditioned on $s(e_1) \oplus \cdots \oplus s(e_d) = s(r)$. The share $s(w)$ of each right vertex $w \in [m]$ is the concatenation of the edge-shares $s(e)$ over all its $\leq n$ incident edges e.

To reconstruct, apply the process in reverse: First distribute $s(w)$ for $w \in [m]$ to its incident edges, then calculate $s(v), v \in [n]$ as $s(e_1) \oplus \cdots \oplus s(e_d)$ and output $f(s(1), \ldots, s(n))$, where f is the reconstruction function of L.

Lemma 2.8. *If G is a (k, ϵ) disperser graph and L is a (n, k) secret sharing scheme then R is a $(m, m - \epsilon m)$ secret sharing scheme with the same reconstruction advantage.*

Proof. It is easy to see that the reconstruction advantage is preserved. Next we argue secrecy.

For contradiction, assume that L is k-secret but R is not $(n - \epsilon n)$-secret. Then there exists a subset $S \subseteq [m]$ of size $\leq m - \epsilon m$ such that the parties in S can distinguish shares of 0 from shares of 1. Consider the joint distribution of the shares assigned to all the edges incident to S. If any vertex $v \in [n]$ has a neighbor outside S, then the edge-shares associated to v's neighbors inside S are uniformly random and independent of all the other edge-shares incident to S (even conditioned on all the values $s(v)$). Therefore, the two distributions must be distinguishable even when restricted to those edges whose right vertices have all their neighbors in S. Let T be the set of all such right vertices. Then the shares of S in L are determined by the shares of T in R. By the disperser property of G, T has size at most k, so the shares in T are indistinguishable, contradicting our assumption. □

We note that Alon et al. [4] applied a similar construction to amplify the distance of linear error-correcting codes, while Damgård et al. [19] used it (in more general form) for improving the tolerance of multiparty computations. Both these applications make use of dispersers (in fact, expanders) G that are balanced (with $m = n$) and of constant degree d. In contrast, we apply it to unbalanced graphs whose left degree is logarithmic in the number of vertices.

If we set $k = n^\alpha$ for some constant $\alpha > 0$, we obtain the following consequence. Here $f \circ XOR_d$ denotes a function that can be computed by composing f by XORs over d inputs.

Theorem 2.9. *Let $\alpha > 0$ be a constant. Suppose that there exists a (n, n^α) secret sharing scheme with reconstruction function $f : \{0,1\}^n \to \{0,1\}$ over alphabet $\{0,1\}$. Then there exists a $(m, (1 - \epsilon)m)$ secret sharing scheme over alphabet $\{0,1\}^n$ with reconstruction function of the type $f \circ XOR_d$ with $d = O((\log n)/\epsilon)$ and $m = \Omega(n^\alpha)$.*

We now have all the pieces to prove Theorem 1.4.

Proof. (of Theorem 1.4) Instantiate Theorem 2.9 with Item 4 in Corollary 2.2. The reconstruction function involves computing parities on $\operatorname{poly} \log n$ bits which can be done in AC^0. To sample the shares efficiently use Theorem C.1. □

Several other schemes are possible. We highlight the following one in which reconstruction is done by a DNF, although it is not perfect.

Corollary 2.10. *For every constant $\epsilon > 0$, there is an explicit $(n, (1 - \epsilon)n)$-secret sharing scheme with reconstruction error ϵ over the alphabet $\{0,1\}^{\operatorname{poly}(n)}$ with reconstruction by a $\operatorname{poly}(n)$-size DNF with terms of size $O(\log n)$.*

Proof. Instantiate Theorem 2.9 with Item 1 in Corollary 2.2. The reconstruction function is an OR of $O((n/\log n)^2)$ XORs of size $O((\log n)/\epsilon)$, which can be computed by a polynomial-size DNF. The shares can be sampled in AC^0 by Theorem 2.4. □

2.3 Reconstruction by a subset of the parties

In this section we give several secret sharing schemes that allow for reconstruction by a subset of the parties. Our starting point is the secret sharing scheme with reconstruction by the OR function.

Claim 2.11. For every r, δ, and n there is an explicit $(n, \Omega(\sqrt{\delta n}), r)$ bit secret sharing scheme with reconstruction by OR with advantage at least $r/n - \delta$.

Here, by OR we mean the class of OR functions on subsets of r input bits. We will need the following fact which is implicit in the proof of Theorem 1.2.

Remark 2.12. *Without loss of generality, the distributions μ and ν can be assumed to have disjoint support.*

Proof. (of Claim 2.11) Let (μ, ν) be any (n, k) bit sharing scheme for OR with reconstruction advantage $1-\delta$. By Remark 2.12 and Claim 2.5 we may assume μ and ν are disjoint and symmetric, so $\nu(0^n) = 1 - \delta$ and all strings in the support of μ have nonzero Hamming weight. For any subset of r parties, the probability that they jointly observe a nonzero entry under ν is then at most δ. By symmetry of μ, the probability that they observe nonzero entry under μ is at least r/n. Therefore $\Pr[f(\mu) = 1] - \Pr[f(\nu) = 1] \geq r/n - \delta$. □

If we set $\delta = r/2n$ we obtain an $(n, \Omega(\sqrt{r}), r)$ bit secret sharing scheme with reconstruction by OR with advantage $\delta = r/2n$. In the next result we make this advantage constant.

We now prove Theorem 1.5, namely the existence of a $(n, \Omega(\sqrt{r}), r)$ bit secret sharing scheme with reconstruction by OR with constant advantage.

Proof. (of Theorem 1.5) First we construct a scheme over alphabet $\{0,1\}^{1/\delta}$ for $\delta = 2n/r$ which we assume to be an integer. To share a zero and a one respectively, sample $1/\delta$ independent shares using the scheme in Claim 2.11 and give the i-th party the i-th bit from each copy. By the proof of Claim 2.11 for any $\Omega(\sqrt{r})$ parties the OR of their i-th copies of their shares of one and zero evaluate to 1 with probability at least $1 - (1-2\delta)^{1/\delta}$ and at most $1 - (1-\delta)^{1/\delta}$, respectively. The difference between these two numbers is always positive and tends to $1/e - 1/e^2$ as $1/\delta$ increases.

To reduce the alphabet to binary, we replace each party's share by the OR of its constituent bits. □

If we allow for more complexity in reconstruction and larger shares, the gap between the secrecy and reconstruction parameters can be improved and the reconstruction error can be made negligible.

Theorem 2.13. *For every pair of constants $0 \leq \sigma < \rho \leq 1$ and sufficiently large m there exists a $(m, \sigma m, \rho m)$ bit secret sharing scheme with reconstruction by circuits of size polynomial in m and depth 4 and advantage $1 - 2^{-m^c}$ for any constant c over alphabet $\Sigma = \{0,1\}^{\mathrm{poly}(m)}$.*

To prove Theorem 2.13, we apply the composition method from §2.2 using a bipartite graph with the following dispersion properties.

Claim 2.14. For all constants $\delta > 0$ and $0 \leq \sigma < \rho \leq 1$, and every sufficiently large n there exist numbers $m = n^{\Omega(1)}$, $r \leq n$, and $d = O(\log n)$ and an $n \times m$ bipartite graph G with left degree d such that

1. For every subset $S \subseteq [m]$ of size at most σm, the set of vertices in $[n]$ all of whose neighbors are in S has size at most r^δ (i.e., G is a $(r^\delta, 1-\sigma)$-disperser), and

2. For every subset $R \subseteq [m]$ of size at least ρm, the set of vertices in $[n]$ all of whose neighbors are in R has size at least r.

We then amplify the reconstruction error in Theorem 1.5 using the following claim.

Claim 2.15. For every integer t, if there exists a (m, k, r) bit secret sharing scheme with reconstruction by size s and depth d circuits and constant advantage over alphabet Σ then there exists a (m, k, r) bit secret scheme with reconstruction by circuits of size $st + \mathrm{poly}(t)$ and depth $d+2$ and advantage $1 - 2^{-\Omega(t)}$ over alphabet Σ^t.

Proof. (of Theorem 2.13) We apply the construction described in §2.2 to the $(n, \Omega(\sqrt{r}), r)$ scheme from Theorem 1.5 and the graph from Claim 2.14 with $\delta = 0.49$. Secrecy follows from Theorem 2.9. Reconstruction proceeds as in §2.2, except that only those parties in $[n]$ that have received all of their shares participate in the process. By property 2 of Claim 2.14, if at least ρm parties on the right participate in the reconstruction then at least r parties on the left receive all their share and the secret is reconstructed with constant advantage. By Claim 2.15 with $t = m^c$, the advantage can be amplified to $1 - 2^{-m^c}$ as desired. □

Proof. (of Claim 2.14) We show that a random graph has both properties with nonzero probability. Choose each of the d neighbors of each left vertex independently and uniformly at random. For a fixed set $S \subseteq [m]$ of size σm, the expected number of left vertices all of whose neighbors are in S equals $n\sigma^d$. By the multiplicative Chernoff-Hoeffding bound and a union bound, the probability that there exists a set S and a set of left vertices of size $2n\sigma^d$ all of whose neighbors are in S is at most $2^m \exp(-n\sigma^d/8)$. By a similar argument, the probability that there exists a set $R \subseteq [m]$ of size ρm such that fewer than $n\rho^d/2$ vertices have all their neighbors in R is at most $2^m \exp(-n\rho^d/3)$.

We set $d = \log_{\rho^\delta/\sigma}(2^{1+\delta} n^{1-\delta})$, $r = (\rho/\sigma)^{d/(1-\delta)}$, and $m = \lfloor r^\delta/2 \rfloor$. This choice of parameters ensures that $n\rho^d/2 = r$, $2n\sigma^d = r^\delta$, and $r, m = n^{\Omega(1)}$. Moreover, both probabilities of interest tend to zero at the rate $\exp(-\Omega(r^\delta)) = \exp(-n^{\Omega(1)})$ so a graph with the desired properties exists for sufficiently large n. □

Proof. (of Claim 2.15) For every pair of constants $0 \leq \ell < h \leq 1$, Ajtai [3] shows the existence of a Boolean function family $ApxMaj$ of depth 3 and size polynomial in its input such that $ApxMaj$ accepts all strings of relative Hamming weight at least h and rejects all strings of relative Hamming weight at most ℓ. These circuits are made explicit in [52].

Let S be the assumed secret sharing scheme. Choose h and ℓ so that the success probability of reconstructing a one from its shares in S bounds h strictly from above and the failure probability of reconstructing a zero in S bounds ℓ strictly from below. To share a bit, sample k independent copies of shares of S and give the i-th party the i-th bit of each copy. To perform the reconstruction, first apply the reconstruction algorithm for S for each copy, then apply $ApxMaj$ to all k reconstructed bits.

The secrecy of S is inherited by construction. We now analyze the probability of correct reconstruction by r parties. By the multiplicative Chernoff bound, the probability that fewer than hk copies of S reconstruct a one correctly, or that more than ℓk copies of S reconstruct a zero incorrectly, is $2^{-\Omega(k)}$. If this does not happen, $ApxMaj$ correctly recovers the secret bit. □

2.4 Limitations

In this section we prove negative results on the existence of secret sharing schemes, or equivalently positive results on functions being fooled by bounded indistinguishability. Our main technical contribution consists of proving negative results that hold even over large alphabets Σ. However, we first start with the case $\Sigma = \{0,1\}$ because this provides motivation and is useful for larger Σ.

In the case $\Sigma = \{0,1\}$ we note an upper bound of $n(1-1/\mathrm{poly}\log n)$ on the approximate-degree of AC^0. While it follows from standard Fourier-analytic techniques, we are not aware that it has been observed before. In terms of secret sharing schemes it shows that the secrecy is at most $n(1-1/\mathrm{poly}\log n)$ if reconstruction is to be done in AC^0.

Claim 2.16. Every function $f\colon \{0,1\}^n \to \{0,1\}$ that has a size s depth d circuit has $n^{-h/2}$-approximate degree $n-h$ for $h = \Omega_d(n/(\log s)^{d-1}(\log n))$.

Proof. We will work with the function $F\colon \{-1,1\}^n \to \{-1,1\}$ given by $F(X) = 1-2f((1+X)/2)$. We construct a polynomial $P\colon \{-1,1\}^n \to \mathbb{R}$ that approximates F pointwise within $2n^{-h/2}$. Let

$$P(X) = \sum\nolimits_{S\subseteq[n],|S|\le n-h} \hat{F}(S) \prod\nolimits_{i\in S} X_i,$$

where $\hat{F}(S) = \mathrm{E}[F(X)\prod_{i\in S} X_i]$ are the Fourier coefficients of F, see e.g. O'Donnell's book [41] for background.

Håstad [28] shows that $|\hat{F}(S)| \le 2^{-c|S|/(\log s)^{d-1}}$, where c is some constant that depends only on d. For every $X \in \{-1,1\}^n$,

$$\begin{aligned} |F(X)-P(X)| &= \Big|\sum\nolimits_{S\colon |S|>n-h} \hat{f}(S) \prod\nolimits_{i\in S} X_i\Big| \\ &\le \sum\nolimits_{S\colon |S|>n-h} |\hat{f}(S)| \le n^h \cdot 2^{-c(n-h+1)/(\log s)^{d-1}}, \end{aligned}$$

which is at most $2n^{-h/2}$ for $h = \min\{n/2, cn/4(\log s)^{d-1}(\log n)\}$. □

The following upper bound on the approximate degree of the OR function was obtained by Kahn, Linial, and Samorodnitsky [33]. The special case $\delta = 1/3$ was first established by Nisan and Szegedy [40].

Lemma 2.17. *For every n and δ, the δ-approximate degree of OR on n bits is $O(\sqrt{n\log(1/\delta)})$.*

It follows from Theorem 1.2 that there does not exist a $(n, \omega(\sqrt{n\log(1/\delta)})$ secret sharing scheme over the alphabet $\{0,1\}$ with reconstruction by OR and advantage δ.

We now derive two negative consequences for secret sharing schemes with more complex reconstruction functions and over alphabets of arbitrary size.

Theorem 2.18. *For every Σ of the form $\{0,1\}^s$ and all n, m, d, h such that $h \le n/(3\ln n \cdot \exp(6\sqrt{\ln(2m)\cdot\ln d}))$ if $f\colon \Sigma^n \to \{0,1\}$ is an OR of m functions each of which depends on at most d inputs then there is no $(n, n-h)$ secret sharing scheme with reconstruction function f and advantage $1/3$.*

In particular, Theorem 2.18 shows that if reconstruction is done by a DNF of size $m = \mathrm{poly}(n)$ and with terms of size $d = n^{o(1)}$ then the secrecy must be at most $n-h = n-n^{1-o(1)}$.

The proof of the theorem relies on the following combinatorial claim.

Claim 2.19. For every N, M, n, m, d, and h such that $h \ln n, M \ln N + 1 \leq n/(3d^M(2m)^{M/N})$ and every collection $\mathcal{S}$ of m subsets of $[n]$, each of size d, there exists a collection $\mathcal{T}$ of N subsets of $[n]$ such that

1. for every set $S \in \mathcal{S}$ there is at least one set $T \in \mathcal{T}$ such that S is a subset of T, and
2. for every M sets $T_1, \ldots, T_M \in \mathcal{T}$, $|T_1 \cup \cdots \cup T_M| < n - h$.

Proof. (of Theorem 2.18) Suppose for contradiction that such a secret sharing scheme S exists. Let $S_i \subseteq [n]$ be the set of variables in the i-th term of f and $\mathcal{S} = \{S_1, \ldots, S_n\}$. For $N = \log_d(2m)$, $M = 2\sqrt{N}$, and sufficiently large n the set system $\mathcal{T} = \{T_1, \ldots, T_N\}$ given by Claim 2.19 exists. Assign to each term t of f a single set $T(t) \in \mathcal{T}$ that covers it as guaranteed by Property 1 of the Claim.

Consider the following N-party secret sharing scheme T for OR. To share, first run the secret sharing for S and evaluate each term t of f using the shares as inputs. Then assign each party i in T the OR of all the terms t such that $T(t) = T_i$. To reconstruct take the OR of all the shares of T. By construction, this equals f evaluated on the shares of S, so T has the same reconstruction advantage as S.

By Property 2 of Claim 2.19, each collection of M parties of T observes fewer than $n - h$ shares of S, so T is an (N, M) secret sharing scheme. By Lemma 2.17 T cannot have reconstruction advantage $1/3$, so neither can S. □

Proof. (of Claim 2.19) We choose the M sets of $\mathcal{T}$ at random such that each element in $[n]$ is included in each set in $\mathcal{T}$ independently with probability $1 - q$ for $q = (1/d)(1/2m)^{1/N}$. On the one hand, by a union bound, the probability that some set $S \in \mathcal{S}$ fails to be covered by any set of $\mathcal{T}$ is at most $m(qd)^N$, which is at most $1/2$ by our choice of q. On the other hand, by a union bound, the probability that property 2 is violated is at most

$$\begin{aligned}\binom{N}{M} \cdot \binom{n}{n-h} \cdot \left(1 - q^M\right)^{n-h} &\leq \exp\left(M \ln N + h \ln n - (n-h)q^M\right) \\ &\leq \exp\left(M \ln N + h \ln n - (2n/3)q^M\right) \\ &\leq 1/e\end{aligned}$$

by the assumed inequality. By a union bound, both desired properties are satisfied with probability at least $1 - 1/2 - 1/e > 0$. □

Next we obtain a stronger negative result in the case in which the reconstruction is done by a decision tree.

Theorem 2.20. *Let $\Sigma = \{0,1\}^s$. If $f \colon \Sigma^n \to \{0,1\}$ has a binary decision tree with at most S leaves then there is no $(n, \omega(\sqrt{n \log(S/\epsilon)}))$-bit secret sharing scheme with reconstruction function f and advantage ϵ.*

In particular, a secret sharing scheme with constant advantage and whose reconstruction function is a polynomial-size decision tree can only be secure against coalitions of $O(\sqrt{n \log n})$ parties.

Proof. First assume f is an OR of a subset of literals. If a secret sharing scheme with reconstrcution function f, secrecy parameter $\omega(\sqrt{n(\log 1/\delta)})$, and advantage δ existed, then a scheme with the

same parameters would exist for a binary alphabet as each party's shares can be replaced by the respective OR of the relevant literals, contradicting Lemma 2.17. By symmetry the same conclusion holds for ANDs of subsets of literals.

If f has a decision tree with $\leq S$ leaves, then we can write f as a sum of at most S ANDs of literals, one for each path in the decision tree that leads to a 1-leaf. This sum is over the reals yet it will always take a boolean value because at most one AND will evaluate to one. If there existed a secret sharing scheme with reconstruction function f, advantage ϵ and the desired properties, by a hybrid argument one of the constituent ANDs would have advantage ϵ/S in the same scheme. Setting $\delta = \epsilon/S$ yields the desired conclusion. □

3 Additional cryptographic applications

In this section we present additional applications of our results on bounded indistinguishability in cryptography. These applications can be viewed as different instances of *leakage-resilient cryptography.*.

The broad goal of leakage-resilient cryptography is to maintain the security of cryptographic primitives even if partial information about their secrets is leaked to an adversary. The type of information being leaked is typically captured by a *leakage function* $L : \{0,1\}^n \to \{0,1\}^t$ taken from a leakage class $\mathcal{L}$, where the input for L represents the internal (secret) state of the primitive and its output represents the partial information available to the adversary. For simplicity we will start by considering the case of single-bit leakage (i.e., $t = 1$) and later extend the results to the more general case.

Our motivating observation is that if two possible distributions of secret states are k-wise indistinguishable, and moreover k-wise indistinguishability implies $\mathcal{L}$-indistinguishability, then obtaining leakage-resilience against $\mathcal{L}$ reduces to obtaining resilience against *k-local* leakage, namely the class of all projection functions $P : \{0,1\}^n \to \{0,1\}^k$. Obtaining provable security against k-local leakage is typically much easier than obtaining provable security against bigger leakage classes, and can be achieved via standard techniques for secret sharing and secure multiparty computation (MPC).

The above observation may be relevant to any cryptographic scheme that maintains a sufficient level of local secrecy. We illustrate its usefulness by presenting applications in the contexts of secret sharing, error detecting codes, and private circuits.

3.1 Leakage-resilience of secret sharing schemes

The implication 1. $\implies$ 2. in Theorem 1.2 can be reformulated in the following equivalent way.

Claim 3.1. Let μ, ν be k-wise indistinguishable distributions over $\{0,1\}^n$. Let $L : \{0,1\}^n \to \{0,1\}$ be a leakage function whose ϵ-approximate degree is at most k. Then

$$|\Pr[L(\mu) = 1] - \Pr[L(\nu) = 1]| \leq \epsilon.$$

Claim 3.1 implies that *every* (m, k) bit secret sharing scheme over $\Sigma = \{0,1\}^\ell$ is resilient against leakage functions $L : \{0,1\}^{m\ell} \to \{0,1\}$ whose approximate degree is at most k. The same holds for secret sharing schemes with bigger secrets.

Many secret sharing schemes from the literature are in fact k-wise independent for a large value of k, in the sense that any k bits in μ and ν are uniformly distributed. This is the case, for instance,

for Shamir's scheme [47] over fields of characteristic 2. In such a case one can appeal to stronger results about bounded independence. For instance, Braverman's theorem [9] implies resilience to every AC^0 leakage function L even when k is polylogarithmic in n, whereas the approximate degree of some AC^0 functions is known to be as big as $\Omega(n^{2/3})$. One could also apply similar results in the case of *biased* k-wise independence, namely μ and ν are k-wise indistinguishable and moreover every k bits are independently distributed (but may each have a different bias). See, e.g., Lemma 5.2 in [14] for the case of OR distinguishers.

However, there are cases in which it is undesirable or even impossible to guarantee a high level of independence. For instance, when considering secret sharing schemes with special properties, such as ones supporting multiplication, bounded independence may come at a significant price [13, 44]. Alternatively, the shares of a k-wise independent secret sharing scheme may be subject to local encoding or to adversarial tampering, after which they are no longer k-wise independent but are still k-wise indistinguishable.

Finally, we extend Claim 3.1 to the case of a leakage function L with t output bits. For convenience, we restate Theorem 1.6 from the Introduction.

Theorem 3.2. *There exists a universal constant C such that the following holds. Let μ, ν be k-wise indistinguishable distributions over $\{0,1\}^n$. Let $L : \{0,1\}^n \to \{0,1\}^t$ be a leakage function such that the $1/3$-approximate degree of each of its t outputs is at most d. Then the statistical distance between $L(\mu)$ and $L(\nu)$ is bounded by δ, provided that $k \geq Cdt(t + \log \frac{1}{\delta})$.*

Proof. Using an indistinguishability variant of Vazirani's statistical XOR lemma (cf. [31, Lemma 1]), it suffices to prove that every $L' : \{0,1\}^n \to \{0,1\}$ obtained by taking the parity of a subset of the outputs of L, we have $|\Pr[L'(\mu) = 1] - \Pr[L'(\nu] = 1]| \leq \delta'$ where $\delta' = \delta \cdot 2^{-t/2}$. Using Lemma B.2, the $1/3$-approximate degree of each such L' is $O(dt)$ and by Lemma B.1 its approximate degree is $O(dt \log \frac{1}{\delta'})$. Applying Claim 3.1, $k = \Omega(dt(t + \log \frac{1}{\delta}))$ suffices to guarantee that the distinguishing advantage of L' is bounded by δ' as required. □

3.2 Private circuits

We now describe an application of Claim 3.1 to *private circuits*, a computational model for leakage-resilient cryptography. We consider the simpler stateless variant of private circuits with encoded inputs and outputs (see, e.g., [32, Section 3] and [29, Section 4.1]) and privacy with respect to a general leakage class $\mathcal{L}$. Informally, such a private circuit is a (possibly randomized) boolean circuit that transforms a randomly encoded input into a randomly encoded output while providing the guarantee that the output of any $\mathcal{L}$-leakage on the n circuit wires reveals essentially nothing about the input. More formally:

Definition 3.3. **($(\mathcal{L}, \epsilon)$-private circuit)** A private circuit for $g : \{0,1\}^{n_i} \to \{0,1\}^{n_o}$ is defined by a triple (I, C, O), where

- $I : \{0,1\}^{n_i} \to \{0,1\}^{\hat{n}_i}$ is a randomized input encoder;
- C is a deterministic or randomized boolean circuit with input $\hat{w} \in \{0,1\}^{\hat{n}_i}$, output $\hat{y} \in \{0,1\}^{\hat{n}_o}$, and n wires;
- $O : \{0,1\}^{\hat{n}_o} \to \{0,1\}^{n_o}$ is a deterministic output decoder.

For a leakage function $L : \{0,1\}^n \to \{0,1\}^t$ and $\epsilon > 0$, we say that (I, C, O) is an (L, ϵ)-*private implementation of* g if the following requirements hold.

- *Correctness:* For any input $w \in \{0,1\}^{n_i}$ we have $\Pr[O(C(I(w))) = g(w)] = 1]$, where the probability is over the randomness of I and (possibly) C.
- *Privacy:* For any $w, w' \in \{0,1\}^{n_i}$, the statistical distance between $L(C[I(w)])$ and $L(C[I(w')])$ is at most ϵ, where $C[x]$ denotes the (randomized) values of the n wires of C on input x.

For a class $\mathcal{L}$ of leakage functions, we say that (I, C, O) is an $(\mathcal{L}, \epsilon)$-*private implementation of* g if it is an (L, ϵ)-private implementation of g for every $L \in \mathcal{L}$, and that it is a k-*private implementation of* g if it is an $(\mathcal{L}, 0)$-private implementation of g for the class $\mathcal{L}$ of projection functions that output k bits of the input.

Without any requirements on I and O, the above definition can be satisfied by having I compute a leakage-resilient secret sharing of the input which is passed by C directly to the decoder. To rule out such a solution we require the encoder and the decoder to be *universal* (i.e., depend only on n_i, n_o and the circuit size of g and not on g itself). Furthermore, we would like the decoder size to be considerably smaller than the circuit size of g. These requirements effectively force C to perform the bulk of the computation in a leakage-resilient manner.

While there are asymptotically efficient constructions of k-private circuits obtained via MPC techniques [32, 18, 29], much less is known about defending against larger leakage classes. We use the connection between approximate degree and bounded indistinguishability to bootstrap from k-private circuits to $(\mathcal{L}, \epsilon)$-private circuits for larger classes $\mathcal{L}$. More accurately, we show that in many cases k-privacy automatically implies $(\mathcal{L}, \epsilon)$-privacy for a large $\mathcal{L}$ and negligible ϵ. A similar result for a special type of leakage called "noisy leakage" was obtained in [23]. The parameters of the leakage-resilient circuits we obtain via bounded indistinguishability are quite limited, since our approach requires the privacy threshold k to be rather close to the circuit size. An interesting research direction is to obtain better parameters by exploiting additional structural properties of the distributions induced by private circuit constructions.

A direct application of Claim 3.1 is the following.

Claim 3.4. Let $L : \{0,1\}^n \to \{0,1\}$ be a leakage function whose ϵ-approximate degree is at most k. Suppose that (I, C, O) is a k-private implementation of g with $|C| = n$. Then (I, C, O) is also an (L, ϵ)-private implementation of g.

To apply the above claim, we combine MPC-based constructions of k-private circuits with known bounds on approximate degree. Any general-purpose n-party MPC protocol with security against k passively corrupted parties can be converted into a k-private circuit in the following way. First, the input encoder I splits the input among $k+1$ virtual parties (or more) with secrecy threshold k. Then C implements the next-message functions of an MPC protocol for computing $k+1$ shares of the output of g from the shares of the input. Finally, the output decoder D recovers the output of g from the shares. The size of C corresponds to the *computational* complexity of the MPC protocol.

To obtain meaningful corollaries, we need k to be quite close to $|C|$. Using a passive-secure variant of an MPC protocol from [18] with $n = O(k)$ parties (see [26]), the size of C can be as small as $|C| = \tilde{O}(s + kh + k^2)$, where s and h are the size and depth of a boolean circuit computing g, respectively. However, even when h is small, this is not good enough for our purposes because the additive k^2 term imposes a super-quadratic gap between k and $|C|$. We eliminate the quadratic

term by replacing a sub-protocol for generating random shared secrets, from which the quadratic term originates, by an input encoder of size $\tilde{O}(s+k)$ that directly produces the outputs of this sub-protocol. This yields the following theorem.

Theorem 3.5. *[18, 29, 26] Any $g : \{0,1\}^{n_i} \to \{0,1\}^{n_o}$ that can be computed by a circuit C_g of size s and depth h admits a k-private implementation (I, C, O), with circuit sizes $|I| = \tilde{O}(s+k)$, $|C| = \tilde{O}(s+kh)$, and $|O| = \tilde{O}(n_o+k)$. Furthermore, the construction is explicit: circuits computing I and O can be computed efficiently given $k, n_i, n_o, |C_g|$, and C can be computed efficiently given k and C_g.*

For functions g with low-depth circuits, the gap between $|C|$ and k can be made small enough to imply privacy against nontrivial leakage classes that have low approximate degree. For simplicity we state the following corollary for functions g in NC. (Here and in the following we abuse notation by using a single function to implicitly denote an infinite class of functions with varying input lengths and circuit sizes.)

Corollary 3.6. *Let $L : \{0,1\}^n \to \{0,1\}$ be a leakage function whose ϵ-approximate degree is at most k. Then any NC-function $g : \{0,1\}^{n_i} \to \{0,1\}^{n_o}$ of circuit size s admits an (L, ϵ)-private implementation (I, C, O), where $|I| = \tilde{O}(s+k)$, $|C| = \tilde{O}(s+k)$, and $|O| = \tilde{O}(n_o+k)$, assuming that $n = |C|$.*

Combining the above corollary with known bounds and conjectures on approximate degree, we obtain private circuits secure against several low-complexity leakage classes whose efficiency improves over previous constructions from [25, 46, 37]. Concretely, in the latter constructions each gate is implemented by a separate gadget whose size grows (at least) quadratically with a statistical security parameter σ. As a result, the size of C grows multiplicatively with σ^2. In contrast, our approach yields private circuits for NC functions that can have a better dependence on σ.

Before formulating the general statements, we illustrate the choice of parameters with an example. Suppose we consider OR-leakage, whose 1/3-approximate degree for input length n is $O(n^{1/2})$. This implies a $2^{-\sigma}$-approximate degree of $k = O(\sigma n^{1/2})$. Applying the previous corollary with this value of k to a boolean NC function g with circuit size s, we can get an $(\mathrm{OR}, 2^{-\sigma})$-private circuit implementation (I, C, O) for g, where I and C are of size $\tilde{O}(s)$ and O is of size $\tilde{O}(\sigma s^{1/2})$, as long as $\sigma \leq s^{1/2}$. (The latter restriction guarantees that we can simultaneously satisfy $|C| = \tilde{O}(s+k)$ and $k = O(\sigma \cdot |C|^{1/2})$.) For $\sigma = s^{1/2}$ we get $2^{-\sigma}$-security with $|C| = \tilde{O}(s)$, whereas previous constructions require $|C| > s \cdot \sigma^2 = s^2$ for achieving the same level of security. A smaller choice of σ can be used to reduce the size of the decoder.

Generalizing the above example, we get the following.

Claim 3.7. Let $L : \{0,1\}^n \to \{0,1\}$ be a leakage function whose 1/3-approximate degree is at most $d(n)$. Then, for every function $\sigma(s)$ such that $\sigma(s) \leq s/d(s)$, every NC-function $g : \{0,1\}^{n_i} \to \{0,1\}^{n_o}$ of circuit size s admits an $(L, 2^{-\sigma(s)})$-private implementation (I, C, O), where $|I| = \tilde{O}(s)$, $|C| = \tilde{O}(s)$, and $|O| = \tilde{O}(n_o + \sigma(s) \cdot d(s))$.

Combining the above claim with known or conjectured upper bounds on approximate degree, we get the following corollary.

Corollary 3.8. *Any NC-function $g : \{0,1\}^{n_i} \to \{0,1\}^{n_o}$ of circuit size s admits an $(\mathcal{L}, 2^{-\sigma})$-private implementation (I, C, O), where $|I| = \tilde{O}(s)$, $|C| = \tilde{O}(s)$, and $|O| = \tilde{O}(n_o + k)$, for the following choices of $\mathcal{L}$, σ, and k:*

1. *$\mathcal{L}$ is the class of decision trees of size S, $k = \sigma\sqrt{s\log(S)}$, and $\sigma \leq \sqrt{s/\log(S)}$.*
2. *$\mathcal{L}$ is the class of read-once DNF (or CNF) formulas, $k = \sigma s^{1/2}$, and $\sigma \leq s^{1/2}$.*
3. *$\mathcal{L}$ is the entire class AC^0, $k = \sigma s^c$, and $\sigma \leq s^{1-c}$, assuming that all AC^0 functions on n-bit inputs have a 1/3-approximate degree of $O(n^c)$ for some constant $c < 1$.*

Proof. The corollary follows from the previous claim, known bounds on approximate degree, and the fact that if the 1/3-approximate degree of L is k then its $2^{-\sigma}$-approximate degree is $O(\sigma k)$ (cf. [20, Claim 3.8]). The degree bounds required for items 1. and 2. follow from Theorem 2.20 and Ambainis et al. [6], respectively. □

Extension to multi-bit leakage. The above corollary can be extended to leakage functions L with t bits of output by relying on Theorem 1.6 instead of Claim 3.1. The general form of the corollary can be obtained by replacing each occurrence of σ with σt^2.

The case of disjunctive leakage. Private circuits that resist *disjunctive leakage*, namely an OR of an arbitrary subset of wires or their negations, have found applications to constant-round secure two-party computation [30]. While it was shown in [30] that every k-private circuit can be transformed into such a disjunction-resilient circuit with a constant multiplicative overhead to the circuit size, this transformation is nontrivial and has a significant concrete cost. We note that for the purpose of this application it is essential that the encoder be small, and thus Corollary 3.8 is not useful even for the case of NC circuits.

Instead, we rely on the following corollary of Claim 3.1 to show that the same k-private circuits to which the transformation from [30] was applied are in fact already resilient against disjunctive leakage.

Claim 3.9. Let μ, ν be k-wise indistinguishable distributions over Σ^n for $\Sigma = \{0,1\}^\ell$. Let $L : \{0,1\}^{\ell n} \to \{0,1\}$ be a disjunctive leakage function. Then

$$|\Pr[L(\mu) = 1] - \Pr[L(\nu) = 1]| \leq 2^{-\Omega(k/\sqrt{n})}.$$

Proof. By decomposing L into n disjunctive functions that operate separately on each ℓ-bit symbol, $L(\mu)$ and $L(\nu)$ can be written as $OR(\mu')$ and $OR(\nu')$ (respectively), where μ' and ν' are k-wise indistinguishable distributions over $\{0,1\}^n$. The claim then follows from Claim 3.1 and the approximate degree of OR. □

The k-private circuits employed in [30] are based on MPC protocols that resist a constant fraction of corrupted parties. As such, they have the property that their N wires can be partitioned into n "symbols" in $\Sigma = \{0,1\}^{N/n}$, such that the wire distributions on different inputs are k-wise indistinguishable over Σ for $k = \Omega(n)$. Thus, Claim 3.9 implies that these k-private circuits achieve a good level of disjunctive resilience without any modification.

3.3 Eliminating selective failure attacks

We illustrate the usefulness of Claim 3.1 with $L = OR$ for eliminating selective failure attacks. While we focus here on a specific application to protecting coding schemes against such attacks, our approach is quite general and can apply to other instances of the selective failure problem.

We start by defining a notion of "fairness" in coding schemes that detect errors in the presence of tampering.

Consider a probabilistic coding scheme $\Pi = (E, D)$ where E encodes an ℓ-bit message x into an m-tuple of symbols $(y_1, \ldots, y_m)$ in Γ^m and D decodes (possibly modified) symbols $(y'_1, \ldots, y'_m)$ into a message x' or a symbol $\perp$ which indicates that an error has been detected. We require that if no tampering occurs, then D outputs the correct message x (with probability 1). For a class $\mathcal{T}$ of tampering attacks $T : \Gamma \to \Gamma$, we say that Π is $(\mathcal{T}, \epsilon)$ *error detecting* if for every $T_1, \ldots, T_m \in \mathcal{T}$ and every message x, we have $\Pr[D(y'_1, \ldots, y'_m) \notin \{x, \perp\}] \leq \epsilon$, where $(y'_1, \ldots, y'_m)$ is obtained by first letting $(y_1, \ldots, y_m) \leftarrow E(x)$ and then letting $y'_i = T_i(y_i)$. For instance, algebraic manipulation detection (AMD) codes [17] can be viewed as a special case of the above definition, where Γ is an Abelian group and $\mathcal{T}$ is the class of functions that add a fixed group element to their input.

The above definition does not rule out a correlation between the event that the decoder D detects an error (by outputting $\perp$) and the message x. Furthermore, such a correlation may depend on the choice of tampering functions T_i. This may be utilized by an attacker to obtain an unfair advantage over honest parties, where the attacker learns partial information about x from the output of the decoder even when the attack causes decoding to fails. Such correlation attacks are harmful in cases where the decoder is uncorrupted and all parties (including the attacker) can only observe its output. This is the case for some natural cryptographic applications of AMD codes (see, e.g., [27]).

To eliminate such correlations, one can impose the following additional *ϵ-fairness* requirement: For every $T_1, \ldots, T_m \in \mathcal{T}$ there is $0 \leq \delta \leq 1$ such that for every message x we have $|\Pr[D(y'_1, \ldots, y'_m) = \perp] - \delta| \leq \epsilon$, where $(y'_1, \ldots, y'_m)$ is as above.

While there are constructions of AMD codes that satisfy the additional fairness requirement with good parameters, it is not clear a-priori that every $(\mathcal{T}, \epsilon)$ error detecting scheme (E, D) can be converted into one that also respects the fairness requirement, let alone with a small loss in efficiency and security.

A natural idea for enforcing fairness is by using an encoding scheme $(\hat{E}, \hat{D})$ defined as follows. To encode a message $x \in \{0,1\}^L$, first apply an (n, k) (string) secret sharing scheme to the message x, resulting in an n-tuple of ℓ-bit shares $(x^1, \ldots, x^n)$, and then apply E independently to each share x^i. To decode, first invoke D on each of the n (possibly modified) m-tuples $(y'^i_1, \ldots, y'^i_m)$. If at least one of the n outputs is $\perp$, output $\perp$; otherwise apply the reconstruction function of the secret sharing scheme to recover x from the n outputs.

Claim 3.10. $(\hat{E}, \hat{D})$ is a $(\mathcal{T}, \epsilon n)$ error detecting scheme with ϵ'-fairness, for $\epsilon' = 2^{-\Omega(k/\sqrt{n})}$.

Proof. First observe that for $\hat{D}$ to output an incorrect message $x' \in \{0,1\}^L$, at least one of the n invocations of D must output an incorrect share. Hence, this probability is bounded by ϵn as required.

We turn to analyze fairness. Given a fixed choice of local tampering functions, let $W(x)$ denote the distribution over $\{0,1\}^n$ obtained by the following experiment: (1) apply $\hat{E}$ to encode the message x; (2) apply to each symbol of the encoding the corresponding tampering function; (3) apply $\hat{D}$ to the corrupted symbols; (4) output $w \in \{0,1\}^n$ such that $w_i = 1$ if the output of the ith instance of D in the invocation of $\hat{D}$ was $\perp$, and $w_i = 0$ otherwise. Note that the probability that $\hat{D}$ outputs $\perp$ on message x is exactly $\Pr[OR(W(x)) = 1]$.

The crucial observation is that for any two messages x, x', the distributions $W(x)$ and $W(x')$ are k-wise indistinguishable. This follows from the k-wise indistinguishability of the shares and the locality of the tampering. Applying Claim 3.1 with $L = OR$ the claim follows. $\square$

Note that by choosing a secret sharing scheme in which each k shares are independent, one could apply a stronger version of Claim 3.1 (see Lemma 5.2 in [14]) that obtains a $2^{-\Omega(k)}$ bound for the case of biased k-wise independence. However, Claim 3.1 applies to an arbitrary secret sharing scheme.

There are quite a few cryptographic protocols that face a similar type of "selective failure" attacks, where the event that some failure is detected may be correlated with secrets that need to be protected. We expect our methodology to be more broadly useful for protecting against such attacks.

Acknowledgements. We thank Daniel Wichs for helpful discussions. Andrej and Emanuele thank Chin Ho Lee for putting them in touch. Emanuele thanks Daniel Wichs for asking whether bounded indistinguishability fools AND, and Mark Bun and Justin Thaler for many discussions about the approximate degree literature.

The first and fourth authors were supported by RGC GRF grants CUHK410113 and CUHK14208215. The second author was supported by ERC starting grant 259426, ISF grant 1709/14, and BSF grant 2012378. Research done in part while visiting the Simons Institute for the Theory of Computing, supported by the Simons Foundation and by the DIMACS/Simons Collaboration in Cryptography through NSF grant #CNS-1523467. Research also supported from a DARPA/ARL SAFEWARE award, NSF Frontier Award 1413955, NSF grants 1228984, 1136174, 1118096, and 1065276. This material is based upon work supported by the Defense Advanced Research Projects Agency through the ARL under Contract W911NF-15-C-0205. The views expressed are those of the author and do not reflect the official policy or position of the Department of Defense, the National Science Foundation, or the U.S. Government. The third author was supported by NSF grant CCF-1319206. Work done in part while a visiting scholar at Harvard University, with support from Salil Vadhan's Simons Investigator grant, and in part while visiting the Simons Institute for the Theory of Computing, supported by the Simons Foundation.

References

[1] S. Aaronson. A counterexample to the Generalized Linial-Nisan Conjecture. Electronic Colloquium on Computational Complexity, Technical Report 109, 2010.

[2] S. Aaronson and Y. Shi. Quantum lower bounds for the collision and the element distinctness problems. *J. of the ACM*, 51(4):595–605, 2004.

[3] M. Ajtai. Approximate counting with uniform constant-depth circuits. In *Advances in computational complexity theory*, pages 1–20. 1993.

[4] N. Alon, J. Bruck, J. Naor, M. Naor, and R. M. Roth. Construction of asymptotically good low-rate error-correcting codes through pseudo-random graphs. *IEEE Transactions on Information Theory*, 38(2):509–516, 1992.

[5] N. Alon, O. Goldreich, and Y. Mansour. Almost k-wise independence versus k-wise independence. *Inf. Process. Lett.*, 88(3):107–110, 2003.

[6] A. Ambainis, A. M. Childs, B. Reichardt, R. Spalek, and S. Zhang. Any AND-OR formula of size N can be evaluated in time $n^{1/2+o(1)}$ on a quantum computer. *SIAM J. on Computing*, 39(6):2513–2530, 2010.

[7] B. Applebaum, Y. Ishai, and E. Kushilevitz. Cryptography in NC^0. *SIAM J. on Computing*, 36(4):845–888, 2006.

[8] L. M. J. Bazzi. Polylogarithmic independence can fool DNF formulas. *SIAM J. Comput.*, 38(6):2220–2272, 2009.

[9] M. Braverman. Polylogarithmic independence fools AC^0 circuits. *J. of the ACM*, 57(5), 2010.

[10] H. Buhrman, I. Newman, H. Röhrig, and R. de Wolf. Robust polynomials and quantum algorithms. *Theory Comput. Syst.*, 40(4):379–395, 2007.

[11] M. Bun and J. Thaler. Dual lower bounds for approximate degree and markov-bernstein inequalities. In Proc. ICALP 2013.

[12] M. Bun and J. Thaler. Dual polynomials for collision and element distinctness, 2015. www.eccc.uni-trier.de/.

[13] I. Cascudo Pueyo, H. Chen, R. Cramer, and C. Xing. Asymptotically good ideal linear secret sharing with strong multiplication over *Any* fixed finite field. In *Proc. of Crypto 2009*, pages 466–486, 2009.

[14] S. Chari, P. Rohatgi, and A. Srinivasan. Improved algorithms via approximations of probability distributions. *J. Comput. System Sci.*, 61(1):81–107, 2000.

[15] S. Cimato, R. D. Prisco, and A. D. Santis. Probabilistic visual cryptography schemes. *The Computer Journal*, 49:97–107, 2006.

[16] R. Cramer, I. B. Damgård, N. Döttling, S. Fehr, and G. Spini. Linear secret sharing schemes from error correcting codes and universal hash functions. In *Proc. of Eurocrypt 2015*, pages 313–336, 2015.

[17] R. Cramer, Y. Dodis, S. Fehr, C. Padró, and D. Wichs. Detection of algebraic manipulation with applications to robust secret sharing and fuzzy extractors. In *Proc. of Eurocrypt 2008*, pages 471–488, 2008.

[18] I. Damgård, Y. Ishai, and M. Krøigaard. Perfectly secure multiparty computation and the computational overhead of cryptography. In *Proc. of Eurocrypt 2010*, pages 445–465, 2010.

[19] I. Damgård, Y. Ishai, M. Krøigaard, J. B. Nielsen, and A. Smith. Scalable Multiparty Computation with Nearly Optimal Work and Resilience. In *Proc. of Crypto 2008*. 2008.

[20] I. Diakonikolas, P. Gopalan, R. Jaiswal, R. A. Servedio, and E. Viola. Bounded independence fools halfspaces. *SIAM J. on Computing*, 39(8):3441–3462, 2010.

[21] I. Diakonikolas, D. Kane, and J. Nelson. Bounded independence fools degree-2 threshold functions. In *Proc. of 51st FOCS*, 2010.

[22] E. Druk and Y. Ishai. Linear-time encodable codes meeting the Gilbert-Varshamov bound and their cryptographic applications. In *Proc. ITCS 2014*, pages 169–182, 2014.

[23] A. Duc, S. Dziembowski, and S. Faust. Unifying leakage models: From probing attacks to noisy leakage. In *Proc. of Eurocrypt 2014*, pages 423–440, 2014.

[24] G. Even, O. Goldreich, M. Luby, N. Nisan, and B. Velickovic. Efficient approximation of product distributions. *Random Struct. Algorithms*, 13(1):1–16, 1998.

[25] S. Faust, T. Rabin, L. Reyzin, E. Tromer, and V. Vaikuntanathan. Protecting circuits from computationally bounded and noisy leakage. *SIAM J. on Computing*, 43(5):1564–1614, 2014.

[26] D. Genkin, Y. Ishai, and A. Polychroniadou. Efficient multi-party computation: From passive to active security via secure SIMD circuits. In *CRYPTO 2015*, pages 721–741, 2015.

[27] S. D. Gordon, Y. Ishai, T. Moran, R. Ostrovsky, and A. Sahai. On complete primitives for fairness. In *Proc. of TCC 2010*, pages 91–108, 2010.

[28] J. Håstad. On the correlation of parity and small-depth circuits. *SIAM J. on Computing*, 43(5):1699–1708, 2014.

[29] Y. Ishai, E. Kushilevitz, X. Li, R. Ostrovsky, M. Prabhakaran, A. Sahai, and D. Zuckerman. Robust pseudorandom generators. In *Proc. of ICALP 2013*, pages 576–588, 2013.

[30] Y. Ishai, E. Kushilevitz, R. Ostrovsky, M. Prabhakaran, and A. Sahai. Efficient non-interactive secure computation. In *Proc. of Eurocrypt 2011*, pages 406–425, 2011.

[31] Y. Ishai, A. Sahai, M. Viderman, and M. Weiss. Zero knowledge LTCs and their applications. In *Proc. of RANDOM 2013*, pages 607–622, 2013.

[32] Y. Ishai, A. Sahai, and D. Wagner. Private circuits: Securing hardware against probing attacks. In *Proc. of Crypto 2003*, pages 463–481, 2003.

[33] J. Kahn, N. Linial, and A. Samorodnitsky. Inclusion-exclusion: Exact and approximate. *Combinatorica*, 16(4):465–477, 1996.

[34] M. Krause and H. Simon. Determining the optimal contrast for secret sharing schemes in visual cryptography. *Combinatorics, Probability & Computing*, 12(3):285–299, 2003.

[35] H. Kuwakado and H. Tanaka. Image size invariant visual cryptography. *IEICE transactions on fundamentals of electronics, communications and computer sciences*, 82(10):2172–2177, 1999.

[36] N. Linial and N. Nisan. Approximate inclusion-exclusion. *Combinatorica*, 10(4):349–365, 1990.

[37] E. Miles and E. Viola. Shielding circuits with groups. In *Proc. of STOC 2013*, 2013.

[38] M. Minsky and S. Papert. *Perceptrons*. MIT Press, Cambridge, MA, 1969.

[39] M. Naor and A. Shamir. Visual cryptography. In *Proc. of Eurocrypt Õ94*, volume 950 of *Lecture Notes in Computer Science*, pages 1–12. Springer Berlin Heidelberg, 1994.

[40] N. Nisan and M. Szegedy. On the degree of Boolean functions as real polynomials. *Computational Complexity*, 4:301–313, 1994.

[41] R. O'Donnell. *Analysis of Boolean Functions*. Cambridge University Press, 2014.

[42] R. O'Donnell and R. A. Servedio. New degree bounds for polynomial threshold functions. *Combinatorica*, 30(3):327–358, 2010.

[43] R. Paturi. On the degree of polynomials that approximate symmetric boolean functions (preliminary version). In *Proc. of STOC Õ92*, pages 468–474, 1992.

[44] H. Randriambololona. Asymptotically good binary linear codes with asymptotically good self-intersection spans. *IEEE Transactions on Information Theory*, 59(5):3038–3045, 2013.

[45] A. A. Razborov. A simple proof of Bazzi's theorem. *ACM Transactions on Computation Theory (TOCT)*, 1(1), 2009.

[46] G. N. Rothblum. How to compute under AC^0 leakage without secure hardware. In *Int. Cryptology Conf. (CRYPTO)*, pages 552–569, 2012.

[47] A. Shamir. How to share a secret. *Commun. ACM*, 22(11):612–613, 1979.

[48] A. A. Sherstov. The pattern matrix method. *SIAM J. on Computing*, 40(6):1969–2000, 2011.

[49] A. A. Sherstov. The power of asymmetry in constant-depth circuits. In *Proc. of FOCS 2015*, 2015.

[50] R. Spalek. A dual polynomial for OR. *CoRR*, abs/0803.4516, 2008.

[51] A. Tal. Tight bounds on The Fourier Spectrum of AC^0. *Electronic Colloquium on Computational Complexity*, Technical Report TR14-174, 2014. www.eccc.uni-trier.de/.

[52] E. Viola. On approximate majority and probabilistic time. *Computational Complexity*, 18(3):337–375, 2009.

[53] E. Viola. The complexity of distributions. *SIAM J. on Computing*, 41(1):191–218, 2012.

[54] D. Zuckerman. Linear degree extractors and the inapproximability of max clique and chromatic number. *Theory of Computing*, 3(1):103–128, 2007.

A Parameters for visual scheme

We demonstrate some specific parameter choices for our visual secret sharing scheme. For given k and α, the corresponding entry in the next table gives the minimum value of n for which an (n, k) bit secret sharing scheme for OR with distinguishing advantage (i.e., contrast) α exists. To compute these exact parameters we formulated the problem as a linear program and used the CVXOPT linear programming solver to perform the calculation. The images were recovered from instantiations of the scheme with parameter settings $k = 8, n = 21$ and $k = 8, n = 46$, respectively.

k	$\alpha = 0.1$	$\alpha = 0.3$
2	3	4
3	4	8
4	7	13
5	9	19
6	13	26
7	16	35
8	21	46
9	26	57

B Useful properties of approximate degree

We rely on the following two lemmas on approximate degree. The first lemma (cf. [20, Claim 3.8]) shows that approximation quality can be traded for degree.

Lemma B.1. *Let $0 < \epsilon' < \epsilon \leq 1/3$. Suppose that the ϵ-approximate degree of f is k. Then the ϵ'-approximate degree of f is $O(k \cdot \log \frac{\epsilon}{\epsilon'})$.*

The second lemma relates the approximate degree of the parity of t functions to a bound on their approximate degree. It follows by composing the functions using a "robust" polynomial for parity of degree $O(t)$ [10]. A simpler bound, obtained by applying Lemma B.1 and multiplying the t approximations, adds an additional $\log t$-factor to the degree.

Lemma B.2. *Let $f_1, f_2, \ldots, f_t$ be boolean functions whose $1/3$-approximate degree is at most k. Then the $1/3$-approximate degree of $f = f_1 \oplus f_2 \oplus \cdots \oplus f_t$ is $O(kt)$.*

C Sharing in AC^0 with perfect secrecy

In this section we describe ways to maintain perfect secrecy while still generating the shares in AC^0. Let p be a distribution over $\{0,1\}^n$. We say that a distribution q over $\{0,1\}^n \cup \{\perp\}$ is ϵ-near p if $\Pr[q = \perp] \leq \epsilon$ and p equals $q|q \neq \perp$, i.e., q conditioned on the event $q \neq \perp$. We think of '$\perp$' as failure and we generally use the word 'near' to indicate sampling with failure.

Theorem C.1. *For schemes 1. to 4. in Corollary 2.2 the following holds. Let μ and ν be the distributions on $\{0,1\}^n$ of the shares of 0 and 1 respectively. Let c be an integer. There exists explicit AC^0 circuits of size polynomial in n that sample distributions $\mu_\perp$ and $\nu_\perp$ such that:*

1. *(Secrecy) If μ and ν are k-wise indistinguishable then so are $\mu_\perp$ and $\nu_\perp$.*
2. *(Reconstruction) $\mu_\perp$ and $\nu_\perp$ are ϵ-near μ and ν, respectively, for $\epsilon = 2^{-n^c}$.*

By Item 1. we achieve perfect secrecy, and Item 2. guarantees that reconstruction works up to a small error.

Proof. (of Theorem C.1) For simplicity let us consider the scheme for OR. As mentioned earlier, in this case μ and ν are symmetric. Let μ' and ν' be the corresponding distributions on Hamming weights. By inspection of the dual polynomial for OR, see [50], the probability mass functions of μ' and ν' is at any point a multiple of $1/m$, where m is an integer with $\mathrm{poly}(n)$ bits.

We now describe the near sampler for μ. First, pick a uniform number in $\{0,1\}^{n'}$ where $n' \geq m$. If the number is bigger than m then output $\perp$. Otherwise, use that to compute a sample i of μ'. This involves computing '$\leq$', which can be done in AC^0. Then the task is to output a uniform string of Hamming weight i. Because AC^0 can nearly sample the uniform distribution of permutations of $[n]$, cf. [53], this uniform string can indeed be sampled. The same process is applied to ν.

Conditioned on not failing, the process is sampling μ and ν as desired. What remains to be seen is that the probability of outputting $\perp$ does not depend on whether we are nearly sampling μ or ν. This holds by inspection. Indeed, in the first step we fail in either case if we obtain a number that is larger than m. In the second the failure probability is that of the sampler of a uniform permutation, which is independent of which distribution we are sampling. □

D Exact vs. almost bounded indistinguishability

In this Appendix we show that k-wise indistinguishability is "robust to noise" in the following sense: Any pair of distributions that are "almost" k-wise indistinguishable is close to a pair of truly k-wise indistinguishable distributions. Alon, Goldreich, and Mansour proved an analogous statement for k-wise independence (Theorem 2.1 in [5]).

Theorem D.1. *Let μ and ν be two distributions on $\{-1,1\}^n$. Suppose that no test $T : \{-1,1\}^k \to \{0,1\}$ on k bits can distinguish μ and ν with advantage bigger than ϵ. Then there exist two distributions μ^* and ν^* such that μ^* has statistical distance $\leq 2\epsilon n^k$ from μ, ν^* has statistical distance $\leq 2\epsilon n^k$ from ν, and μ^* and ν^* are k-wise indistinguishable.*

Proof. For a subset I of $[n]$ let $\chi_I : \{-1,1\}^n \to \{-1,1\}$ be $\chi_I(x) = \prod_{i \in I} x_i$. It suffices to prove the conclusion for the tests χ_I where $|I| \leq k$. This is because if $\sum_x (\mu'(x) - \nu'(x))T(x) \geq \alpha$, then writing T in Fourier expansion we have $\sum_I \hat{T}_I \sum_x (\mu'(x) - \nu'(x))\chi_I(x) \geq \alpha$, and so there exists a test χ_I giving advantage at least $\alpha/2^k$.

For a function $f : \{-1,1\}^n \to \mathbb{R}$ we write $[f, I]$ for $\sum_x f(x)\chi_I(x)$, and call it the I coefficient of f. We "adjust" the coefficients of μ and ν by repeating the following step. Let $I \subseteq [n]$ be a non-empty subset of size at most k. By hypothesis, $|[\mu - \nu, I]| = \alpha \leq \epsilon$. Without loss of generality let $[\mu, I] \leq [\nu, I]$. Set $\mu' := \mu + \alpha(\chi_I + 1)/2^n$, and $\nu' := \nu + \alpha/2^n$. Now we have $[\mu' - \nu', I] = 0$, while $[\mu' - \nu', J] = [\mu - \nu, J]$ for $J \neq I$. Moreover, note that $\sum_x |\mu'(x)| = \sum_x |\mu(x)| + \alpha$, $\sum_x |\mu(x) - \mu'(x)| = \alpha$, and that the same holds for ν.

Repeating the adjustment $\leq n^k$ times, we get two non-negative functions μ' and ν' such that $[\mu' - \nu', I] = 0$ for every I of size at most k, and $\sum_x |\mu(x) - \mu'(x)| \leq \epsilon n^k$, and the same for ν', and also $\sum_x |\mu'(x)| = \sum_x |\nu'(x)| = 1 + \sigma$, for some $0 \leq \sigma \leq \epsilon n^k$.

Finally, let $\mu^* = \mu/(1+\sigma)$ and $\nu^* = \nu/(1+\sigma)$. We have $[\mu^* - \nu^*, I] = 0$ for every I of size at most k. The distance of $\mu*$ from μ is $\leq (1+\sigma)^{-1}(\sum_x |\mu(x) - \mu^*(x)| + \sigma \sum_x \mu(x)) \leq 2\epsilon n^k$, and the same for ν. □