On the Key Generation Rate of Physically Unclonable Functions

Yitao Chen, Muryong Kim, and Sriram Vishwanath
University of Texas at Austin
Austin, TX 78701 USA
Email: {yitaochen, muryong}@utexas.edu, sriram@ece.utexas.edu

***Abstract*—THIS PAPER IS ELIGIBLE FOR THE STUDENT PAPER AWARD.**
In this paper, an algebraic binning based coding scheme and its associated achievable rate for key generation using physically unclonable functions (PUFs) is determined. This achievable rate is shown to be optimal under the generated-secret (GS) model for PUFs. Furthermore, a polar code based polynomial-time encoding and decoding scheme that achieves this rate is also presented.

I. Introduction

Physically unclonable functions (PUFs) form a promising innovative primitive that are increasingly gaining traction in the domains of authentication and secret key storage [1]–[3]. Instead of storing secrets in digital memory, PUFs derive a secret from the physical characteristics of an integrated circuit (IC) that form an inherent part of the device. Such a PUF can be obtained, as even though the mask and manufacturing process is relatively similar among ICs built for a particular purpose, each IC is actually unique due to normal manufacturing variability.

This unique behavior after manufacturing stems from a *static randomness* due to technological dispersion. This static randomness was characterized by Pelgrom [4], and is known to follow a normal distribution. Unfortunately, PUF outputs are also subject to *dynamic randomness* due to measurement noise, which is detrimental to the reliability of a PUF as a source for cryptographic elements.

In this paper, we understand the information theoretic limits of key generation using PUFs, given this static and dynamic randomness in the system. As discussed in [1], [2], one of the central use-cases for PUFs is secret key generation, where this key is subsequently utilized in a variety of cryptographic algorithms. A higher key generation rate implies greater security guarantees for the overall system, and therefore, our focus is to understand its limits, and to characterize coding schemes that approach these limits.

A. Related Work and Our Contributions

There is already a considerable body of work on combining PUFs with error correction coding schemes to obtain reliable keys or secrets [3]. Conventionally, these have combined BCH/RS codes with PUFs. More recently, [8] presents simulation results on the combination of a polar code with a PUF, setting the stage for such a combination to be understood analytically. In parallel work to this paper, the authors et al. [9] uncover the connection between PUF key generation problem and Wyner-Ziv problem [15]. And they study a nested polar codes construction scheme based on [10].

In this work, we present a PUF key generation scheme based on a previously well studied model called *generated-secret* (GS) model. In [6], the authors present the region of achievable secret-key vs. privacy-leakage (key vs. leakage) rates for the GS model. In this paper, we show that the optimal key generation rate is achievable using algebraic binning with linear codes, and uncover the relation between PUF key generation problem and Slepian-Wolf problem. Further, we present encoding and decoding algorithms using polar codes that achieve the optimal rate. Finally, we present simulation results to showcase the performance of our scheme.

Compared to existing literature, we find that our scheme results in a relatively straightforward interpretation of the PUF key generation problem, and results in a key generation rate that is optimal for the GS model for PUFs. This is further expanded on in later sections of the paper.

The remainder of this paper is organized as follows. Section II provides the system model for PUF and formally defines the problem. Section III shows the algebraic binning method and polar code construction achieving the maximal key generation rate. Section IV compares our method to the existing other methods. Section V presents the simulation result. Finally, Section VI concludes the paper and gives future directions.

II. System Model

Upper case letters represent random variables and lower case letters their realizations. A superscript denotes a vector of variables, e.g., $X^n = X_1, X_2, \dots, X_n$, and a subscript denotes the position of a variable in a vector. Calligraphic letters such as $\mathcal{X}$ denote sets, and set sizes are written as $\|\mathcal{X}\|$. $H_b(x) = -x \log x - (1-x)\log(1-x)$ is the binary entropy function. The $*$-operator is defined as $p * x = p(1-x) + (1-p)x$. The operator $\oplus$ represents the element-wise modulo-2 summation. A binary symmetric channel (BSC) with crossover probability p is denoted by BSC(p). $Q_b(X)$ represents a binary quantizer that quantizes $X > 0$ to 1 and $X < 0$ to 0.

A physically unclonable function (PUF) can be mathematically represented fairly simply as

$$\tilde{Y} = \tilde{X} + \tilde{Z},$$

where $\tilde{Y}$ is the PUF output, $\tilde{X} \sim \mathcal{N}(0,P)$ is the *static randomness* and $\tilde{Z} \sim \mathcal{N}(0,N)$ the *dynamic randomness* independent of X. As stated earlier in the introduction, $\tilde{X}$ is the desired “signal", which is corrupted by the “noise” $\tilde{Z}$ when observed at the output of a PUF.

In most conventional systems today, the PUF output is quantized immediately after observation. Most models in literature assume the output passes through a binary quantizer $Q_b(\cdot)$. It can be easily shown that $Q_b(\tilde{Y})$, $Q_b(\tilde{X})$ are distributed as $Bernoulli(0.5)$, and $Q_b(\tilde{Y}) = Q_b(\tilde{X}) \oplus Z'$, where Z' is independent of $Q_b(\tilde{X})$, distributed as Bernoulli(p), where p is a function of P and N. From now on, we use Y to represent $Q_b(\tilde{Y})$, X to represent $Q_b(\tilde{X})$ and Z to represent Z'. Thus we have our PUF model with binary quantization as

$$Y = X \oplus Z, \tag{1}$$

where Y is the PUF output, $X \sim Bernoulli(0.5)$ and $Z \sim Bernoulli(p)$. Note that from the nature of PUF we do not have access to X. Indeed, in the real world applications, both Y and X are PUF outputs, which gives a different crossover probability $p{*}p$ in the model. But since it does not change the model (BSC), in the remaining part of this paper, we abuse the notations that we stick to the above model while both Y and X represent PUF outputs and p is the parameter that measures the noise between two PUF outputs.

As mentioned earlier, we follow the *generated-secret* (GS) model for key generation of PUFs, as depicted in Figure 1. For a given sequence X^n, our task is to design an encoder $\phi : \mathbb{F}_2^n :\to (\mathbb{F}_2^{n-k}, \mathbb{F}_2^k)$ that generates a *helper sequence* S^{n-k} and a key W^k and a decoder $\psi : \mathbb{F}_2^n \times \mathbb{F}_2^{n-k} :\to \mathbb{F}_2^k$ that authenticates the key. This is such that, for a particular PUF, the probability of successful authentication goes to 1 as n goes to infinity. Let $(S^{n-k}, W^k) = \phi(X^n)$ and $\hat{W}^k = \psi(Y^n, S^{n-k})$,

$$\Pr(\hat{W}^k \neq W^k) \to 0 \quad \text{as } n \to \infty.$$

Define the key generation rate as $R = k/n$, we desire to determine the maximal key generation rate

$$\max_{\phi,\psi} R \quad \text{s.t. } \Pr(\hat{W}^k \neq W^k) \to 0. \tag{2}$$

III. PUF System Design Using Polar Codes

In this section, we first state our main theorem and show the optimal key generation rate is achievable with algebraic binning using linear codes. Note that, the result can also be obtained by random binning, but algebraic binning offers greater insights for PUF key generation system design. Therefore, we choose to use an algebraic binning framework going forward.

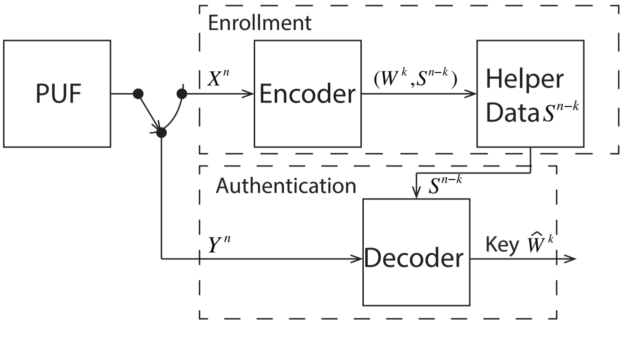

Fig. 1. The GS model for PUF key generation system

A. Algebraic Binning Using Linear Codes

Theorem 1. *Given a PUF and an associated key generation rate $R < I(X;Y)$, there exists a linear code such that*

$$\Pr(\hat{W}^k \neq W^k) \to 0 \quad \textit{as } n \to \infty.$$

Proof. The basic idea is to generate the bins as the cosets of a “good" parity-check code. Let an (n,k) binary parity check code specified by the $(n-k) \times n$ binary parity-check matrix H. The code $\mathcal{C} = \{c^n\}$ contains all n-length binary vectors c^n whose syndrome $s^{n-k} \triangleq Hc^n$ is equal to zero, where here multiplication and addition are modulo 2. Assuming that all rows of H are linearly independent, there are 2^k codewords in $\mathcal{C}$, so the code rate is $(\log|\mathcal{C}|)/n = k/n$. Given some general syndrome $s^{n-k} \in \{0,1\}^{n-k}$, the set of all n-length vectors x^n satisfying $Hx^n = s^{n-k}$ is called a coset $\mathcal{C}_s$. Define a decoding function $f(s^{n-k})$, where $f : \{0,1\}^{n-k} \to \{0,1\}^n$, is equal to the vector $v^n \in \mathcal{C}_s$ with the minimum Hamming weight, where ties are broken evenly. It follows from linearity that the coset is a shift of the code $\mathcal{C}$ by the vector v^n, i.e.,

$$\mathcal{C}_s \triangleq \{x^n : Hx^n = s^{n-k}\} = \{c^n \oplus v^n : c^n \in \mathcal{C}\} \triangleq \mathcal{C}^v,$$

where the n-length vector $v^n = f(s^{n-k})$ is the coset leader.

Decoding of this parity-check code amounts to quantizing y^n to the nearest vector in $\mathcal{C}$ with respect to the Hamming distance. This vector, $\hat{x}^n$, can be computed by syndrome decoding using the function f

$$\hat{x}^n = y^n \oplus \hat{z}^n, \quad \hat{z}^n = f(Hy^n) = y^n \bmod \mathcal{C}. \tag{3}$$

We may view the decoder above as a partition of $\{0,1\}^n$ to 2^k decision cells of size 2^{n-k} each, which are all shifted versions of the basic “Voronoi” set

$$\{z^n : z^n \oplus f(Hz^n) = 0\} \triangleq \Omega_0.$$

Each of the 2^{n-k} members of Ω_0 is a coset leader for a different coset. The enrollment and authentication procedures

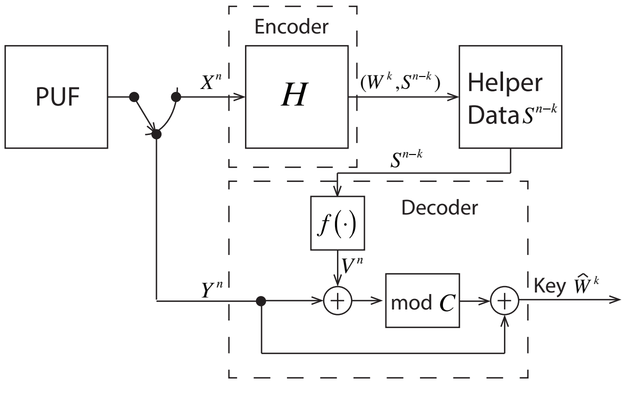

Fig. 2. The GS model for PUF key generation system with algebraic binning

with this algebraic binning are summarized in Algorithm 1. And the corresponding system model is shown in Figure 2.

Since our PUF model $Y = X \oplus Z$ is a BSC with crossover probability p. We are interested in "good" parity check codes over BSC(p) that are capacity achieving, i.e., they have a rate R arbitrarily close to $1 - H(p)$ for n large enough. Note that $H(X|Y) = H(p)$ and $H(X) = 1$, together with Algorithm 1 will grant us the desired result.

Algorithm 1 Algebraic Binning

```
procedure ENROLLMENT($x^n$)
    Store the syndrome $s^{n-k} = Hx^n$ as helper data.
end procedure
procedure AUTHENTICATION($y^n, s^{n-k}$)
    Find the coset leader $v^n = f(s^{n-k})$.
    Find the error between $y^n$ and the coset $\mathcal{C}_s$
    $\hat{z}^n = (v^n \oplus y^n) \bmod \mathcal{C}$.
    Reconstruct the key $\hat{w}^k = y^n \oplus \hat{z}^n \oplus v^n$.
end procedure
```

First, we show for any enrollment sequence x^n and the corresponding PUF output y^n, the block error probability is vanishingly small. Note that the decoding computation in Algorithm 1 is unique, so unlike in random binning we never have ambiguous decoding. Hence, noting from the PUF model $z^n = x^n \oplus y^n$ and from (3) that $\hat{z}^n = f(H(x^n \oplus y^n)) = f(Hz^n)$, a decoding error event amounts to $\{\hat{w}^k \neq w^k\} \leftrightarrow \{\hat{x}^n \neq x^n\} \leftrightarrow \{\hat{z}^n \neq z^n\}$ so the probability of decoding error is

$$\Pr\{\hat{W}^k \neq W^k\} = \Pr\{\hat{X}^n \neq X^n\} = \Pr\{f(HZ^n) \neq Z^n\}$$

which by good BSC(p) code is smaller than ϵ.

Next, we show the optimal rate is $I(X;Y)$. Because the total number of typical sequences are $2^{nH(X)}$, maximizing the key generation rate R is equivalent to minimize the number of bins (cosets)

$$\max R = \max\ \frac{k}{n} = \max\ 1 - \frac{n-k}{n}. \tag{4}$$

Here we need the following Slepian-Wolf bound for distributed source coding.

Theorem 2 (Slepian-Wolf [5]). *For the distributed source coding problem for the source (X, Y) drawn i.i.d.$\sim p(x, y)$, the achievable rate region is given by*

$$R_1 \geq H(X|Y),$$
$$R_2 \geq H(Y|X),$$
$$R_1 + R_2 \geq H(X,Y).$$

To establish the connection between GS model and Slepian-Wolf problem, we see the two PUF outputs X and Y are the correlated source for Slepian-Wolf problem, and the number of bins in GS model is equivalent to the rate of the first source in Slepian-Wolf problem

$$\frac{n-k}{n} = R_1 \geq H(X|Y). \tag{5}$$

Combing Equ.(4) and (5), we have the optimal rate as $1 - H(X|Y) = H(X) - H(X|Y) = I(X;Y)$. The optimality is guaranteed by the Slepian-Wolf bound. □

Remark 1. *The proof shows the key generation rate $I(X;Y)$ is achievable with a "good" coset partition, in a sense that each coset is a "good" parity check code over BSC(p). It is a general statement, as long as one can find the "good" coset partition with each coset a "good" parity check code for some channel, the key generation rate $I(X;Y)$ is achievable for that channel.*

B. Polar Codes for PUFs

Polar codes are popular linear block codes, introduced by Arikan in [7]. A binary polar code can be specified by $(N, K, \mathcal{F}, u^{\mathcal{F}})$, where $N = 2^n$ is the block length, K is the number of information bits encoded per codeword, $\mathcal{F}$ is the set of indices of the $N - K$ frozen bits and $u^{\mathcal{F}}$ is a vector of frozen bits, which is known to both encoder and decoder.

1) Encoding of Polar Codes: For an $(N, K, \mathcal{F}, u^{\mathcal{F}})$ polar code, the encoding operation for a message vector u^N, is performed using a generator matrix,

$$G_N = B_N G_2^{\otimes \log N},$$

where B_N is a *bit-reversal* permutation matrix, $G_2 = \begin{bmatrix} 1 & 0 \\ 1 & 1 \end{bmatrix}$ and $\otimes$ denotes the Kronecker product.

Given a message vector u^N, the codewords are generated as

$$x^N = u^{\mathcal{F}^c}(G_N)_{\mathcal{F}^c} \oplus u^{\mathcal{F}}(G_N)_{\mathcal{F}},$$

where $\mathcal{F}^c \triangleq \{1, 2, \ldots, N\} \backslash \mathcal{F}$ corresponds to the information bits indices. So $u^{\mathcal{F}^c}$ are the information bits and $u^{\mathcal{F}}$ are the frozen bits.

2) Decoding of Polar Codes: Polar codes achieve the channel capacity asymptotically in code length, when decoding is done using the successive-cancellation (SC) decoding algorithm. The SC decoder observes $(y^N, u^{\mathcal{F}})$ and generates an estimate $\hat{u}^N$ of u^N. The ith bit of the estimate $\hat{u}^N$ depends on the channel output y^N and the previous bit decisions $\hat{u}_1, \hat{u}_2, \ldots, \hat{u}_{i-1}$, denoted by $\hat{u}^{i-1}$. It uses the following decision rules,

$$\hat{u}_i = \begin{cases} u_i & \text{if } i \in \mathcal{F} \\ 0 & \text{if } i \in \mathcal{F}^c \text{ and } L^{(i)}(y^N, \hat{u}^{i-1}) \geq 1 \\ l & \text{if } i \in \mathcal{F}^c \text{ and } L^{(i)}(y^N, \hat{u}^{i-1}) < 1 \end{cases}$$

where $L_N^{(i)}(y^N, \hat{u}^{i-1}) = \mathbb{P}(y^N, \hat{u}^{i-1}|0)/\mathbb{P}(y^N, \hat{u}^{i-1}|1)$ is the ith likelihood ratio (LR) at length N. We omit further details in SC decoding for limited space, readers can get the full knowledge of SC decoding in [7].

C. Applying Polar Codes to PUFs

Given the PUF model as a BSC(p), block length N and rate K/N, we have the polar code with parameters $(N, K, \mathcal{F})$. And the algebraic binning with polar code is shown in Algorithm 2.

Algorithm 2 Algebraic Binning with Polar Code Construction

```
procedure ENROLLMENT($x^N, (N, K, \mathcal{F})$)
    Store the syndrome $s^{N-K} = ((G_N^T)^{-1} x^N)_{\mathcal{F}}$ as helper data.
end procedure
procedure AUTHENTICATION($y^N, s^{N-K}, (N, K, \mathcal{F})$)
    Reconstruct the key $\hat{w}^K = SCdec(y^n, s^{N-K})$.
end procedure
```

Theorem 3. *For PUF, every key generation rate $R < I(X;Y)$, there exist a polar encoder and decoder, such that*

$$\sum_{s^{N-K} \in \mathcal{X}^{N-K}} \frac{1}{2^{N-K}} \sum_{w^K \in \mathcal{X}^K} \frac{1}{2^K} \Pr(\hat{w}^K \neq w^K) = O(N^{-\frac{1}{4}}). \tag{6}$$

Proof. As introduced in [7], polar code can be represented as

$$\begin{aligned} x^N &= u^{\mathcal{F}^c}(G_N)_{\mathcal{F}^c} \oplus u^{\mathcal{F}}(G_N)_{\mathcal{F}} \\ &= w^K (G_N)_{\mathcal{F}^c} \oplus s^{N-K}(G_N)_{\mathcal{F}}. \end{aligned}$$

By inversion of G_N, we have the syndrome and the key as

$$\begin{aligned} s^{N-K} &= ((G_N^T)^{-1} x^N)_{\mathcal{F}}, \\ w^K &= ((G_N^T)^{-1} x^N)_{\mathcal{F}^c}. \end{aligned}$$

Now for each PUF observation x^N, we treat it as a codeword of polar code with parameters $(N, K, \mathcal{F}, s^{N-K})$. So we use a set of polar code $\mathcal{C} = \{C(N, K, \mathcal{F}, s^{N-K}) : s^{N-K} \in \mathbb{F}_2^{N-K}\}$. Because G_N has full rank, for any $x^N \in \mathbb{F}_2^N$, $x^N \in \mathcal{C}$, and $C \subseteq \mathbb{F}_2^N$. So $\mathcal{C} = \mathbb{F}_2^N$. We proved that C is a coset partition of $\mathbb{F}_2^N$ and each coset code $C(N, K, \mathcal{F}, s^{N-K})$ with coset leader s^{N-K} is a channel code for the channel. According to the Theorem 3 in [7], we have for rate $R < I(X;Y)$, the block error probability for polar coding under successive cancellation decoding satisfies

$$\sum_{s^{N-K} \in \mathbb{F}_2^{N-K}} \frac{1}{2^{N-K}} \sum_{w^K \in \mathbb{F}_2^K} \frac{1}{2^K} \Pr(\hat{w}^K \neq w^K) = O(N^{-\frac{1}{4}}).$$

Although polar codes cannot guarantee each coset code is a good channel code such that

$$\Pr(\hat{W}^K \neq W^K) = o(1),$$

on average, we obtain a good coset partition as required by Theorem 1. □

D. Achievable Scheme for Unquantized PUFs: The Gaussian Case

As mentioned earlier, a vast majority of PUF outputs are quantized to a binary alphabet right after generation. However, for the unquantized PUF model $\tilde{Y} = \tilde{X} + \tilde{Z}$ (again, we abuse the notations such that $\tilde{Y}, \tilde{X}$ are outputs of the PUF), we use a lattice based coding scheme as below.

Definitions: Lattice Λ is a discrete subgroup of $\mathbb{R}^n$. Quantization with respect to Λ is $Q_\Lambda(\tilde{x}^n) = \arg\min_{\lambda \in \Lambda} \|\tilde{x}^n - \lambda\|$. Fundamental Voronoi region of Λ is $\mathcal{V}(\Lambda) = \{\tilde{x}^n : Q_\Lambda(\tilde{x}^n) = 0\}$. Volume of the Voronoi region of Λ is $V(\Lambda) = \int_{\mathcal{V}(\Lambda)} d\tilde{x}^n$. Normalized second moment of Λ is $G(\Lambda) = \frac{\sigma^2(\Lambda)}{V(\Lambda)^{2/n}}$ where $\sigma^2(\Lambda) = \frac{1}{nV(\Lambda)} \int_{\mathcal{V}(\Lambda)} \|\tilde{x}^n\|^2 d\tilde{x}^n$. A pair of Lattices (Λ, Λ_0) are said to be nested if $\Lambda \subseteq \Lambda_0$.

We use nested lattices $\Lambda \subseteq \Lambda_0$ for coset partitioning and algebraic binning. The encoder block with input $\tilde{x}^n$ and output $\tilde{d}^n$ is implemented by lattice modulo operation

$$\tilde{d}^n = [\tilde{x}^n] \bmod \Lambda_0 = \tilde{x}^n - Q_{\Lambda_0}(\tilde{x}^n).$$

We use $\tilde{d}^n$ as a helper data. As in Figure 2, for the decoder block with input $\tilde{y}^n$, helper data $\tilde{d}^n$, and output $\tilde{v}^n$, we perform

$$\tilde{v}^n = Q_{\Lambda_0}([\tilde{y}^n - \tilde{d}^n] \bmod \Lambda).$$

Since

$$\tilde{y}^n = \tilde{x}^n + \tilde{z}^n = \tilde{t}^n + \tilde{d}^n + \tilde{z}^n$$

where $\tilde{t}^n = Q_{\Lambda_0}(\tilde{x}^n) \in \Lambda_0$, the decoder output is

$$\tilde{v}^n = Q_{\Lambda_0}([\tilde{t}^n + \tilde{z}^n] \bmod \Lambda).$$

If we use nested lattices satisfying $\tilde{z}^n \in \mathcal{V}(\Lambda_0)$ with high probability, it follows that

$$Q_{\Lambda_0}([\tilde{t}^n + \tilde{z}^n] \bmod \Lambda) = Q_{\Lambda_0}(\tilde{t}^n)$$

with high probability. Since

$$Q_{\Lambda_0}(\tilde{t}^n) = \tilde{t}^n = Q_{\Lambda_0}(\tilde{x}^n),$$

it also means that

$$Q_{\Lambda_0}([\tilde{y}^n - \tilde{d}^n] \bmod \Lambda) = Q_{\Lambda_0}(\tilde{x}^n)$$

with high probability. In other words, the helper data cancels the effect of noise $\tilde{z}^n$.

The lattice codebook is defined by the set $\Lambda_0 \cap \mathcal{V}(\Lambda)$. The code rate is given by $R = \frac{1}{n} \log\left(\frac{V(\Lambda)}{V(\Lambda_0)}\right)$ where $V(\cdot)$ is the volume of the fundamental Voronoi region of a lattice. We use nested lattices with parameters $\sigma^2(\Lambda) = P$, $G(\Lambda) = \frac{1}{2\pi e}$, and $V(\Lambda) = (2\pi e P)^{\frac{n}{2}}$. Nested lattices good for Gaussian channel coding [14] can be used to achieve a rate up to $R = \frac{1}{2} \log\left(\frac{P}{N}\right)$ with vanishing error probability. In practice, polar lattices [12], [13] can be used for polynomial-time processing.

IV. Comparisons with Existing Methods

There are several existing method proposed for the GS model.

The authors et al. [9] establish the connection between Wyner-Ziv problem and the GS model. They describe the key-leakage-storage region for GS model. However, for GS model, according to the definitions, storage rate and privacy rate are the same since $I(X^n; W) = H(W) - H(W|X^n) = H(W)$, where W is a function of X^n in GS model. It is also reflected in Theorem 1 in [9] as R_l and R_w have the same bound. So the key-leakage-privacy region can be treated as key-leakage region or key-storage region. We describe the optimal point of the key-storage region by the algebraic binning argument. The authors et al. [9] also show a polar code construction based on the nested polar code in [10] to achieve the key-leakage-storage region, which give the maximal key generation rate as $1 - H_b(q * p)$ for given PUF noise as a BSC(p), where $q \in [0, 0.5]$ is a chosen parameter for the first step *vector quantization* (VQ) in the nested polar code. Since $H_b(q*p) \geq H_b(p)$, we have our optimal rate greater than their optimal rate $1-H_b(p) \geq 1-H_b(q*p)$, and the storage $H_b(p) \leq H_b(q*p)$. Notice that the both equalities can be achieved if $q = 0$, but at this point they lose the nested polar code construction. The reason for the degradation in their result is that there exists a gap between Wyner-Ziv problem having distortion (reflected as the first step VQ in the nested polar code construction) and the GS model requiring an exact recovery of the key. So the VQ step introducing the distortion is not necessary for GS model. In all, we offer a better rate with a simpler implementation.

The authors et al. [11] offer an LDPC based scheme for PUF. Their scheme does not optimize the key generation rate since the LDPC does not necessarily form a coset partition.

V. Simulation Results

We simulate the system in Figure 2 with the polar code construction in Section III-C with MATLAB. If we use PUFs in a field programmable gate array (FPGA) as the randomness source, we must satisfy a block error probability P_B of at most 10^{-6} [16]. Consider a BSC(p) with crossover probability $p = 0.15$, which is a common value for SRAM PUFs.

First, we consider the block length $N = 1024$ and we design polar code with rate $K/N = 128/1024 = 0.125$ for the BSC(p) channel. We evaluate the block error probability performance of this code with SC decoder and SC list (SCL) decoder with list size 8 respectively for a BSC with a range of crossover probability, as shown in Figure. 3. It shows the SCL decoder has better performance, and achieves a block error probability of $P_B = 10^{-6}$ at a crossover probability 0.2. For comparison, we achieve the key generation rate 0.125 with crossover probability and block error probability $(0.2, 10^{-6})$, better than the crossover probability and block error probability tuple $(0.1819, 10^{-6})$ in [9].

VI. Conclusion

By algebraic binning, we show that "good" coset partition is needed to achieve the optimal key generation rate for PUFs. Thus we offer a principle in general for PUF key generation system design. And we design a polar code-based system for PUFs that achieve better key generation rate than existing methods. In future work, we will further study the "good" code for unquantized PUFs.

Acknowledgment

This work was supported by the NSF and the ONR.

References

[1] Gassend, Blaise, Dwaine Clarke, Marten Van Dijk, and Srinivas Devadas. "Silicon physical random functions." In *Proceedings of the 9th ACM conference on Computer and communications security*, pp. 148-160. ACM, 2002.

[2] Suh, G. Edward, and Srinivas Devadas. "Physical unclonable functions for device authentication and secret key generation." In *Proceedings of the 44th annual Design Automation Conference*, pp. 9-14. ACM, 2007.

[3] Guajardo, Jorge, Sandeep S. Kumar, Geert Jan Schrijen, and Pim Tuyls. "FPGA intrinsic PUFs and their use for IP protection." In *CHES*, vol. 4727, pp. 63-80. 2007.

[4] Pelgrom, Marcel JM, Aad CJ Duinmaijer, and Anton PG Welbers. "Matching properties of MOS transistors." *IEEE Journal of solid-state circuits* vol. 24, no. 5 (1989): 1433-1439.

[5] Cover, Thomas M., and Joy A. Thomas. *Elements of information theory*. John Wiley & Sons, 2012.

[6] Ignatenko, Tanya, and Frans MJ Willems. "Biometric systems: Privacy and secrecy aspects." *IEEE Transactions on Information Forensics and Security* vol. 4, no. 4 (2009): 956-973.

[7] Arikan, Erdal. "Channel polarization: A method for constructing capacity-achieving codes for symmetric binary-input memoryless channels." *IEEE Transactions on Information Theory* vol. 55, no. 7 (2009): 3051-3073.

[8] B. Chen, T. Ignatenko, F. M. Willems, R. Maes, E. van der Sluis, and G. Selimis, "A robust SRAM-PUF key generation scheme based on polar codes," July 2017, [Online]. Available: arxiv.org/abs/1701.07320.

[9] Onur Gunlu, Onurcan Iscan, Vladimir Sidorenko, and Gerhard Kramer, "Wyner-Ziv Coding for Physical Unclonable Functions and Biometric Secrecy Systems," Sep 2017, [Online]. Available: https://arxiv.org/abs/1709.00275

[10] Korada, Satish Babu, and Rudiger Urbanke. "Polar codes for slepian-wolf, wyner-ziv, and gelfand-pinsker." In *Information Theory (ITW 2010, Cairo), 2010 IEEE Information Theory Workshop on*, pp. 1-5. IEEE, 2010.

[11] Muelich, Sven, and Martin Bossert. "A New Error Correction Scheme for Physical Unclonable Functions." In *SCC 2017; 11th International ITG Conference on Systems, Communications and Coding; Proceedings of*, pp. 1-6. VDE, 2017.

[12] Yan, Yanfei, Cong Ling, and Xiaofu Wu. "Polar lattices: where Ar?kan meets Forney." In *Information Theory Proceedings (ISIT), 2013 IEEE International Symposium on*, pp. 1292-1296. IEEE, 2013.

[13] Y. Yan, L. Liu, and C. Ling, "Polar lattices for strong secrecy over the mod-Λ Gaussian wiretap channel," Jan. 2014. [Online] Available: arXiv:1401.4532 [cs.IT]

[14] U. Erez and R. Zamir, "Achieving $\frac{1}{2}\log(1 + \mathrm{SNR})$ on the AWGN channel with lattice encoding and decoding," *IEEE Trans. Inf. Theory*, vol. 50, no. 10, pp. 2293–2314, Oct. 2004.

[15] Wyner, Aaron, and Jacob Ziv. "The rate-distortion function for source coding with side information at the decoder." *IEEE Transactions on information Theory* vol. 22, no. 1 (1976): 1-10.

[16] Bosch, Christoph, et al. "Efficient helper data key extractor on FPGAs." *Cryptographic Hardware and Embedded Systems CHES 2008* (2008): 181-197.

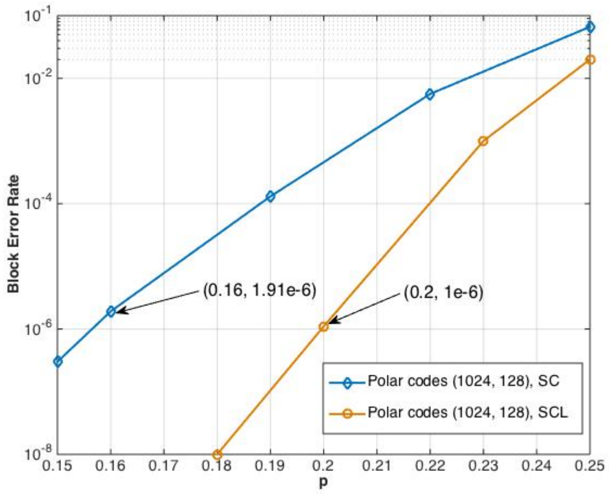

Fig. 3. Block error probability over a BSC(p) for polar codes with SC decoder and SCL decoder, respectively.