Secret Sharing with Binary Shares

Fuchun Lin[*] Mahdi Cheraghchi[†] Venkatesan Guruswami[‡]
Reihaneh Safavi-Naini[§] Huaxiong Wang[*]

September 8, 2018

Abstract

Shamir's celebrated secret sharing scheme provides an efficient method for encoding a secret of arbitrary length ℓ among any $N \leq 2^\ell$ players such that for a threshold parameter t, (i) the knowledge of any t shares does not reveal any information about the secret and, (ii) any choice of $t+1$ shares fully reveals the secret. It is known that any such threshold secret sharing scheme necessarily requires shares of length ℓ, and in this sense Shamir's scheme is optimal. The relaxed notion of ramp schemes requires the reconstruction of secret from any $t+1+g$ shares, for a gap parameter $g > 0$. Ramp secret sharing is possible with share lengths depending only on the gap ratio g/N.

In this work, we study secret sharing in the extremal case of bit-long shares, where even ramp secret sharing becomes impossible. We show, however, that a slightly relaxed but equally effective notion of semantic security for the secret, and negligible reconstruction error probability, eliminates the impossibility. Moreover, we provide explicit constructions of such schemes. Our relaxation results in separation of adaptive and non-adaptive adversaries. For non-adaptive adversaries, we explicitly construct secret sharing schemes that provide secrecy against any τ fraction of observed shares, and reconstruction from any κ fraction of shares, for any choices of $0 \leq \tau < \kappa \leq 1$. Our construction achieves secret length $N(\kappa - \tau - o(1))$ which we show to be optimal. Finally, we construct explicit schemes against adaptive adversaries attaining a secret length $\Omega(N(\kappa - \tau))$. Our work makes a new connection between secret sharing and coding theory, this time wiretap codes, that was not known before, and raises new interesting open questions.

1 Introduction

Secret sharing, introduced independently by Blakley [3] and Shamir [19], is one of the most fundamental cryptographic primitives with far-reaching applications; e.g., a major tool in secure multiparty computation (cf. [10]). The general goal in secret sharing is to encode a secret s into a number of *shares* $\mathsf{X}_1, \ldots, \mathsf{X}_N$ that are distributed among N players such that only certain *authorized subsets* of the players can reconstruct the secret. An *authorized* subset of players is a set $A \subseteq [N]$ such that the shares with indices in A can collectively be used to reconstruct the secret s (*perfect reconstructiblity*). On the other hand, A is an *unauthorized* subset if the knowledge of

[*] Division of Mathematical Sciences, School of Physical and Mathematical Sciences, Nanyang Technological University, SG

[†] Department of Computing, Imperial College London, UK

[‡] Computer Science Department, Carnegie Mellon University, USA

[§] Department of Computer Science, University of Calgary, CA

the shares with indices in A reveals no information about the secret (*perfect privacy*). The set of authorized and unauthorized sets define an access structure, of which the most widely used is the so-called *threshold structure.* A secret sharing scheme with threshold access structure, is defined with respect to an integer parameter t and satisfies the following property: Any set $A \subseteq [N]$ with $|A| \leq t$ is an unauthorized set. That is, the knowledge of any t shares or fewer does not reveal any information about the secret. On the other hand, any set A with $|A| > t$ is an authorized set; namely, the knowledge of any $t+1$ or more shares completely reveals the secret.

Shamir's secret sharing scheme [19] gives an elegant construction for threshold schemes that can be interpreted as the use of Reed-Solomon codes for encoding the secret. Suppose the secret s is an ℓ-bit string and $N \leq 2^{\ell}$. Then, Shamir's scheme encodes the secret as an element of the finite field $\mathbb{F}_q$, where $q = 2^{\ell}$, padded with t uniformly random and independent elements from the same field. The resulting vector over $\mathbb{F}_q^{t+1}$ is then encoded using a Reed-Solomon code of length N, providing N shares of length ℓ bits each. The fact that a Reed-Solomon code is MDS can then be used to show that the threshold guarantee is satisfied.

Remarkably, Shamir's scheme is optimal for threshold secret sharing in the following sense: Any threshold secret sharing scheme sharing ℓ-bit secrets necessarily requires shares of length at least ℓ, and Shamir's scheme attains this lower bound [21].

It is natural to ask whether secret sharing is possible at share lengths below the secret length. Of course, in this case, the threshold guarantee that requires all subsets of participants be either authorized, or unauthorized, can no longer be attained. Instead, the notion can be relaxed to *ramp* secret sharing which allows some subset of participants to learn some information about the secret. A ramp scheme is defined with respect to two threshold parameters, t and $k > t+1$. As in threshold scheme, the knowledge of any t shares or fewer does not reveal any information about the secret. On the other hand, any k shares can be used to reconstruct the secret. The subsets of size $\geq t+1$ or $\leq k-1$, may learn some information about the secret. Ideally, one would like to obtain as small a choice of k as possible for a given t. We consider a family of secret sharing schemes with relative privacy and reconstructibility parameters $\tau := t/N$ and $\kappa := k/N$, respectively, for constants $0 \leq \tau < \kappa \leq 1$, and wish to maintain a small relative gap $\gamma := \kappa - \tau$. There is no constraint on the secret length and the number of players, both can grow to infinity. Ideally, for any arbitrarily small relative gap γ, we wish to explicitly construct such family of schemes with a fixed share size q.

It is straightforward to verify via using Reed-Solomon code interpretation of Shamir's approach applied to a random linear code that, for every fixed gap γ, there is a constant q only depending on γ such that a ramp secret sharing scheme with share size q exists. Such schemes can actually be constructed by using explicit algebraic geometry codes instead of random linear codes. In fact, this dependence of share size q on relative gap γ is inherent for ramp schemes. It is shown in [5] that for a ramp scheme with constant share size q, threshold gap g, privacy threshold t and unconstrained number of players N, the following always holds.

$$g \geq (N - t + 1)/q. \tag{1}$$

This means that for $t = \tau N$, the relative gap $\gamma = g/N$ should satisfy $\gamma \geq (1 - \tau + 1/N)/q$. This completely rules out the possibility of achieving our goal using ramp schemes.

Our results in this work is to show that an equally effective guarantee can be ensured by a slight relaxation of privacy and reconstructbility definitions. To circumvent the impossibility of ramp schemes with fixed share size and arbitrarily small relative gap γ, we allow slightly *imperfect privacy* and a negligible *reconstruction error.* That is, for negligible leakage error ε and reconstruction error δ, we relax the scheme to satisfy the following: The secret can be reconstructed correctly from any

k shares with probability $1 - \delta$ over the randomness of the encoder. Furthermore, any set of t shares only reveal a negligible amount of information about the secret, measured with respect to the chosen parameter ε. More precisely, semantic security holds in the following sense: For any two different choices of the secret, the distribution of the content of the t shares revealed to an adversary is ε-close in statistical (total variation) distance.

1.1 Related work

Ramp secret sharing with constant share lengths has been studied as an extension of Shamir's original scheme utilizing general families of algebraic geometry codes instead of Reed-Solomon code (see [6] and the extension to robust secret sharing in [11, 8]). However, although this approach achieves constant share lengths, the share length depends on the ramp gap $\kappa - \tau$ and cannot be lowered to an absolute constant such as a single bit [5]

In [7], binary secret sharing with adaptive and non-adaptive adversaries, as well as their robust analogue where the adversary can tamper with shares are considered. The goal there is to relax privacy to achieve large secrets over small alphabets. However the paper considers only a privacy threshold, and reconstruction is from the full share set (and not a reconstruction threshold (instead of $N\kappa$ shares). Similar to our work, they also need to consider adaptive adversary ([7, Theorem 1.2]).

Perhaps the closest notion to binary secret sharing schemes is that of *wiretap codes*, first studied in information theory. In the basic wiretap channel model of Wyner and its extension to broadcast channel with confidential messages [23, 13], there is a point-to-point *direct* channel between a sender and a receiver that has partial leakage to the adversary, and the leakage of information is modelled by a second point-to-point *wiretapper* channel between the sender and the adversary. To goal of the sender is to encode messages in such a way that the receiver can decode them, while the adversary does not learn much about them [23]. The highest information rate of wiretap codes is called the *secrecy capacity.* A Binary Erasure Channel (BEC_p) is a probabilistic transformation that maps inputs 0 and 1 to a symbol ? that denotes erasure with probability p and to the inputs themselves with probability $1-p$. For a pair (BEC_{p_m},BEC_{p_w}) of BEC's, such that $p_m < p_w$, it is known that the secrecy capacity is the difference of the respective channel capacities: $(1-p_m)-(1-p_w) = p_w - p_m$.

It is important to note the distinctions between the erasure wiretap model above, and binary secret sharing. First, as is customary in information theory, the guarantees of wiretap codes are required to only hold for random messages, whereas in secret sharing, the cryptographic convention of security for worst case messages is required. Second, in the standard wiretap model, the notion of secrecy is information-theoretic as well and typically measured in terms of the mutual information measure. Namely, for a random ℓ-bit message M, and letting W denote the information delivered to the adversary (with randomness of W depending on the randomness of M, the two channels, as well as the internal randomness of the encoder), information-secrecy is satisfied in the weak (resp., strong) sense if $\mathsf{I}(\mathsf{M};\mathsf{W}) \leq \varepsilon\ell$ (resp., $\mathsf{I}(\mathsf{M};\mathsf{W}) \leq \varepsilon$) for an arbitrarily small constant ε. For secret sharing, on the other hand, either perfect secrecy or semantic secrecy (negligible leakage with respect to statistical distance) is a requirement.

The notion of privacy in wiretap codes has evolved over years. More recently the strong notion of semantic security for wiretap model has been introduced [2], which allows arbitrary message distribution (including the one chosen by the adversary) and is shown to be equivalent to negligible leakage with respect to statistical distance.

There remains one distinction between semantically secure wiretap model and binary secret sharing. That is the nature of the direct and wiretap channels are typically stochastic (e.g., the erasure channel with random i.i.d. erasures), whereas for secret sharing a worst-case guarantee for the erasure patterns is also required. Namely, in secret sharing, reconstruction is required for every choice of k or more shares (albeit independent of the randomness of the encoder). Furthermore, privacy of the secret is required for every (adaptive or non-adaptive) choice of the t shares delivered to the adversary.

1.2 Our contributions

We motivate the study of secret sharing scheme with constant share lengths, and study the extremal case of binary shares. Our goal is to show that even in this extremely restrictive case, a slight relaxation of the security notion of ramp secret sharing guarantees construction of families of ramp schemes with constant relative privacy and reconstruction thresholds. Namely, for any constants $0 \leq \tau < \kappa \leq 1$, it can be guaranteed that any τN or fewer shares reveal essentially no information about the secret, whereas any κN or more shares can reconstruct the exact secret with a negligible failure probability (over the internal randomness of the encoder). While we only focus on the extremal special case $q = 2$ in this presentation, all our results can be extended to any constant q (see Section 6).

Note that secret sharing with constant-length shares necessarily imposes certain restrictions that are not common in standard threshold secret sharing. Unlike secret sharing with share length dependent on the secret (for threshold) or gap (for ramp schemes), binary sharing of an ℓ-bit secret obviously requires at least ℓ shares to capture the secret information. For a family of ramp secret sharing schemes with fixed thresholds and fixed length share, as N grows the absolute gap length $(\kappa - \tau)N$ grows, and the length of the secret is expected to grow and so the notion of the ratio $\ell/N \in (0, 1]$ becomes a key parameter of interest for the family. As is customary in coding theory (we then call it *coding rate*), it is desired to characterize the maximum possible rate of a binary secret sharing scheme. This will give the minimum number of shares necessary for encoding an ℓ-bit secret.

We consider binary sharing of an ℓ-bit secret and for this work focus on the asymptotic case where the length ℓ (equivalently, the number of players N) is sufficiently large. For given constant relative thresholds $0 \leq \tau < \kappa \leq 1$, we allow the adversary to learn a τ fraction of the shares and require reconstruction of the secret from any κ fraction of the shares (up to a negligible probability of reconstruction error over the randomness of the encoder).

We replace perfect privacy with semantic secrecy, the strongest cryptographic notion of security. That is, for any two secrets (possibly chosen by the adversary), we require the adversary's view (as a random variable depending on the randomness of the encoder) to be statistically indistinguishable. The notion of indistinguishability that we use is total variation (statistical) distance bounded by a (negligible) leakage parameter $\varepsilon > 0$. Using non-perfect privacy creates a distinction between non-adaptive and adaptive secrecy. A non-adaptive adversary chooses any τ fraction of the N players at once, and receives their corresponding shares. An adaptive adversary however, selects share holders one by one, receives their shares and uses its available information to make its next choice. When $\varepsilon = 0$; i.e., when perfect privacy holds, non-adaptive secrecy automatically implies adaptive secrecy as well. However, this is not necessarily the case when $\varepsilon > 0$ and we thus study the two cases separately.

We use the relation between a binary secret sharing family with relative threshold (τ, κ) and codes for a Wyner wiretap channel with two binary erasure channels to derive a coding rate upper bound of $\kappa - \tau$ for binary secret sharing (see Lemma 11).

Our main technical contribution is explicit constructions of binary secret sharing schemes in both the non-adaptive and adaptive models, and proving optimality of non-adaptive construction. Namely, we prove the following:

Theorem 1 (informal summary of Corollary 22, Corollary 18, and Lemma 11)**.** For any choice of $0 \leq \tau < \kappa \leq 1$, and large enough N, there is an explicit construction of a binary secret sharing scheme with N players that provides (adaptive or non-adaptive) privacy against leakage of any τN or fewer shares, as well as reconstruction from any κN or more of the shares (achieving semantic secrecy with negligible error and negligible reconstruction error). For non-adaptive secrecy, the scheme encodes shares of length $\ell \geq (\kappa - \tau - o(1))N$, which is asymptotically optimal. For adaptive secrecy, the scheme attains a coding rate of $\Omega(\kappa - \tau)$.

As a side contribution, our findings unify the Wyner wiretap model and its adversarial analogue. Ozarow and Wyner proposed the wiretap channel II, where an adversary observes arbitrary t out of the total N bits of the communication [17]. This is the adversarial analogue of a binary erasure channel with erasure probability $(N - t)/N$. Our capacity-achieving construction of binary secret sharing for non-adaptive adversaries implies that the secrecy capacity of the adversarial analogue of erasure Wyner wiretap channel is similarly characterized the erasure ratios of the two channels. Moreover the capacity is achievable with semantic security.

This answers an open question posted in [1]. The authors studied a generalisation of the wiretap II model, where the adversary chooses t bits to observe and erases them. They showed that the rate $1 - \tau - h_2(\tau)$ (while the quantity is non-negative) can be achieved and left open the question of whether a higher rate is achievable. Our result specialized to their setting shows that, the rate $1 - 2\tau$ can be explicitly achieved.

1.3 Our approach and techniques

Our explicit constructions follow the paradigm of invertible randomness extractors formalized in [9]. Invertible extractors were used in [9] for explicit construction of optimal wiretap coding schemes in the *Wiretap II* model of Ozarow and Wyner [17]. This corresponds to the special case of the erasure wiretap problem in which the direct channel is noiseless (i.e., reconstruction is only required when all shares are available).

As in [9], we rely on invertible affine extractors as our primary technical tool. Such an extractor is an explicit function $\mathsf{AExt}\colon \{0,1\}^n \to \{0,1\}^m$ such that, for any random variable X uniformly distributed over an unknown k-dimensional subspace of $\mathbb{F}_2^n$, the distribution of $\mathsf{AExt}(\mathsf{X})$ is close to uniform (in statistical distance). Furthermore, the invertibility guarantee provides an efficient algorithm for (approximately) sampling a uniform element from $\mathsf{AExt}^{-1}(\mathsf{s})$ for any given output s.

It is then natural to consider the affine extractor's uniform inverter as a candidate encoder for a secret sharing scheme. Intuitively, if the message s is chosen uniformly at random, we have the guarantee that for any choice of a bounded number of encoded bits revealed to the adversary, the extractor's output (i.e., the secret s) remains uniform (and thus unaffected in distribution) given the information revealed to the adversary. Consequently, secrecy should at least hold in an information-theoretic sense. This is what was formalized and used in [9] for the construction of Wiretap II codes.

For non-adaptive adversaries, in fact it is possible to use invertible *seeded extractors* rather than affine extractors described in the above construction. Recall that a (strong) seeded extractor assumes, in addition to the main input, an independent seed as an auxiliary input and ensures uniformity of the output for most fixings of the seed. The secret sharing encoder appends a randomly chosen seed to the encoding and inverts the extractor with respect to the chosen seed. Then, the above argument would still hold even if the seed is completely revealed to the adversary.

It is important to note that the adversary is non-adaptive. An adaptive adversary can read the seed first and choose the rest of the reading positions according to it. In this case, the extractor seed and the extractor source induced by the chosen positions are not independent, violating the seeded extractor definition.

The interest in the use of seeded, as opposed to seedless affine, extractors is twofold. First, nearly optimal and very efficient constructions of seeded extractors are known in the literature that extract nearly the entire source entropy with only a short seed. This allows us to attain nearly optimal rates for the non-adaptive case. Furthermore, and crucially, such nearly optimal extractor constructions (in particular, Trevisan's extractor [22, 18]) can in fact be linear functions for every fixed choice of the seed (in contrast, seedless affine extractors can never be linear functions). We take advantage of the linearity of the extractor in a crucial way and use a rather delicate analysis to show that in fact the linearity of the extractor can be utilized to prove that the resulting secret sharing scheme provides the stringent worst-case secret guarantee which is a key requirement distinguishing secret sharing schemes (a cryptographic primitive) from wiretap codes (an information-theoretic notion).

In order to provide reconstructibility from a subset (of size k) of the shares, we naturally compose the encoding obtained from the extractor's inversion routine with a linear erasure-correcting code. The linearity of the code ensures that the extractor's input subject to the adversary's observation (which now can consist of linear combinations of the original encoding) remains uniform on some affine space, thus preserving the privacy guarantee. Furthermore, the linearity of the code crucially ensures that the privacy guarantee with respect to worst-case messages can be preserved as well.

However, since by the known rate-distance trade-offs of binary error-correcting codes, no deterministic coding scheme can correct more than a $1/2$ fraction of erasures (a constraint that would limit the choice of κ), the relaxed notion of *stochastic coding schemes* is necessary for us to allow reconstruction for all choices of $\kappa \in (0, 1]$. Intuitively, a stochastic code is a randomized encoder with a deterministic decoder, that allows the required fraction of errors to be corrected. We utilize what we call a *stochastic affine* code. Such codes are equipped with encoders that are affine functions of the message for every fixing of the encoder's internal randomness. We show that such codes are as suitable as deterministic linear codes for providing the linearity properties that that our construction needs.

To construct capacity-achieving stochastic affine erasure codes, i.e., those that correct every $1 - \kappa$ fraction of erasures at asymptotic rate κ, we utilize a construction of stochastic codes due to Guruswami and Smith [15] for bit-flip errors. We observe that this construction can be modified to yield capacity-achieving erasure codes. Roughly speaking, this is achieved by taking an explicit linear code in the Shannon erasure model (against independent erasures) and pseudo-randomly shuffling the codeword positions. Combined with a delicate encoding of hidden "control information" to communicate the choice of the permutation to the decoder in a robust manner, the construction transforms robustness against random erasures to worst-case erasures at the cost of making the encoder randomized.

Our binary secret sharing schemes with adaptive security is similarly constructed by a direct composition of an invertible (seedless) affine extractor with an affine stochastic erasure correcting code. We prove that such a composition guarantees both security for worst-case messages and against adaptive adversaries if the affine extractor provides the strong guarantee of having a nearly uniform output with respect to the ℓ_∞ measure rather than ℓ_1 (statistical distance). However, this comes at the cost of the extractor not being able to extract the entire entropy of the source, leading to ramp secret sharing schemes with slightly sub-optimal rates, albeit still achieving rates within a constant factor of the optimum, and therefore, $\Omega(N)$ share lengths. As a proof of concept, we utilize a simple padding and truncation technique to convert any off-the-shelf seedless affine extractor (such as those of Bourgain [4] or Li [16]) to one that satisfies the stronger uniformity condition that we require. Altogether, this results in an explicit construction of adaptively-secure binary secret sharing schemes for sharing $\Omega(N)$ bits of secret among N parties, and for all choice of the parameters $0 \leq \tau < \kappa \leq 1$.

1.4 Organization of the paper

Section 2 contains brief introduction to the two building blocks for our constructions. In Section 3, we formally define the binary secret sharing model and prove a coding rate upper bound. Section 4 contains a capacity-achieving construction with privacy against non-adaptive adversaries. Section 5 contains a constant rate construction with privacy against adaptive adversaries. We conclude the paper and discuss open problems in Section 6.

2 Preliminaries and definitions

In this section, we review the necessary facts and results about randomness extractors, both the seeded and seedless affine variants, as well as the stochastic erasure correcting codes.

Randomness extractors. Randomness extractors extract close to uniform bits from input sequences that are not uniform but have some guaranteed entropy. The closeness to uniform of the extractor output is measured by the statistical distance (half the ℓ_1-norm). For two random variables $\mathsf{X}, \mathsf{Y} \leftarrow \Omega$, the statistical distance between X and Y (or their distributions) is defined as,

$$\mathsf{SD}(\mathsf{X};\mathsf{Y}) = \frac{1}{2}\sum_{\omega\in\Omega}\left|\mathsf{Pr}(\mathsf{X}=\omega) - \mathsf{Pr}(\mathsf{Y}=\omega)\right|.$$

We say X and Y are ε-close if $\mathsf{SD}(\mathsf{X},\mathsf{Y}) \leq \varepsilon$. A *randomness source* is a random variable with lower bound on its min-entropy, which is defined by $\mathsf{H}_\infty(\mathsf{X}) = -\log\max_{\mathsf{x}}\{\mathsf{Pr}[\mathsf{X}=\mathsf{x}]\}$. We say a random variable $\mathsf{X} \leftarrow \{0,1\}^n$ is a (n,k)*-source* if $\mathsf{H}_\infty(\mathsf{X}) \geq k$.

For well structured sources, there exist deterministic functions that can extract close to uniform bits. An *affine* (n,k)-source is a random variable that is uniformly distributed on an affine translation of some k-dimensional sub-space of $\{0,1\}^n$. Let U_m denote the random variable uniformly distributed over $\{0,1\}^m$.

Definition 2. A function $\mathsf{AExt}\colon \{0,1\}^n \to \{0,1\}^m$ is an affine (k,ε)-extractor if for any affine (n,k)-source X, we have

$$\mathsf{SD}(\mathsf{AExt}(\mathsf{X});\mathsf{U}_m) \leq \varepsilon.$$

For general (n,k)-sources, there does not exist a deterministic function that can extract close to uniform bits from all of them simultaneously. A family of deterministic functions are needed.

Definition 3. A function $\mathsf{Ext}\colon \{0,1\}^d \times \{0,1\}^n \to \{0,1\}^m$ is a strong seeded (k,ε)-extractor if for any (n,k)-source X, we have

$$\mathsf{SD}(\mathsf{S},\mathsf{Ext}(\mathsf{S},\mathsf{X});\mathsf{S},\mathsf{U}_m) \leq \varepsilon,$$

where S is chosen uniformly from $\{0,1\}^d$. A seeded extractor $\mathsf{Ext}(\cdot,\cdot)$ is called linear if for any fixed seed $\mathsf{S}=\mathsf{s}$, the function $\mathsf{Ext}(\mathsf{s},\cdot)$ is a linear function.

We will use Trevisan's extractor [22] in our first construction. In particular, we use the following improvement of this extractor due to Raz, Reingold and Vadhan [18].

Lemma 4 ([18])**.** There is an explicit linear strong seeded (k, ε)-extractor $\mathsf{Ext}\colon \{0,1\}^d \times \{0,1\}^n \to \{0,1\}^\ell$ with $d = O(\log^3(n/\varepsilon))$ and $\ell = k - O(d)$.

We will use Bourgain's affine extractor in our second construction[1].

Lemma 5 ([4])**.** For every constant $0 < \mu \leq 1$, there is an explicit affine extractor $\mathsf{AExt}\colon \{0,1\}^n \to \{0,1\}^m$ for sources with min-entropy $n\mu$ with output length $m = \Omega(n)$ and error at most $2^{-\Omega(n)}$.

Explicit constructions of randomness extractors have efficient forward direction of extraction. In some applications, we usually need to efficiently invert the process: Given an extractor output, sample a random pre-image.

Definition 6 ([9])**.** Let f be a mapping from $\{0,1\}^n$ to $\{0,1\}^m$. For $v \geq 0$, a function $\mathsf{Inv}\colon \{0,1\}^m \times \{0,1\}^r \to \{0,1\}^n$ is called a v-inverter for f if the following conditions hold:

- (Inversion) Given $\mathsf{y} \in \{0,1\}^m$ such that its pre-image $f^{-1}(\mathsf{y})$ is nonempty, for every $\mathsf{r} \in \{0,1\}^r$ we have $f(\mathsf{Inv}(\mathsf{y}, \mathsf{r})) = \mathsf{y}$.
- (Uniformity) $\mathsf{Inv}(\mathsf{U}_m, \mathsf{U}_r)$ is v-close to U_n.

A v-inverter is called efficient if there is a randomized algorithm that runs in worst-case polynomial time and, given $\mathsf{y} \in \{0,1\}^m$ and r as a random seed, computes $\mathsf{Inv}(\mathsf{y}, \mathsf{r})$. We call a mapping v-invertible if it has an efficient v-inverter, and drop the prefix v from the notation when it is zero. We abuse the notation and denote the inverter of f by f^{-1}.

Stochastic codes. A stochastic code (also known as a *coding scheme* in cryptography literature) has a randomised encoder and a deterministic decoder. The encoder $\mathsf{Enc}\colon \{0,1\}^k \times \mathcal{R} \to \{0,1\}^n$ uses local randomness $\mathsf{R} \leftarrow \mathcal{R}$ to encode a message $\mathsf{m} \in \{0,1\}^k$. The decoder is a deterministic function $\mathsf{Dec}\colon \{0,1\}^n \to \{0,1\}^k \cup \{\perp\}$. The decoding probability is defined over the encoding randomness $\mathsf{R} \leftarrow \mathcal{R}$. Stochastic codes are known to explicitly achieve the capacity of some adversarial channels [15].

Affine source plays an important role in our constructions. We define a general requirement for the stochastic code used in the outer layer that facilitates an affine distribution of its random message.

Definition 7 (Stochastic Affine codes)**.** Let $\mathsf{Enc}\colon \{0,1\}^m \times \mathcal{R} \to \{0,1\}^n$ be the encoder of a stochastic code. We say it is a stochastic affine code if for any $\mathsf{r} \in \mathcal{R}$, the encoding function $\mathsf{Enc}(\cdot, \mathsf{r})$ specified by r is an affine function of the message. That is we have

$$\mathsf{Enc}(\mathsf{m}, \mathsf{r}) = \mathsf{m}G_{\mathsf{r}} + \Delta_{\mathsf{r}},$$

where $G_{\mathsf{r}} \in \{0,1\}^{m \times n}$ and $\Delta_{\mathsf{r}} \in \{0,1\}^n$ are determined by the code and the randomness r.

We then adapt a construction in [15] to obtain the following capacity-achieving Stochastic Affine-Erasure Correcting Code (SA-ECC). In particular, we show for any $p \in [0,1)$, there is an explicit stochastic affine code that corrects p fraction of adversarial erasures and achieves the rate $1 - p$ (see Appendix A for more details).

[1] We note, however, that we could have used other explicit extractors for this purpose, such as [16].

Lemma 8 (Adapted from [15])**.** For every $p \in [0, 1)$, and every $\xi > 0$, there is an efficiently encodable and decodable stochastic affine code $(\mathsf{Enc}, \mathsf{Dec})$ with rate $R = 1 - p - \xi$ such that for every $\mathsf{m} \in \{0, 1\}^{NR}$ and erasure pattern of at most p fraction, we have $\Pr[\mathsf{Dec}(\widetilde{\mathsf{Enc}(\mathsf{m})}) = \mathsf{m}] \geq 1 - \exp(-\Omega(\xi^2 N / \log^2 N))$, where $\widetilde{\mathsf{Enc}(\mathsf{m})}$ denotes the partially erased random codeword and N denotes the length of the codeword.

3 Binary secret sharing schemes

In this section, we define our model of nearly-threshold binary secret sharing schemes. We begin with a description of the two models of non-adaptive and adaptive adversaries which can access up to t of the N shares.

A *leakage oracle* is a machine $\mathcal{O}(\cdot)$ that takes as input an N-bit string $\mathbf{c} \in \{0, 1\}^N$ and then answers the *leakage queries* of the type I_j, for $I_j \subset [N]$, $j = 1, 2, \ldots, q$. Each query I_j is answered with $\mathbf{c}_{I_j}$. An interactive machine $\mathcal{A}$ that issues the leakage queries is called a *leakage adversary.* Let $A_{\mathbf{c}} = \cup_{j=1}^{q} I_j$ denote the union of all the index sets chosen by $\mathcal{A}$ when the oracle input is $\mathbf{c}$. The oracle is called t-bounded, denoted by $\mathcal{O}_t(\cdot)$, if it rejects leakage queries from $\mathcal{A}$ if there exists some $\mathbf{c} \in \{0, 1\}^N$ such that $|A_{\mathbf{c}}| > t$. An adaptive leakage adversary decides the index set I_{j+1} according to the oracle's answers to all previous queries $I_1, \ldots, I_j$. A non-adaptive leakage adversary has to decide the index set $A_{\mathbf{c}}$ before any information about $\mathbf{c}$ is given. This means that for a non-adaptive adversary, given any oracle input $\mathbf{c} \in \{0, 1\}^N$, we always have $A_{\mathbf{c}} = A$ for some $A \subset [N]$. Let $\mathsf{View}_{\mathcal{A}}^{\mathcal{O}_t(\cdot)}$ denote the view of the leakage adversary $\mathcal{A}$ interacting with a t-bounded leakage oracle. When $\mathcal{A}$ is non-adaptive, we use the shorthand $\mathsf{View}_{\mathcal{A}}^{\mathcal{O}_t(\cdot)} = (\cdot)_A$, for some $A \subset [N]$ of size $|A| \leq t$. A concrete example showing the distinction between these two types of adversaries is our construction in Section 4. It is semantically secure against a non-adaptive adversary but fails against an adaptive adversary (see Remark 17).

Definition 9. For any $0 \leq \tau < \kappa \leq 1$, an $(\varepsilon(N), \delta(N))$-SSS with relative threshold pair (τ, κ) is a pair of polynomial-time algorithms $(\mathsf{Share}, \mathsf{Recst})$,

$$\mathsf{Share}\colon \{0, 1\}^{\ell(N)} \times \mathcal{R} \to \{0, 1\}^N,$$

where $\mathcal{R}$ denote the randomness set, and

$$\mathsf{Recst}\colon \widetilde{\{0, 1\}}^N \to \{0, 1\}^{\ell(N)} \cup \{\perp\},$$

where $\widetilde{\{0, 1\}}^N$ denotes the subset of $(\{0, 1\} \cup \{?\})^N$ with at least $N\kappa$ components not equal to the erasure symbol "?", that satisfy the following properties.

- Reconstruction: Given $k(N) = N\kappa$ correct shares of a share vector $\mathsf{Share}(\mathsf{s})$, the reconstruct algorithm Recst reconstructs the secret s with probability at least $1 - \delta(N)$.

  When $\delta(N) = 0$, we say the SSS has perfect reconstruction.

- Privacy (non-adaptive/adaptive):

  - Non-adaptive: for any $\mathsf{s}_0, \mathsf{s}_1 \in \{0, 1\}^{\ell(N)}$, any $A \subset [N]$ of size $|A| \leq t(N) = N\tau$,

$$\mathsf{SD}(\mathsf{Share}(\mathsf{s}_0)_A; \mathsf{Share}(\mathsf{s}_1)_A) \leq \varepsilon(N). \tag{2}$$

- Adaptive: for any $\mathsf{s}_0, \mathsf{s}_1 \in \{0,1\}^{\ell(N)}$ and any adaptive adversary $\mathcal{A}$ interacting with a $t(N)$-bounded leakage oracle $\mathcal{O}_{t(N)}(\cdot)$ for $t(N) = N\tau$,

$$\mathsf{SD}\left(\mathsf{View}_{\mathcal{A}}^{\mathcal{O}_{t(N)}(\mathsf{Share}(\mathsf{s}_0))}; \mathsf{View}_{\mathcal{A}}^{\mathcal{O}_{t(N)}(\mathsf{Share}(\mathsf{s}_1))}\right) \leq \varepsilon(N). \tag{3}$$

When $\varepsilon(N) = 0$, we say the SSS has perfect privacy.

The difference $\gamma = \kappa - \tau$ is called the relative gap, since $N\gamma = k(N) - t(N)$ is the threshold gap of the scheme. When clear from context, we write $\varepsilon, \delta, t, k, \ell$ instead of $\varepsilon(N), \delta(N), t(N), k(N), \ell(N)$. When the parameters are not specified, we call a (ε, δ)-SSS simply a binary SSS.

Coding rate of binary SSS. In this work, we are concerned with binary SSS that shares a secret of length ℓ, which is a constant fraction of N ($\ell = \Omega(N)$).

Definition 10. For any $0 \leq \tau < \kappa \leq 1$, a coding rate $R \in [0,1]$ is achievable if there exists a family of (ε, δ)-SSS with relative threshold pair (τ, κ) such that ε and δ are both negligible in N and $\frac{\ell}{N} \to R$. The highest achievable coding rate of (ε, δ)-SSS for a pair (τ, κ) is called its capacity.

By relating binary SSS with relative threshold pair (τ, κ) to Wyner wiretap codes for a pair of binary erasure channels, we obtain the following coding rate upper bound for binary SSS.

Lemma 11. *For $0 \leq \tau < \kappa \leq 1$, the coding rate capacity of binary SSS with relative threshold pair (τ, κ) is asymptotically upper-bounded by $\kappa - \tau$.*

Proof. Let $(\mathsf{Share}, \mathsf{Recst})$ be a non-adaptive binary SSS with relative threshold pair (τ, κ). We use Share as the encoder and Recst as the decoder, and verify in the following that we obtain a Wyner wiretap code for a BEC_{p_m} main channel and a BEC_{p_w} wiretapper channel, where $p_m = 1 - \kappa - \xi$ and $p_w = 1 - \tau + \xi$, respectively, for arbitrarily small $\xi > 0$. Erasure in binary SSS is worst case, while it is probabilistic in the Wyner wiretap model. We however note that asymptotically, the number of random erasures of BEC_{p_m} and BEC_{p_w} approaches Np_m and Np_w, respectively, with overwhelming probability, and so a code that protects against worst case erasure can be used as a wiretap code with probabilistic erasure. In our proof we also take into account the difference in the secrecy notion in SSS and in the case of Wyner wiretap code.

The N-bit output $\mathsf{Y} = \mathsf{Y}_1, \ldots, \mathsf{Y}_N$ of a BEC_p has a distribution where each bit is identically independently erased with probability p. By the Chernoff-Hoeffding bounds, the fraction η of erasures satisfies the following. For arbitrarily small $\xi > 0$,

$$\begin{cases} \mathsf{Pr}[\eta \geq p + \xi] & \leq \left(\left(\frac{p}{p+\xi}\right)^{p+\xi}\left(\frac{1-p}{1-p-\xi}\right)^{1-p-\xi}\right)^N; \\ \mathsf{Pr}[\eta \leq p - \xi] & \leq \left(\left(\frac{p}{p-\xi}\right)^{p-\xi}\left(\frac{1-p}{1-p+\xi}\right)^{1-p+\xi}\right)^N. \end{cases}$$

Applying the two inequalities to BEC_{p_m} and BEC_{p_w}, respectively, we obtain the following conclusions. The probability that BEC_{p_m} has at most $p_m + \xi = 1 - \kappa$ fraction of erasures and the probability that BEC_{p_w} has at least $p_w - \xi = 1 - \tau$ fraction of erasures are both at most $\exp(-\Omega(N))$ for arbitrarily small $\xi > 0$.

We are ready to prove the Wyner wiretap reliability and secrecy properties as defined in [23, 13].

- We show correct decoding with probability $1-o(1)$. When the erasures are below $p_m+\xi=1-\kappa$ fraction, it follows directly from the reconstructability of SSS that the decoding error is bounded from above by δ, which is arbitrarily small for big enough N, where the probability is over the randomness of the encoder. When the erasures are not below $p_m+\xi=1-\kappa$ fraction, we do not have correct decoding guarantee. But as argued above, this only occurs with a negligible probability over the randomness of the BEC_{p_m}. Averaging over the channel randomness of the BEC_{p_m}, we have correct decoding with probability $1-o(1)$.

- We show random message equivocation secrecy $\mathsf{H}(\mathsf{S}|\mathsf{W}) \geq \ell(1-o(1))$, where S is a uniform secret and $\mathsf{W}=\mathsf{BEC}_{p_w}(\mathsf{Share}(\mathsf{S}))$ is the view of the wiretapper. We in fact first prove the wiretap indistinguishability security as defined in [2] and then deduce that it implies Wyner wiretap secrecy as defined in [23, 13]. For each of the erasure patterns (say $A \subset [N]$ are not erased) of BEC_{p_w} that exceeds $p_w-\xi=1-\tau$ fraction (equivalently, $|A| \leq N\tau$), the binary SSS privacy gives that for any two secrets, the corresponding views $\mathsf{W}|(\mathsf{S}=\mathsf{s}_0, A \text{ not erased})$ and $\mathsf{W}|(\mathsf{S}=\mathsf{s}_1, A \text{ not erased})$ are indistinguishable with error ε, which is arbitrarily small for big enough N. The distribution $(\mathsf{W}|\mathsf{S}=\mathsf{s}_0)$ and $(\mathsf{W}|\mathsf{S}=\mathsf{s}_1)$ are convex combinations of $\mathsf{W}|(\mathsf{S}=\mathsf{s}_0, A \text{ not erased})$ and $\mathsf{W}|(\mathsf{S}=\mathsf{s}_1, A \text{ not erased})$, respectively, for all the erasure patterns A of BEC_{p_w}. As argued before, the probability that the erasures does not exceed $p_w-\xi=1-\tau$ fraction is negligible. We average over the channel randomness of the wiretapper channel BEC_{p_w} and claim that the statistical distance of $(\mathsf{W}|\mathsf{S}=\mathsf{s}_0)$ and $(\mathsf{W}|\mathsf{S}=\mathsf{s}_1)$ is arbitrarily small for big enough N. According to [2], this is strictly stronger than the Wyner wiretap secrecy.

Finally we use the coding rate upper bound of the Wyner wiretap code to bound the coding rate of binary SSS. We have shown that a binary SSS with relative threshold pair (τ,κ) is a wiretap code for the pair (BEC_{p_m},BEC_{p_w}). According to [23, 13], the achievable coding rate for the Wyner wiretap code is $(1-p_m)-(1-p_w)=p_w-p_m=\kappa-\tau+2\xi$. Since this holds for arbitrarily small $\xi>0$, we obtain an upper bound of $\kappa-\tau$ for binary SSS with relative threshold pair (τ,κ). □

In the rest of the paper, we give two constant rate constructions of binary SSS against non-adaptive adversary and adaptive adversary, respectively. The non-adaptive adversary construction is optimal in the sense that the coding rate achieves the upper bound in Lemma 11.

4 Secret sharing against non-adaptive adversaries

We first present our construction of (capacity-achieving) binary secret sharing schemes that are private against non-adaptive adversaries, using invertible seeded extractors and optimal rate stochastic erasure correcting codes. The following theorem describes the construction using these components.

Theorem 12. *Let* $\mathsf{Ext}\colon \{0,1\}^d \times \{0,1\}^n \to \{0,1\}^\ell$ *be a linear strong seeded* $(n-t,\frac{\varepsilon}{8})$*-extractor and* $\mathsf{Ext}^{-1}(\mathsf{z},\cdot)\colon \{0,1\}^\ell \times \mathcal{R}_1 \to \{0,1\}^n$ *be the inverter of the function* $\mathsf{Ext}(\mathsf{z},\cdot)$ *that maps an* $\mathsf{s} \in \{0,1\}^\ell$ *to one of its pre-images chosen uniformly at random. Let* $(\mathsf{SA\text{-}ECCenc},\mathsf{SA\text{-}ECCdec})$ *be a stochastic affine-erasure correcting code with encoder* $\mathsf{SA\text{-}ECCenc}\colon \{0,1\}^{d+n} \times \mathcal{R}_2 \to \{0,1\}^N$ *that tolerates* $N-k$ *erasures and decodes with success probability at least* $1-\delta$*. Then the following coding scheme* $(\mathsf{Share},\mathsf{Recst})$ *is a non-adaptive* (ε,δ)*-SSS with threshold pair* (t,k)*.*

$$\begin{cases} \mathsf{Share}(\mathsf{s}) & = \mathsf{SA\text{-}ECCenc}(\mathsf{Z}||\mathsf{Ext}^{-1}(\mathsf{Z},\mathsf{s})), \text{where } \mathsf{Z} \xleftarrow{\$} \{0,1\}^d; \\ \mathsf{Recst}(\tilde{\mathsf{v}}) & = \mathsf{Ext}(\mathsf{z},\mathsf{x}), \text{where } (\mathsf{z}||\mathsf{x}) = \mathsf{SA\text{-}ECCdec}(\tilde{\mathsf{v}}). \end{cases}$$

Here $\tilde{\mathsf{v}}$ denotes an incomplete version of a share vector $\mathsf{v} \in \{0,1\}^N$ with some of its components replaced by erasure symbols.

We first prove a general property of a linear strong (k, ε)-extractor $\mathsf{Ext}\colon \{0,1\}^d \times \{0,1\}^n \to \{0,1\}^m$. Roughly speaking, we prove that the pre-images (the pre-image of $\mathsf{m} \in \{0,1\}^m$ is a random variable tuple (Z, X) that satisfies the condition $\mathsf{Ext}(\mathsf{Z}, \mathsf{X}) = \mathsf{m}$) of any two extractor outputs can not be distinguished by any affine function $f_A\colon \{0,1\}^{d+n} \to \{0,1\}^t$ with $t \leq n-k$. The privacy of the SSS in Theorem 12 then follows trivially as a natural consequence of this property. For the property to hold, we in fact only need the extractor to be able to extract from affine sources. But since seeded extractors for general sources with good parameters are not more difficult to construct than that for affine sources, we state the property with a condition stronger than necessary.

Lemma 13. *Let $\mathsf{Ext}\colon \{0,1\}^d \times \{0,1\}^n \to \{0,1\}^m$ be a linear strong (k, ε)-extractor. Let $f_A\colon \{0,1\}^{d+n} \to \{0,1\}^t$ be any affine function with output length $t \leq n-k$. For any $\mathsf{m}, \mathsf{m}' \in \{0,1\}^m$, let $(\mathsf{Z}, \mathsf{X}) = (\mathsf{U}_d, \mathsf{U}_n) | (\mathsf{Ext}(\mathsf{U}_d, \mathsf{U}_n) = \mathsf{m})$ and $(\mathsf{Z}', \mathsf{X}') = (\mathsf{U}_d, \mathsf{U}_n) | (\mathsf{Ext}(\mathsf{U}_d, \mathsf{U}_n) = \mathsf{m}')$. We have*

$$\mathsf{SD}(f_A(\mathsf{Z}, \mathsf{X}); f_A(\mathsf{Z}', \mathsf{X}')) \leq 8\varepsilon. \tag{4}$$

Proof. Without loss of generality, we assume that the linear function $\mathsf{Ext}(\mathsf{z}, \cdot) : \{0,1\}^n \to \{0,1\}^m$, for every seed z, has the entire $\{0,1\}^m$ as its image [2]. Without loss of generality, it suffices to assume that f_A is of the form $f_A(\mathsf{Z}, \mathsf{X}) = (\mathsf{Z}, W(\mathsf{X}))$ for some affine function $W\colon \{0,1\}^n \to \{0,1\}^t$ (this is because for any arbitrary f_A, the information contained in $f_A(\mathsf{Z}, \mathsf{X})$ can be obtained from $(\mathsf{Z}, W(\mathsf{X}))$ for a suitable choice of W).

Let $\mathcal{D}$ be the uniform distribution on the image of W. For the above pairwise guarantee (4) to hold, it suffices to show that for every fixed choice of $\mathsf{m} \in \{0,1\}^m$, the distribution of $f_A(\mathsf{Z}, \mathsf{X})$ is (4ε)-close to $\mathsf{U}_d \times \mathcal{D}$, where U_d is the uniform distribution on $\{0,1\}^d$.

Let $\mathsf{K} \leftarrow \{0,1\}^n$ be a random variable uniformly distributed over the kernel of the linear transformation defined by W, and note that it has entropy at least $n - t \geq k$. The extractor Ext thus guarantees that $\mathsf{Ext}(\mathsf{Z}, \mathsf{K})$, for a uniform and independent seed Z, is ε-close to uniform. By averaging, it follows that for at least $1 - 4\varepsilon$ fraction of the choices of the seed $\mathsf{z} \in \{0,1\}^d$, the distribution of $\mathsf{Ext}(\mathsf{z}, \mathsf{K})$ is $(1/4)$-close to uniform. We now use the following claim.

Claim 14. *Let U be uniformly distributed on $\{0,1\}^m$ and U' be any affine source that is not uniform on $\{0,1\}^m$. Then, the statistical distance between U and U' is at least $1/2$.*

Since Ext is a linear function for every seed, the distribution of $\mathsf{Ext}(\mathsf{z}, \mathsf{K})$ for any seed z is an affine source. Therefore, the above claim allows us to conclude that for at least $1 - 4\varepsilon$ fraction of the choices of z, the distribution of $\mathsf{Ext}(\mathsf{z}, \mathsf{K})$ is *exactly* uniform. Let $\mathcal{G} \subseteq \{0,1\}^d$ be the set of such choices of the seed. Observe that if $\mathsf{Ext}(\mathsf{z}, \mathsf{K})$ is uniform for some seed z, then for any affine translation of K, namely, $\mathsf{K} + \mathsf{v}$ for any $\mathsf{v} \in \{0,1\}^n$, we have that $\mathsf{Ext}(\mathsf{z}, \mathsf{K} + v)$ is uniform as well. This is due to the linearity of the extractor.

Recall that our goal now is to show that $f_A(\mathsf{Z}, \mathsf{X}) = (\mathsf{Z}, W(\mathsf{X}))$ is (4ε)-close to $\mathsf{U}_d \times \mathcal{D}$. The distribution $(\mathsf{Z}, W(\mathsf{X}))$ can be obtained as $(\mathsf{U}_d, W(\mathsf{U}_n))|(\mathsf{Ext}(\mathsf{U}_d, \mathsf{U}_n) = \mathsf{m})$. We take two steps to reach our goal. Step one, we find out the distribution $(\mathsf{U}_d, W(\mathsf{U}_n))|(\mathsf{Ext}(\mathsf{U}_d, \mathsf{U}_n) = \mathsf{m}, \mathsf{U}_d = \mathsf{z})$ for a particular seed z (Proposition 15). Step two, the distribution $(\mathsf{Z}, W(\mathsf{X}))$ is a convex combination of the distributions obtained in Step one (Proposition 16).

[2] If this condition is not satisfied for some choice z of the seed, there must be linear dependencies between the m output bits of $\mathsf{Ext}(\mathsf{z}, \cdot)$. Therefore, for this choice $\mathsf{Ext}(\mathsf{z}, \cdot)$ can never be an extractor and arbitrarily changing $\mathsf{Ext}(\cdot, \mathsf{z})$ to be an arbitrary full rank linear function will not change the overall performance of the extractor.

Consider a uniformly distributed random variable $\mathsf{M} \overset{\$}{\leftarrow} \{0,1\}^m$, and an independent and uniform $\mathsf{Z} \overset{\$}{\leftarrow} \{0,1\}^d$. Let (Z,Y) be the pre-image of the random variable M and define the shorthand $\mathsf{W} := W(\mathsf{Y})$. The rest of the proof is focus on the three random variables W, M and Z.

Proposition 15. *Let* $\mathsf{z} \in \mathcal{G}$ *and consider any* $\mathsf{m} \in \{0,1\}^m$*. Then, the conditional distribution of* $\mathsf{W}|(\mathsf{Z}=\mathsf{z},\mathsf{M}=\mathsf{m})$ *is exactly* $\mathcal{D}$*.*

Note that the distribution of (Z,Y) is uniform on $\{0,1\}^{d+n}$. Therefore, the distribution of (Z,W) is exactly $\mathsf{U}_d \times \mathcal{D}$. In particular, for any $\mathsf{z} \in \{0,1\}^d$, the conditional distribution $\mathsf{W}|(\mathsf{Z}=\mathsf{z})$ is exactly $\mathcal{D}$.

Fix any $\mathsf{z} \in \mathcal{G}$ and let $\mathsf{w} \in \{0,1\}^t$ be any element in the image of W. The conditional distribution $\mathsf{Y}|(\mathsf{Z}=\mathsf{z})$ is uniform over $\{0,1\}^n$ and the conditional distribution $\mathsf{Y}|(\mathsf{Z}=\mathsf{z},\mathsf{W}=\mathsf{w})$ is uniform over a translation of K. By the above argument, recalling $\mathsf{M} = \mathsf{Ext}(\mathsf{Z},\mathsf{Y})$, we therefore know that the conditional distribution of $\mathsf{M}|(\mathsf{Z}=\mathsf{z},\mathsf{W}=\mathsf{w})$ is exactly uniform over $\{0,1\}^m$. Since the conditional distribution of $\mathsf{W}|(\mathsf{Z}=\mathsf{z})$ is $\mathcal{D}$, this means that the conditional distribution of $(\mathsf{M},\mathsf{W})|(\mathsf{Z}=\mathsf{z})$ is exactly $\mathsf{U}_m \times \mathcal{D}$. We have therefore proved Proposition 15.

Proposition 16. *For any* $\mathsf{m} \in \{0,1\}^m$*, the conditional distribution of* $(\mathsf{Z},\mathsf{W})|(\mathsf{M}=\mathsf{m})$ *is* (4ε)*-close to* $\mathsf{U}_d \times \mathcal{D}$*.*

It suffices to note that the distribution of $(\mathsf{Z},\mathsf{W})|(\mathsf{M}=\mathsf{m})$ is a convex combination of the distributions $(\mathsf{Z},\mathsf{W})|(\mathsf{M}=\mathsf{m},\mathsf{Z}=\mathsf{z})$ and then use the result of Proposition 15 along with the fact that $\Pr[\mathsf{Z} \notin \mathcal{G}] \le 4\varepsilon$. A detailed derivation follows.

Recall that for any $\mathsf{z} \in \{0,1\}^d$, the conditional distribution of $\mathsf{W}|(\mathsf{Z}=\mathsf{z})$ is exactly $\mathcal{D}$ (since $\mathsf{Y}|(\mathsf{Z}=\mathsf{z})$ is uniform over $\{0,1\}^n$). Consider any event $\mathcal{E} \subseteq \{0,1\}^{d+t}$ and let $p := \Pr[(\mathsf{Z},\mathsf{W}) \in \mathcal{E}]$. Since Z and W are independent, we have that

$$p = 2^{-d} \sum_{(\mathsf{z},\mathsf{w})\in\mathcal{E}} \mathcal{D}(\mathsf{w}),$$

where $\mathcal{D}(\mathsf{w})$ denotes the probability assigned to the outcome w by $\mathcal{D}$. On the other hand, we shall write down the same probability in the conditional probability space $\mathsf{M}=\mathsf{m}$ and show that it is different from p by at most 4ε, concluding the claim on the statistical distance. We have

$$\begin{aligned}\Pr[(\mathsf{Z},\mathsf{W}) \in \mathcal{E}|\mathsf{M}=\mathsf{m}] &= \sum_{(\mathsf{z},\mathsf{w})\in\mathcal{E}} \Pr[\mathsf{Z}=\mathsf{z},\mathsf{W}=\mathsf{w}|\mathsf{M}=\mathsf{m}] \\ &= \sum_{(\mathsf{z},\mathsf{w})\in\mathcal{E},\mathsf{z}\in\mathcal{G}} \Pr[\mathsf{Z}=\mathsf{z},\mathsf{W}=\mathsf{w}|\mathsf{M}=\mathsf{m}] + \sum_{(\mathsf{z},\mathsf{w})\in\mathcal{E},\mathsf{z}\notin\mathcal{G}} \Pr[\mathsf{Z}=\mathsf{z},\mathsf{W}=\mathsf{w}|\mathsf{M}=\mathsf{m}].\end{aligned}$$

Note that

$$\eta := \sum_{(\mathsf{z},\mathsf{w})\in\mathcal{E},\mathsf{z}\notin\mathcal{G}} \Pr[\mathsf{Z}=\mathsf{z},\mathsf{W}=\mathsf{w}|\mathsf{M}=\mathsf{m}] \le \Pr[\mathsf{Z}\notin\mathcal{G}|\mathsf{M}=\mathsf{m}] \le 4\varepsilon,$$

since M and Z are independent. Therefore,

$$\begin{aligned}\Pr[(\mathsf{Z},\mathsf{W})\in\mathcal{E}|\mathsf{M}=\mathsf{m}] &= \sum_{(\mathsf{z},\mathsf{w})\in\mathcal{E},\mathsf{z}\in\mathcal{G}} \Pr[\mathsf{Z}=\mathsf{z},\mathsf{W}=\mathsf{w}|\mathsf{M}=\mathsf{m}]+\eta \\ &= 2^{-d}\sum_{(\mathsf{z},\mathsf{w})\in\mathcal{E},\mathsf{z}\in\mathcal{G}} \Pr[\mathsf{W}=\mathsf{w}|\mathsf{M}=\mathsf{m},\mathsf{Z}=\mathsf{z}]+\eta \qquad (5) \\ &= 2^{-d}\sum_{(\mathsf{z},\mathsf{w})\in\mathcal{E},\mathsf{z}\in\mathcal{G}} \mathcal{D}(\mathsf{w})+\eta \qquad (6) \\ &= 2^{-d}\Big(\sum_{(\mathsf{z},\mathsf{w})\in\mathcal{E}} \mathcal{D}(\mathsf{w}) - \sum_{(\mathsf{z},\mathsf{w})\in\mathcal{E},\mathsf{z}\notin\mathcal{G}} \mathcal{D}(\mathsf{w})\Big)+\eta\end{aligned}$$

where (5) uses the independence of W and Z and (6) follows from Proposition 15. Observe that

$$\eta' := 2^{-d}\sum_{(\mathsf{z},\mathsf{w})\in\mathcal{E},\mathsf{z}\notin\mathcal{G}} \mathcal{D}(\mathsf{w}) = 2^{-d}\sum_{\mathsf{z}\notin\mathcal{G}}\sum_{\substack{\mathsf{w}:\\(\mathsf{z},\mathsf{w})\in\mathcal{E}}} \mathcal{D}(\mathsf{w}) \le 2^{-d}(2^d-|\mathcal{G}|)\le 4\varepsilon.$$

Therefore,

$$\Pr[(\mathsf{Z},\mathsf{W})\in\mathcal{E}|\mathsf{M}=\mathsf{m}] = p+\eta-\eta' = p\pm 4\varepsilon = \Pr[(\mathsf{Z},\mathsf{W})\in\mathcal{E}]\pm 4\varepsilon,$$

since $0\le\eta\le 4\varepsilon$ and $0\le\eta'\le 4\varepsilon$. The claim follows. □

Proof of Theorem 12. The sharing algorithm of the SSS (before applying the stochastic affine code) takes a secret, which is a particular extractor output $\mathsf{s}\in\{0,1\}^{\ell}$, and uniformly samples a seed $\mathsf{z}\in\{0,1\}^d$ of Ext before uniformly finds an $\mathsf{x}\in\{0,1\}^n$ such that $\mathsf{Ext}(\mathsf{z},\mathsf{x})=\mathsf{s}$. This process of obtaining (z,x) is the same as sampling $(\mathsf{U}_d,\mathsf{U}_n)\overset{\$}{\leftarrow}\{0,1\}^{d+n}$ and then restrict to $\mathsf{Ext}(\mathsf{U}_d,\mathsf{U}_n)=\mathsf{s}$. We define the random variable tuple

$$(\mathsf{Z},\mathsf{X}) := (\mathsf{U}_d,\mathsf{U}_n)|\left(\mathsf{Ext}(\mathsf{U}_d,\mathsf{U}_n)=\mathsf{s}\right) \qquad (7)$$

and refer to it as the pre-image of s.

Let $\Pi_A:\{0,1\}^N\to\{0,1\}^t$ be the projection function that maps a share vector to the t shares with index set $A\subset[N]$ chosen by the non-adaptive adversary. Observe that the combination $(\Pi_A\circ\mathsf{SA\text{-}ECCenc}):\{0,1\}^{d+n}\to\{0,1\}^t$ (for any fixed randomness r of $\mathsf{SA\text{-}ECCenc}$) is an affine function. So the view of the adversary is simply the output of the affine function $f_A=(\Pi_A\circ\mathsf{SA\text{-}ECCenc})$ applied to the random variable tuple (Z,X) defined in (7).

We can now formulate the privacy of the SSS in this context. We want to prove that the statistical distance of the views of the adversary for a pair of secrets can be made arbitrarily small. The views of the adversary are the outputs of the affine function f_A with inputs (Z,X) and $(\mathsf{Z}',\mathsf{X}')$ for the secret s and s', respectively. According to Lemma 13, we then have that the privacy error is $8\times\frac{\varepsilon}{8}=\varepsilon$. □

Remark 17. Note that the same argument cannot be made if the set A is chosen according to the seed z, in which case the source (induced by the set A) of Ext depends on its seed z, violating the definition of a seeded extractor. To see this, recall that the source of Ext is uniform n-bit conditioned on the output of the affine function $f_A=(\Pi_A\circ\mathsf{SA\text{-}ECCenc})$. The choice of A then can affect this source. In a real life adaptive attack, the adversary can first spend some reading budget on figuring out the value of the seed z of Ext, and then decide the rest of the reading positions according to the seed value. Finding out the value of z is possible because the encoding randomness of the $\mathsf{SA\text{-}ECCenc}$ is included as part of the share vector. The adversary can choose to read the encoding randomness r of the $\mathsf{SA\text{-}ECCenc}$. After r is revealed, the $\mathsf{SA\text{-}ECCenc}$ is just a deterministic affine function. It is then possible to recover information about z by solving linear equations.

Instantiations of the construction.

We now analyze the coding rate of the (ε,δ)-SSS with relative threshold pair $(\frac{t}{N},\frac{k}{N})$ constructed in Theorem 12 when instantiated with the $\mathsf{SA\text{-}ECC}$ from Lemma 8 and the Ext from Lemma 4. The secret length is $\ell = n - t - O(d)$, where the seed length is $d = O(\log^3(2n/\varepsilon))$. The $\mathsf{SA\text{-}ECC}$ encodes $d+n$ bits to N bits and with coding rate $R_{ECC} = \kappa - \xi$ for a small ξ determined by δ (satisfying the relation $\delta = \exp(-\Omega(\xi^2 N/\log^2 N))$ according to Lemma 8). We then have $n = N(\kappa-\xi) - d$, resulting in the coding rate

$$R = \frac{\ell}{N} = \frac{n-t-O(d)}{N} = \frac{N(\kappa-\xi)-t-O(d)}{N} = \kappa - \tau - (\xi + \frac{O(d)}{N}) = \kappa - \tau - o(1).$$

Corollary 18. *For any $0 \le \tau < \kappa \le 1$, there is an explicit construction of non-adaptive (ε,δ)-SSS with relative threshold pair (τ,κ) achieving coding rate $\kappa - \tau - o(1)$ for large N.*

5 Secret sharing against adaptive adversaries

In this section, we will describe our construction which achieves privacy against adaptive adversaries, using affine extractors. We start with the specific extraction property needed from our affine extractors.

Definition 19. An affine extractor $\mathsf{AExt}\colon \{0,1\}^n \to \{0,1\}^m$ is called (k,ε)-almost perfect if for any affine (n,k)-source X,

$$\left|\mathsf{Pr}[\mathsf{AExt}(\mathsf{X}) = \mathsf{y}] - \frac{1}{2^m}\right| \le 2^{-m}\cdot\varepsilon, \text{ for any } \mathsf{y} \in \{0,1\}^m.$$

Almost perfect property can be trivially achieved by requiring an exponentially (in m) small error in statistical distance, using the relation between ℓ_∞-norm and ℓ_1-norm.

Theorem 20. *Let $\mathsf{AExt}\colon \{0,1\}^n \to \{0,1\}^\ell$ be an invertible $(n-t,\frac{\varepsilon}{2})$-almost perfect affine extractor and $\mathsf{AExt}^{-1}\colon \{0,1\}^\ell \times \mathcal{R}_1 \to \{0,1\}^n$ be its v-inverter that maps an $\mathsf{s} \in \{0,1\}^\ell$ to one of its preimages chosen uniformly at random. Let $(\mathsf{SA\text{-}ECCenc}, \mathsf{SA\text{-}ECCdec})$ be a stochastic affine-erasure correcting code with encoder $\mathsf{SA\text{-}ECCenc}\colon \{0,1\}^n \times \mathcal{R}_2 \to \{0,1\}^N$ that tolerates $N-k$ erasures and decodes with success probability at least $1-\delta$. Then the $(\mathsf{Share}, \mathsf{Recst})$ defined as follows is an adaptive (ε,δ)-SSS with threshold pair (t,k).*

$$\begin{cases} \mathsf{Share}(\mathsf{s}) & = \mathsf{SA\text{-}ECCenc}(\mathsf{AExt}^{-1}(\mathsf{s})); \\ \mathsf{Recst}(\tilde{\mathsf{v}}) & = \mathsf{AExt}(\mathsf{SA\text{-}ECCdec}(\tilde{\mathsf{v}})), \end{cases}$$

where $\tilde{\mathsf{v}}$ denotes an incomplete version of a share vector $\mathsf{v} \in \{0,1\}^N$ with some of its components replaced by erasure symbols.

Proof. The (k,δ)-reconstructability follows directly from the erasure correcting capability of the $\mathsf{SA\text{-}ECC}$. For any $\tilde{\mathsf{v}}$ with at most $N-k$ erasure symbols and the rest of its components consistent with a valid codeword $\mathsf{v} \in \{0,1\}^N$, the $\mathsf{SA\text{-}ECC}$ decoder identifies the unique codeword v with probability $1-\delta$ over the encoder randomness. The corresponding $\mathsf{SA\text{-}ECC}$ message of v is then inputted to AExt and the original secret s is reconstructed with the same probability.

We next prove the (t,ε)-privacy. For any $\mathsf{r} \in \mathcal{R}_2$, the affine encoder of $\mathsf{SA\text{-}ECC}$ is characterised by a matrix $G_{\mathsf{r}} \in \{0,1\}^{n\times N}$ and an offset Δ_{r}. For n unknowns $\mathbf{x} = (x_1, \ldots, x_n)$, we have

$$\mathsf{SA\text{-}ECCenc}(\mathbf{x}) = \mathbf{x}G_{\mathsf{r}} + \Delta_{\mathsf{r}} = (\mathbf{x}G_1, \ldots, \mathbf{x}G_N) + \Delta_{\mathsf{r}},$$

where $G_i = (g_{1,i}, \ldots, g_{n,i})^T$ (here the subscript "$_{\mathsf{r}}$" is omitted to avoid double subscripts) denotes the ith column of G_{r}, $i = 1, \ldots, N$. This means that knowing a component c_i of the $\mathsf{SA\text{-}ECC}$ codeword is equivalent to obtaining a linear equation $c_i \oplus \Delta_i = \mathbf{x}G_i = g_{1,i}x_1 + \cdots + g_{n,i}x_n$ about the n unknowns $x_1, \ldots, x_n$, where Δ_i (again, the subscript "$_{\mathsf{r}}$" is omitted) denotes the ith component of Δ_{r}.

Now, we investigate the distribution of $\mathsf{View}_{\mathcal{A}}^{\mathcal{O}_t(\mathsf{Share}(\mathsf{s}))}$ for any secret $\mathsf{s} \in \{0,1\}^{\ell}$ by finding the probability $\Pr[\mathsf{View}_{\mathcal{A}}^{\mathcal{O}_t(\mathsf{Share}(\mathsf{s}))} = \mathsf{w}]$ for arbitrary w. We then have

$$\begin{aligned}
\Pr[\mathsf{View}_{\mathcal{A}}^{\mathcal{O}_t(\mathsf{Share}(\mathsf{s}))} = \mathsf{w}] &= \Pr[\mathsf{View}_{\mathcal{A}}^{\mathcal{O}_t(\mathsf{SA\text{-}ECCenc}(\mathsf{X}))} = \mathsf{w} | \mathsf{AExt}(\mathsf{X}) = \mathsf{s}] \\
&= \frac{\Pr[\mathsf{AExt}(\mathsf{X}) = \mathsf{s} | \mathsf{View}_{\mathcal{A}}^{\mathsf{SA\text{-}ECCenc}(\mathsf{X})} = \mathsf{w}] \cdot \Pr[\mathsf{View}_{\mathcal{A}}^{\mathsf{SA\text{-}ECCenc}(\mathsf{X})} = \mathsf{w}]}{\Pr[\mathsf{AExt}(\mathsf{X}) = \mathsf{s}]} \\
&\underset{=}{\underline{(i)}} \frac{(1 \pm \frac{\varepsilon}{2})2^{-\ell} \cdot \Pr[\mathsf{View}_{\mathcal{A}}^{\mathsf{SA\text{-}ECCenc}(\mathsf{X})} = \mathsf{w}]}{\Pr[\mathsf{AExt}(\mathsf{X}) = \mathsf{s}]} \\
&\underset{=}{\underline{(ii)}} \frac{(1 \pm \frac{\varepsilon}{2})2^{-\ell} \cdot \frac{2^{n-\mathrm{rank}(\mathcal{A})}}{2^n}}{2^{-\ell}} \\
&= (1 \pm \frac{\varepsilon}{2}) \cdot 2^{-\mathrm{rank}(\mathcal{A})},
\end{aligned}$$

where notations X, ±, $\mathrm{rank}(\mathcal{A})$ and $(i), (ii)$ are explained as follows. In above, we first use the fact that $\Pr[\mathsf{View}_{\mathcal{A}}^{\mathcal{O}_t(\mathsf{Share}(\mathsf{s}))} = \mathsf{w}]$ can be seen as the probability of uniformly selecting X from $\{0,1\}^n$, with the condition that $\mathsf{AExt}(\mathsf{X}) = \mathsf{s}$. This is true because the sets $\mathsf{AExt}^{-1}(\mathsf{s})$ for all s, partition $\{0,1\}^n$ and the rest follows from Definition 6. The shorthand "$y = 1 \pm \frac{\varepsilon}{2}$" denotes "$1 - \frac{\varepsilon}{2} \le y \le 1 + \frac{\varepsilon}{2}$". The shorthand "$\mathrm{rank}(\mathcal{A})$" denotes the rank of the up to t columns of G corresponding to the index set A adaptively chosen by $\mathcal{A}$. The equality (i) follows from the fact that AExt is an $(n-t, 2^{-(\ell+1)}\varepsilon)$-affine extractor and the uniform X conditioned on at most t linear equations is an affine source with at least $n-t$ bits entropy. The equality (ii) holds if and only if w is in the set $\{\mathsf{SA\text{-}ECCenc}(\mathsf{x})_A : \mathsf{x} \in \{0,1\}^n\}$. Indeed, consider X as unknowns for equations, the number of solutions to the linear system $\mathsf{SA\text{-}ECCenc}(\mathsf{X})_A = \mathsf{w}$ is either 0 or equal to $2^{n-\mathrm{rank}(\mathcal{A})}$.

The distribution of $\mathsf{View}_{\mathcal{A}}^{\mathcal{O}_t(\mathsf{Share}(\mathsf{s}))}$ for any secret s is determined by the quantity $\mathrm{rank}(\mathcal{A})$, which is independent of the secret s. Let W be the uniform distribution over the set $\{\mathsf{SA\text{-}ECCenc}(\mathsf{x})_A : \mathsf{x} \in \{0,1\}^n\}$. Then by the triangular inequality, we have

$$\begin{aligned}
\mathsf{SD}\left(\mathsf{View}_{\mathcal{A}}^{\mathcal{O}_t(\mathsf{Share}(\mathsf{s}_0))}; \mathsf{View}_{\mathcal{A}}^{\mathcal{O}_t(\mathsf{Share}(\mathsf{s}_1))}\right) &\le \mathsf{SD}\left(\mathsf{View}_{\mathcal{A}}^{\mathcal{O}_t(\mathsf{Share}(\mathsf{s}_0))}; \mathsf{W}\right) + \mathsf{SD}\left(\mathsf{W}; \mathsf{View}_{\mathcal{A}}^{\mathcal{O}_t(\mathsf{Share}(\mathsf{s}_1))}\right) \\
&\le \tfrac{\varepsilon}{2} + \tfrac{\varepsilon}{2} \\
&= \varepsilon.
\end{aligned}$$

This concludes the privacy proof. □

Instantiations of the construction.

There are explicit constructions of binary affine extractors that, given a constant fraction of entropy, outputs a constant fraction of random bits with exponentially small error (see Lemma 5) There are known methods for constructing an invertible affine extractor AExt' from any affine extractor AExt such that the constant fraction output size and exponentially small error properties are preserved. A simple method is to let $\mathsf{AExt}'(\mathsf{U}_n||M) := \mathsf{AExt}(\mathsf{U}_n) \oplus M$ (see Appendix B for a discussion). This is summarized in the lemma below.

Lemma 21. *For any $\delta \in (0, 1]$, there is an explicit seedless $(\delta n, \varepsilon)$-almost perfect affine extractor $\mathsf{AExt}\colon \{0,1\}^n \to \{0,1\}^m$ where $m = \Omega(n)$ and $\varepsilon = \exp(-\Omega(n))$. Moreover, there is an efficiently computable ε-inverter for the extractor.*

Proof. Let $f\colon \{0,1\}^n \to \{0,1\}^m$ be Bourgain's affine extractor (Lemma 5) for entropy rate μ, output length $m = \Omega(n)$, and achieving exponentially small error $\varepsilon = \exp(-\Omega(n))$. Using the one-time pad trick (Appendix B), we construct an invertible variant achieving output length $m' = \Omega(m) = \Omega(n)$ and exponentially small error. Finally, we simply truncate the output length of the resulting extractor to $m'' = \Omega(m') = \Omega(n)$ bits so that the ℓ_∞ guarantee of the distance to uniformity required for almost-perfect extraction is satisfied. The truncated extractor is still invertible since the inverter can simply pad the given input with random bits and invoke the original inverter function. □

Note that for simplicity, Theorem 20 assumes that the inverter of AExt is perfect, namely, $\mathsf{AExt}^{-1}(\mathsf{U}_\ell) = \mathsf{U}_n$. In the case when there is an exponentially small error, as in the case of Lemma 21, the privacy error parameter will be slightly affected (increase by 2 times) due to a hybrid that replaces the ideal U_n with the real $\mathsf{AExt}^{-1}(\mathsf{U}_\ell)$.

It now suffices to instantiate Theorem 20 with the explicit construction of SA-ECC and the invertible affine extractor AExt of Lemma 21. Let R_{ECC} denote the rate of the SA-ECC. Then we have $R_{ECC} = \frac{n}{N}$, where n is the input length of the affine extractor AExt and N is the number of players. The intuition of the construction in Theorem 20 is that if a uniform secret is shared and conditioning on the revealed shares the secret still has a uniform distribution (being the output of a randomness extractor), then no information is leaked. In fact, the proof of Theorem 20 above is this intuition made exact, with special care on handling the imperfectness of the affine extractor. So as long as the "source" of the affine extractor AExt has enough entropy, privacy is guaranteed. Here the "source" is the distribution U_n conditioned on the adversary's view, which is the output of a t-bit affine function. The "source" then is affine and has at least $n - \tau N = n(1 - \frac{\tau}{R_{ECC}})$ bits of entropy. Now as long as $\tau < R_{ECC}$, using the AExt from Lemma 5 (more precisely, an invertible affine extractor $\mathsf{AExt}' : \{0,1\}^{n'} \to \{0,1\}^\ell$ constructed from AExt) with $\mu = 1 - \frac{\tau}{R_{ECC}}$, a constant fraction of random bits can be extracted with exponentially small error. This says that privacy is guaranteed for $\tau \in [0, R_{ECC})$. The stochastic affine ECC in Lemma 8 asymptotically achieves the rate $1 - (1 - \kappa) = \kappa$. We then have the following corollary.

Corollary 22. For any $0 \le \tau < \kappa \le 1$, there is an explicit constant coding rate adaptive (ε, δ)-SSS with relative threshold pair (τ, κ).

The construction above achieves a constant coding rate for any (τ, κ) pair satisfying $0 \le \tau < \kappa \le 1$. Since the binary affine extractor in Lemma 5 does not extract all the entropy from the source (meaning the output length is equal to the source entropy minus a non-negligible amount) and moreover the step that transforms an affine extractor into an invertible affine extractor incurs non-negligible overhead, the coding rate of the above construction does not approach $\kappa - \tau$.

6 Conclusion

We studied the problem of sharing *arbitrary long secrets* using constant length shares and required a near threshold access structure. By near threshold we mean a ramp scheme with arbitrarily small gap to number of players ratio. We show that by replacing perfect privacy and reconstructibility with slightly relaxed notions and inline with similar strong cryptographic notions, one can explicitly construct such near threshold schemes. We gave two constructions with security against non-adaptive and adaptive adversaries, respectively, and proved optimality of the former. Our work also make a new connection between secret sharing and wiretap coding.

Our constructions are not linear: even the explicit non-adaptive construction that uses an affine function for every fixing of the encoder's randomness does not result in a linear secret sharing. Linearity is an essential property in applications such as multiparty computation and so explicit constructions of *linear* secret sharing schemes in our model will be an important achievement. Yet another important direction for future work is deriving bounds and constructing optimal codes for finite length (N) case. Such result will also be of high interest for wiretap coding.

We presented our model and constructions for the extremal case of binary shares. However, we point out that the model and our constructions can be extended to shares over any desired alphabet size q. Using straightforward observations (such as assigning multiple shares to each player), this task reduces to extending the constructions over any prime q. In this case, the building blocks that we use; namely, the stochastic error-correcting code, seeded and seedless affine extractors need to be extended to the q-ary alphabet. The constructions that we use [22, 18, 15], however, can be extended to general alphabets with straightforward modifications. The only exception is Bourgain's seedless affine extractor [4] which needs some work to be extended to arbitrary alphabets. This has been accomplished in a work by Yehudayoff [24].

References

[1] Vaneet Aggarwal, Lifeng Lai, A. Robert Calderbank, and H. Vincent Poor. Wiretap channel type II with an active eavesdropper. *IEEE International Symposium on Information Theory, ISIT 2009*, pages 1944–1948, 2009.

[2] Mihir Bellare, Stefano Tessaro, and Alexander Vardy. Semantic security for the wiretap channel. In *Advances in Cryptology - CRYPTO 2012*, volume 7417 of *Lecture Notes in Computer Science*, pages 294–311. Springer, 2012.

[3] George R. Blakley. Safeguarding cryptographic keys. In *Proceedings of the 1979 AFIPS National Computer Conference*, pages 313–317, 1979.

[4] Jean Bourgain. On the construction of affine extractors. *Geometric and Functional Analysis*, 17(1):33–57, 2007.

[5] Ignacio Cascudo Pueyo, Ronald Cramer, and Chaoping Xing. Bounds on the threshold gap in secret sharing and its applications. *IEEE Trans. Information Theory*, 59(9):5600–5612, 2013.

[6] Hao Chen and Ronald Cramer. Algebraic geometric secret sharing schemes and secure multi-party computations over small fields. In *Advances in Cryptology - CRYPTO 2006*, volume 4117 of *Lecture Notes in Computer Science*, pages 521–536. Springer, 2006.

[7] Kuan Cheng, Yuval Ishai, and Xin Li. Near-optimal secret sharing and error correcting codes in AC0. In *Theory of Cryptography - TCC 2017, Part II*, pages 424–458, 2017.

[8] Mahdi Cheraghchi. Nearly optimal robust secret sharing. *IEEE International Symposium on Information Theory, ISIT 2016*, pages 2509–2513, 2016.

[9] Mahdi Cheraghchi, Frédéric Didier, and Amin Shokrollahi. Invertible extractors and wiretap protocols. *IEEE Trans. Information Theory*, 58(2):1254–1274, 2012.

[10] Ronald Cramer, Ivan Damgård, and Jesper Buus Nielsen. *Secure Multiparty Computation and Secret Sharing*. Cambridge University Press, 2015.

[11] Ronald Cramer, Ivan Bjerre Damgård, Nico Döttling, Serge Fehr, and Gabriele Spini. Linear secret sharing schemes from error correcting codes and universal hash functions. In *Advances in Cryptology - EUROCRYPT 2015*, volume 9057 of *Lecture Notes in Computer Science*, pages 313–336. Springer, 2015.

[12] Ronald Cramer, Yevgeniy Dodis, Serge Fehr, Carles Padró, and Daniel Wichs. Detection of algebraic manipulation with applications to robust secret sharing and fuzzy extractors. In *Advances in Cryptology - EUROCRYPT 2008*, volume 4965 of *Lecture Notes in Computer Science*, pages 471–488. Springer, 2008.

[13] Imre Csiszár and Janos Körner. Broadcast channels with confidential messages. *IEEE Trans. Information Theory*, 24(3):339–348, 1978.

[14] Venkatesan Guruswami. List decoding from erasures: bounds and code constructions. *IEEE Trans. Information Theory*, 49(11), 2003.

[15] Venkatesan Guruswami and Adam D. Smith. Optimal rate code constructions for computationally simple channels. *J. ACM*, 63(4):35:1–35:37, 2016.

[16] Xin Li. A new approach to affine extractors and dispersers. *IEEE Conference on Computational Complexity, CCC 2011*, pages 137–147, 2011.

[17] Lawrence H. Ozarow and Aaron D. Wyner. Wire-tap channel II. *The Bell System Technical Journal*, 63:2135–2157, Dec 1984.

[18] Ran Raz, Omer Reingold, and Salil P. Vadhan. Extracting all the randomness and reducing the error in Trevisan's extractors. *J. Comput. Syst. Sci.*, 65(1):97–128, 2002.

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

[20] Adam D. Smith. Scrambling adversarial errors using few random bits, optimal information reconciliation, and better private codes. In *ACM-SIAM Symposium on Discrete Algorithms SODA '07*, pages 395–404. Society for Industrial and Applied Mathematics, 2007.

[21] Douglas R. Stinson. An explication of secret sharing schemes. *Des. Codes Cryptography*, 2(4):357–390, 1992.

[22] Luca Trevisan. Extractors and pseudorandom generators. *J. ACM*, 48(4):860–879, 2001.

[23] Aaron D. Wyner. The wire-tap channel. *The Bell System Technical Journal*, 54(8):1355–1387, Oct 1975.

[24] Amir Yehudayoff. Affine extractors over prime fields. *Combinatorica*, 31(2):245, 2011.

A Proof Sketch of Lemma 8

Proof. We defer the details of the construction to the full version of this paper. For now we refer to [15, Theorem 6.1] and point out the adaptations needed. There are six building blocks involved in the construction: SC, RS, Samp, KNR, POLY_t and REC. We replace the first and last building blocks.

The first building block is a *Stochastic Code* (SC). We need two properties from this building block: detect (output $\perp$) when the codeword is masked by a random offset and correct from erasures of no more than $1-\kappa+\varepsilon$ fraction. While the former property is always satisfied by the original SC used in [15], the latter property might not hold. When $1-\kappa$ is small, we can let the decoder of the SC used in [15] set the erased bits to 0 and decode it as errors. But when $1-\kappa > \frac{1}{2}$, this trick no longer works. We then combine a systematic AMD in [12] and an erasure list-decodable code [14] (sub-optimal suffices since SC encodes the control information which is negligible) to obtain the SC in our construction of $\mathsf{SA\text{-}ECC}$.

The last building block is a *Random Error Code* (REC). We also need two properties from this building block: correct from random erasures of $1-\kappa$ fraction and the encoder is a linear function. We need the latter property for affine property of the $\mathsf{SA\text{-}ECC}$ constructed. Explicit linear codes at rate $1-p$ that correct p fraction of random erasures are known. We can use any explicit construction of capacity achieving codes for $\mathrm{BEC}_{1-\kappa}$ for REC and use a similar argument of [20].

We now refer to the **Algorithm 1.** in the proof of [15, Theorem 6.1] and show that, with the SC and REC replaced accordingly, we do have a $\mathsf{SA\text{-}ECC}$. The error correction capability and optimal rate follow similarly as in the proof of [15, Theorem 6.1]. We next show affine property. **Phase 1** and **Phase 2** are about the *control information*, which are part of the encoding randomness r of the $\mathsf{SA\text{-}ECC}$ to be fixed to constant value in the analysis of affine property. During **Phase 3**, the message $\mathbf{m}$ is linearly encoded (our REC is linear) and then permuted, followed by adding a translation term Δ_{r}. Since permutation is a linear transformation, we combine the two linear transformations and write $\mathbf{m}G_{\mathsf{r}} + \Delta_{\mathsf{r}}$, where G_{r} is a binary matrix. Finally, during **Phase 4**, some blocks that contain the control information are inserted into $\mathbf{m}G_{\mathsf{r}} + \Delta_{\mathsf{r}}$. We add dummy zero columns into G_{r} and zero blocks into Δ_{r} to the corresponding positions where the control information blocks are inserted. Let $\mathbf{m}\hat{G}_{\mathsf{r}} + \hat{\Delta}_{\mathsf{r}}$ be the vector after padding dummy zeros. Let $\hat{\Delta}'_{\mathsf{r}}$ be the vector obtained from padding dummy zero blocks, complementary to the padding above, to the control information blocks. We then write the final codeword of the $\mathsf{SA\text{-}ECC}$ in the form $\mathbf{m}\hat{G}_{\mathsf{r}} + (\hat{\Delta}_{\mathsf{r}} + \hat{\Delta}'_{\mathsf{r}})$, which is indeed an affine function of the message $\mathbf{m}$.

□

B One-Time-Pad trick of inverting extractors

There is a well known way to transform an efficient function into one that is also efficiently invertible through a "One-Time-Pad" trick. We give a proof for the special case of affine extractors, for completeness.

Lemma 23. Let $\mathsf{AExt}\colon \{0,1\}^n \to \{0,1\}^m$ be an affine (n,k)-extractor with error ε. Then $\mathsf{AExt}' : \{0,1\}^{n+m} \to \{0,1\}^m$ defined as follows is a ε-invertible affine $(n+m, k+m)$-extractor with error ε.

$$\mathsf{AExt}'(\mathsf{z}) = \mathsf{AExt}'(\mathsf{x}||\mathsf{y}) = \mathsf{AExt}(\mathsf{x}) + \mathsf{y},$$

where the input $\mathsf{z} \in \{0,1\}^{n+m}$ is separated into two parts: $\mathsf{x} \in \{0,1\}^n$ and $\mathsf{y} \in \{0,1\}^m$.

Proof. Let Z be a random variable with flat distribution supported on an affine subspace of $\{0,1\}^{n+m}$ of dimension at least $k+m$. Separate Z into two parts $\mathsf{Z} = (\mathsf{X}||\mathsf{Y})$, where $\mathsf{X} \in \{0,1\}^n$ and $\mathsf{Y} \in \{0,1\}^m$. Then conditioned on any $\mathsf{Y} = \mathsf{y}$, X has a distribution supported on an affine subspace of $\{0,1\}^n$ of dimension at least k. This asserts that conditioned on any $\mathsf{Y} = \mathsf{y}$,

$$\mathsf{SD}(\mathsf{AExt}(\mathsf{X}) + \mathbf{y}; \mathsf{U}_{\{0,1\}^m}) \leq \varepsilon.$$

Averaging over the distribution of Y concludes the extractor proof.

We next show an efficient inverter AExt'^{-1} for AExt'. For any $\mathsf{s} \in \{0,1\}^m$, define

$$\mathsf{AExt}'^{-1}(\mathsf{s}) = (\mathsf{U}_n||\mathsf{AExt}(\mathsf{U}_n) + \mathsf{s}).$$

The randomised function AExt'^{-1} is efficient and $\mathsf{AExt}'^{-1}(\mathsf{U}_m) \overset{\varepsilon}{\sim} \mathsf{U}_{n+m}$. □