2019 57th Annual Allerton Conference on Communication, Control, and Computing
(Allerton)
Allerton Park and Retreat Center
Monticello, IL, USA, September 24-27, 2019

Entropy Estimation of Physically Unclonable Functions via Chow Parameters

Alexander Schaub Olivier Rioul
LTCI, Telecom Paris, Institut Polytechnique de Paris, France
firstname.lastname@telecom-paris.fr

Joseph J. Boutros
Texas A&M University, 23874 Doha, Qatar
boutros@tamu.edu

***Abstract*— A physically unclonable function (PUF) is an electronic circuit that produces an intrinsic identifier in response to a challenge. These identifiers depend on uncontrollable variations of the manufacturing process, which make them hard to predict or to replicate. Various security protocols leverage on such intrinsic randomness for authentification, cryptographic key generation, anti-counterfeiting, etc. Evaluating the entropy of PUFs (for all possible challenges) allows one to assess the security properties of such protocols.**

In this paper, we estimate the probability distribution of certain kinds of PUFs composed of n delay elements. This is used to evaluate relevant Rényi entropies and determine how they increase with n. Such a problem was known to have extremely high complexity (in the order of 2^{2^n}) and previous entropy estimations were carried out up to $n = 7$. Making the link with the theory of Boolean threshold functions, we leverage on the representation by Chow parameters to estimate probability distributions up to $n = 10$. The resulting Shannon entropy of the PUF is close to the max-entropy, which is asymptotically quadratic in n.

I. Introduction

Physically unclonable functions, or PUFs, are electronic devices that are used to produce unique identifiers. Small variations of the manufacturing process are exploited so that any two devices, built according to the same description, will likely produce different identifiers. Moreover, since such process variations are intrinsically random, they cannot be controlled to replicate the behavior of another device, hence the name *physically unclonable* functions. PUFs find many applications: the identifier can be used to generate a unique cryptographic key, which cannot be easily extracted from the device; it can be recorded during manufacturing into a whitelist to prevent counterfeiting or overproduction; and it can also be employed in the implementation of challenge-response protocols at a low cost. This is especially valuable on devices where implementing asymmetric cryptography primitives is too computationally expensive.

There are several ways to build PUFs. SRAM-PUFs [1] exploit the states of SRAM cells after powering up, while ring-oscillator (RO) PUFs [2] exploit delay differences of signals in electronic circuits. In this paper, we analyze another delay PUF, called loop-PUF, first proposed in [3]. Our analysis will also be valid for the RO-sum PUF [4], which shares essentially the same mathematical model, as well as the arbiter PUF [5]. In the remainder of this paper, we will write PUF as a short-hand for loop-PUF, RO-sum PUF or arbiter PUF.

A. Modelization and Notations

A PUF of *size* n generates one identifier bit, or *response* bit, when queried with a *challenge* $c = (c_1, \dots, c_n) \in \{\pm 1\}^n$, a sequence of n values $+1$ or -1. The PUF is characterized by n weights, denoted by $x = (x_1, \dots, x_n) \in \mathbb{R}^n$ that represent *delay differences* of the PUF circuit. As explained in [6], for each challenge $c \in \{\pm 1\}^n$, the response bit of the PUF of parameters $x = (x_1, \dots, x_n)$ is equal to $\operatorname{sgn}(c \cdot x) = \operatorname{sgn}(c_1 x_1 + \cdots + c_n x_n) \in \{\pm 1\}$.

The base for all logarithms in this paper is equal to 2, and all entropies are given in bits.

Due to manufacturing process variations, the weights x_i are modeled as realizations of random variables X_i. In [6], a Gaussian model was analyzed, where the Gaussian nature of the variables $X_i \sim \mathcal{N}(0, 1)$ is justified by simulations of process variations in electronic circuits [7].

More generally, our analysis is valid for any $X = (X_1, X_2, \dots, X_n)$ whose components X_i are i.i.d. continuous variables with symmetric densities about 0 (whose support contains 0). The i.i.d. assumption is justified by the fact that delays are caused by "identical" circuit elements that lie in different locations in the circuit and can, therefore, be considered independent. In particular, each x_i is the difference between two delays caused by such "identical" independent elements, which justifies the symmetry assumption. Simulations in Section V will be made in the Gaussian model, for which the weight distribution is centered isotropic.

B. Problem Statement

The security of PUFs is related to Rényi entropies H_α of various orders α [8].

The min-entropy $H_\infty = \log(1/P_{\max})$ is related to the maximum (worst-case) probability $P_{\max}$ of successfully cloning a given PUF. Therefore, min-entropy H_∞ should be as large as possible to ensure a given worst-case security level.

The collision entropy $H_2 = \log(1/P_{\text{eq}})$ is related to the average probability P_{eq} that two randomly chosen PUFs have the same identifier. Therefore, H_2 should also be as large as possible to ensure a given average security level against collision.

The classical Shannon's entropy H_1 is known to provide a resistance criterion against modeling attacks—which predict the response to a new challenge given previous responses to other challenges [9]. Again H_1 should be as large as possible.

The max-entropy H_0 is simply the logarithm of the total number of PUFs. H_0 upper bounds all the other entropies H_α. Theoretically, it is possible to choose a non i.i.d. weight distribution such that all PUFs are equiprobable, yielding $H_\alpha = H_0$ for every α. In this case it is sufficient to count PUFs. In practice, however, due to the assumption of i.i.d. weights (typically Gaussian), the upper bound H_0 will not be attained. Therefore, it is important to derive a efficient method to estimate the various Rényi entropies.

Estimating the various Rényi entropies typically requires estimating the entire PUF probability distribution. However, because a PUF of size n is determined by 2^n response bits, there can be as many as 2^{2^n} PUFs of size n. The naive complexity increases very rapidly with n, in the order of 2^{2^n}.

C. Outline

In this paper, we link the analysis of PUFs to the theory of Boolean Threshold Functions (BTF) and build an algorithm that accurately estimates the PUF probability distribution and entropies up to order $n = 10$. Our algorithm relies on determining equivalent classes of PUFs with the same probability, and then estimating the probability within each class. The classes are determined using Chow parameters from BTF theory. The remainder of the paper is thus organized as follows. Section II recalls known results from the theory of BTFs which we adapt to PUFs. The key results on the equivalence classes are proved in Section III. Section IV describes the simulation algorithm that allows us in Section V to determine the PUF distributions and entropies up to order 10. Finally, Section VI concludes.

II. The Chow Parameters of PUFs

Definition 1 (PUF): Let $x \in \mathbb{R}^n$ be such that for all $c \in \{\pm 1\}^n$, $c \cdot x \neq 0$. The PUF of *size* n and *weight* sequence x is the function $f_x : \{-1, +1\}^n \to \{-1, +1\}$ defined as

$$f_x(c) = \operatorname{sgn}(c \cdot x) \tag{1}$$

where $c \cdot x = \sum_{i=1}^n c_i x_i$ is the usual scalar product.

This definition coincides with so-called "self-dual" BTFs of n variables [10]. BTFs have been studied since the 1950's as building blocks for Boolean circuits [11] and also find applications in machine learning [12]. Leveraging the correspondence between PUFs and BTFs, we adapt fundamental results from BTF theory to conveniently characterize PUFs.

A. All PUFs are Attainable

Recall that in our framework, the PUF parameters $x \in \mathbb{R}^n$ are realizations of a random vector $X \in \mathbb{R}^n$. Under this probabilistic model a PUF becomes a randomized mapping f_X such that $f_X(c) = \operatorname{sgn}(c \cdot X)$ for any (deterministic) challenge $c \in \{\pm 1\}^n$.

Lemma 1: For every PUF f_x, we have $\mathbb{P}(f_X = f_x) > 0$. In other words, every PUF f_x can be reached by a realization of weights X with positive probability (even though one has $\mathbb{P}(X = x) = 0$).

Proof: By assumption all components of X are i.i.d. with symmetric density of support S containing 0. Hence the support S^n of the density of X is an n-dimensional manifold containing the origin in its interior. Let $x \in \mathbb{R}^n$ be fixed and let C_x be the cone (scale-invariant set) of all $y \in \mathbb{R}^n$ such that $f_x = f_y$. This cone C_x has apex 0 and contains the intersection of all half-spaces $\{y \mid \operatorname{sgn}(c \cdot y) = \operatorname{sgn}(c \cdot x)\}$ where $c \in \{\pm 1\}^n$. Therefore, it is a n-dimensional manifold which intersects S^n with positive volume. Hence $\mathbb{P}(f_X = f_x) = \mathbb{P}(X \in C_x \cap S^n) > 0$. ∎

B. Chow Parameters Characterize PUFs

First introduced by Chow [13] and later studied by Winder [11] who gave them their name, the so-called Chow parameters uniquely define a Boolean threshold function. Their definition is especially simple for PUFs:

Definition 2 (Chow parameters): The *Chow parameters* $p = (p_1, \ldots, p_n) \in \mathbb{Z}^n$ of a PUF f of size n is defined as

$$p = \sum_{c \mid f(c)=1} c \tag{2}$$

where the vector sum is carried out componentwise.

We remark that for $n \geq 2$, all Chow parameters are *even integers*. This is due to the fact that a sum of even number of elements ± 1 must be even. More precisely,

$$p_i \bmod 2 \equiv \sum_{c \mid f(c)=1} c_i \bmod 2 \equiv \sum_{c \mid f(c)=1} 1 \bmod 2 \tag{3}$$

$$\equiv 2^{n-1} \bmod 2 \equiv 0 \bmod 2. \tag{4}$$

Theorem 1 (Chow's theorem [13]): Two PUFs with the same Chow parameters are identical.

For completeness, we give a new proof of Chow's theorem rewritten in our PUF framework. Such proof turns out to be very simple.

Proof: Let f_x and f_y be two PUFs with identical Chow parameters:

$$\sum_{c \mid f_x(c)=1} c = \sum_{c \mid f_y(c)=1} c. \tag{5}$$

Simplifying this expression by $\sum_{\substack{c \mid f_x(c)=1,\\ f_y(c)=1}} c$, we obtain

$$\sum_{\substack{c \mid f_x(c)=1,\\ f_y(c)=-1}} c = \sum_{\substack{c \mid f_x(c)=-1,\\ f_y(c)=1}} c, \tag{6}$$

which is equivalent to

$$\sum_{c \mid f_x(c) \neq f_y(c)} f_x(c) c = 0. \tag{7}$$

Taking the scalar product with x, we get

$$\sum_{c \mid f_x(c) \neq f_y(c)} f_x(c) c \cdot x = \sum_{c \mid f_x(c) \neq f_y(c)} |c \cdot x| = 0 \tag{8}$$

which implies $c \cdot x = 0$ whenever $f_x(c) \neq f_y(c)$. Now we assumed that $c \cdot x$ is never zero by Def. 1. Thus $f_x = f_y$. ∎

C. Consequence on the Max-Entropy

An upper bound on the max-entropy can be easily deduced from Chow's theorem.

Corollary 1: There are no more than 2^{n^2} PUFs of size n, i.e., the max-entropy of the PUF of size n satisfies

$$H_0(n) \leq n^2 \qquad (\forall n \geq 2). \tag{9}$$

A more refined version, which can be rewritten as $H_0(n) \leq (n-1)^2+1$ for $n > 1$, can be found in [14, Corollary 10.2]. The proof of (9) is again particularly simple for PUFs.

Proof: The Chow parameters p_i, $i = 1 \dots n$, satisfy

$$p_i = \sum_{c|f(c)=1} c_i \leq \sum_{\substack{c|f(c)=1,\\ c_i=1}} 1 \leq 2^{n-1} \tag{10}$$

and similarly, $p_i \geq -2^{n-1}$. Since there are $2^{n-1}+1$ even integers between -2^{n-1} and 2^{n-1}, there can only be $(2^{n-1}+1)^n \leq 2^{n^2}$ different values taken by the Chow parameters. The conclusion follows from Chow's Theorem 1. ∎

A lower bound on H_0 is also easily found from the representation of Definition 1, as given by the following Proposition. The corresponding bound for the number of BTFs was first established independently by Smith [15] and Yajima et al. [16] in the 1960s.

Proposition 1: The max-entropy satisfies

$$H_0(n) > \frac{(n-2)^2}{2} \quad (\forall n \geq 2). \tag{11}$$

Proof: Recall from Lemma 1 that every PUF f_x can be reached by a realization of weights X with positive probability. Hence it is sufficient to consider all f_x for all $x \in \mathbb{R}^n$ in order to lower-bound the total number of PUFs.

Let f_x a PUF of size n. Applying some small perturbation on x if necessary (without affecting f_x) we may always assume that all the $c \cdot x$ ($c \in \{\pm 1\}^n$) take distinct values.

Now let $x_{n+1} \in \mathbb{R}$ be such that $2x_{n+1}$ is different from all the $c \cdot x$, and define $x' = (x_1, \cdots, x_{n-1}, x_n - x_{n+1}, x_{n+1})$. For any challenge $c' = (c_1, \dots, c_n, c_{n+1})$, we have

$$f_{x'}(c') = \begin{cases} f_x(c_1, \dots, c_n) & \text{if } c_n = c_{n+1} \\ \operatorname{sgn}(\sum_{i=1}^n c_i x_i - 2c_n x_{n+1}) & \text{otherwise.} \end{cases} \tag{12}$$

Depending on how many of the 2^{n-1} values of $c \cdot x$ are smaller/larger than $2c_n x_{n+1}$, we can construct $2^{n-1}+1$ different PUF functions of size $n+1$. Hence each PUF of size n gives rise to more than 2^{n-1} PUFs of size $n+1$. Therefore, $H_0(n+1) > n - 1 + H_0(n)$. The result follows by finite induction:

$$H_0(n) > \frac{(n-1)(n-2)}{2} + H_0(2) > \frac{(n-2)^2}{2}. \qquad ∎$$

More recently, Zuev [17] has shown that, asymptotically, $H_0(n) > n^2(1 - \frac{10}{\ln(n)})$. Therefore, for the max-entropy, we have that $H_0(n) \sim n^2$. As a result, instead of evaluating the probabilities of 2^{2^n} different PUFs, we will only have to evaluate about 2^{n^2}.

As apparent in the proof of Zuev [17, Theorem 1] although through different geometrical considerations on normal vectors of hyperplanes, we can further reduce the number of PUFs to be considered down by a factor of about $2^n n!$. Section III will derive the exact compression factor using the equivalence classes on Chow parameters.

D. Order and Sign Stability of Chow Parameters

An important property of the Chow parameters p is that their share the same signs and relative order as the weights x.

Lemma 2: Let $f = f_x$ be a PUF with weight $x \in \mathbb{R}^n$, and $p \in \mathbb{Z}^n$ be the corresponding Chow parameters. Then

- $x_i \geq 0 \implies p_i \geq 0$ and $x_i \leq 0 \implies p_i \leq 0$.
- $x_i \leq x_j \implies p_i \leq p_j$.

A similar result was shown by Chow in [13], although with another definition of Chow parameters. Again we give a simplified proof in the PUF framework.

Proof: We first prove that $x_i \geq 0 \implies p_i \geq 0$, the other case $x_i \leq 0 \implies p_i \leq 0$ being similar. Suppose that $x_i \geq 0$. Let E_i^+ (resp. E_i^-) be the set $\{c \mid f(c) = 1, c_i = 1\}$ (resp. $\{c \mid f(c) = 1, c_i = -1\}$). By definition,

$$p_i = \sum_{c|f(c)=1} c_i = |E_i^+| - |E_i^-|. \tag{13}$$

We show the existence of an injective mapping from E_i^- to E_i^+. Consider the one-to-one mapping $\phi : \{\pm 1\}^n \to \{\pm 1\}^n$ defined by

$$\phi(c)_j = \begin{cases} c_j, & j \neq i \\ -c_j, & j = i \end{cases} \tag{14}$$

For any $c \in E_i^-$, $c_i = -1$, $\phi(c)_i = +1$ and

$$\sum_{j=1}^n \phi(c)_j x_j = \sum_{j \neq i}^n c_j x_j + x_i \tag{15}$$

$$= \underbrace{\sum_{j=1}^n c_j x_j}_{>0} + \underbrace{2x_i}_{\geq 0} > 0. \tag{16}$$

Therefore, $f(\phi(c)) = 1$ and $\phi(c) \in E_i^+$. Hence, the bijection ϕ induces an injection from E_i^- to E_i^+. This implies that $|E_i^+| \geq |E_i^-|$ hence $p_i \geq 0$.

To prove the second part, assume that $x_i \leq x_j$ for $j \neq i$. Let $f' : \{\pm 1\}^{n-1} \to \{\pm 1\}$ be a PUF given by $f'(c') = \operatorname{sgn}(c' \cdot x')$, where $c' \in \{\pm 1\}^{n-1}$ is obtained from c by dropping c_i, $x'_\ell = x_\ell$ for any $\ell \neq j$, and $x'_j = x_j - x_i \geq 0$. Say the Chow parameters of f' is p'. According to the first part of this lemma, we have $p'_j \geq 0$. Now, expand the expression of $p_j - p_i$ as

$$p_j - p_i = \sum_{c|f(c)=1} c_j - \sum_{c|f(c)=1} c_i \tag{17}$$

$$= 2 \sum_{c|f(c)=1, c_j=-c_i} c_j \tag{18}$$

$$= 2 \sum_{c'|f'(c')=1} c'_j = 2p'_j \geq 0. \tag{19}$$

∎

III. Equivalence Classes and Chow Parameters

Since the X_i are i.i.d. symmetric random variables, the joint probability distribution of the weights $X = (X_1, \ldots, X_n)$ is invariant under permutations and sign changes. Therefore, all PUFs f_x that can be obtained from one another by permuting or changing signs of their weights $x_1, x_2, \ldots, x_n$ can be clustered together into equivalence classes of PUFs with the same probability $\mathbb{P}(f_X = f_x)$.

We now establish several properties of these equivalence classes for PUFs, known as "self-dual" classes [10] in the context of BTFs. Zuev [17] had already mentioned $2^n n!$ elements per class in a special case. Our generalization (Theorem 3) is mentionned in a different form in [18, § 3.1.2] for calculating the total number of BTFs, yet we couldn't find formal proofs published in the literature.

We give a formal definition of the equivalence classes by the action of the group

$$G_n = S_n \times \{-1, +1\}^n \tag{20}$$

where S_n is the symmetric group of order $n!$. An element $g = (\sigma, s) \in G_n$ is determined by the permutation $\sigma \in S_n$ and the sign changes $s \in \{-1, +1\}^n$.

Proposition 2: For any $x = (x_1, \ldots, x_n) \in \mathbb{R}^n$ and $g = (\sigma, s) \in G_n$ define $g \cdot x : G_n \times \mathbb{R}^n \to \mathbb{R}^n$ such that

$$(g \cdot x)_i = s_i x_{\sigma(i)}. \tag{21}$$

This defines a group action of G_n on $\mathbb{R}^n$, where the inner product in G_n is defined by

$$(\sigma_1, s^1) \cdot (\sigma_2, s^2) = (\sigma_1 \circ \sigma_2, (s^1_i s^2_{\sigma_1(i)})_i). \tag{22}$$

Proof: G_n is clearly a group with identity $e = (id, (1, \cdots, 1))$. For any $(\sigma_1, s^1), (\sigma_2, s^2) \in G_n$ and $x \in \mathbb{R}^n$,

$$\begin{aligned}
(\sigma_1, s^1) \cdot ((\sigma_2, s^2) \cdot x) &= (\sigma_1, s^1) \cdot (s^2_i x_{\sigma_2(i)})_i && (23)\\
&= (s^1_i s^2_{\sigma_1(i)} x_{\sigma_1(\sigma_2(i))})_i && (24)\\
&= (\sigma_1 \circ \sigma_2, (s^1_i s^2_{\sigma_1(i)})_i) \cdot x && (25)\\
&= ((\sigma_1, s^1) \cdot (\sigma_2, s^2)) \cdot x. && (26)
\end{aligned}$$

This shows that $g \cdot x$ defines a group action of G_n on $\mathbb{R}^n$. ∎

Thus we can say that the group G_n acts on the PUFs of size n, the action being defined as

$$g \cdot f_x = f_{g \cdot x}. \tag{27}$$

In keeping with Lemma 2, we now show that the group action is carried over to Chow parameters:

Theorem 2: Let f_x a PUF of Chow parameters p, and let $g \in G_n$. The Chow parameters of $f_{g \cdot x}$ is $g \cdot p$.

Proof: Let $g = (\sigma, s) \in G_n$. For any challenge c, we have that $f_x(g^{-1} \cdot c) = f_{g \cdot x}(c)$. Thus,

$$\begin{aligned}
\sum_{c | f_{g \cdot x}(c) = 1} c_i &= \sum_{c | f_x(g^{-1} \cdot c) = 1} c_i = \sum_{c | f_x(c) = 1} (g \cdot c)_i && (28)\\
&= \sum_{c | f_x(c) = 1} s_i c_{\sigma(i)} = s_i p_{\sigma(i)} = (g \cdot p)_i. && (29)
\end{aligned}$$

∎

Changing the signs of the weights or permuting them is reflected by the same operation on the Chow parameters. This allows us to compute the size of the equivalence classes:

Theorem 3: Let f be a PUF with Chow parameters p. Let $m_p(k)$ be the number of Chow parameters equal to k or $-k \in \mathbb{Z}$, and let $\mathrm{Orb}(f) = \{g \cdot f \mid g \in G_n\}$ the *orbit* of f by G_n, that is, the equivalence class containing f. Then

$$|\mathrm{Orb}(f)| = 2^n n! \Big(2^{m_p(0)} \prod_{k \in \mathbb{N}} m_p(k)!\Big)^{-1}. \tag{30}$$

Proof: By applying the well-known *orbit-stabilizer theorem* (see for instance [19, p. 89]), we have

$$|\mathrm{Orb}(f)| = \frac{|G_n|}{|\mathrm{Stab}(f)|} = \frac{|\{\pm 1\}^n| \times |S_n|}{|\mathrm{Stab}(f)|} = \frac{2^n n!}{|\mathrm{Stab}(f)|} \tag{31}$$

where $\mathrm{Stab}(f) = \{g \in G_n \mid g \cdot f = f\}$ is the *stabilizer* of f. The size of the orbit of f can therefore be deduced from the size of its stabilizer. Now the latter can be easily computed: Let $g = (\sigma, s) \in G_n$ such that $g \cdot f = f$. Since $g \cdot p = p$, we have $\sigma(i) = j \iff p_i = s_i \cdot p_j$ and $s_i = \mathrm{sgn}(p_i) \cdot \mathrm{sgn}(p_{\sigma(i)})$ except when $p_i = 0$, in which case s_i is unconstrained. The number of such g is exactly $2^{m_p(0)} \prod_{k \in \mathbb{N}} m_p(k)!$. ∎

IV. Monte-Carlo Algorithm

As seen in the introduction to the previous section, all PUFs in one equivalence class have the same probability. It follows that the probability of any particular PUF can be deduced from the probability of the class to which it belongs. Therefore, to determine the various entropies, it suffices to find a method that estimates the probabilities of the various equivalence classes.

In this section, we propose an algorithm that exploits a definition of a *canonical* PUF in any equivalence class in such a way that for given any PUF, it is trivial to determine the corresponding canonical PUF. As expected, only about $2^{n^2}/2^n n!$ probabilities need to be estimated, instead of approximatively 2^{n^2}.

Definition 3 (Canonical PUF): A *canonical* PUF of n variables is a PUF whose Chow parameters satisfy

$$p_1 \geq p_2 \geq \cdots \geq p_n \geq 0. \tag{32}$$

The *canonical form* of a PUF f is the canonical PUF belonging to the same class, i.e., $f' = g \cdot f$ where $g \in G_n$ is such that f' is canonical.

This notion was first introduced by Winder [11] and is related to the concept of "prime" functions independently studied by Chow [13].

Proposition 3 (Unicity of the canonical PUF): Two canonical PUFs in the same class are equal.

Proof: Since f and f' are in the same equivalence class, their Chow parameters are identical up to sign changes and order. Since both are canonical, the signs and order are fixed. Their Chow parameters are thus identical and $f = f'$. ∎

Proposition 4: Let $x = (x_1, \ldots, x_n)$ be a weight sequence of a PUF $f = f_x$, and let $g \in G_n$ such that $g \cdot x = (x'_1, \ldots x'_n)$ satisfies

$$x'_1 \geq x'_2 \geq \ldots \geq x'_n \geq 0. \tag{33}$$

Then $g \cdot f$ is the canonical form of the PUF f.

Proof: Let us denote by p (resp p') the Chow parameters of f (resp $g \cdot f$). The PUF obtained from weights x' is $g \cdot f$. From Lemma 2, the p'_i satisfy the same ordinal relations and have the same signs as the x'_i. Therefore, f' is a canonical PUF. ∎

These results allow us to efficiently estimate the PUF distribution by Monte-Carlo methods, as described in Algorithm 1. Such an algorithm can be used for any i.i.d. weight distribution with symmetric densities (not necessarily Gaussian).

Algorithm 1: How to estimate the PUF distribution.

Data: $n > 0, nbRounds > 0$
Result: Estimation of PUF probability distribution
Initialize HashMaps counts, proba, size;
for $i \leftarrow 1$ **to** $nbRounds$ **do**
  Generate n realizations $x_1, \ldots, x_n$;
  Sort the absolute values of the x_i to obtain x';
  Compute the Chow parameters p of $f_{x'}$;
  if $p \in$ counts **then**
    counts $[p] \leftarrow$ counts $[p]$ + 1;
  else
    counts $[p] \leftarrow 1$;
  end
end
for $p \in$ counts **do**
  $\text{size}[p] \leftarrow \dfrac{2^n n!}{2^{m_p(0)} \prod_k m_p(k)!}$;
  $\text{proba}[p] \leftarrow \dfrac{\text{counts}[p]}{\text{size}[p] * nbRounds}$;
end
return (proba, size) ;

V. Entropies Estimation

In this section, we present the simulation results in the Gaussian case where the weights X_i are i.i.d. $\sim \mathcal{N}(0,1)$. Exact values were already determined up to $n = 4$ in [20].

A. Estimating the Max-Entropy H_0

According to Lemma 1, every PUF can be attained by some realization of weights. Therefore, the max-entropy of the PUF distribution is simply the logarithm of the total number of PUFs with n weights. This number is equal to the total number of BTFs of $n-1$ variables and has been computed up to $n = 10$ in [18, § 3.1.2], see Table I.

B. Estimating the Shannon Entropy H_1

For any PUF f, let $[f]$ denote the equivalence class of f with cardinality $|[f]|$, $\mathbb{P}(f)$ its probability, F_n the set of all PUFs and F_n/G_n the quotient group induced by the action of the group G_n. Then, letting $\mathbb{P}([f']) = \sum_{f \in [f']} \mathbb{P}(f)$, one has

$$H_1(n) = -\sum_{f \in F_n} \mathbb{P}(f) \log(\mathbb{P}(f)) \tag{34}$$

TABLE I
Exact values of H_0

n	# PUFs	H_0 (bits)
1	2	1
2	4	2
3	14	3.8074…
4	104	6.7004…
5	1882	10.8781…
6	94572	16.5291…
7	15028134	23.8411…
8	8378070864	32.9640…
9	17561539552946	43.9974…
10	144130531453121108	57.0001…

$$= -\sum_{f' \in F_n/G_n} \sum_{f \in [f']} \mathbb{P}(f) \log(\mathbb{P}(f)) \tag{35}$$

$$= -\sum_{f' \in F_n/G_n} \mathbb{P}([f']) \log(\mathbb{P}(f')) \tag{36}$$

$$= -\sum_{f' \in F_n/G_n} \mathbb{P}([f']) \log(\mathbb{P}([f'])) + \mathbb{E}[\log(|[f_X]|)]. \tag{37}$$

In other words, the Shannon entropy of the PUF distribution is simply the sum of the entropy of the equivalence classes and the average of their logarithmic size. The latter term can be estimated using the unbiased empirical mean, where a confidence interval can be determined using Student's t-distribution [21]. The former term, however, is an entropy, for which no unbiased estimator exists [22]. The NSB estimator [23] has a reduced bias and a low variance. However, because we generated much more PUFs than equivalence classes (by a factor of at least 100000), the plug-in estimator, based on the empirical frequency estimates, performs quite well: Its bias can be upper bounded as described in [22] and was found to be less than 0.01 bit. The results are summarized in Table II.

TABLE II
Confidence intervals at the 95% level for H_1
(exact values up to $n = 4$).

n	PUF Sample size	H_1 (bits)
1	—	1
2	—	2
3	—	3.6655…
4	—	6.2516…
5	10^{10}	10.0134 – 10.0156
6	10^{10}	15.1903 – 15.1925
7	10^{10}	21.9856 – 21.9879
8	$2 \cdot 10^{10}$	30.5628 – 30.5645
9	$2 \cdot 10^{10}$	41.0367 – 41.0384
10	$3 \cdot 10^{12}$	53.4737 – 53.4740

C. Estimating the Collision Entropy H_2

The collision entropy was estimated using an unbiased estimator adapted from [24, § 1.4.2]. Let $N_{[f]}$ be the number of PUF samples that belong to the equivalence class of $[f]$ among a number of Poisson-distributed PUFs with parameter

N, and $N^2_{[f]} = N_{[f]} \cdot (N_{[f]} - 1)$. We can compute

$$\mathbb{E}\Big[\sum_{f\in F_n/G_n} \frac{N^2_{[f]}}{|[f]|N^2}\Big] = \sum_{f\in F_n/G_n} \mathbb{E}\Big[\frac{N^2_{[f]}}{N^2}\Big]\frac{1}{|[f]|} \tag{38}$$

$$= \sum_{f\in F_n/G_n} \frac{\mathbb{P}([f])^2}{|[f]|} \tag{39}$$

$$= \sum_{f\in F_n} \mathbb{P}(f)^2 \tag{40}$$

where we used the fact that $\mathbb{E}[\frac{N^2_{[f]}}{N^2}] = \mathbb{P}([f])^2$ from [24, § 2.2]. It follows that

$$\sum_{f\in F_n/G_n} \frac{N^2_{[f]}}{|[f]|N^2}. \tag{41}$$

is an unbiased estimator for the power-sum $\sum_{f\in F_n} \mathbb{P}(f)^2$. As can be also checked, the variance of this estimator admits the same upper bound as the one described in [24, § 1.4.2]. This allows us to determine confidence intervals for the collision entropy as shown in Table III.

TABLE III
CONFIDENCE INTERVALS AT THE 95% LEVEL FOR H_2
(EXACT VALUES UP TO $n = 4$)

n	PUF Sample size	H_2 (bits)
1	—	1
2	—	2
3	—	3.5462…
4	—	5.7105…
5	10^{10}	8.4551 – 8.4568
6	10^{10}	11.5977 – 11.6023
7	10^{10}	14.8819 – 14.89805
8	$2 \cdot 10^{10}$	18.5201 – 18.5753
9	$2 \cdot 10^{10}$	22.0309 – 22.4067
10	$3 \cdot 10^{12}$	25.9070 – 26.1983

D. Estimating the Min-Entropy H_∞

In order to determine the min-entropy of the PUF distribution, one needs to estimate the probability of the most likely PUF. Our experiments, as well as those of Delvaux et al. [25], strongly suggest that for a Gaussian distribution of the weights, the most likely PUFs are the $2n$ PUFs corresponding to the Boolean functions c_i and $\overline{c_i}$, $i = 1...n$.

The maximum likelihood estimator of that probability is simply the sample frequency, which is an unbiased estimator. A confidence interval for this estimator can be obtained using the Wilson score interval [26], which yields a confidence interval for the min-entropy H_∞.

Because we have already determined that there are exactly $2n$ PUFs in the equivalence class of the most likely PUF, we only need to estimate a confidence interval on the sample frequency of the *equivalence class*. Once such an interval was obtained, for instance $[p_-, p_+]$, then the confidence interval for the min-entropy is given by

$$[-\log_2(p_+) + \log_2(2n), -\log_2(p_-) + \log_2(2n)].$$

TABLE IV
CONFIDENCE INTERVALS AT THE 95% LEVEL FOR H_∞
(EXACT VALUES UP TO $n = 4$)

n	PUF Sample size	H_∞ (bits)
1	—	1
2	—	2
3	—	3.2086…
4	—	4.5850…
5	10^{10}	6.1006 – 6.1008
6	10^{10}	7.7352 – 7.7354
7	10^{10}	9.4731 – 9.4735
8	$2 \cdot 10^{10}$	11.3020 – 11.3024
9	$2 \cdot 10^{10}$	13.2123 – 13.2132
10	$3 \cdot 10^{12}$	15.1899 – 15.1901

The confidence intervals of the min-entropy are presented in Table IV.

The results of the simulation, up to $n = 10$, are presented in Figure 1. The results show that the Shannon entropy is close to the max-entropy, which as seen in Section II is asymptotically equivalent to n^2 as n increases.

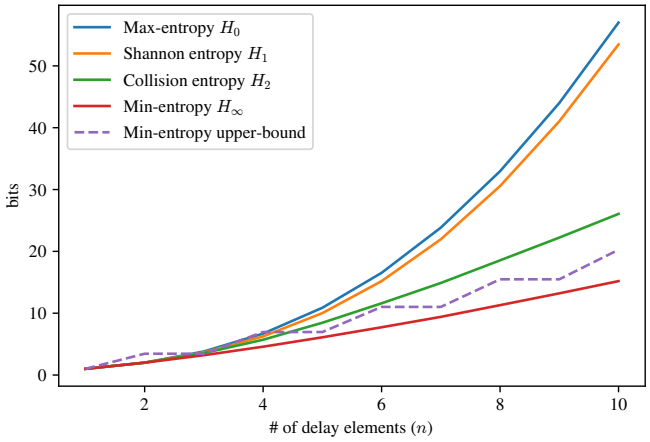

Fig. 1. Entropy estimates for $n \leq 10$. The upper bound of the min-entropy (dashed line) is taken from [25].

VI. CONCLUSIONS AND PERSPECTIVES

While it had been previously shown [6] that the entropy of the loop-PUF of n elements could exceed n, the exact values were only known for very small values of n. Making the link with BTF theory using Chow parameters, we have extended these results to provide accurate approximations up to $n = 10$. Our results suggest that the entropy of the loop-PUF might be *quadratic* in n: This would be a very positive result for circuit designers, since it implies that the PUF has a very good resistance to machine learning attacks. However, because the min-entropy and collision entropy are much smaller (on the order of n) the resistance to cloning may not be as high as expected.

Two interesting theoretical aspects of the PUF entropy are still open: First, to what extent does the entropy of the PUF stay close to the max-entropy for larger values of n? Second, is it possible to obtain a quasi-quadratic entropy in n

when choosing a small subset of all 2^n possible challenges? The latter point is of great practical interest since it would reduce the time required to obtain the PUF identifier while maintaining a high resistance to machine learning attacks.

For values of n larger than 10, our method seems to become too costly in space and time to produce accurate estimates of the PUF probability distributions under reasonable conditions. One could perhaps have recourse to entropy estimation methods that dispense with learning the distribution itself, such as the NSB estimation [23]. This could be used to check the predicted trend of the PUF entropy for increasing n.

ACKNOWLEDGMENT

The authors would like to thank Prof. Gadiel Seroussi, who first suggested a possible link between our problem and BTF theory at the LAWCI'2018 conference in Campinas, Brazil.

REFERENCES

[1] D. E. Holcomb, W. P. Burleson, and K. Fu, "Power-up SRAM state as an identifying fingerprint and source of true random numbers," *IEEE Transactions on Computers*, vol. 58, no. 9, pp. 1198–1210, 2009.

[2] G. E. Suh and S. Devadas, "Physical unclonable functions for device authentication and secret key generation," in *44th ACM/IEEE Design Automation Conference*, 2007, pp. 9–14.

[3] Z. Cherif, J.-L. Danger, S. Guilley, and L. Bossuet, "An easy-to-design PUF based on a single oscillator: The Loop PUF," in *15th Euromicro Conference on Digital System Design (DSD)*. IEEE, 2012, pp. 156–162.

[4] M.-D. M. Yu and S. Devadas, "Recombination of physical unclonable functions," in *35th Annual GOMACTech Conference*, 2010.

[5] B. Gassend, D. Clarke, M. Van Dijk, and S. Devadas, "Delay-based circuit authentication and applications," in *Proceedings of the 2003 ACM Symposium on Applied Computing*. ACM, 2003, pp. 294–301.

[6] O. Rioul, P. Solé, S. Guilley, and J.-L. Danger, "On the entropy of physically unclonable functions," in *IEEE International Symposium on Information Theory (ISIT)*, July 2016, pp. 2928–2932.

[7] H. Chang and S. S. Sapatnekar, "Statistical timing analysis considering spatial correlations using a single PERT-like traversal," in *Proceedings of the 2003 IEEE/ACM International Conference on Computer-Aided Design*. IEEE Computer Society, 2003, p. 621.

[8] A. Rényi, "On measures of entropy and information," in *Proceedings of the Fourth Berkeley Symposium on Mathematical Statistics and Probability*. Berkeley: University of California Press, 1961.

[9] A. Vijayakumar, V. C. Patil, C. B. Prado, and S. Kundu, "Machine learning resistant strong PUF: Possible or a pipe dream?" in *IEEE International Hardware Oriented Security and Trust*, 2016, pp. 19–24.

[10] E. Goto and H. Takahasi, "Some theorems useful in threshold logic for enumerating boolean functions." in *IFIP Congress*, 1962, pp. 747–752.

[11] R. O. Winder, "Single stage threshold logic," in *Symposium on Switching Circuit Theory and Logical Design*. IEEE, 1961, pp. 321–332.

[12] T. M. Cover, "Geometrical and statistical properties of systems of linear inequalities with applications in pattern recognition," *IEEE Transactions on Electronic Computers*, no. 3, pp. 326–334, 1965.

[13] C.-K. Chow, "On the characterization of threshold functions," in *Symposium on Switching Circuit Theory and Logical Design*, 1961, pp. 34–38.

[14] S.-T. Hu, *Threshold Logic*. Univ of California Press, 1965.

[15] D. R. Smith, "Bounds on the number of threshold functions," *IEEE Transactions on Electronic Computers*, no. 3, pp. 368–369, June 1966.

[16] S. Yajima and T. Ibaraki, "A lower bound of the number of threshold functions," *IEEE Transactions on Electronic Computers*, vol. EC-14, no. 6, pp. 926–929, Dec 1965.

[17] Y. A. Zuev, "Methods of geometry and probabilistic combinatorics in threshold logic," *Discrete Mathematics and Applications*, vol. 2, no. 4, pp. 427–438, 1992.

[18] N. Gruzling, "Linear separability of the vertices of an n-dimensional hypercube," Master's thesis, University of Northern British Columbia, 2008.

[19] T. W. Hungerford, *Algebra*, ser. Graduate Texts in Mathematics. New York: Springer-Verlag, 1980, vol. 73.

[20] A. Schaub, O. Rioul, J. J. Boutros, J.-L. Danger, and S. Guilley, "Challenge codes for physically unclonable functions with Gaussian delays: A maximum entropy problem," *Latin American Week on Coding and Information, UNICAMP-Campinas, Brazil*, pp. 22–27, 2018.

[21] Student, "The probable error of a mean," *Biometrika*, pp. 1–25, 1908.

[22] L. Paninski, "Estimation of entropy and mutual information," *Neural computation*, vol. 15, no. 6, pp. 1191–1253, 2003.

[23] I. Nemenman, F. Shafee, and W. Bialek, "Entropy and inference, revisited," in *Advances in Neural Information Processing Systems*, 2002, pp. 471–478.

[24] J. Acharya, A. Orlitsky, A. T. Suresh, and H. Tyagi, "The complexity of estimating Rényi entropy," in *Proceedings of the Twenty-Sixth Annual ACM-SIAM Symposium on Discrete Algorithms*. SIAM, 2014, pp. 1855–1869.

[25] J. Delvaux, D. Gu, and I. Verbauwhede, "Upper bounds on the min-entropy of RO sum, arbiter, feed-forward arbiter, and S-ArbRO PUFs," in *Hardware-Oriented Security and Trust (AsianHOST), IEEE Asian*. IEEE, 2016, pp. 1–6.

[26] E. B. Wilson, "Probable inference, the law of succession, and statistical inference," *Journal of the American Statistical Association*, vol. 22, no. 158, pp. 209–212, 1927.