PROPERTY TESTING OF PHYSICALLY UNCLONABLE FUNCTIONS

CHRISTOPH GRAEBNITZ

Degree-One Weight as Metric for Predictability

Institut für Informatik
Fachbereich Mathematik und Informatik
Freie Universität Berlin

April 9, 2018

Christoph Graebnitz: *Property Testing of Physically Unclonable Functions,* Degree-One Weight as Metric for Predictability, © April 9, 2018

Supervisors:
Prof. Dr. Marian Margraf

Location:
Berlin

Time Frame:
From November 1, 2017 to April 9, 2018

Herewith I would like to thank everyone who supported me in the development of this thesis.

Special thanks to Marian Margraf, Tudor Soroceanu, Wolfgang Studier, Nils Wisiol and Benjamin Zengin.

Dedicated to the most important person in my life.
K. O.

ABSTRACT

This thesis deals with the properties that define a Physically Unclonable Function (PUF). Since a PUF should not be copyable, it makes sense to consider the predictability of a PUF through algorithms from the field of machine learning. PUFs can be interpreted as Boolean functions in their Fourier extension. This allows the determination of the degree-one weight of a PUF. Further, the degree-one weight of a Boolean function can serve as a metric that helps to identify the predictability of a Boolean function. For this reason, a central point of this work is the theoretical consideration of various probabilistic methods which can be used to approximate the degree-one weight of a Boolean function. The number of randomly selected inputs is essential to maintain an absolute error with a certain probability. The empirical studies carried out in this thesis partially prove, that under certain circumstances a smaller number of inputs is sufficient to maintain a particular absolute error than shown in theory.

ZUSAMMENFASSUNG

Diese Arbeit beschäftigt sich mit den Eigenschaften, welche eine Physically Unclonable Function (PUF) definieren. Da eine PUF nicht kopierbar sein sollte, ist es sinnvoll die Vorhersagbarkeit durch Algorithmen aus dem Bereich des maschinellen Lernens zu betrachten. PUFs können als Boolsche Funktionen in ihrer Fourier Erweiterung interpretiert werden. Dies ermöglicht die Bestimmung des degree-one weights einer PUF. Das degree-one weight einer Boolschen Funktionen kann als Metrik dienen, die zur Erkennung von Vorhersagbarkeit einer Boolschen Funktion beiträgt. Aus diesem Grund ist ein zentraler Punkt dieser Arbeit, die theoretische Betrachtung verschiedener probabilistischer Verfahren, welche zur Approximation des degree-one weights einer Boolschen Funktion verwendet werden können. Dabei ist besonders die Anzahl an zufällig gewählten Eingaben interessant, um mit einer gewissen Wahrscheinlichkeit einen absoluten Fehler einzuhalten. Die in dieser Arbeit durchgeführten empirischen Studien belegen teilweise, dass unter gewissen Umständen eine geringere Anzahl von Eingaben, als in der Theorie gezeigt, für die Einhaltung eines bestimmten absoluten Fehlers ausreichen.

CONTENTS

1 INTRODUCTION 1

2 PRELIMINARIES 5

2.1 Boolean Functions 5

2.2 Properties of Physically Unclonable Functions 14

2.2.1 Reliability and Uniqueness 16

3 DEGREE-ONE WEIGHT AS METRIC FOR PREDICTABILITY 21

4 DEGREE-ONE WEIGHT AND LINEAR THRESHOLD FUNCTIONS 23

4.1 $^2/\pi$ Theorem 23

4.2 Distance to Unbiased Linear Threshold Functions . . . 30

5 FOURIER COEFFICIENT APPROXIMATION 33

5.1 Low-Degree Algorithm 33

5.2 Empirical Fourier Coefficient Approximation 36

6 DEGREE-ONE WEIGHT APPROXIMATION 41

6.1 Empirical Degree-One Weight Approximation 41

6.2 Approximate Sums of Powers of Fourier Coefficients . 47

6.3 Biased Distance to an Unbiased Linear Threshold Function 54

7 SURVEY OF DEGREE-ONE WEIGHT APPROXIMATION 57

7.1 Experiment Structure 57

7.1.1 Conducting the Experiment 57

7.1.2 Used Tools 59

7.2 Results Empirical Degree-One Weight Approximation 59

7.3 Results Approximate Sums of Powers of Fourier Coefficients 63

8 CONCLUSION 67

BIBLIOGRAPHY 69

LIST OF FIGURES

Figure 2.2.1 Relationship of PUF defining properties 16

Figure 2.2.2 The number of random variables needed to guarantee an absolute error $\mu \in [0.05, 0.1]$ with confidence $0.99 = 1 - \delta$ according to Theorem 2.10. 19

Figure 3.0.1 Degree-one weight as predictability indicator of Boolean functions. The dots are marking the degree-one weight which indicates the claim displayed at the x-axis. 22

Figure 4.2.1 Distance for an $(\varepsilon, 1)$-regular Boolean function f between 0 and 1. The distance is calculated by (4.2.1) with
$c = 0.5129$ according to Korolev and Shevtsova [KS10]. 31

Figure 5.1.1 The degree-one weights $\mathbf{W}^1[f] = 1 - \varepsilon/2$ of an Boolean function in relation to the absolute error of the degree-one Fourier coefficient approximation through the Low-Degree algorithm, supposing a concentration according to (5.1.1).
Further, $\varepsilon \in [0, 2]$ and the black horizontal line marks $2/\pi$. 35

Figure 6.1.1 Number of queries needed by Theorem 6.2 to guarantee with confidence $1 - \delta = 0.99$ an absolute error $\mu \in [0.1, 0.5]$ of the degree-one weight for different input lengths. 46

Figure 6.2.1 A comparison between the approximation with Theorem 6.2 which depends on the number of input bits n and Corollary 6.1 which is independent of n. The lines displaying the number of queries to guarantee with confidence $0.99 = 1 - \delta$ an absolute error μ. The solid lines represent Theorem 6.2 the dashed line displays Corollary 6.1. 52

Figure 7.2.1 Statistical results of 1000 controlled experiments for the empirical degree-one weight approximation with 1 to 10^5 queries. The black horizontal line marks the ideal degree-one weight. 62

Figure 7.3.1 Statistical results of 1000 controlled experiments for the degree-one weight approximation through Algorithm 6.1 with $\mu = 0.05$ and 3 to $12 \cdot 10^5$ queries. The black horizontal line marks the ideal degree-one weight. 65

LIST OF TABLES

Table 7.1 The boundary of queries for the empirical degree-one weight approximation experiments, such that a certain absolute error was maintained. Further, the hyphen indicates that one or more experiments exceeded the absolute error. . . . 61

Table 7.2 The boundary of queries for the sum of the powers of degree-one Fourier coefficients approximation experiments, such that a certain absolute error was maintained. To ensure that the theoretical number of queries is maintained in order to guarantee an absolute error with confidence $0.99 = 1 - \delta$, a round to the next higher whole number was performed. Further, the hyphen indicates that one or more experiments exceeded the absolute error. 63

1 INTRODUCTION

Hiding information from unwanted access, i.e., protecting confidentiality, is an integral part of modern cryptography and can be assured to a certain extent, for example, by encrypting a plain text using state of the art methods such as the Advanced Encryption Standard (AES) by Daemen and Rijmen [DR02]. Other objectives worthy of protection are integrity, authenticity and non-repudiation of information. Which of the protection goals are worth protecting always depends on the system under consideration. For a payment system, e.g., how it comes to use at banks, all of the objectives mentioned here would be worth protecting. Various special mechanisms are used to protect these goals.

In the case of signatures or encryption methods, keys or key pairs consisting of private and public keys are used. One point here is the secure storage of the private key. A method to store a private key is to utilize a smart card, e.g., bank card. For extraction or manipulation of data from a smart card there are various possibilities shown by Skorobogatov [Sko05]. There are also some measures against attempts to extract or manipulate information from smart cards [Aar16]. This kind of action and reaction regarding attack and protection is not desirable.

Another topic is the key generation, which is visible to the manufacturer, such that manipulations of the key would be possible.

Physically Unclonable Functions are a type of device that aims to store keys securely. The idea behind PUFs is that device-specific properties are used to generate keys at runtime instead of storing keys. Furthermore, the key generating features of the PUF should be selected in such a way that they are ideally unique and uncopyable from device to device.
One of the first devices was an optical PUF published by Pappu et al. [Pap+02]. This optical PUF is a device which projects laser through a light-transmissive material from certain angles and then creates a key from the resulting unique speckle pattern. By changing the angle of the laser emission, it creates different keys.

In the same year, Gassend et al. [Gas+02] published a paper proposing PUFs utilizing integrated circuits (ICs). They presented a network of circuits, which influences the delay of two signals running through. The individual manufacturing inaccuracies of the different components, which affect the speed of the signal, are assumed to be unique for a PUF instance. Based on this circuit Gassend et al. [Gas+04] invented the Arbiter PUF which uses an arbiter to recognize which signal arrives first and generates a related output.

Independently of this, PUFs are usually divided into two categories in the literature [Gua+07; RBK10].

Strong PUFs are characterized by an exponentially large challenge space and must fulfill the following security features. First of all, it must be unfeasible for anybody to build another device which challenge-response behavior acts indistinguishably from the original PUF. Secondly, within a limited time frame and unlimited access to the challenge response behavior, the extraction of all CRPs must be impossible. Furthermore, even if some CRPs are known, it must be difficult to simulate the response behavior of a PUF [Rüh+10].

If the number of different CRPs is rather small, the PUF is referred to as a Weak PUF. Hence, the responses of a Weak PUF should not leave a secure environment to prevent the extraction of all responses. Some examples for Weak PUFs are SRAM PUF [Gua+07], Latch PUF [Yam+11] or Anderson PUF [And10]. For the sake of completeness, some examples of Strong PUFs are XOR Arbiter PUF [SD07], Bistable Ring (BR) PUF [Che+11] or Lightweight PUF [MKP08].

However, since the focus of this thesis is not on the design of PUFs, the PUFs presented here will not be discussed further.

The primary focus of this thesis is to propose a metric for Strong PUFs, which is related to the predictability of Boolean functions. This is of interest because according to the definition of Strong PUFs it should not be possible to simulate the behavior of a PUF by utilizing a set of CRPs with high probability by so-called modeling attacks, for instance, done by Ruehrmair et al. [Rüh+10]. The benefit of a metric to recognize predictability is to gain information about the applicability of specific machine learning algorithms, e.g., the Perceptron algorithm for learning linear threshold functions (LTFs). This metric for predictability is based on the degree-one weight of Boolean functions in its representation as a Fourier expansion.

For the description of the degree-one weight metric, it is crucial to know the Fourier expansion of Boolean functions, which is defined in Section 2.1. Section 2.2 then provides a brief introduction to the properties that define a PUF. In Chapter 3, the degree-one weight of Boolean functions is presented as a metric for predictability. An essential part of the degree-one weight metric is the determination of the proximity of Boolean functions to functions from the unbiased LTF class. For this purpose, an analysis using the $2/\pi$ theorem is carried out in Chapter 4. The degree-one Fourier coefficients are an essential part of the degree-one weight of a Boolean function and therefore Chapter 5 is dedicated to their approximation. Chapter 6 introduces two ways of approximating degree-one weights and describes the influence of the absolute approximation error on the $2/\pi$ theorem. This is followed by surveys in Chapter 7, which examine the theoretical behavior of the degree-one weight approximation method in practice. Finally, Chapter 8 concludes on the usability of the degree-one weight metric as an indicator of the predictability of a PUF.

2 PRELIMINARIES

BOOLEAN FUNCTIONS

The reason why theories about Boolean functions are discussed here is that noiseless PUFs can be considered as Boolean functions and thus provide insightful analytical approaches to PUFs. At this point, it should be said that in this thesis PUFs are considered noiseless unless explicitly marked as noisy. This limitation is indeed not realistic from a practical point of view, but it simplifies the theoretical analysis of this work considerably. The most common form of a Boolean function is

$$f : \mathbb{F}_2^n \rightarrow \{0,1\},$$

where $\mathbb{F}_2$ describes a finite field. This representation is near to the logic of a computer but in the following sections, the notation of the book Analysis of Boolean Functions is used [OD014]. A real-valued Boolean function $f : \{-1,1\}^n \rightarrow \mathbb{R}$ can be expressed as its Fourier expansion, which is a multilinear polynomial.

First of all essential aspects of a polynomial are coefficients and monomials. The coefficients of the monomial $\chi_S(x)$ are expressed as $\widehat{f}(S) \in \mathbb{R}$. In addition to that, the association between $\widehat{f}(S)$ and $\chi_S(x)$ is realized through a subset S of the index set $[n] = \{1, 2, \ldots, n\}$.

The encoding $\chi : \mathbb{F}_2^n \rightarrow \mathbb{R}$ is selected to act as a bridge from ordinary Boolean functions with $\mathbb{F}_2^n$ inputs to inputs from $\{-1,1\}^n$ and vice versa

$$\chi\left(0_{\mathbb{F}_2}\right) = -1^0 = +1, \chi\left(1_{\mathbb{F}_2}\right) = -1^1 = -1,$$

where

$$\chi\left(x\right) = \left(-1\right)^{\sum_{i=1}^n x_i}.$$

Definition 2.1. Let $S \subseteq [n]$ be an index set then the parity function $\chi_S(x) : \{-1,1\}^n \rightarrow \{-1,1\}$ can be defined as

$$\chi_S(x) = \prod_{i \in S} x_i. \qquad \text{(where } \chi_\emptyset\left(x\right) = 1\text{)}$$

As an alternative representation the parity function can also be defined for inputs from $\mathbb{F}_2^n$ such that

$$\chi_S\left(x\right) = \prod_{i \in S} \chi\left(x_i\right) = \left(-1\right)^{\sum_{i \in S} x_i}.$$

Finally all pieces can be put together and realize the Fourier expansion.

Definition 2.2. (Fourier expansion [OD014, Theorem 1.1.])
The Fourier expansion of a function $f : \{-1,1\}^n \to \mathbb{R}$ defined by the multilinear polynomial

$$f(x) = \sum_{S \subseteq [n]} \widehat{f}(S)\chi_S(x),$$

where $[n] = \{1, 2, \ldots, n\}$ is related to 2^n the different number of terms.

The following is an alternative notation for single elements $i \in [n]$,

$$\widehat{f}\left(\{i\}\right) = \widehat{f}\left(i\right) \text{ and } \chi_{\{i\}}\left(x\right) = \chi_i\left(x\right).$$

An essential characteristic of a polynomial is its degree which is given by

$$\deg\left(f\right) = \max_{S \subseteq [n]} \left\{|S| : \widehat{f}\left(S\right) \neq 0\right\}$$

where f is a real-valued Boolean function [OD014, Exercise 1.10].

Another topic is the probability theory of two Boolean functions in the context of Fourier expansions. Throughout the thesis, $\mathbf{x} \sim \{-1,1\}^n$ denotes that $\mathbf{x} \in \{-1,1\}^n$ is an uniform at random drawn string. Further, the bits $\mathbf{x}_i$ are independently picked with

$$\Pr_{\mathbf{x}}\left[\mathbf{x}_i = 1\right] = \Pr_{\mathbf{x}}\left[\mathbf{x}_i = -1\right] = \frac{1}{2}.$$

Sometimes it is necessary to distinguish between several random strings and the superscript $\mathbf{i}$ is meant to be the index of $\mathbf{x}^{\mathbf{i}} \sim \{-1,1\}^n$.

Throughout this thesis, the expected value of the parity function even for non-uniformly distributed inputs is of interest. The following theorem describes the influence of non-uniformly distributed input bits.

Theorem 2.1. *Let $\chi_S : \{-1,1\}^n \to \{-1,1\}$ be the parity function and $S \subseteq [n]$. If inputs $\mathbf{x}$ are chosen independently at random from $\{-1,1\}^n$ with $\Pr_{\mathbf{x}}\left[\mathbf{x}_i = 1\right] = \frac{1}{2}$ for $i \notin S$ and $\Pr_{\mathbf{x}}\left[\mathbf{x}_i = 1\right] = \frac{1}{2} + \frac{\mu}{2}$ for $i \in S$ then*

$$\mathop{\mathbf{E}}_{\mathbf{x}}\left[\chi_S\left(\mathbf{x}\right)\right] = \begin{cases} 1 & \text{if } S = \emptyset, \\ \mu^{|S|} & \text{if } S \neq \emptyset \end{cases}$$

and for $T \subseteq [n] \setminus S$

$$\mathop{\mathbf{E}}_{\mathbf{x}}\left[\chi_T\left(\mathbf{x}\right)\right] = \begin{cases} 1 & \text{if } T = \emptyset, \\ 0 & \text{if } T \neq \emptyset. \end{cases}$$

Proof. (Theorem 2.1)
The proof uses the Definition 2.1 of the parity function.

$$\begin{aligned}\mathop{\mathbf{E}}_{\mathbf{x}}\left[\chi_S(\mathbf{x})\right] &= \mathop{\mathbf{E}}_{\mathbf{x}}\left[\prod_{i\in S}\mathbf{x}_i\right] && (S \neq \emptyset)\\ &= \prod_{i\in S}\mathop{\mathbf{E}}_{\mathbf{x}}\left[\mathbf{x}_i\right] && \text{(independent bits)}\end{aligned}$$

The expected value for one bit is

$$\begin{aligned}\mathop{\mathbf{E}}_{\mathbf{x}}\left[\mathbf{x}_i\right] &= (+1)\cdot\left(\frac{1}{2}+\frac{\mu}{2}\right)+(-1)\cdot\left(1-\left(\frac{1}{2}+\frac{\mu}{2}\right)\right)\\ &= \frac{1}{2}+\frac{\mu}{2}-1+\frac{1}{2}+\frac{\mu}{2}\\ &= \mu\end{aligned}$$

Hence the expected value of the parity function for the set S is

$$\mathop{\mathbf{E}}_{\mathbf{x}}\left[\chi_S(\mathbf{x})\right] = \mu^{|S|}.$$

Analogously the expected value of the parity function for the set $T \neq \emptyset$ can be calculated, thus

$$\mathop{\mathbf{E}}_{\mathbf{x}}\left[\chi_T(\mathbf{x})\right] = \left((+1)\cdot\frac{1}{2}+(-1)\cdot\frac{1}{2}\right)^{|T|} = 0.$$

The $\mathop{\mathbf{E}}_{\mathbf{x}}\left[\chi_\emptyset(\mathbf{x})\right] = 1$ follows from the Definition 2.1 of the parity function. □

Throughout the thesis, the parity function is expected to deal with equal distributed bits ($\mu = 0$) unless they are explicitly marked as unequally distributed.

An important tool for the calculation of the expected value is the inner product.

Definition 2.3. (Inner product [OD014, Definition 1.3.])
For two functions $f, g : \{-1,1\}^n \to \mathbb{R}$ the inner product $\langle\cdot,\cdot\rangle$ is defined by

$$\langle f,g\rangle = 2^{-n}\sum_{x\in\{-1,1\}^n} f(x)g(x) = \mathop{\mathbf{E}}_{\mathbf{x}}\left[f(\mathbf{x})g(\mathbf{x})\right]$$

The inner product defines the expected value but without considering the Fourier expansion. For more information about the relation to the Fourier coefficients, it is necessary to introduce Plancherel's Theorem.

Theorem 2.2. *(Plancherel's [OD014, Plancherels Theorem.])*
For any functions $f, g : \{-1,1\}^n \to \mathbb{R}$,

$$\langle f,g\rangle = \mathop{\mathbf{E}}_{\mathbf{x}}\left[f(\mathbf{x})g(\mathbf{x})\right] = \sum_{S\subseteq[n]}\widehat{f}(S)\,\widehat{g}(S).$$

Let's verify Plancherel's Theorem explicitly. To proof Plancherel's Theorem, it is helpful to formulate a theorem which examines the inner product of the monomials of the Fourier expansion.

Theorem 2.3. *([OD014, Theorem 1.5.])*
The 2^n functions $\chi_S : \{-1,1\}^n \to \{-1,1\}$ form an orthonormal basis for the vector space V of functions $\{-1,1\}^n \to \mathbb{R}$ where

$$\langle \chi_S, \chi_T \rangle = \begin{cases} 1 & \textit{if } S = T, \\ 0 & \textit{if } S \neq T. \end{cases}$$

Proof. (Theorem 2.3)
This proof uses two facts about the monomials of Fourier expansions. First of all,

$$\chi_S(x)\,\chi_T(x) = \prod_{i \in S \triangle T} x_i \prod_{i \in S \cap T} x_i^2 = \prod_{i \in S \triangle T} x_i = \chi_{S \triangle T}(x). \tag{2.1.1}$$

Secondly, lets have a look at the expected value of ,

$$\mathop{\mathbf{E}}_{\mathbf{x}}[\chi_S(\mathbf{x})] = \mathop{\mathbf{E}}_{\mathbf{x}}\left[\prod_{i \in S} \mathbf{x_i}\right] = \begin{cases} 1 & \text{if } S = \emptyset, \\ 0 & \text{if } S \neq \emptyset. \end{cases} \tag{2.1.2}$$

Equation (2.1.2) follows from Theorem 2.1 with $\mu = 0$.
Finally Theorem 2.3 can be proven.

$$\begin{aligned} \langle \chi_S, \chi_T \rangle =& \mathop{\mathbf{E}}_{\mathbf{x}}[\chi_S(\mathbf{x})\chi_T(\mathbf{x})] && \text{(Definition 2.3)} \\ =& \mathop{\mathbf{E}}_{\mathbf{x}}[\chi_{S \triangle T}(\mathbf{x})] && \text{(2.1.1)} \\ =& \begin{cases} \mathop{\mathbf{E}}_{\mathbf{x}}[\chi_{\emptyset}(\mathbf{x})] = 1 & \text{if } S = T, \\ \mathop{\mathbf{E}}_{\mathbf{x}}[\chi_{S \triangle T}(\mathbf{x})] = 0 & \text{if } S \neq T. \end{cases} && \text{(2.1.2)} \end{aligned}$$

□

With the help of Theorem 2.2, it is possible to show that the inner product of two real-valued Boolean functions f and g is equivalent to the expected value of the product of f and g.

Proof. (Theorem 2.2)

$$\begin{aligned}
\langle f,g\rangle =&2^{-n}\sum_{x\in\{-1,1\}^n}\left(\sum_{S\subseteq[n]}\widehat{f}(S)\chi_S(x)\cdot\sum_{T\subseteq[n]}\widehat{g}(T)\chi_T(x)\right)\\
=&2^{-n}\sum_{x\in\{-1,1\}^n}\left(\sum_{S,T\subseteq[n]}\widehat{f}(S)\chi_S(x)\cdot\widehat{g}(T)\chi_T(x)\right)\\
=&\sum_{S,T\subseteq[n]}\left(\widehat{f}(S)\cdot\widehat{g}(T)\cdot 2^{-n}\sum_{x\in\{-1,1\}^n}\chi_S(x)\cdot\chi_T(x)\right)\\
=&\sum_{S,T\subseteq[n]}\widehat{f}(S)\widehat{g}(T)\langle\chi_S,\chi_T\rangle\\
=&\sum_{S\subseteq[n]}\widehat{f}(S)\widehat{g}(S) && \text{(Theorem 2.3)}
\end{aligned}$$

□

In addition to that, the parity function for an arbitrary but fixed subset of $[n]$ can also be interpreted as a real-valued Boolean function. Furthermore, the following proposition shows that the inner product of the parity function $\chi_S(x)$ and f will map to the Fourier coefficient $\widehat{f}(S)$.

Proposition 2.1. *([ODo14, Proposition 1.8.])*
Let $f:\{-1,1\}^n\to\mathbb{R}$ and $S\subseteq[n]$, then the Fourier coefficient of f on S is given by

$$\widehat{f}(S)=\langle f,\chi_S\rangle=\mathop{\mathbf{E}}_{\mathbf{x}}\left[f(\mathbf{x})\,\chi_S(\mathbf{x})\right].$$

Proof. (Proposition 2.1)

$$\begin{aligned}
\langle f,\chi_S\rangle &=\left\langle\sum_{T\subseteq[n]}\widehat{f}(T)\,\chi_T,\chi_S\right\rangle\\
&=\sum_{T\subseteq[n]}\widehat{f}(T)\,\langle\chi_T,\chi_S\rangle\\
&=\widehat{f}(S) && \text{(Theorem 2.3)}
\end{aligned}$$

□

Plancherel's Theorem showed the interaction of Fourier coefficients concerning the inner product of two different real-valued Boolean functions. Further, the impact of the inner product of an arbitrary real-valued Boolean function with itself is also important.

Theorem 2.4. *(Parseval's Theorem. [ODo14, Parsevals Theorem.])*
For any $f : \{-1,1\}^n \to \mathbb{R}$,

$$\langle f,f\rangle = \mathop{\mathbf{E}}_{\mathbf{x}}\left[f(\mathbf{x})^2\right] = \sum_{S\subseteq[n]} \widehat{f}(S)^2. \tag{2.1.3}$$

If $f : \{-1,1\}^n \to \{-1,1\}$ is a Boolean function, then

$$\sum_{S\subseteq[n]} \widehat{f}(S)^2 = 1. \tag{2.1.4}$$

Proof. (Theorem 2.4)
First of all, (2.1.3) follows from Theorem 2.2. In addition to that, (2.1.4) is also true because of $f(x)\cdot f(x) = 1$. □

Moreover, the sum of squared Fourier coefficients that occurred in (2.1.3) is an essential part of some analytical facts and empirical results of this thesis and is called Fourier weight.

Definition 2.4. (Fourier weight [ODo14, Definition 1.19.])
For $f : \{-1,1\}^n \to \mathbb{R}$ and $0 \leq k \leq n$, the Fourier weight of f at degree k is

$$\mathbf{W}^k[f] = \sum_{\substack{S\subseteq[n]\\|S|=k}} \widehat{f}(S)^2.$$

The degree-one weight of Boolean functions is of special interest for the verification of a particular property of a PUF. Therefore it is essential to know how to calculate the Fourier weight of a Boolean function.

Theorem 2.5. *Let $f : \{-1,1\}^n \to \{-1,1\}$ be a Boolean function then the Fourier weight of f at degree k can be exactly calculated using all 2^n inputs of $x \in \{-1,1\}^n$.*

Proof. (Theorem 2.5)

$$\begin{aligned}\widehat{f}(S) &= \langle f,\chi_S\rangle && \text{(Proposition 2.1)}\\ &= 2^{-n}\sum_{x\in\{-1,1\}^n} f(x)\cdot\chi_S(x) && \text{(Definition 2.3)}\end{aligned}$$

The calculation of an arbitrary $\widehat{f}(S)$ needs all possible inputs of f. Hence $\mathbf{W}^k[f]$ can be calculated with all 2^n inputs of f. □

Sometimes only specific parts of Fourier expansions are analyzed.

Definition 2.5. Let $f : \{-1,1\}^n \to \{-1,1\}$ be a Boolean function and $S \subseteq [n]$, then

$$f^{=k}(x) = f^{=k} = \sum_{|S|=k} \widehat{f}(S)\chi_S(x)$$

is called the degree-k part of f.

The next definition introduces a shortcut for the relation of ε to the degree-one Fourier coefficients.

Definition 2.6.
A function $f : \{-1,1\}^n \to \mathbb{R}$ is $(\varepsilon, 1)$-regular if $\left|\widehat{f}(i)\right| \leq \varepsilon$ for all $i \in [n]$.

It is noteworthy that for a sufficient ε all Boolean functions are $(\varepsilon, 1)$-regular . Chapter 4 shows that if the degree-one weight of an $(\varepsilon, 1)$-regular Boolean function f lies within the interval

$$\frac{2}{\pi} - \varepsilon \leq \mathbf{W}^1[f] \leq \frac{2}{\pi} + O(\varepsilon),$$

then the distance to an LTF can be estimated.

Definition 2.7. ([OD014, Definition 2.5.])
A function $f : \{-1,1\}^n \to \{-1,1\}$ is called a linear threshold function if it is expressible as

$$f(x) = \operatorname{sgn}(a_0 + a_1 x_1 + \ldots + a_n x_n)$$

for some weights $a_0, \ldots, a_n \in \mathbb{R}$ (for definiteness $\operatorname{sgn}(0) = 1$). A LTF is called unbiased if $a_0 = 0$.

Another important class of functions (whose degree-one weight is known) are bent functions which were first introduced in O. S. Rothaus [Rot76]. To maintain consistency in the notation of Boolean functions the definition of bent functions are taken from O'Donnell [OD014].

Definition 2.8. ([OD014, Definition 6.26.])
A function $f : \mathbb{F}_2^n \to \{-1,1\}$ with n even is called bent if $\left|\widehat{f}(\gamma)\right| = \frac{1}{\sqrt{2^n}}$ for all $\gamma \in \widehat{\mathbb{F}_2^n}$.

The Fourier expansion of the functions from Definition 2.8 differs from Boolean functions according to Definition 2.2. The difference lies in their indexing of the Fourier coefficients and in their input space for the parity function. Recalling the Definition 2.1, for inputs from $\mathbb{F}_2^n$ the parity function changes from

$$\chi_S(x) = \prod_{i \in S} x_i$$

to

$$\chi_\gamma(x) = (-1)^{\gamma \cdot x},$$

where $x \in \mathbb{F}_2^n$ and $\gamma \in \widehat{\mathbb{F}_2^n}$ with the dot product $\gamma \cdot x$ being carried out in $\mathbb{F}_2^n$ such that

$$f(x) = \sum_{\gamma \in \widehat{\mathbb{F}_2^n}} \widehat{f}(\gamma) \chi_\gamma(x).$$

An important fact is that the Fourier coefficients remain the same.

Theorem 2.6. *Let $f : \mathbb{F}_2^n \to \{-1, 1\}$ be a bent function then its degree-one weight is*

$$\mathbf{W}^1[f] = \frac{n}{2^n}.$$

Proof. (Theorem 2.6)
Since by definition all Fourier coefficients have the same value, the degree-one weight is simply determined by

$$\mathbf{W}^1[f] = \sum_{i \in [n]} \widehat{f}(i)^2 = n \cdot \left(\left| \frac{1}{\sqrt{2^n}} \right| \right)^2 = \frac{n}{2^n}.$$

□

The bent function, used in this thesis, to easily apply the theory of degree-one weight approximation is defined for inputs of the set $\{-1, 1\}^n$ and is named inner product modulo two.

Definition 2.9. The inner product modulo two function
$IP_n : \{-1, 1\}^n \to \{-1, 1\}$, for an even $n \geq 2$, is defined by

$$IP_n(x) = \max(x_1, x_2) \cdot \max(x_3, x_4) \cdot \ldots \cdot \max(x_{n-1}, x_n).$$

For $n = 2$ it is only the maximum function.

To be able to show that the inner product modulo two function is a bent function, it is helpful to use the following theorem, which allows a successive definition of bent functions

Theorem 2.7. *([OD014, Proposition 6.27.])*
Let $f : \mathbb{F}_2^n \to \{-1, 1\}$ and $g : \mathbb{F}_2^{n'} \to \{-1, 1\}$ be bent. Then $f \oplus g : \mathbb{F}_2^{n+n'} \to \{-1, 1\}$ defined by $(f \oplus g)(x, x') = f(x) g(x')$ is also bent.

Proof. (Theorem 2.7)
To guarantee that $f(x) g(x')$ is a bent function it is necessary to show that the Fourier coefficients $\widehat{f \oplus g}(\gamma) = \pm \frac{1}{\sqrt{2^{n+n'}}}$.
We write $\gamma = (\gamma_1, \gamma_2)$:

$$\begin{aligned}
\widehat{f \oplus g}(\gamma_1, \gamma_2) &= \mathop{\mathbf{E}}_{\mathbf{x}, \mathbf{x}'} \left[f(\mathbf{x}) g(\mathbf{x}') \chi_{\gamma_1, \gamma_2}(\mathbf{x}, \mathbf{x}') \right] \\
&= \mathop{\mathbf{E}}_{\mathbf{x}, \mathbf{x}'} \left[f(\mathbf{x}) \chi_{\gamma_1}(\mathbf{x}) g(\mathbf{x}') \chi_{\gamma_2}(\mathbf{x}') \right] \\
&= \mathop{\mathbf{E}}_{\mathbf{x}} [f(\mathbf{x}) \chi_{\gamma_1}(\mathbf{x})] \cdot \mathop{\mathbf{E}}_{\mathbf{x}'} \left[g(\mathbf{x}') \chi_{\gamma_2}(\mathbf{x}') \right] \\
&= \pm \frac{1}{\sqrt{2^n}} \cdot \pm \frac{1}{\sqrt{2^{n'}}} \quad \text{(Definition 2.8 and Proposition 2.1)} \\
&= \pm \frac{1}{\sqrt{2^{n+n'}}}
\end{aligned}$$

□

O'Donnell shows that the Fourier expansion of the maximum function $\max_2 : \{-1,1\}^2 \rightarrow \{-1,1\}$ has only Fourier coefficients of size $\pm\frac{1}{2}$ [OD014, Sec. 1.2.]. Additionally, the corresponding function for inputs from $\mathbb{F}_2^2$ would be the logical AND, i.e., the multiplication function.

Theorem 2.8. *The function $IP_n : \{-1,1\}^n \rightarrow \{-1,1\}$ is a bent function.*

Proof. The inner product modulo two function is by Definition 2.9 the product of $\max_2$ functions. The $\max_2$ function is a bent function. Through the successive application of Theorem 2.7, the inner product modulo two function is a bent function. □

Last but not least, dictator functions are the remaining Boolean functions which are utilized in this thesis.

Definition 2.10. ([OD014, Definition 2.3.])
The i-th dictator function $\chi_i : \{-1,1\}^n \rightarrow \{-1,1\}$ is defined by

$$\chi_i(x) = x_i.$$

This type of function is very simple, since it only uses the i-th input bit as the result and discards the others. Followed by this fact the degree-one weight of a i-th dictator function can be simply determined.

Theorem 2.9. *The degree-one weight of the i-th dictator function $\chi_i : \{-1,1\}^n \rightarrow \{-1,1\}$ is*

$$\mathbf{W}^1[\chi_i] = 1.$$

Proof. (Theorem 2.9)
The i-th Fourier coefficient of a dictator function can be calculated by Proposition 2.1

$$\widehat{f}(i) = \mathop{\mathbf{E}}_{\mathbf{x}}[\chi_i \cdot \chi_i(\mathbf{x})] = 1.$$

From this and Theorem 2.4 it follows that the degree-one weight of a dictator function is

$$\mathbf{W}^1[\chi_i] = \sum_{i\in[n]} \widehat{f}(i)^2 = \widehat{f}(i)^2 = 1.$$

□

PROPERTIES OF PHYSICALLY UNCLONABLE FUNCTIONS

When is a PUF a PUF? An answer to this question requires a definition of properties that are essential for a PUF. To this end, this section briefly introduces the PUF defining properties. Reference is continuously made to the metrics that are used in the current literature.

Section 2.2.1 presents the two most popular metrics that are commonly used in the development of new PUF designs.

Further, Chapter 3 proposes a metric, with the help of which one can recognize the predictability of a PUF concerning specific algorithms.

Maes defines a comprehensive explanation of properties and metrics, which deal with both construction and security from a cryptographic point of view [Mae12]. Armknecht et al. [Arm+16] focuses in particular on the establishment of a security model for PUFs and some of the properties presented are equivalent to those from Maes. Since this thesis considers PUFs regarding authentication and identification, the inappropriate properties are neglected.

Figure 2.2.1 shows the properties that define PUFs according to Maes and their relationship to each other. In the original figure from Maes, the characteristics of a PUF are divided into PUF defining and nice-to-have. This thesis does not use this strict separation to avoid the impression that some properties are less critical for the PUF.

The following is a brief discussion of what is shown in Figure 2.2.1. Of particular interest are the metrics available for measuring these properties.

Constructability indicates whether it is possible to construct an instance of a class of PUFs, depending of course on whether the PUF is evaluable. The practical feasibility could be measured, for instance, in the case of IC PUFs in the number of logic modules. In the concrete example of PUFs for field-programmable gate arrays (FPGAs), the number of look-up tables used could be a metric for constructability [HHS17].

Several other properties more or less strongly influence the evaluability of a PUF. A common metric for this is the time required for the evaluation. Improvements to related properties usually lead to an increase in the time needed for evaluation. In the case of the Majority XOR Arbiter PUF by Wisiol et al., the poor reproducibility of the XOR Arbiter PUF responses, resulting from the use of many Arbiter PUFs whose noisy responses are linked to XOR, is improved by a majority vote on the individual Arbiter PUFs [Wis+]. This improvement in reproducibility inevitably leads to an extended evaluation period. The reproducibility can be measured with the metric of reliability presented in Definition 2.11.

The uniqueness of a PUF describes how unique the answers are compared to other PUFs of the same class. Since PUFs can be regarded as a memory of cryptographic keys, it is essential that the keys generated by PUF instances are as different as possible. Definition 2.14 gives a popular metric to determine the uniqueness of a PUF.

From reproducibility and uniqueness follows the identifiability of the PUF and thus could be measured by suitable metrics given by Definition 2.11 and Definition 2.14.

Further, the tamper evidence describes the protection against manipulation, e.g., the side-channel attack by laser or the acquisition of physically intrinsic characteristics such as the delays of the Arbiter PUF by the photon emission attack of Tajik et al. [Taj+15; Taj+17]. The measure of tamper evidence would then be the security against all known attacks relevant to the PUF class.

The one-wayness of a PUF describes the probability that someone can imply from a response to the related challenge. Further, the size of the challenge space significantly influences the one-wayness of the PUF [Arm+16].

Furthermore, unpredictability describes the learnability of a PUF of which only a limited number of CRPs are available for the learning process. The metric to measure this property is just resistance to all known learning attacks such as Ruehrmair et al. [RS14] or Becker [Bec15].

Similar to unpredictability, mathematical unclonability is a property that describes the learnability or reproducibility of a PUF under unrestricted physical access. Further, this means that not only CRPs can be used for learning but also, for instance, evaluation delays.

The physical unclonability describes the resistance of the PUF class against manipulations in the manufacturing process, such that two or more PUF instances differ according to the uniqueness property even after a manipulation. If a PUF is physically unclonable, even the manufacturer of the PUF does not have to be trusted.

Besides, the true unclonability ultimately follows from the fact of mathematical and physical unclonability.

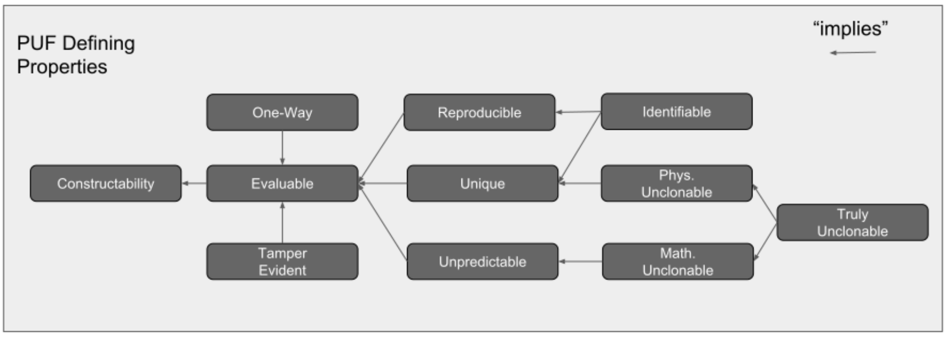

Figure 2.2.1: Relationship of PUF defining properties

Reliability and Uniqueness

The most popular terms in the context of PUF property metrics are the reliability and uniqueness. These metrics are straightforward to calculate and may show unacceptable characteristics of a PUF. In literature, the reliability of a PUF instance is described as the arithmetic mean of the hamming distances of the repeated evaluations of the same challenge [Che+11; Rah+14; MS09]. There are publications which investigate environmental variations such as temperature, voltage or aging [MKD10; Che+12; ML14]. These physical conditions are known to have an impact on certain PUF instances. An author might be interested in characterizing the reliability and uniqueness of a PUF instance under specific conditions. For the sake of simplicity physical effects are not considered in this thesis.

In the entire thesis, $\mathrm{H}(x, y)$ denotes the hamming distance between two strings from the set $\{-1, 1\}^{\eta}$, where η defines the ouput bits of a PUF.

In some papers, it is common practice to define reliability such that, 0 defines the highest and 1 the worst reliability. This definition is not very intuitive, so this work reverses the commonly known reliability. The reliability metric only makes sense in the context of noisy PUFs.

Definition 2.11. The reliability for a challenge $x \in \{-1,1\}^n$ and r evaluations of a PUF instance $f : \{-1,1\}^n \rightarrow \{-1,1\}^\eta$ is

$$\mathrm{R}(f,x,r) = 1 - \frac{1}{r}\sum_{j=1}^{r} \frac{\mathrm{H}(f(x), f^{(j)}(x))}{\eta},$$

where $f^{(j)}(x)$ is the j-th evaluation result and $f(x)$ is the actual response.

In practical situations, the actual response $f(x)$ of a PUF instance can be approximated by majority voting over a set of evaluation results. If all responses to the same challenge are equal, the reliability is ideal ($\mathrm{R}(f,x,r) = 1$). The function described in Definition 2.11 only measures the reliability of one PUF instance and one challenge. Thus, a scalable statistical experiment is needed to get more insight into the reliability behavior of a class of PUFs.

Definition 2.12. A statistical PUF experiment is a tuple $S = (F, X, r)$. Where

- $F = \{f_1, \dots, f_m\}$ is a set of PUF instances $f_i : \{-1,1\}^n \rightarrow \{-1,1\}^\eta$,
- X is a multiset of Challenges $x \in \{-1,1\}^n$,
- r is the number of evaluations : $r \in \mathbb{N}$.

The tuple S can be used to perform more comprehensive statistical analyses. An application of tuple S is described in the following definition.

Definition 2.13. The reliability multiset S_R of $S = (F, X, r)$ is given by

$$S_\mathrm{R} = \{\mathrm{d} \mid \forall f \in F, \forall x \in X : \mathrm{d} = \mathrm{R}(f,x,r)\}.$$

The reliability multiset can now be used to calculate the minimum, maximum, median, and estimate the mean and sample variance.

The uniqueness of a PUF is described by the average response inter-distance between different PUF instances.

Definition 2.14. For a challenge $x \in \{-1,1\}^n$ and $F = \{f_1, \dots, f_m\}$ a set of PUF instances $f_i : \{-1,1\}^n \rightarrow \{-1,1\}^\eta$ the uniqueness is expressed as

$$\mathrm{U}(F,x) = \frac{2}{m(m-1)} \sum_{u=1}^{m-1} \sum_{v=u+1}^{m} \frac{\mathrm{H}(f_u(x), f_v(x))}{\eta}.$$

An optimal uniqueness value for a set of PUF instances is 0.5. The idea of comparing PUF instance distances of Definition 2.14 can be used as a basis for deeper statistical analysis similar to Definition 2.13.

Definition 2.15. For a tuple $S = (F, X, r)$ the uniqueness multiset S_{U} is defined by

$$S_{\mathrm{U}} = \{\mathrm{d}_{x,1}, \ldots, \mathrm{d}_{x,r} \mid \forall i \in \{1, 2, \ldots, r\}, \forall x \in X : \mathrm{d}_{x,i} = \mathrm{U}(F, x)\},$$

where $\mathrm{d}_{x,i}$ denotes the i-th uniqueness calculation for challenge x.

Like the reliability multiset, the uniqueness multiset is also a set of real numbers. Hence all the statistical functions presented in this section can be applied to the uniqueness multiset too. The uniqueness and reliability are essential properties, such that a PUF should meet the optimal values to be a PUF rather than a particular case of a random number generator. For a small set of PUF instances, in terms of number and scale, the presented techniques can be used to characterize these attributes in an acceptable amount of time and precision.

Due to the unknown distribution of the samples of the reliability multiset or uniqueness multiset, the confidence for the arithmetic means used in both statistics can be calculated by Hoeffding's inequality [Hoe63]. Furthermore, the Hoeffding bound of Theorem 2.10 is also applicable for the arithmetic mean of the reliability supposing the addends $\mathrm{H}(f(x), f^{(j)}(x))/\eta$ are independent random variables.

Theorem 2.10. *([Hoe63, Theorem 1.]) Let $\mathbf{X} = \mathbf{X}_1, \ldots \mathbf{X}_N$ be random variables drawn independently from the same distribution with $\mathbf{X}_i \in [0, 1]$ and $\mathbf{T} = \frac{1}{N}\sum_{i=1}^{N} \mathbf{X}_i$. Then for $\lambda > 0$,*

$$\Pr_{\mathbf{X}_i}\left[\left|\mathbf{T} - \mathop{\mathbf{E}}_{\mathbf{X}_i}[\mathbf{T}]\right| \geq \lambda\right] \leq e^{-2N\lambda^2}.$$

Theorem 2.10 can be used to determine a lower bound for the number of samples needed to guarantee with confidence $1 - \delta$ that the absolute error of the arithmetic mean of the samples is smaller than λ.

Theorem 2.11. *Let $\mathbf{X} = \mathbf{X}_1, \ldots \mathbf{X}_N$ be random variables drawn independently from the same distribution with $\mathbf{X}_i \in [0, 1]$ and $\mathbf{T} = \frac{1}{N}\sum_{i=1}^{N} \mathbf{X}_i$. Then $N \geq O\left(\log(1/\delta)/\lambda^2\right)$ variables are needed to guarantee with confidence $1 - \delta$ and $\lambda > 0$ that*

$$\left|\mathbf{T} - \mathop{\mathbf{E}}_{\mathbf{X}_i}[\mathbf{T}]\right| \leq \lambda$$

holds.

Proof. (Theorem 2.11)
From Theorem 2.10 it follows

$$\begin{aligned}\Pr_{\mathbf{X}_i}\left[\left|\mathbf{T} - \mathop{\mathbf{E}}_{\mathbf{X}_i}[\mathbf{T}]\right| \geq \lambda\right] \leq e^{-2N\lambda^2} &\leq \delta \\ -2N\lambda^2 &\leq \log(\delta) \\ N &\geq \frac{\log\left(\frac{1}{\delta}\right)}{2\lambda^2} \\ N &\geq O\left(\frac{\log\left(\frac{1}{\delta}\right)}{\lambda^2}\right).\end{aligned}$$

N denotes the number of random variables. Hence, with probability $1-\delta$ the absolute error of the arithmetic mean $\mathbf{T}$ is lower equal to λ. □

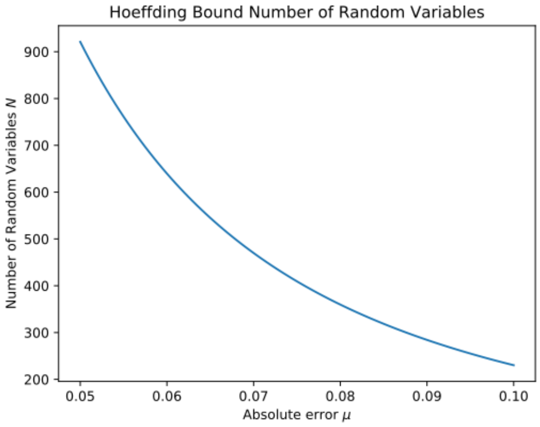

Figure 2.2.2: The number of random variables needed to guarantee an absolute error $\mu \in [0.05, 0.1]$ with confidence $0.99 = 1 - \delta$ according to Theorem 2.10.

Figure 2.2.2 shows that with slightly more than 900 samples an error of 0.05 with confidence $0.99 = 1 - \delta$ is maintained.

3 DEGREE-ONE WEIGHT AS METRIC FOR PREDICTABILITY

Section 2.2 introduces the PUF defining properties.
One security relevant property is the unpredictability, which is sometimes neglected when introducing a new PUF design. In the case of the BR PUF by Chen et al., it is suspected that the design is challenging to model and thus unpredictable [Che+11]. However, Xu et al. prove this suspicion wrong by learning a 2 XOR BR PUF to 95 percent accuracy, using support vector machine learning [Xu+15].

An idea to describe the unpredictability of a PUF design would be to investigate the predictability of a PUF class with all known learning algorithms for Boolean functions in experiments for particular instances. The results of learning algorithms, such as the logistic regression used by Ruehrmair et al. [Rüh+10] are based on probabilistic parameter initialization. Hence, the number of experiments that would have to be carried out would probably go beyond the scope of a scientific paper.

Therefore it makes more sense to consider which classes of Boolean functions are predictable from known learning algorithms and then test the PUF on the membership of a specific class of functions or to identify properties that provide indications for the predictability of Boolean functions.

For this reason, this work suggests the degree-one weight of Boolean functions as a metric that can be used to discover predictability. It should be said that this is only one step in the direction of the recognition of predictability of Boolean functions. A proof about unpredictability cannot be achieved with the degree-one weight.

Figure 3.0.1 shows indicators for the predictability of a Boolean function. If the degree-one weight of an $(\varepsilon, 1)$-regular Boolean function f fulfills ${}^{2}/_{\pi} - \varepsilon \leq \mathbf{W}^{1}[f] \leq {}^{2}/_{\pi} + O(\varepsilon)$, then according to Theorem 4.3 it can be assumed that f is $O\left(\sqrt{\varepsilon}\right)$ close to an unbiased LTF.

Further, according to Matulef et al. there is a test which recognizes whether a Boolean function is an $(\varepsilon, 1)$-regular unbiased LTF [Mat+10, Theorem 28.]. The test by Matulef et al. uses an approximated degree-one weight and is not deepened further for the benefit of a more detailed analysis of the approximation of degree-one weights in Chapter 6 and the ${}^{2}/_{\pi}$ theorem.

Rojas shows that LTFs are by definition linear separable and learnable with the Perceptron algorithm within a finite number of steps [Roj13, Ch. 4 Definition 3.; Ch. 4 Proposition 8.]. Furthermore, the

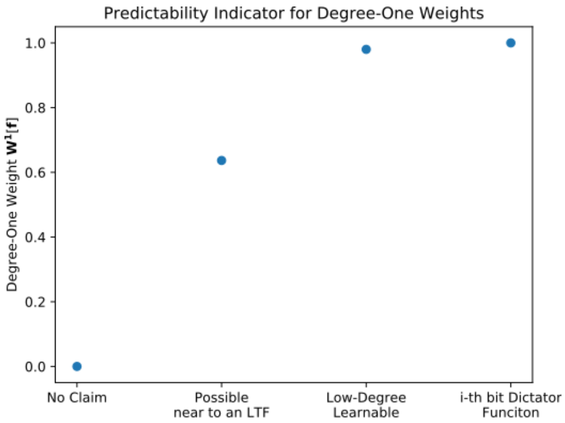

Figure 3.0.1: Degree-one weight as predictability indicator of Boolean functions. The dots are marking the degree-one weight which indicates the claim displayed at the x-axis.

learning of linear separable spaces is a well-studied field of research and offers several learning algorithms, for instance, support vector machine [Bis06; CV95].

Furthermore, if the degree-one weight is sufficiently large, then the Low-Degree algorithm presented in Section 5.1 generates a sufficiently accurate approximation of the real Boolean function with high probability.

Apart from that, by Definition 2.10 the i-th bit dictator functions have a degree-one weight of exactly 1 and are easy to learn with $O(n)$ steps by fixing one input bit and observing changes to the output after inverting it.

Finally, if the degree-one weight of a PUF meets one of those values, the reconsideration of the PUF design is strongly recommended.

4 DEGREE-ONE WEIGHT AND LINEAR THRESHOLD FUNCTIONS

This chapter presents a method to check the distance of a Boolean function to an unbiased LTF. For this purpose, the $2/\pi$ theorem is used, which describes an upper bound for the distance of a Boolean function to an unbiased LTF. An essential variable for the $2/\pi$ theorem is the degree-one weight and the magnitude of the degree-one Fourier coefficients of the investigated Boolean function.

In addition to that, Section 4.2 shows how small the degree-one Fourier coefficients must be to get a convincing distance measure.

$2/\pi$ THEOREM

This section describes the distance of the degree-one weight of a Boolean function f to an unbiased LTF whose weights are the degree-one Fourier coefficients of f. The theorem that describes this connection is the $2/\pi$ theorem, which consists of two parts, where the first part is invented by Khot et al. and the second part is developed by Matulef et. al [Kho+07; Mat+10]. This thesis uses the combination of both parts shown by O'Donnell [OD014, The Two Over Pi Theorem.].

An additional goal of this section is to reproduce the proof of the $2/\pi$ theorem, with the intention to be more extensive. The proof of the $2/\pi$ theorem uses some essential non-trivial theorems, which will be discussed next.

First of all, the central limit theorem with error bounds is essential to show.

Theorem 4.1. *(Berry-Esseen Central Limit [Ber41; OD014])*
Let $\mathbf{X}_1, \ldots, \mathbf{X}_n$ be independent random variables with $\mathbf{E}[\mathbf{X}_i] = 0$ and $\mathbf{VAR}[\mathbf{X}_i] = \sigma_i^2$, and assume $\sum_{i=1}^n \sigma_i^2 = 1$. Let $\mathbf{S} = \sum_{i=1}^n \mathbf{X}_i$ and let $\mathbf{Z} \sim \mathcal{N}(0,1)$ be a standard Gaussian. Then for all $u \in \mathbb{R}$,

$$|\mathbf{Pr}[\mathbf{S} \leq u] - \mathbf{Pr}[\mathbf{Z} \leq u]| \leq c\gamma,$$

where

$$\gamma = \sum_{i=1}^n \|\mathbf{X}_i\|_3^3 = \sum_{i=1}^n \mathbf{E}\left[|\mathbf{X}_i|^3\right]$$

and c is a universal constant.

Remark 4.1. ([ODo14, Remark 5.15.])
If all of the $\mathbf{X}_i$'s satisfy $|\mathbf{X}_i| \leq \varepsilon$ with probability 1, then the bound

$$\gamma = \sum_{i=1}^{n} \mathbf{E}\left[|\mathbf{X}_i|^3\right] \leq \varepsilon \cdot \sum_{i=1}^{n} \mathbf{E}\left[|\mathbf{X}_i|^2\right] = \varepsilon \cdot \sum_{i=1}^{n} \sigma_i^2 = \varepsilon$$

can be used.

Theorem 4.1 offers an alternative representation of the core statement, which is used to proof the $2/\pi$ theorem.

$$\begin{aligned} &|\mathbf{Pr}\left[\mathbf{S} \leq u\right] - \mathbf{Pr}\left[\mathbf{Z} \leq u\right]| \leq c\gamma \\ \iff & -c\gamma + \mathbf{Pr}\left[\mathbf{Z} \leq u\right] \leq \mathbf{Pr}\left[\mathbf{S} \leq u\right] \leq c\gamma + \mathbf{Pr}\left[\mathbf{Z} \leq u\right] \end{aligned} \tag{4.1.1}$$

The following theorem gives information about the error bound of the expected value of the sum of independent random variables of a particular form.

Theorem 4.2. *([ODo14, Theorem 5.16])*
Let $a_1, \ldots, a_n \in \mathbb{R}$ with $\sum_i a_i^2 = 1$ and $|a_i| \leq \varepsilon$ for $i \in [n]$.

$$\left| \mathop{\mathbf{E}}_{\mathbf{x}}\left[\left|\sum_{i=1}^{n} a_i \mathbf{x}_i\right|\right] - \sqrt{\frac{2}{\pi}} \right| \leq c\varepsilon$$

where c is a universal constant.

Proof. (Theorem 4.2)
Based on theorem 4.1 the distribution function of $\mathbf{S}$ converges pointwise towards the distribution function of standard normal distributed random variable $\mathbf{Z} \sim \mathcal{N}(0,1)$ with an upper bound error.
From this, it follows that the difference between the expected values of $\mathbf{S}$ and $\mathbf{Z}$ have the same upper bound.

$$\begin{aligned} \left| \mathop{\mathbf{E}}_{\mathbf{x}}\left[\sum_{i=1}^{n} a_i \mathbf{x}_i\right] - \mathbf{E}\left[\mathbf{Z}\right] \right| &\leq c\gamma && (|\mathbf{X}_i| = |a_i \mathbf{x}_i|) \\ \left| \mathop{\mathbf{E}}_{\mathbf{x}}\left[\left|\sum_{i=1}^{n} a_i \mathbf{x}_i\right|\right] - \mathbf{E}\left[|\mathbf{Z}|\right] \right| &\leq c\gamma && \text{(triangle inequality)} \\ \left| \mathop{\mathbf{E}}_{\mathbf{x}}\left[\left|\sum_{i=1}^{n} a_i \mathbf{x}_i\right|\right] - 2 \cdot \int_0^\infty z \cdot \frac{1}{\sqrt{2\pi}} e^{-z^2/2} dz \right| &\leq c\gamma \\ \left| \mathop{\mathbf{E}}_{\mathbf{x}}\left[\left|\sum_{i=1}^{n} a_i \mathbf{x}_i\right|\right] - \left(\sqrt{\frac{2}{\pi}} \cdot \int_0^\infty z \cdot e^{-z^2/2} dz \right) \right| &\leq c\gamma \end{aligned} \tag{4.1.2}$$

Integral calculation by substituting with $u = -\frac{z^2}{2}$ and $du = -zdz$:

$$\begin{aligned}
\sqrt{\frac{2}{\pi}} \cdot \int_0^{\infty} z \cdot e^{-z^2/2} dz &= -\sqrt{\frac{2}{\pi}} \cdot \int_0^{-\infty} e^u du \\
&= \sqrt{\frac{2}{\pi}} \cdot \int_{-\infty}^{0} e^u du && (\cdot\,(-1)) \\
&= \sqrt{\frac{2}{\pi}} \cdot e^u |_{-\infty}^{0} \\
&= \sqrt{\frac{2}{\pi}} && (4.1.3)
\end{aligned}$$

From (4.1.2) and (4.1.3) it follows

$$\begin{aligned}
\left| \mathop{\mathbf{E}}_{\mathbf{x}} \left[\left| \sum_{i=1}^{n} a_i \mathbf{x}_i \right| \right] - \sqrt{\frac{2}{\pi}} \right| &\leq c\gamma \\
\left| \mathop{\mathbf{E}}_{\mathbf{x}} \left[\left| \sum_{i=1}^{n} a_i \mathbf{x}_i \right| \right] - \sqrt{\frac{2}{\pi}} \right| &\leq c\varepsilon. && \text{(Remark 4.1)}
\end{aligned}$$

□

Remark 4.2. An alternative form of Theorem 4.2 which is used to proof the 2/π theorem is

$$\begin{aligned}
& \left| \mathop{\mathbf{E}}_{\mathbf{x}} \left[\left| \sum_{i=1}^{n} a_i \mathbf{x}_i \right| \right] - \sqrt{\frac{2}{\pi}} \right| \leq c\varepsilon \\
\iff \quad & -c\varepsilon \leq \mathop{\mathbf{E}}_{\mathbf{x}} \left[\left| \sum_{i=1}^{n} a_i \mathbf{x}_i \right| \right] - \sqrt{\frac{2}{\pi}} \leq c\varepsilon \\
\iff \quad & \sqrt{\frac{2}{\pi}} - c\varepsilon \leq \mathop{\mathbf{E}}_{\mathbf{x}} \left[\left| \sum_{i=1}^{n} a_i \mathbf{x}_i \right| \right] \leq c\varepsilon + \sqrt{\frac{2}{\pi}} \\
\implies \quad & \mathop{\mathbf{E}}_{\mathbf{x}} \left[\left| \sum_{i=1}^{n} a_i \mathbf{x}_i \right| \right] \leq \sqrt{\frac{2}{\pi}} + c\varepsilon.
\end{aligned}$$

The next lemma is an useful tool for the proof of the 2/π theorem which simplifies the dealing with ε.

Lemma 4.1. *Let $f : \{-1,1\}^n \to \{-1,1\}$ be a Boolean function with $\left|\widehat{f}(i)\right| \leq \varepsilon$ for all $i \in [n]$ and $\lambda > 0$. If $\mathbf{W}^1[f] \geq 2/\pi - \lambda$ then*

$$\mathbf{W}^1[f] \geq \frac{2}{\pi} - \varepsilon$$

where $\varepsilon = \min(\lambda, 2/\pi)$.

Proof. (Lemma 4.1)
Due to the definition of $\varepsilon = \min\left(\lambda, {}^{2}/_{\pi}\right)$ the proof of this lemma must consider two cases for $\lambda > 0$.
Case $\lambda \leq {}^{2}/_{\pi}$:
The following

$$\begin{aligned} \mathbf{W}^1[f] &\geq \frac{2}{\pi} - \varepsilon \\ &\geq \frac{2}{\pi} - \lambda && (\varepsilon = \min\left(\lambda, {}^{2}/_{\pi}\right)) \end{aligned}$$

is true according to hypothesis $\mathbf{W}^1[f] \geq {}^{2}/_{\pi} - \lambda$.
Case $\lambda > {}^{2}/_{\pi}$:

$$\begin{aligned} \mathbf{W}^1[f] &\geq \frac{2}{\pi} - \lambda \\ &\geq \frac{2}{\pi} - \frac{2}{\pi} && (\varepsilon = \min\left(\lambda, {}^{2}/_{\pi}\right)) \\ &\geq 0 && (4.1.4) \end{aligned}$$

The inequality (4.1.4) also holds because $\mathbf{W}^1[f]$ is the sum of squares of real numbers according to Definition 2.4. □

Finally, all pieces can put together and the ${}^{2}/_{\pi}$ theorem can be introduced.

Theorem 4.3. *([ODo14, The $\frac{2}{\pi}$ Theorem])*
Let $f : \{-1,1\}^n \to \{-1,1\}$ satisfy $\left|\widehat{f}(i)\right| \leq \varepsilon$ for all $i \in [n]$, then

$$\mathbf{W}^1[f] \leq \frac{2}{\pi} + O(\varepsilon).$$

Further, if $\mathbf{W}^1[f] \geq \frac{2}{\pi} - \varepsilon$ then f is $O(\sqrt{\varepsilon})$-close to the linear threshold function $\operatorname{sgn}\left(f^{=1}\right)$.

Proof. (Theorem 4.3)
The following proof is based on ideas from Khot et al. [Kho+07] and Matulef et. al [Kho+07; Mat+10]. This thesis rephrases the proof according to O'Donnell [ODo14, Proof of the $\frac{2}{\pi}$ Theorem]. Additionally, c is the universal constant from Theorem 4.1 and can be set w.l.o.g. to 0.5129 (Korolev and Shevtsova [KS10]). Assume w.l.o.g. according to Lemma 4.1 that $\varepsilon \in [0, {}^{2}/_{\pi}]$.
Let

$$\sigma = \sqrt{\mathbf{W}^1[f]} \geq \frac{1}{2} \tag{4.1.5}$$

and

$$l(x) = \frac{1}{\sigma} f^{=1} = \frac{1}{\sigma} \sum_{i \in [n]} \widehat{f}(i)\, \chi_i(x), \tag{4.1.6}$$

such that $\sqrt{\langle l, l \rangle} = 1$ and $\left|\widehat{l}\,(i)\right| \leq 2\varepsilon$ for all $i \in [n]$.
Theorem 4.3 consists of two statements. Therefore, the proof is carried out in two parts. First of all, it is reasonable to proof the statement

$$\mathbf{W}^1\,[f] \leq \frac{2}{\pi} + O(\varepsilon).$$

Case $\sqrt{\mathbf{W}^1\,[f]} \leq 1/2 \leq 2/\pi + O\,(\varepsilon)$:

Due to $\varepsilon \in [0, 2/\pi]$ degree-one weights $\mathbf{W}^1\,[f] \leq 1/2$ are smaller $2/\pi + \varepsilon$ anyway $(1/2 \leq 2/\pi)$.

Case $1/2 \leq \sqrt{\mathbf{W}^1\,[f]} \leq 2/\pi + O\,(\varepsilon)$:

$$\begin{aligned}
\sigma &= \sqrt{\mathbf{W}^1\,[f]} && (4.1.5)\\
&= \frac{\mathbf{W}^1\,[f]}{\sqrt{\mathbf{W}^1\,[f]}}\\
&= \frac{1}{\sigma} \sum_{i \in [n]} \widehat{f}\,(i) \cdot \widehat{f}\,(i) && \text{(Definition 2.4 and (4.1.5))}\\
&= \langle f, l \rangle && \text{(Theorem 2.2 and (4.1.6))}\\
&= \mathop{\mathbf{E}}_{\mathbf{x}}\,[f(\mathbf{x}) l(\mathbf{x})] && \text{(Theorem 2.2)}\\
&\leq \mathop{\mathbf{E}}_{\mathbf{x}}\,[|f(\mathbf{x})| \cdot |l(\mathbf{x})|] && (4.1.7)\\
&= \mathop{\mathbf{E}}_{\mathbf{x}}\,[1 \cdot |l(\mathbf{x})|] && (4.1.8)\\
&\leq \sqrt{\frac{2}{\pi}} + 2c\varepsilon. && \text{(Remark 4.2 where } a_i = \widehat{f}(i)/\sigma,\ \mathbf{x}_i = \chi_i\,(\mathbf{x}))
\end{aligned}$$

After squaring σ the first statement is proven.

$$\Longrightarrow \sigma^2 = \mathbf{W}^1\,[f] \leq \frac{2}{\pi} + O(\varepsilon). \tag{4.1.9}$$

Note that from (4.1.7) and (4.1.8) it follows

$$\sigma = \langle f, l \rangle \leq \mathop{\mathbf{E}}_{\mathbf{x}}\,[|l\,(\mathbf{x})|]\,. \tag{4.1.10}$$

Furthermore, the second statement,

$$\mathbf{W}^1\,[f] \geq \frac{2}{\pi} - \varepsilon \Rightarrow \mathop{\mathbf{Pr}}_{\mathbf{x}}\left[f\,(\mathbf{x}) \neq \operatorname{sgn}\left(f^{=1}\,(\mathbf{x})\right)\right] \leq O(\sqrt{\varepsilon}) \tag{4.1.11}$$

of the theorem requires a proof too. Statement (4.1.11) is established through a proof by contradiction using an upper bound for the expected value

$$\mathop{\mathbf{E}}_{\mathbf{x}}\,[(\operatorname{sgn}\,(l\,(\mathbf{x})) - f\,(\mathbf{x})) \cdot l\,(\mathbf{x})]\,. \tag{4.1.12}$$

However, before this can be done the relation of (4.1.12) and σ has to be investigated. Suppose that (4.1.11) is true for f. From this it follows

$$\begin{aligned}
\sqrt{\frac{2}{\pi}-\varepsilon} &\leq \sigma \leq \sqrt{\frac{2}{\pi}}+2c\varepsilon \\
\sqrt{\frac{2}{\pi}-\varepsilon} &\leq \sigma \leq \mathop{\mathbf{E}}_{\mathbf{x}}\left[\left|l\left(\mathbf{x}\right)\right|\right] \leq \sqrt{\frac{2}{\pi}}+2c\varepsilon && (4.1.8) \\
\sqrt{\frac{2}{\pi}}-2\varepsilon &\leq \sigma \leq \mathop{\mathbf{E}}_{\mathbf{x}}\left[\left|l\left(\mathbf{x}\right)\right|\right] \leq \sqrt{\frac{2}{\pi}}+2c\varepsilon. && (\text{For } \varepsilon \in [0, {}^{2}/_{\pi}])
\end{aligned}$$

From 4.1.10 it follows

$$\begin{aligned}
\mathop{\mathbf{E}}_{\mathbf{x}}\left[\left|l\left(\mathbf{x}\right)\right|\right]-\langle f,l\rangle &\leq \sqrt{\frac{2}{\pi}}+2c\varepsilon-\sqrt{\frac{2}{\pi}}+2\varepsilon \\
&\leq 2\varepsilon\left(c+1\right). && (4.1.13)
\end{aligned}$$

The following rearrangement shows the relation between the expected value (4.1.12) and the upper bound (4.1.13).

$$\begin{aligned}
\mathop{\mathbf{E}}_{\mathbf{x}}\left[\left|l\left(\mathbf{x}\right)\right|\right]-\langle f,l\rangle &= \mathop{\mathbf{E}}_{\mathbf{x}}\left[\left|l\left(\mathbf{x}\right)\right|\right]-\mathop{\mathbf{E}}_{\mathbf{x}}\left[f\left(\mathbf{x}\right)\cdot l\left(\mathbf{x}\right)\right] && (\text{Theorem } 2.2) \\
&= \mathop{\mathbf{E}}_{\mathbf{x}}\left[\left|l\left(\mathbf{x}\right)\right|-f\left(\mathbf{x}\right)\cdot l\left(\mathbf{x}\right)\right] \\
&= \mathop{\mathbf{E}}_{\mathbf{x}}\left[\operatorname{sgn}\left(l\left(\mathbf{x}\right)\right)\cdot l\left(\mathbf{x}\right)-f\left(\mathbf{x}\right)\cdot l\left(\mathbf{x}\right)\right] \\
&= \mathop{\mathbf{E}}_{\mathbf{x}}\left[\left(\operatorname{sgn}\left(l\left(\mathbf{x}\right)\right)-f\left(\mathbf{x}\right)\right)\cdot l\left(\mathbf{x}\right)\right] && (4.1.14)
\end{aligned}$$

From (4.1.14) and (4.1.13) it follows

$$\mathop{\mathbf{E}}_{\mathbf{x}}\left[\left(\operatorname{sgn}\left(l\left(\mathbf{x}\right)\right)-f\left(\mathbf{x}\right)\right)\cdot l\left(\mathbf{x}\right)\right] \leq 2\varepsilon\left(c+1\right). \qquad (4.1.15)$$

The next step estimates an upper bound for the probability of $\left|l\left(\mathbf{x}\right)\right|$ to be smaller than a constant value, or in other words to be very small. For any constant $K \geq 1$:

$$\mathop{\mathbf{Pr}}_{\mathbf{x}}\left[\left|l\left(\mathbf{x}\right)\right| \leq K\sqrt{\varepsilon}\right] = \mathbf{Pr}\left[\left|\sum_{i=1}^{n}\mathbf{X}_i\right| \leq K\sqrt{\varepsilon}\right]. \qquad (\mathbf{X}_i = \widehat{f}(i)\chi_i(\mathbf{x})/\sigma)$$

Due to $\left|\widehat{l}\left(i\right)\right| \leq 2\varepsilon$, the random variables fulfill $\left|\mathbf{X}_i\right| \leq 2\varepsilon$.
From (4.1.1) and Remark 4.1 it follows

$$\begin{aligned}
\mathbf{Pr}\left[\left|\sum_{i=1}^{n}\mathbf{X}_i\right| \leq K\sqrt{\varepsilon}\right] &\leq \mathbf{Pr}\left[\left|\mathbf{Z}\right| \leq K\sqrt{\varepsilon}\right]+c\cdot 2\varepsilon \\
&= \mathbf{Pr}\left[-K\sqrt{\varepsilon} \leq \mathbf{Z} \leq K\sqrt{\varepsilon}\right]+c\cdot 2\varepsilon \\
&\leq \frac{1}{\sqrt{2\pi}}\cdot 2K\sqrt{\varepsilon}+c\cdot 2\varepsilon && (\text{integral estimation}) \\
&\leq \left(\frac{1}{\sqrt{2\pi}}+\frac{\sqrt{\varepsilon}\cdot c}{K}\right)\cdot 2K\sqrt{\varepsilon} \\
&\leq \left(\frac{1}{2}+\frac{1}{2}\right)2K\sqrt{\varepsilon} && (\varepsilon \in [0, {}^{2}/_{\pi}]) \\
&\leq 2K\sqrt{\varepsilon}. && (4.1.16)
\end{aligned}$$

The proof by contradiction uses these implications

$$\begin{aligned}
&\Pr_{\mathbf{x}}\left[f(\mathbf{x}) \neq \operatorname{sgn}(l(\mathbf{x}))\right] \geq 3K\sqrt{\varepsilon} \\
\Rightarrow &\Pr_{\mathbf{x}}\left[f(\mathbf{x}) \neq \operatorname{sgn}(l(\mathbf{x})) \wedge |l(\mathbf{x})| \geq K\sqrt{\varepsilon}\right] \geq K\sqrt{\varepsilon} \\
\Rightarrow &\mathop{\mathbf{E}}_{\mathbf{x}}\left[(\operatorname{sgn}(l(\mathbf{x})) - f(\mathbf{x})) \cdot l(\mathbf{x})\right] \geq 2K^2\varepsilon
\end{aligned} \tag{4.1.17}$$

and it follows an in-depth derivation of them.
For $K \geq 1$ assume

$$\Pr_{\mathbf{x}}\left[f(\mathbf{x}) \neq \operatorname{sgn}(l(\mathbf{x}))\right] \geq 3K\sqrt{\varepsilon}. \tag{4.1.18}$$

For better readability it is useful to define

$$\Pr_{\mathbf{x}}\left[f(\mathbf{x}) \neq \operatorname{sgn}(l(\mathbf{x})) \wedge |l(\mathbf{x})| \geq K\sqrt{\varepsilon}\right] = \Pr_{\mathbf{x}}\left[A \cap B\right], \tag{4.1.19}$$

where

$$\begin{aligned}
A &= \{\mathbf{x} : f \neq \operatorname{sgn}(l(\mathbf{x}))\} \\
B &= \left\{\mathbf{x} : |l(\mathbf{x})| \geq K\sqrt{\varepsilon}\right\} \\
B^c &= \left\{\mathbf{x} : |l(\mathbf{x})| \leq K\sqrt{\varepsilon}\right\}.
\end{aligned}$$

Since $\mathbf{Pr}_{\mathbf{x}}[B]$ is not known, the law of total probability is used to estimate a lower bound for (4.1.19).

$$\begin{aligned}
\Pr_{\mathbf{x}}[A] &= \Pr_{\mathbf{x}}[A \mid B] \cdot \Pr_{\mathbf{x}}[B] + \Pr_{\mathbf{x}}[A \mid B^c] \cdot \Pr_{\mathbf{x}}[B^c] \\
&= \Pr_{\mathbf{x}}[A \cap B] + \Pr_{\mathbf{x}}[A \mid B^c] \cdot \Pr_{\mathbf{x}}[B^c] \\
\Leftrightarrow \Pr_{\mathbf{x}}[A \cap B] &= \Pr_{\mathbf{x}}[A] - \Pr_{\mathbf{x}}[A \mid B^c] \cdot \Pr_{\mathbf{x}}[B^c] \\
\Pr_{\mathbf{x}}[A \cap B] &\geq 3K\sqrt{\varepsilon} - \Pr_{\mathbf{x}}[A \mid B^c] \cdot 2K\sqrt{\varepsilon} && ((4.1.18) \text{ and } (4.1.16)) \\
&\geq 3K\sqrt{\varepsilon} - 2K\sqrt{\varepsilon} && (\mathbf{Pr}_{\mathbf{x}}[A \mid B^c] \in [0,1]) \\
&= K\sqrt{\varepsilon} && (4.1.20)
\end{aligned}$$

Now the necessary lower bound for the expected value (4.1.12) can be estimate. Let $\mathbf{y} \in \left\{\mathbf{x} : f(\mathbf{x}) \neq \operatorname{sgn}(l(\mathbf{x})) \wedge |l(\mathbf{x})| \geq K\sqrt{\varepsilon}\right\}$ such that,

$$\begin{aligned}
&\mathop{\mathbf{E}}_{\mathbf{x}}\left[(\operatorname{sgn}(l(\mathbf{x})) - f(\mathbf{x})) \cdot l(\mathbf{x})\right] \\
=&\mathop{\mathbf{E}}_{\mathbf{x}}\left[(\operatorname{sgn}(l(\mathbf{x})) - f(\mathbf{x})) \cdot \operatorname{sgn}(l(\mathbf{x})) \cdot |l(\mathbf{x})|\right] \\
\geq&\mathop{\mathbf{E}}_{\mathbf{y}}\left[(\operatorname{sgn}(l(\mathbf{y})) - f(\mathbf{y})) \cdot \operatorname{sgn}(l(\mathbf{y})) \cdot |l(\mathbf{y})|\right] \\
=&\mathop{\mathbf{E}}_{\mathbf{y}}\left[2 \cdot |l(\mathbf{y})|\right] && (f(\mathbf{y}) \neq \operatorname{sgn}(l(\mathbf{y}))) \\
\geq&K\sqrt{\varepsilon} \cdot 2 \cdot K\sqrt{\varepsilon} && ((4.1.20) \text{ and } |l(\mathbf{x})| = K\sqrt{\varepsilon}) \\
=&2K^2\varepsilon.
\end{aligned}$$

For $K = \sqrt{c+2} \approx 1.6$ the expected value $2\sqrt{c+2}^2 \varepsilon = 5.0258\varepsilon$ contradicts the upper bound (4.1.13) $2\varepsilon\,(c+1) = 2.0258\varepsilon$ ((4.1.17) is wrong). Thus

$$\begin{aligned}\Pr_{\mathbf{x}}\left[f\left(\mathbf{x}\right) \neq \operatorname{sgn}\left(l\left(\mathbf{x}\right)\right)\right] \leq & 3K\sqrt{\varepsilon} \\ \leq & 3\sqrt{c+2}\sqrt{\varepsilon} \\ \leq & O(\sqrt{\varepsilon})\end{aligned} \tag{4.1.21}$$

holds. □

DISTANCE TO UNBIASED LINEAR THRESHOLD FUNCTIONS

From a theoretical point of view, the $2/\pi$ theorem provides an excellent way to estimate the distance of any Boolean function to an unbiased LTF. To do so, only all degree-one Fourier coefficients of the Boolean function are needed. If one wants to determine the distance of a PUF instance to an unbiased LTF, then one probably has no information about the Fourier coefficients or the $(\varepsilon, 1)$-regularity of this PUF. Therefore, the next Section 5.2 describes the approximation of degree-one Fourier coefficients, to be able to get insights about the $(\varepsilon, 1)$-regularity and to approximate the degree-one weight. Besides, in Section 6, the estimate of the degree-one weights is examined in two different ways.

The approaches presented in this thesis to approximate degree-one Fourier coefficients and degree-one weights are probabilistic algorithms that require a certain number of CRPs to achieve an absolute error for the approximation with a sufficient probability. With this background, it is interesting to see how significant the absolute error can be to describe a meaningful correlation between the degree-one Fourier coefficients and the degree-one weight and the distance of a Boolean function to an unbiased LTF, utilizing the $2/\pi$ theorem. For this purpose, Figure 4.2.1 shows how small the ε of an $(\varepsilon, 1)$-regular Boolean function f must be to determine an upper bound for the distance from f to an unbiased LTF, which lies between 0 and 1. The inequality (4.2.1) shows that for $(\varepsilon, 1)$-regularity parameters of a Boolean function, with

$$1 \leq 3\sqrt{c+2}\sqrt{\varepsilon} \tag{4.1.21}$$

$$\frac{1}{9 \cdot (c+2)^2} \leq \varepsilon \tag{4.2.1}$$

a meaningful distance to an LTF can be determined. From this it follows, that the absolute error in the approximation of the degree-one Fourier coefficients and the Fourier coefficients them selfs should fall below the limit in inequality (4.2.1). Due to the nature of the approximation errors, the distance to an unbiased LTF may be over- or underestimated. Therefore, the distance estimated with approximated

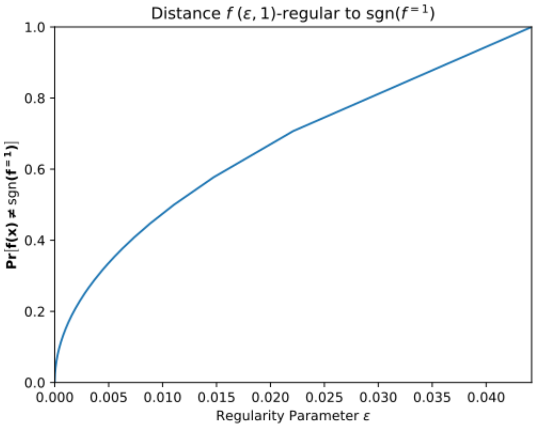

Figure 4.2.1: Distance for an $(\varepsilon, 1)$-regular Boolean function f between 0 and 1. The distance is calculated by (4.2.1) with $c = 0.5129$ according to Korolev and Shevtsova [KS10].

degree-one Fourier coefficients or degree-one weights should not be regarded as unambiguous. Section 6.3 discusses the relationship between the absolute error of a degree-one weight approximation and the distance to an unbiased LTF. Furthermore, it is shown in Section 6 that the upper bounds proven in this thesis require a large number of CRPs in order to specify a provable and meaningful distance of a Boolean function to an unbiased LTF.

5 FOURIER COEFFICIENT APPROXIMATION

The Definition 2.4 says that the degree-one weight of a Boolean function is the sum of the squared degree-one Fourier coefficients. Therefore, this chapter presents two probabilistic ways of estimating degree-one Fourier coefficients. The first variant uses the Low-Degree algorithm of Linial et al. [LMN93], a probabilistic algorithm, which learns a Boolean function with a specific absolute error for the Fourier coefficients. Further, the Low-Degree algorithm requires a degree-one weight that is too high to give a meaningful distance estimation according to Theorem 4.3. Hence, a second variant of approximation is considered. The second variant of the estimation of the degree-one Fourier coefficients uses, similar to the Low-Degree algorithm, the approximation of the Fourier coefficients by the arithmetic mean, based on Theorem 2.5. An essential quantity when approximating Fourier coefficients is the resulting absolute error and the two methods diverge especially in the number of queries to guarantee a sufficiently small absolute error.

LOW-DEGREE ALGORITHM

This section presents the Low-Degree algorithm which is an algorithm to approximate a Boolean function through its Fourier expansion. The focus will be on the Fourier coefficients which are essential for determining the epsilon bound of the degree-one weight according to Theorem 4.3. The Low-Degree algorithm was invented by Linial et al. [LMN89]. Additional information about this algorithm is shown by O'Donnell too, but this thesis will use a derived definition from Linial et al. *[LMN93; ODo14].*

Definition 5.1. (Low-Degree algorithm [LMN93, Section 4])
The Low-Degree algorithm takes the tuple $L^\circ = (f, k, \varepsilon, \delta)$ as input. Where $f : \{-1,1\}^n \to \{-1,1\}$ is a Boolean function with the concentration

$$\sum_{|S|>k} \widehat{f}(S)^2 \leq \frac{\varepsilon}{2}, \tag{5.1.1}$$

the error constant $\varepsilon > 0$ and the confidence $\delta \in [0,1]$.

1. Generate $m \geq 4\left(2n^k/\varepsilon\right)\ln\left(2n^k/\delta\right)$ random inputs $\mathbf{x^1}, \ldots, \mathbf{x^m} \sim \{-1,1\}^n$.
2. For all $S \subseteq [n]$, with $|S| \leq k$, calculate the approximated Fourier coefficients
$$\widetilde{f}(S) = \frac{1}{m}\sum_{i=1}^{m} f\left(\mathbf{x^i}\right) \cdot \chi_S\left(\mathbf{x^i}\right).$$
3. Return the hypothesis function $h(x) = \operatorname{sgn}\left(\sum_{\substack{S\subseteq[n]\\|S|\leq k}} \widetilde{f}(S)\chi_S(x)\right)$.

The Low-Degree algorithm learns a function f and returns a hypothesis function h, such that h disagrees with f on no more than an ε fraction of the inputs, with confidence of at least $1-\delta$ [LMN93, Proof of Theorem 1.]. Besides, the core of the Low-Degree algorithm is the approximation of the Fourier coefficients. In the algorithm's proof, Linial et al. demonstrated that the algorithm approximates the Fourier coefficients $\widetilde{f}(S)$ where the following constraint

$$\left|\widetilde{f}(S) - \widehat{f}(S)\right| \leq \sqrt{\frac{\epsilon}{2n^k}}, \tag{5.1.2}$$

holds with confidence of at least $1-\delta$ for all subsets $S \subseteq [n]$ with $|S| \leq k$ [LMN93, Lemma 8]. Further, another essential attribute of this algorithm is its runtime. In order to ensure the probability parameters and the maximal degree of the Fourier coefficients, the approximation needs $m \geq 4(2n^k/\varepsilon)\ln(2n^k/\delta)$ challenge-response pairs. Hence, the challenge-response pair generation requires $O(n^k/\varepsilon \cdot \ln(2n^k/\delta))$ steps. Another quantity which impacts the execution time is the degree k. Concerning the degree k, the number of subsets of $[n]$ which are used for the Fourier coefficent approximation is bounded by $O(n^k)$. Furthermore, the coefficient approximation uses all m random inputs. Hence, the algorithm runs in $O\left(n^{2k}/\varepsilon \cdot \ln(2n^k/\delta)\right)$ steps.

A downside of the Low-Degree algorithm with the intent to check the closeness to an unbiased LTF is that the algorithm supposes an concentration according to (5.1.1), which influences the absolute error (5.1.2). For testing the LTF closeness through Theorem 4.3, it is necessary to approximate the degree-one Fourier coefficients with a small absolute error.

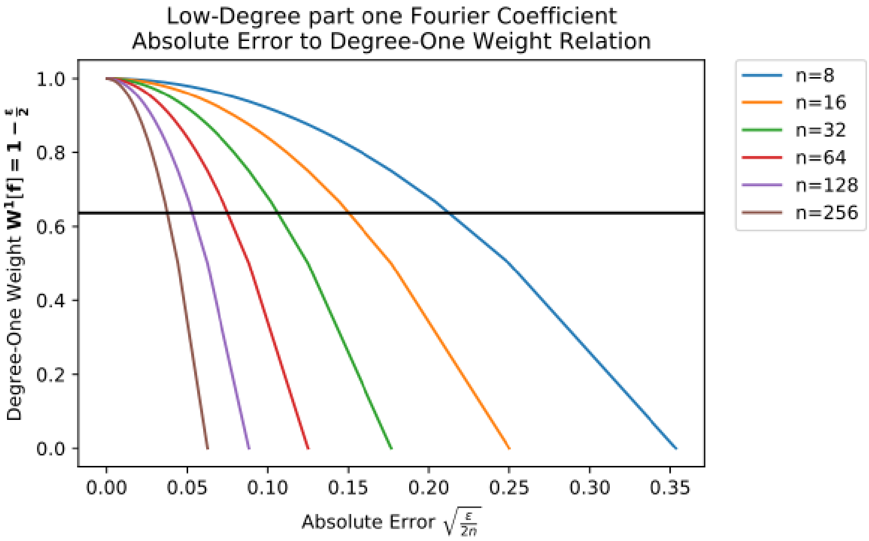

Figure 5.1.1: The degree-one weights $\mathbf{W}^1[f] = 1 - \varepsilon/2$ of an Boolean function in relation to the absolute error of the degree-one Fourier coefficient approximation through the Low-Degree algorithm, supposing a concentration according to (5.1.1).
Further, $\varepsilon \in [0, 2]$ and the black horizontal line marks $2/\pi$.

As a result, Figure 5.1.1 shows the degree-one weights which are needed to guarantee a sufficiently small absolute error of the approximated Fourier coefficients. Additionally, Figure 5.1.1 uses the fact from Theorem 2.4 that the sum of all squared Fourier coefficients is 1. From this, it follows that if f is concentrated according to (5.1.1) with $k = 1$, the degree-one weight is at least $1 - \varepsilon/2$, with an absolute error lower or equal to $\sqrt{\varepsilon/2n}$ and $1 - \delta$ confidence.

Furthermore, Figure 5.1.1 shows an interesting relation between the number of input bits, the absolute error $\left|\widehat{f}(S) - \widetilde{f}(S)\right|$. Derived by Figure 5.1.1 the possible absolute approximation error decreases with growing n. In addition to that, the degree-one weight must be sufficiently large in order to guarantee a particular small absolute error.

Theorem 6.1 defines a upper bound for the absolute error which occurs when calculating the degree-one weight with approximated degree-one Fourier coefficients. Using the upper bound from Theorem 6.1 makes the Low-Degree algorithm impractical because the worst case degree-one concentration would be far away from $2/\pi$ which is not close to an unbiased LTF. Hence, in the context of PUFs, the Low-Degree algorithm is unfeasible to determine the closeness to an unbiased LTF.

EMPIRICAL FOURIER COEFFICIENT APPROXIMATION

Section 5.1 displayed how to learn a Boolean function and approximate the Fourier coefficients with an error ε and confidence $1-\delta$. A necessary assumption to use the Low-Degree algorithm is that the function's weights are concentrated accroding to (5.1.1). O'Donnell shows that it is possible to empirically estimate a specific Fourier coefficient of an arbitrary Boolean function with $O\left(\ln\left(1/\delta\right)\cdot 1/\lambda^2\right)$ random inputs regarding a success probability $1-\delta$ and λ a two-sided error bound [OD014, Proposition 3.30.]. This section presents an rigorous proof for O'Donnell's argument about the approximation of Fourier coefficients of an arbitrary Boolean function.

Proposition 2.1 stated that the Fourier coefficients of a real-valued Boolean function f are the expected value of the product of f and the monomials of S for uniformly random chosen inputs. From Proposition 2.1 it follows that it is possible to approximate the Fourier coefficients through an arithmetic mean.

Proposition 5.1. *Let $f:\{-1,1\}^n\to\{-1,1\}$ be a Boolean function, $S\subseteq[n]$ and $\mathbf{x^1},\ldots,\mathbf{x^N}\sim\{-1,1\}^n$ then*

$$\widetilde{f}\left(S\right)=\frac{1}{N}\sum_{i=1}^{N}f\left(\mathbf{x}^{i}\right)\chi_S\left(\mathbf{x}^{i}\right)$$

is an approximation of $\widehat{f}\left(S\right)$.

Due to the probabilistic behavior of Proposition 5.1 the number of inputs is essential, to guarantee with a certain probability, that all approximated Fourier coefficients have a particular distance from the original Fourier coefficients. A tool to estimate the number of challenges for certain probability and error parameters for the arithmetic mean of independent random variables was presented in Theorem 2.10. Due to the Fourier Expansion 2.2 of Boolean functions, the random variables of the Fourier coefficient approximation are -1 or 1. Hoeffding also defined a general bound for $\mathbf{X}_i\in[a_i,b_i]$ which makes it possible to estimate a probabilistic absolute error for the Fourier coefficient approximation [Hoe63, Theorem 2]. For the sake of simplicity, this thesis uses the following lemma whose proof can be found in [Zen17].

Lemma 5.1. *(Hoeffding bound [Zen17, Lemma 2.4.]) Let $\mathbf{X}=\mathbf{X}_1,\ldots\mathbf{X}_n$ be random variables drawn independently from the same distribution with $\mathbf{X}_i\in[-1,1]$ and $\mathbf{T}=\frac{1}{N}\sum_{i=1}^{N}\mathbf{X}_i$. Then for $\lambda>0$,*

$$\Pr_{\mathbf{X}_i}\left[\left|\mathbf{T}-\mathop{\mathbf{E}}_{\mathbf{X}_i}\left[\mathbf{T}\right]\right|\geq\lambda\right]\leq 2e^{-\lambda^2N/2}.$$

The next lemma utilizes Lemma 5.1 to estimate the number of queries to guarantee an absolute error bound $\lambda>0$.

Lemma 5.2. *Let* $\mathbf{X} = \mathbf{X}_1, \dots \mathbf{X}_n$ *be random variables drawn independently from the same distribution with* $\mathbf{X}_i \in [-1,1]$ *and* $\mathbf{T} = \frac{1}{N}\sum_{i=1}^{N}\mathbf{X}_i$*. Then* $N \geq O\left(\log(1/\delta)/\lambda^2\right)$ *random variables are needed to guarantee with probability* $1-\delta$ *that*

$$\left|\mathbf{T} - \mathop{\mathbf{E}}_{\mathbf{X}_i}[\mathbf{T}]\right| \leq \lambda$$

holds.

Proof. (Lemma 5.2)
The requirements of Lemma 5.2 fulfill the conditions of Lemma 5.1. From this it follows

$$\begin{aligned}
\mathop{\mathbf{Pr}}_{\mathbf{X}_i}\left[\left|\mathbf{T} - \mathop{\mathbf{E}}_{\mathbf{X}_i}[\mathbf{T}]\right| \geq \lambda\right] \leq 2e^{-\lambda^2 N/2} &\leq \delta \\
-\lambda^2\frac{N}{2} &\leq \log\left(\frac{\delta}{2}\right) \\
-\lambda^2\frac{N}{2} &\leq \log(\delta) - \log(2) \\
\frac{N}{2} &\geq \frac{\log\left(\frac{1}{\delta}\right) + \log(2)}{\lambda^2} \\
N &\geq 2 \cdot \left(\frac{\log\left(\frac{1}{\delta}\right) + \log(2)}{\lambda^2}\right) \\
N &\geq O\left(\frac{\log\left(\frac{1}{\delta}\right)}{\lambda^2}\right).
\end{aligned}$$

The Lower bound $O\left(\log(1/\delta)/\lambda^2\right)$ for the number of random variables guarantees with confidence $1-\delta$ that

$$\left|\mathbf{T} - \mathop{\mathbf{E}}_{\mathbf{X}_i}[\mathbf{T}]\right| \leq \lambda.$$

□

Combining Proposition 5.1 and Lemma 5.1 leads to the following theorem.

Theorem 5.1. *Let* $f : \{-1,1\}^n \to \{-1,1\}$ *be a Boolean function,* $S \subseteq [n]$ *an index set, then the estimation* $\widetilde{f}(S)$ *introduced by Proposition 5.1 needs*

$$O\left(\frac{\log\left(\frac{1}{\delta}\right)}{\lambda^2}\right)$$

samples to guarantee with probability $1-\delta$ *that the absolute error parameter of the estimation is lower* $\lambda > 0$.

Proof. (Theorem 5.1)
First of all it is necessary to recall the relationship of the Fourier coefficients and the parity function from Proposition 2.1

$$\widehat{f}(S) = \langle f, \chi_S \rangle = \mathop{\mathbf{E}}_{\mathbf{x}}[f(\mathbf{x})\chi_S(\mathbf{x})]. \tag{5.2.1}$$

Secondly, remember the definition of the Fourier coefficient approximation for sufficient large N,

$$\widetilde{f}\,(S) = \frac{1}{N}\sum_{i=1}^{N} f\left(\mathbf{x^i}\right)\chi_S\left(\mathbf{x^i}\right). \tag{5.2.2}$$

Equation 5.2.1 and 5.2.2 have the equivalent term

$$f(\mathbf{x^i})\chi_S\left(\mathbf{x^i}\right). \tag{5.2.3}$$

The product $f(\mathbf{x^i})\chi_S\left(\mathbf{x^i}\right)$ depends on an independently uniform at random drawn input which makes it to a random variable $\mathbf{X}_i$. Further, the random variable $\mathbf{X}_i$ is in the interval $[-1,1]$. From this, it follows that Lemma 5.1 is applicable and the estimate

$$\begin{aligned}\mathop{\mathbf{Pr}}_{\mathbf{X_i}}\left[\left|\frac{1}{N}\sum_{i=1}^{N} f(\mathbf{x^i})\chi_S\left(\mathbf{x^i}\right) - \mathop{\mathbf{E}}_{\mathbf{X}_i}\left[\frac{1}{N}\sum_{i=1}^{N} f(\mathbf{x^i})\chi_S\left(\mathbf{x^i}\right)\right]\right|\right] &\le 2e^{-\lambda^2 N/2}\\ \mathbf{Pr}\left[\left|\widetilde{f}\,(S) - \widehat{f}\,(S)\right| \ge \lambda\right] &\le 2e^{-\lambda^2 N/2} \le \delta\\ \mathbf{Pr}\left[\left|\widetilde{f}\,(S) - \widehat{f}\,(S)\right| \ge \lambda\right] &\le \delta \qquad (5.2.4)\end{aligned}$$

can be done. From (5.2.4) it follows that

$$\mathbf{Pr}\left[\left|\widetilde{f}\,(S) - \widehat{f}\,(S)\right| \le \lambda\right] \ge 1-\delta. \tag{5.2.5}$$

Last but not least, it is necessary to estimate the number of random samples. Since the conditions of Lemma 5.1 are met, Lemma 5.2 can be used to. According to Lemma 5.2 $O\left(\log(1/\delta)/\lambda^2\right)$ random inputs are sufficient to guarantee 5.2.5 with confidence $1-\delta$. □

Algorithm 5.1 degree-one Fourier coefficient approximation

```
Input
    • Boolean function $f : \{-1,1\}^n \to \{-1,1\}$
    • Confidence $1-\delta$
    • Approximation error $\lambda > 0$
    • Random samples $\mathbf{x^1}, \dots, \mathbf{x^N} \sim \{-1,1\}^n$
Execution
$N = \left\lceil 2 \cdot \left(\frac{\log(n)+\log(2)-\log(\delta)}{\lambda^2}\right)\right\rceil$
foreach $S \in \{1, \dots n\}$ do
    • Compute $\widetilde{f}\,(S) = \frac{1}{N}\sum_{i=1}^{N} f(\mathbf{x^i})\chi_S\left(\mathbf{x^i}\right)$
end
Output $\widetilde{f}\,(1), \dots, \widetilde{f}\,(n)$
```

With some facts from the proof of Theorem 5.1 it is possible to form the Algorithm 5.1 which can be used to approximate the degree-one Fourier coefficients with an error $\lambda > 0$ and confidence $1 - \delta$ to satisfy the error bound for every single Fourier coefficient.

Theorem 5.2. *The degree-one Fourier coefficient approximation Algorithm 5.1 approximates the degree-one Fourier coefficients of a Boolean function $f : \{-1, 1\}^n \rightarrow \{-1, 1\}$ for $N \geq O\left(\frac{\log(n) - \log(\delta)}{\lambda^2}\right)$ random samples with an absolute error $\lambda > 0$ and confidence $1 - \delta$.*

Proof. (Theorem 5.2)
The proof of Theorem 5.1 uses Lemma 5.2 to estimate the number of samples needed to guarantee an absolute error $\lambda > 0$ with confidence $1 - \delta$. The estimation is based on Lemma 5.1 and is a lower bound of random inputs for one Fourier coefficient approximation to have an absolute error of $\lambda > 0$. The probability that all n degree-one Fourier coefficients have an absolute error of $\lambda > 0$ can be estimated with the union bound and utilizing Lemma 5.1

$$\mathbf{Pr}\left[\bigcup_{i=1}^{n} \left|\widetilde{f}(i) - \widehat{f}(i)\right| \geq \lambda\right] \leq n \cdot 2e^{-\lambda^2 N/2} \leq \delta. \tag{5.2.6}$$

From (5.2.6) it follows

$$\begin{aligned}
2e^{-\lambda^2 N/2} &\leq \frac{\delta}{n} \\
e^{-\lambda^2 N/2} &\leq \frac{\delta}{2n} \\
\frac{-\lambda^2 N}{2} &\leq \log\left(\frac{\delta}{2n}\right) \\
\frac{N}{2} &\geq -\frac{\log\left(\frac{\delta}{2n}\right)}{\lambda^2} \\
\frac{N}{2} &\geq \frac{\log(2n) - \log(\delta)}{\lambda^2} \\
N &\geq 2 \cdot \left(\frac{\log(n) + \log(2) - \log(\delta)}{\lambda^2}\right) \\
N &\geq O\left(\frac{\log(n) - \log(\delta)}{\lambda^2}\right).
\end{aligned}$$

□

With the degree-one Fourier coefficients calculated by Algorithm 5.1, it is possible to determine an approximated ε for an $(\varepsilon, 1)$-regular Boolean function. In the case of unbiased degree-one Fourier coefficients, ε is at least the $\max\left\{\left|\widehat{f}(1)\right|, \ldots, \left|\widehat{f}(n)\right|\right\} \leq \varepsilon$. Utilizing the approximated degree-one Fourier coefficients to estimate ε it could happen, that due to the absolute error λ the $\max\left\{\left|\widetilde{f}(1)\right|, \ldots, \left|\widetilde{f}(n)\right|\right\}$

under- or overestimates ε. However, this is a first step to be able to determine whether the function is likely to be close to an unbiased LTF according to Theorem 4.3.

An equally important part of Theorem 4.3, but also as property metric even for PUFs, is the degree-one weight. Since the degree-one weight of a Boolean function offers even more information about the possible membership, e.g., to the class of dictator functions, a more in-depth discussion of the approximated degree-one Fourier coefficients concerning Theorem 4.3 is omitted.

In Section 6.3 it is shown that utilizing Theorem 4.3, the distance of a Boolean function f to an unbiased LTF can be estimated by using the approximated degree-one weight of f.

6 DEGREE-ONE WEIGHT APPROXIMATION

Since this thesis suggests the degree-one weight of Boolean functions as a property metric for PUFs, it is essential to know how to calculate the degree-one weight practically. Since this thesis considers only probabilistic approximations to the real degree-one weight of a Boolean function, quality criteria such as the absolute error, and thus the numbers of random inputs are of interest, which are not likely to exceed a specific variance from the real degree-one weight. For this purpose, two methods for approximating the degree-one weight are presented in this section. They differ significantly in the lower bound for the number of random inputs required to guarantee a theoretically meaningful approximation.

The first method presented is based on the Fourier coefficient approximation method shown in Section 5.2. The second method is based on the idea of Matulef et al. [Mat+10, Lemma 15.] which estimates the degree-one weight directly.

A general term for the approximated degree-one weight of a Boolean function serves the following definition.

Definition 6.1. Let $f : \{-1,1\}^n \to \{-1,1\}$ be a Boolean function then call $\widetilde{\mathbf{W}}^1[f]$ the approximated degree-one weight of f and

$$\left|\widetilde{\mathbf{W}}^1[f] - \mathbf{W}^1[f]\right| \leq \mu$$

defines an upper bound for the absolute estimation error $\mu > 0$.

In Definition 6.1, it is irrelevant which method generates the approximated degree-one weight.

EMPIRICAL DEGREE-ONE WEIGHT APPROXIMATION

This section considers the use of approximated degree-one Fourier coefficients from Section 5.2 to estimate the degree-one weight of a Boolean function.

Definition 6.2 introduces a name for the Fourier coefficients calculated by Algorithm 5.1.

Definition 6.2. Let $f : \{-1,1\}^n \to \{-1,1\}$ be a Boolean function, $\lambda > 0$ the maximal absolute error and $1-\delta$ the probability that the degree-one Fourier coefficient approximation Algorithm 5.1 returns the correct approximated Fourier coefficients $\widetilde{f}(1), \ldots, \widetilde{f}(n)$. Then call $\widetilde{f}(1), \ldots, \widetilde{f}(n)$ the (λ, δ) Fourier coefficients.

Definition 2.4 describes the composition of the Fourier weights. In the case of degree-one weight, it is only the summed squares of the degree-one Fourier coefficients. For completeness, the following definition describes the generation of an approximated degree-one weight by (λ, δ) Fourier coefficients.

Definition 6.3. Let $f : \{-1, 1\}^n \to \{-1, 1\}$ be a Boolean function and $\widetilde{f}(1), \ldots, \widetilde{f}(n)$ the (λ, δ) Fourier coefficients for f then

$$\widetilde{\mathbf{W}}^1[f] = \sum_{i=1}^{n} \widetilde{f}(i)^2$$

is an approximated degree-one weight.

Further, the impact of the approximation error of the degree-one weight according to Definition 6.3 concerning the actual degree-one Fourier weight is essential for the use as property metric.

Theorem 6.1. *Let $f : \{-1, 1\}^n \to \{-1, 1\}$ be a Boolean function, $\lambda > 0$ and confidence $1 - \delta$ then*

$$\left| \widetilde{\mathbf{W}}^1[f] - \mathbf{W}^1[f] \right| \leq \lambda \cdot (2\sqrt{n} + n) \leq \mu$$

is an upper bound for the absolute error $\mu > 0$ of the degree-one weight according to Definition 6.3.

Theorem 6.1 shows a theoretical upper limit for the absolute error of approximation, which is mainly caused by the squaring and the number of (λ, δ) Fourier coefficients. It should be said that the bound presented in Theorem 6.1 is a worst case estimation assuming the maximum possible absolute error, given by the (λ, δ) Fourier coefficients. Indeed, as in the example of the Low-Degree algorithm, one could suppose a particular concentration of weights or assume that the function is $(\varepsilon, 1)$-regular with a sufficient small ε to reduce the theoretical error bound. But the idea behind Theorem 6.1 is to make as little restrictive assumptions as possible to include as many as possible Boolean functions.

Proof. (Theorem 6.1)
The goal of this proof is to find an upper and lower bound for the difference between the approximated degree-one weight and the degree-one weight to be able to specify an absolute error bound.

$$\widetilde{\mathbf{W}}^1[f] - \mathbf{W}^1[f] = \sum_{i=1}^{n} \widetilde{f}(i)^2 - \sum_{i=1}^{n} \widehat{f}(i)^2 \qquad \text{(Def. 6.3 and Def.2.4)}$$

The (λ, δ) Fourier coefficients $\widetilde{f}(1), \ldots, \widetilde{f}(n)$ are approximated by Algorithm 5.1 with $\lambda > 0$ and confidence $1 - \delta$. Further, the Algorithm 5.1 is based on Theorem 5.2. From this it follows

$$\begin{aligned} \left|\widetilde{f}\,(i) - \widehat{f}\,(i)\right| &\leq \lambda \\ \Leftrightarrow -\lambda \leq \quad \widetilde{f}\,(i) - \widehat{f}\,(i) &\leq \lambda. \\ \Leftrightarrow \widehat{f}\,(i) - \lambda \leq \quad \widetilde{f}\,(i) &\leq \widehat{f}\,(i) + \lambda. \end{aligned} \tag{6.1.1}$$

Two different absolute error bounds for $\widetilde{f}\,(i)$ are introduced by (6.1.1). The idea to determine an upper an lower bound is to plug in the bounds from (6.1.1) into

$$\sum_{i=1}^{n} \widetilde{f}\,(i)^2 - \sum_{i=1}^{n} \widehat{f}\,(i)^2 \tag{6.1.2}$$

and use a case analysis. First of all, the sum of all Fourier coefficients occurs in all cases and must be estimated in order to continue.

$$\begin{aligned} \sum_{i=1}^{n} \widehat{f}\,(i) &\leq \sqrt{\sum_{i=1}^{n} \widehat{f}\,(i)^2} \cdot \sqrt{\sum_{i=1}^{n} 1^2} && \text{(Cauchy-Schwarz)} \\ &\leq \sqrt{\sum_{i=1}^{n} \widehat{f}\,(i)^2} \cdot \sqrt{n} \\ &\leq 1 \cdot \sqrt{n} && \text{(2.1.4)} \\ &\leq \sqrt{n} && \text{(6.1.3)} \end{aligned}$$

Due to (2.1.4) and $\widehat{f}\,(i) \in [-1, 1]$, from (6.1.3) also follows

$$-\sqrt{n} \leq \sum_{i=1}^{n} \widehat{f}\,(i) \leq \sqrt{n}. \tag{6.1.4}$$

The next step is to combine the bounds from (6.1.4) and (6.1.1) with (6.1.2).
Case $\widetilde{f}\,(i) = \widehat{f}\,(i) + \lambda$ and $\sum_{i=1}^{n} \widehat{f}\,(i) = \sqrt{n}$:

$$\begin{aligned} \sum_{i=1}^{n} \widetilde{f}\,(i)^2 - \sum_{i=1}^{n} \widehat{f}\,(i)^2 &= \sum_{i=1}^{n} \left(\widehat{f}\,(i) + \lambda\right)^2 - \sum_{i=1}^{n} \widehat{f}\,(i)^2 \\ &= \sum_{i=1}^{n} \widehat{f}\,(i)^2 + 2\widehat{f}\,(i)\,\lambda + \lambda^2 - \sum_{i=1}^{n} \widehat{f}\,(i)^2 \\ &= \sum_{i=1}^{n} 2\widehat{f}\,(i)\,\lambda + \lambda^2 \\ &= n\lambda^2 + 2\lambda \sum_{i=1}^{n} \widehat{f}\,(i) \\ &= n\lambda^2 + 2\lambda\sqrt{n} \end{aligned}$$

Case $\widetilde{f}(i) = \widehat{f}(i) - \lambda$ and $\sum_{i=1}^{n} \widehat{f}(i) = \sqrt{n}$:

$$\begin{aligned}
\sum_{i=1}^{n} \widetilde{f}(i)^2 - \sum_{i=1}^{n} \widehat{f}(i)^2 &= \sum_{i=1}^{n} \left(\widehat{f}(i) - \lambda\right)^2 - \sum_{i=1}^{n} \widehat{f}(i)^2 \\
&= \sum_{i=1}^{n} \widehat{f}(i)^2 - 2\widehat{f}(i)\,\lambda + \lambda^2 - \sum_{i=1}^{n} \widehat{f}(i)^2 \\
&= \sum_{i=1}^{n} -2\widehat{f}(i)\,\lambda + \lambda^2 \\
&= n\lambda^2 - 2\lambda \sum_{i=1}^{n} \widehat{f}(i) \\
&= n\lambda^2 - 2\lambda\sqrt{n}
\end{aligned}$$

Case $\widetilde{f}(i) = \widehat{f}(i) + \lambda$ and $\sum_{i=1}^{n} \widehat{f}(i) = -\sqrt{n}$:

$$\begin{aligned}
\sum_{i=1}^{n} \widetilde{f}(i)^2 - \sum_{i=1}^{n} \widehat{f}(i)^2 &= \sum_{i=1}^{n} \left(\widehat{f}(i) + \lambda\right)^2 - \sum_{i=1}^{n} \widehat{f}(i)^2 \\
&= \sum_{i=1}^{n} \widehat{f}(i)^2 + 2\widehat{f}(i)\,\lambda + \lambda^2 - \sum_{i=1}^{n} \widehat{f}(i)^2 \\
&= \sum_{i=1}^{n} 2\widehat{f}(i)\,\lambda + \lambda^2 \\
&= n\lambda^2 + 2\lambda \sum_{i=1}^{n} \widehat{f}(i) \\
&= n\lambda^2 - 2\lambda\sqrt{n}
\end{aligned}$$

Case $\widetilde{f}(i) = \widehat{f}(i) - \lambda$ and $\sum_{i=1}^{n} \widehat{f}(i) = -\sqrt{n}$:

$$\begin{aligned}
\sum_{i=1}^{n} \widetilde{f}(i)^2 - \sum_{i=1}^{n} \widehat{f}(i)^2 &= \sum_{i=1}^{n} \left(\widehat{f}(i) - \lambda\right)^2 - \sum_{i=1}^{n} \widehat{f}(i)^2 \\
&= \sum_{i=1}^{n} \widehat{f}(i)^2 - 2\widehat{f}(i)\,\lambda + \lambda^2 - \sum_{i=1}^{n} \widehat{f}(i)^2 \\
&= \sum_{i=1}^{n} -2\widehat{f}(i)\,\lambda + \lambda^2 \\
&= n\lambda^2 - 2\lambda \sum_{i=1}^{n} \widehat{f}(i) \\
&= n\lambda^2 + 2\lambda\sqrt{n}
\end{aligned}$$

The different worst case estimation show that (6.1.2) is bounded by

$$\begin{aligned}
n\lambda^2 - 2\lambda\sqrt{n} \leq \sum_{i=1}^{n} \widetilde{f}(i)^2 - \sum_{i=1}^{n} \widehat{f}(i)^2 &\leq n\lambda^2 + 2\lambda\sqrt{n} \\
\Rightarrow -n\lambda^2 - 2\lambda\sqrt{n} \leq \sum_{i=1}^{n} \widetilde{f}(i)^2 - \sum_{i=1}^{n} \widehat{f}(i)^2 &\leq n\lambda^2 + 2\lambda\sqrt{n} \\
\Leftrightarrow \left| \sum_{i=1}^{n} \widetilde{f}(i)^2 - \sum_{i=1}^{n} \widehat{f}(i)^2 \right| &\leq n\lambda^2 + 2\lambda\sqrt{n} \qquad (6.1.5)
\end{aligned}$$

From (6.1.5) it follows

$$\begin{aligned}\left|\widetilde{\mathbf{W}}^1[f]-\mathbf{W}^1[f]\right| &\leq n\lambda^2+2\lambda\sqrt{n}\\ &\leq n\lambda+2\lambda\sqrt{n}\\ &\leq \lambda\cdot\left(2\sqrt{n}+n\right).\end{aligned}$$

□

Next, the number of inputs is essential to ensure that the approximated degree-one weight specified in Definition 6.3 meets specific quality criteria.

Theorem 6.2. *Let $f:\{-1,1\}^n\to\{-1,1\}$ be a Boolean function then the approximated degree-one weight $\widetilde{\mathbf{W}}^1[f]$ can be calculated with*

$$O\left(n^2\cdot\left(\frac{\log(n)-\log(\delta)}{\mu^2}\right)\right)$$

queries, confidence $1-\delta$ and an absolute error μ.

Proof. (Theorem 6.2)
Theorem 6.1 says that the absolute error $\mu>0$ of approximated degree-one weights according to Definition 6.3 of an Boolean function $f:\{-1,1\}^n\to\{-1,1\}$ can be estimated by

$$\left|\mathbf{W}^1[f]-\widetilde{\mathbf{W}}^1[f]\right|\leq\lambda\cdot\left(2\sqrt{n}+n\right)\leq\mu. \tag{6.1.6}$$

From (6.1.6) it follows that μ depends on the absolute Fourier coefficient approximation error $\lambda>0$. Hence, for fixed n the absolute error μ has the following impact on every absolute Fourier coefficient approximation error

$$\lambda\leq\frac{\mu}{2\sqrt{n}+n}. \tag{6.1.7}$$

Theorem 5.2 specifies a lower bound of the number of queries N to approximate the degree-one Fourier coefficient of f with confidence $1-\delta$,

$$N\geq 2\cdot\left(\frac{\log(n)+\log(2)-\log(\delta)}{\lambda^2}\right). \tag{6.1.8}$$

To obtain a lower bound for the number of queries needed to approximated the degree-one weight it is sufficient to combine (6.1.7) and (6.1.8) into

$$
\begin{aligned}
N &\geq 2 \cdot \left(\frac{\log(n) + \log(2) - \log(\delta)}{\frac{\mu^2}{\left(2\sqrt{n}+n\right)^2}} \right) \\
&= 2 \cdot \left(2\sqrt{n} + n\right)^2 \cdot \left(\frac{\log(n) + \log(2) - \log(\delta)}{\mu^2} \right) \\
&= 2 \cdot (4n + 4\sqrt{n}n + n^2) \cdot \left(\frac{\log(n) + \log(2) - \log(\delta)}{\mu^2} \right) \\
&= n^2 \cdot \left(\frac{8}{n} + \frac{8}{\sqrt{n}} + 2 \right) \cdot \left(\frac{\log(n) + \log(2) - \log(\delta)}{\mu^2} \right) \\
&= O\left(n^2 \cdot \left(\frac{\log(n) - \log(\delta)}{\mu^2} \right) \right).
\end{aligned}
$$

□

Figure 6.1.1 illustrates the upper bound of queries needed to guarantee with confidence $1 - \delta = 0.99$ an absolute error μ of the degree-one weight approximation obtained through Theorem 6.2. Apparently, the number of required queries does not make sense for all combinations of the approximation parameter. If the number of inputs is larger than the input space itself, then Theorem 2.5 shows that 2^n queries are sufficient to calculate the degree-one weight correctly.

The number of inputs needed to guarantee with a sufficiently large confidence and a useful small absolute error is unusable in practice. Further, the reason for this high bound for the number of queries is the use of the union bound in Theorem 5.2 and the worst case estimation from Theorem 6.1. The experiments in section 7.2 reveal that in particular cases the estimated degree-one weight according Theorem 6.2 approaches the ideal degree-one weight with much fewer queries.

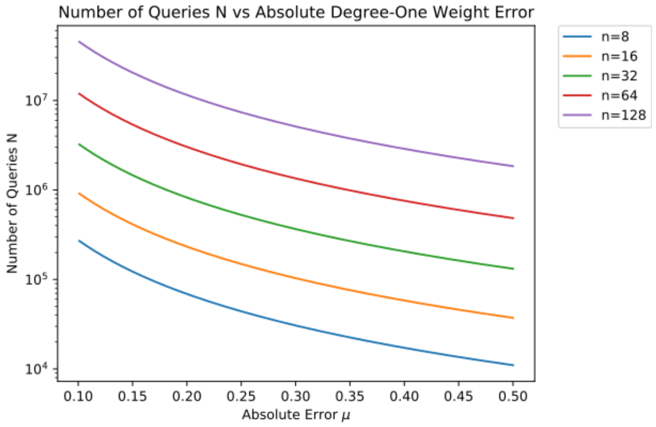

Figure 6.1.1: Number of queries needed by Theorem 6.2 to guarantee with confidence $1 - \delta = 0.99$ an absolute error $\mu \in [0.1, 0.5]$ of the degree-one weight for different input lengths.

approximate sums of powers of fourier coefficients

This section presents a probabilistic method from Kevin Matulef, Ryan O'Donnell, and Ronitt Rubinfeld to estimate the sum of products of Fourier coefficients with $O\left(p \cdot \frac{\log\left(\frac{1}{\delta}\right)}{\mu^4}\right)$ queries [Mat+10], where $p \geq 2$ is the number of functions used. Concerning $p = 2$ and the Boolean functions are equal, this is precisely the approximated degree-one weight of a Boolean function within an absolute error μ. For this purpose Lemma 6.1 introduces a relation between the expected value of specific products of Boolean functions and the sum of the Fourier coefficients of these functions.

Lemma 6.1. *([Mat+10, Lemma 14.]) For a fixed $p \geq 2$ and $T \subseteq [n]$, let $f_1, \ldots f_p : \{-1,1\}^n \to \{-1,1\}$ be p functions and $\mathbf{x^1}, \ldots, \mathbf{x^{p\text{-}1}}$ be independent uniform random strings in $\{-1,1\}^n$. Let $\mathbf{y}$ be a random string whose bits are independently chosen with $\mathbf{Pr_y}\left[\mathbf{y}_i = 1\right] = \frac{1}{2}$ for $i \notin T$ and $\mathbf{Pr_y}\left[\mathbf{y}_i = 1\right] = \frac{1}{2} + \frac{\mu}{2}$ for $i \in T$. Let $\odot$ denote the coordinate-wise multiplication then*

$$\underset{\mathbf{x^{\mathit{i}}},\mathbf{y}}{\mathbf{E}}\left[f_1\left(\mathbf{x^1}\right) f_2\left(\mathbf{x^2}\right) \ldots f_{p-1}\left(\mathbf{x^{\mathit{p}\text{-}1}}\right) f_p\left(\mathbf{x^1} \odot \mathbf{x^2} \odot \cdots \odot \mathbf{x^{\mathit{p}\text{-}1}} \odot \mathbf{y}\right)\right]$$
$$= \sum_{S \subseteq T} \mu^{|S|} \widehat{f_1}(S)\, \widehat{f_2}(S) \ldots \widehat{f_p}(S)\,.$$

Note in particular the choice of non-equally distributed inputs bits for an evaluation of the Boolean function f_p, which makes it possible that the expected value is not $\widehat{f_1}(\emptyset) \cdot \ldots \cdot \widehat{f_p}(\emptyset)$.

Proof. (Lemma 6.1)
The proof of Lemma 6.1 uses the ideas of Matulef et al. but is more detailed [Mat+10, Proof. Lemma 14.]. Writing the functions $f_1, \ldots f_p$ as Fourier expansions leads to

$$\underset{\mathbf{x^i},\mathbf{y}}{\mathbf{E}}\left[f_1\left(\mathbf{x^1}\right) f_2\left(\mathbf{x^2}\right) \ldots f_{p-1}\left(\mathbf{x^{p\text{-}1}}\right) f_p\left(\mathbf{x^1} \odot \mathbf{x^2} \odot \cdots \odot \mathbf{x^{p\text{-}1}} \odot \mathbf{y}\right)\right]$$
$$= \underset{\mathbf{x^i},\mathbf{y}}{\mathbf{E}}\left[\sum_{S_1,\ldots,S_p \subseteq [n]} \widehat{f_1}(S_1) \ldots \widehat{f_p}(S_p)\, \chi_{S_1}\left(\mathbf{x^1}\right) \ldots \chi_{S_{p-1}}\left(\mathbf{x^{p\text{-}1}}\right) \chi_{S_p}\left(\mathbf{x^1} \odot \cdots \mathbf{x^{p\text{-}1}} \odot \mathbf{y}\right)\right] \quad (6.2.1)$$

Now evaluate successive for all $\chi_{S_i}\left(\mathbf{x^i}\right)$ starting with $i=1$ the product

$$\begin{aligned}
&\chi_{S_1}\left(\mathbf{x^1}\right)\cdot\chi_{S_p}\left(\mathbf{x^1}\odot\cdots\odot\mathbf{x^{p-1}}\odot\mathbf{y}\right) \\
=&\prod_{j\in S_1}\mathbf{x}_j^{\mathbf{1}}\cdot\prod_{j\in S_p}\mathbf{x}_j^{\mathbf{1}}\cdot\ldots\cdot\mathbf{x}_j^{\mathbf{p-1}}\cdot\mathbf{y}_j && \text{(Definition 2.1)} \\
=&\prod_{j\in S_1\triangle S_p}\mathbf{x}_j^{\mathbf{1}}\cdot\prod_{j\in S_p}\mathbf{x}_j^{\mathbf{2}}\cdot\ldots\cdot\mathbf{x}_j^{\mathbf{p-1}}\cdot\mathbf{y}_j && \text{(2.1.1)} \\
=&\chi_{S_1\triangle S_p}\left(\mathbf{x^1}\right)\cdot\chi_{S_p}\left(\mathbf{x^2}\odot\cdots\odot\mathbf{x^{p-1}}\odot\mathbf{y}\right)
\end{aligned}$$

in order to obtain

$$\begin{aligned}
&\chi_{S_1}\left(\mathbf{x^1}\right)\ldots\chi_{S_{p-1}}\left(\mathbf{x^{p-1}}\right)\chi_{S_p}\left(\mathbf{x^1}\odot\cdots\mathbf{x^{p-1}}\odot\mathbf{y}\right) \\
=&\chi_{S_1\triangle S_p}\left(\mathbf{x^1}\right)\cdot\ldots\cdot\chi_{S_{p-1}\triangle S_p}\left(\mathbf{x^{p-1}}\right)\cdot\chi_{S_p}\left(\mathbf{y}\right).
\end{aligned}\tag{6.2.2}$$

Recalling (6.2.1) with functions $f_1,\ldots f_p$ as Fourier expansions and combining it with fact (6.2.2) leads to

$$\begin{aligned}
&\mathop{\mathbf{E}}_{\mathbf{x^i},\mathbf{y}}\left[\sum_{S_1,\ldots,S_p\subseteq[n]}\widehat{f_1}\left(S_1\right)\ldots\widehat{f_p}\left(S_p\right)\chi_{S_1\triangle S_p}\left(\mathbf{x^1}\right)\cdot\ldots\cdot\chi_{S_{p-1}\triangle S_p}\left(\mathbf{x^{p-1}}\right)\cdot\chi_{S_p}\left(\mathbf{y}\right)\right] \\
=&\sum_{S_1,\ldots,S_p\subseteq[n]}\widehat{f_1}\left(S_1\right)\ldots\widehat{f_p}\left(S_p\right)\cdot\mathop{\mathbf{E}}_{\mathbf{x^i},\mathbf{y}}\left[\chi_{S_1\triangle S_p}\left(\mathbf{x^1}\right)\cdot\ldots\cdot\chi_{S_{p-1}\triangle S_p}\left(\mathbf{x^{p-1}}\right)\cdot\chi_{S_p}\left(\mathbf{y}\right)\right].
\end{aligned}$$

As next step it is necessary to focus on

$$\mathop{\mathbf{E}}_{\mathbf{x^i},\mathbf{y}}\left[\chi_{S_1\triangle S_p}\left(\mathbf{x^1}\right)\cdot\ldots\cdot\chi_{S_{p-1}\triangle S_p}\left(\mathbf{x^{p-1}}\right)\cdot\chi_{S_p}\left(\mathbf{y}\right)\right].\tag{6.2.3}$$

Since the random variables $\mathbf{x^1},\ldots,\mathbf{x^{p-1}},\mathbf{y}$ are independent, from (6.2.3) it follows

$$\begin{aligned}
&\mathop{\mathbf{E}}_{\mathbf{x^i},\mathbf{y}}\left[\chi_{S_1\triangle S_p}\left(\mathbf{x^1}\right)\cdot\ldots\cdot\chi_{S_{p-1}\triangle S_p}\left(\mathbf{x^{p-1}}\right)\cdot\chi_{S_p}\left(\mathbf{y}\right)\right] \\
=&\mathop{\mathbf{E}}_{\mathbf{x^1}}\left[\chi_{S_1\triangle S_p}\left(\mathbf{x^1}\right)\right]\cdot\ldots\cdot\mathop{\mathbf{E}}_{\mathbf{x^{p-1}}}\left[\chi_{S_{p-1}\triangle S_p}\left(\mathbf{x^{p-1}}\right)\right]\cdot\mathop{\mathbf{E}}_{\mathbf{y}}\left[\chi_{S_p}\left(\mathbf{y}\right)\right].
\end{aligned}\tag{6.2.4}$$

Since the bits of $\mathbf{x^1},\ldots,\mathbf{x^{p-1}}\sim\{-1,1\}^n$ are distributed with $\mathbf{Pr}_{\mathbf{x^i}}\left[\mathbf{x}_j^{\mathbf{i}}=1\right]=1/2=\mathbf{Pr}_{\mathbf{x^i}}\left[\mathbf{x}_j^{\mathbf{i}}=-1\right]$, from Theorem 2.1 it follows that the only case when (6.2.4) is not 0 is when $S_1=\ldots=S_p=S\subseteq T$ which can expressed in

$$\begin{aligned}
&\mathop{\mathbf{E}}_{\mathbf{x^1}}\left[\chi_{S_1\triangle S_p}\left(\mathbf{x^1}\right)\right]\cdot\ldots\cdot\mathop{\mathbf{E}}_{\mathbf{x^{p-1}}}\left[\chi_{S_{p-1}\triangle S_p}\left(\mathbf{x^{p-1}}\right)\right]\cdot\mathop{\mathbf{E}}_{\mathbf{y}}\left[\chi_{S_p}\left(\mathbf{y}\right)\right] \\
=&1\cdot\ldots\cdot1\cdot\mathop{\mathbf{E}}_{\mathbf{y}}\left[\chi_{S_p}\left(\mathbf{y}\right)\right] \\
=&\mathop{\mathbf{E}}_{\mathbf{y}}\left[\chi_{S_p}\left(\mathbf{y}\right)\right].
\end{aligned}\tag{6.2.5}$$

Combining fact (6.2.5) with

$$\sum_{S_1,\ldots,S_p\subseteq[n]} \widehat{f_1}\left(S_1\right)\ldots\widehat{f_p}\left(S_p\right)\cdot \mathop{\mathbf{E}}_{\mathbf{x^i},\mathbf{y}}\left[\chi_{S_1\triangle S_p}\left(\mathbf{x^1}\right)\cdot\ldots\cdot\chi_{S_{p-1}\triangle S_p}\left(\mathbf{x^{p-1}}\right)\cdot\chi_{S_p}\left(\mathbf{y}\right)\right]$$

leads to

$$\sum_{S\subseteq T}\widehat{f_1}\left(S\right)\ldots\widehat{f_p}\left(S\right)\cdot\mathop{\mathbf{E}}_{\mathbf{y}}\left[\chi_S\left(\mathbf{y}\right)\right]$$

where $S_1 = \ldots = S_p = S \subseteq T$. Due to $\mathbf{y}$ is a random string whose bits are independently chosen with $\mathbf{Pr}_{\mathbf{y}}\left[\mathbf{y}_i = 1\right] = \frac{1}{2}$ for $i \notin T$ and $\mathbf{Pr}_{\mathbf{y}}\left[\mathbf{y}_i = 1\right] = \frac{1}{2} + \frac{\mu}{2}$ for $i \in T$, applying Theorem 2.1 leads to

$$\sum_{S\subseteq T}\mu^{|S|}\cdot\widehat{f_1}\left(S\right)\ldots\widehat{f_p}\left(S\right).$$

□

Lemma 6.1 gives an analytical relation between the expected value of the product of certain Boolean functions and their Fourier coefficients. The idea was to choose the inputs of function f_p non-uniformly distributed.

The next Theorem 6.3 shows how many queries are needed to guarantee with confidence $1-\delta$ that the sum of the products of the Fourier coefficients of $f_1\ldots f_p$ is within an additive $\pm\mu$.

Theorem 6.3. *([Mat+10, Lemma 15.]) For a fixed $p \geq 2$ and a $T \subseteq [n]$. Let $f_1\ldots f_p : \{-1,1\}^n \rightarrow \{-1,1\}$ have black-box access. Then the sum of the products of degree-one Fourier coefficients*

$$\sum_{i\in T}\widehat{f_1}\left(i\right)\ldots\widehat{f_p}\left(i\right)$$

can be estimated within an additive $\pm\mu$, with confidence $1-\delta$ using $O\left(p\cdot{}^{\log(1/\delta)}/_{\mu^4}\right)$ queries.

Proof. (Theorem 6.3)
The proof of Theorem 6.3 uses the ideas of Matulef et al. [Mat+10, Proof. Lemma 15.] but is more detailed. Further, the general idea of this proof is to approximate the sum of products of Fourier coefficients and subtract the sum by specific values until the approximated sum of products of degree-one Fourier coefficients is left.
Let $\mathbf{x^1},\ldots,\mathbf{x^{p-1}}$ be independent uniform random strings in $\{-1,1\}^n$ and $\mathbf{y}$ be chosen as described in Lemma 6.1.

Empirical estimate

$$\begin{aligned}
&\mathop{\mathbf{E}}_{\mathbf{x^i}}\left[f_1\left(\mathbf{x^1}\right)\cdot\ldots\cdot f_p\left(\mathbf{x^p}\right)\right] \\
=&\mathop{\mathbf{E}}_{\mathbf{x^1}}\left[f_1\left(\mathbf{x^1}\right)\right]\cdot\ldots\cdot\mathop{\mathbf{E}}_{\mathbf{x^p}}\left[f_p\left(\mathbf{x^p}\right)\right] && \text{(independence of } \mathbf{x^i}\text{)} \\
=&\widehat{f_1}\left(\emptyset\right)\cdot\ldots\cdot\widehat{f_p}\left(\emptyset\right) && \text{(Proposition 2.1 with } S=\emptyset\text{)}
\end{aligned}$$

and

$$\begin{aligned}
&\mathop{\mathbf{E}}_{\mathbf{x^i},\mathbf{y}}\left[f_1\left(\mathbf{x^1}\right)f_2\left(\mathbf{x^2}\right)\ldots f_{p-1}\left(\mathbf{x^{p-1}}\right)f_p\left(\mathbf{x^1}\odot\mathbf{x^2}\odot\cdots\odot\mathbf{x^{p-1}}\odot\mathbf{y}\right)\right] \\
&=\sum_{S\subseteq T}\mu^{|S|}\cdot\widehat{f_1}\left(S\right)\ldots\widehat{f_p}\left(S\right). && \text{(Lemma 6.1)}
\end{aligned}$$

with an absolute error of μ^2. From Lemma 5.2 it follows that the estimation needs $O\left(\log(1/\delta)/\mu^4\right)$ samples for each random variable to obtain an absolute error μ^2 with confidence $1-\delta$.
Hence, $O\left((p+1)\cdot\log(1/\delta)/\mu^4\right)$ queries are needed in total. Now subtracting the Fourier coefficients of the empty set from the sum obtained by Lemma (6.1) leads to

$$\begin{aligned}
&\left(\sum_{S\subseteq T}\mu^{|S|}\cdot\widehat{f_1}\left(S\right)\ldots\widehat{f_p}\left(S\right)\right)-\widehat{f_1}\left(\emptyset\right)\cdot\ldots\cdot\widehat{f_p}\left(\emptyset\right) \\
&=\sum_{\substack{S\subseteq T,\\ |S|>0}}\mu^{|S|}\cdot\widehat{f_1}\left(S\right)\ldots\widehat{f_p}\left(S\right) && (6.2.6)
\end{aligned}$$

within an additive $\pm\mu^2$. In order to obtain the sum of products of degree-one Fourier coefficients the next step is to estimate the Fourier coefficients of degree greater one.

$$\begin{aligned}
&\sum_{\substack{S\subseteq T,\\ |S|>1}}\mu^{|S|}\cdot\widehat{f_1}\left(S\right)\ldots\widehat{f_p}\left(S\right) \\
&\leq\mu^2\sum_{\substack{S\subseteq T,\\ |S|>1}}\widehat{f_1}\left(S\right)\ldots\widehat{f_p}\left(S\right) && \text{(Due to Lemma 6.1 is w.l.o.g. } \mu\in(\text{-}1,1)\text{)} \\
&\leq\mu^2\sqrt{\sum_{\substack{S\subseteq T,\\ |S|>1}}\widehat{f_1}\left(S\right)^2}\cdot\sqrt{\sum_{\substack{S\subseteq T,\\ |S|>1}}\widehat{f_2}\left(S\right)^2\ldots\widehat{f_p}\left(S\right)^2} && \text{(Chauchy-Schwarz)} \\
&\leq\mu^2\cdot\sqrt{\sum_{\substack{S\subseteq T,\\ |S|>1}}\widehat{f_2}\left(S\right)^2\ldots\widehat{f_p}\left(S\right)^2} && \text{(Theorem 2.4)}
\end{aligned}$$

The repeated application of the Cauchy-Schwarz inequality and Theorem 2.4 leads to

$$\sum_{\substack{S\subseteq T,\\ |S|>1}}\mu^{|S|}\cdot\widehat{f_1}\left(S\right)\ldots\widehat{f_p}\left(S\right)\leq\mu^2. \tag{6.2.7}$$

For reasons of space in the following the sum $\sum_S \mu^{|S|} \cdot \widehat{f_1}(S) \cdot \ldots \cdot \widehat{f_p}(S)$ is replaced by $\sum_S \mu^{|S|} \cdot \widehat{S}$. Theorem 6.3 approximates the sum of the products of the degree-one Fourier coefficients with an additive $\pm\mu$. Recalling sum (6.2.6) which is approximated with an additive $\pm\mu^2$ then the following estimation can be done

$$\sum_{\substack{S \subseteq T,\\ |S|>0}} \mu^{|S|} \cdot \widehat{S} - \mu^2 \leq \sum_{\substack{S \subseteq T,\\ |S|>0}} \mu^{|S|} \cdot \widehat{S} \leq \sum_{\substack{S \subseteq T,\\ |S|>0}} \mu^{|S|} \cdot \widehat{S} + \mu^2.$$

From (6.2.7) it follows

$$\begin{aligned}
&\sum_{\substack{S \subseteq T,\\ |S|=1}} \mu \cdot \widehat{S} \leq \sum_{\substack{S \subseteq T,\\ |S|=1}} \mu \cdot \widehat{S} + \mu^2 \leq \sum_{\substack{S \subseteq T,\\ |S|=1}} \mu \cdot \widehat{S} + 2\mu^2 \\
\Leftrightarrow &\sum_{\substack{S \subseteq T,\\ |S|=1}} \mu \cdot \widehat{S} - \mu^2 \leq \sum_{\substack{S \subseteq T,\\ |S|=1}} \mu \cdot \widehat{S} \leq \sum_{\substack{S \subseteq T,\\ |S|=1}} \mu \cdot \widehat{S} + \mu^2 \\
\Leftrightarrow &\sum_{i \in T} \widehat{f_1}(i) \ldots \widehat{f_p}(i) - \mu \leq \sum_{i \in T} \widehat{f_1}(i) \ldots \widehat{f_p}(i) \leq \sum_{i \in T} \widehat{f_1}(i) \ldots \widehat{f_p}(i) + \mu.
\end{aligned}$$

□

With Theorem 6.3 and if $f_1 = \ldots = f_i = \ldots = f_p$ it is possible to approximate the sum of powers of Fourier coefficients.

Corollary 6.1. *([Mat+10, Corollary 16.]) For a fixed $p \geq 2$ and $T \subseteq [n]$. Suppose black-box access to $f : \{-1,1\}^n \to \{-1,1\}$, then it is possible to estimate*

$$\sum_{i \in T} \widehat{f}(i)^p$$

to an additive $\pm\mu$, with confidence $1-\delta$, using $O\left(p \cdot \log(1/\delta)/\mu^4\right)$ queries.

If choosing $p = 2$ in Corollary 6.1, it is possible to approximate the degree-one weight of a Boolean function with it.

Compared with the approximation given by Theorem 6.2 the estimate in Corollary 6.1 is independent of the input dimension of the Boolean function and for certain absolute errors Corollary 6.1 needs less queries to guarantee with confidence $1-\delta$ that the absolute error does not exceed a certain bound. Due to the independence of the number of input bits n of the Boolean function, Corollary 6.1 does need much fewer queries than Theorem 6.2 for certain absolute errors and n.

Figure 6.2.1 shows that for adequately small absolute error Theorem 6.2 needs fewer queries than Corollary 6.1. Last but not least, Figure 6.2.1 shows that for low absolute errors both variants of degree-one weight approximation exceeding the number of possible different inputs and in this case both approximations does not give a guarantee better than brute force computation using 2^n queries according to Theorem 2.5.

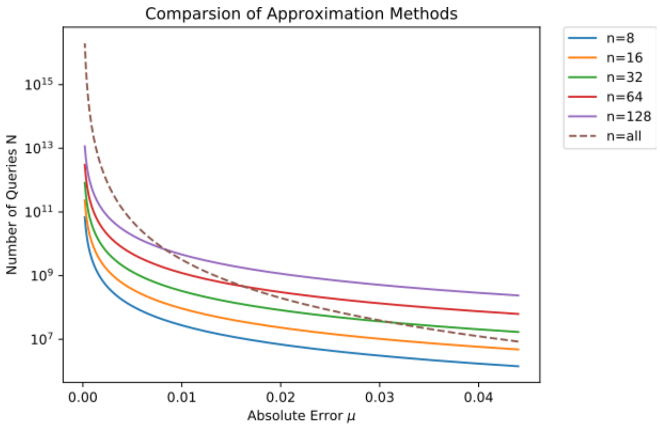

Figure 6.2.1: A comparison between the approximation with Theorem 6.2 which depends on the number of input bits n and Corollary 6.1 which is independent of n. The lines displaying the number of queries to guarantee with confidence $0.99 = 1 - \delta$ an absolute error μ. The solid lines represent Theorem 6.2 the dashed line displays Corollary 6.1.

The Algorithm 6.1 is a explicit form of the proceeding in the proof of Theorem 6.3 and should display how to empirical estimate the sum of powers of degree-one Fourier coefficients.

Algorithm 6.1 Estimation of sum of powers of degree-one Fourier coefficients through Corollary 6.1

Input

- $p \geq 2$
- Boolean function $f : \{-1,1\}^n \to \{-1,1\}$
- Index set $T \subseteq \{1,\ldots,n\}$
- Absolute error μ

Execution

1. $N = \left\lceil 2 \cdot \left(\frac{\log\left(\frac{1}{\delta}\right)+\log(2)}{\mu^4} \right) \right\rceil$
2. Generate $\mathbf{X^1},\ldots,\mathbf{X^p}$ where $\mathbf{X^i} = \left(\mathbf{x^1},\ldots,\mathbf{x^N}\right)$ and $\mathbf{x^i} \sim \{-1,1\}^n$
3. Generate $\mathbf{Y} = \left(\mathbf{y^1},\ldots,\mathbf{y^N}\right)$ with $\mathbf{y^i} \in \{-1,1\}^n$ where
$$\Pr\left[\mathbf{y}_i^{\mathbf{i}} = 1\right] = \begin{cases} \frac{1}{2} & \text{if } i \notin T, \\ \frac{1}{2} + \frac{\mu}{2} & \text{if } i \in T \end{cases}$$
4. $\text{est}_{\emptyset} = \frac{1}{N}\sum_{j=1}^{N} f\left(\mathbf{X}_j^{\mathbf{1}}\right) \cdot \ldots \cdot f\left(\mathbf{X}_j^{\mathbf{p}}\right)$
5. $\text{est}_{S\subseteq T} = \frac{1}{N}\sum_{j=1}^{N} f\left(\mathbf{X}_j^{\mathbf{1}}\right) \cdot \ldots \cdot f\left(\mathbf{X}_j^{\mathbf{p}-1}\right) \cdot f\left(\mathbf{X}_j^{\mathbf{1}} \odot \ldots \odot \mathbf{X}_j^{\mathbf{p-1}} \odot \mathbf{Y}_j\right)$
6. $\text{est}_{S,|S|>0} = \text{est}_{S\subseteq T} - \text{est}_{\emptyset}$
7. $\text{est}_{S,|S|=1} = \frac{\text{est}_{S,|S|>0} - \mu^2}{\mu}$

Output

$$\text{est}_{S,|S|=1} = \sum_{i \in T} \widehat{f}(i)^p$$

BIASED DISTANCE TO AN UNBIASED LINEAR THRESHOLD FUNCTION

This section shows that it possible to approximate the distance of a Boolean function to an unbiased LTF which is described in Theorem 4.3. The first theorem of this section shows how the Theorem 4.3 relates to an approximated degree-one weight a Boolean function.

Theorem 6.4. *Let $f : \{-1,1\}^n \to \{-1,1\}$ be an $(\varepsilon,1)$-regular Boolean function and $\widetilde{\mathbf{W}}^1[f]$ a degree-one weight approximation with an absolute error μ. If f fulfills both hypotheses of Theorem 4.3, then the approximated degree-one weight lies in the interval*

$$\frac{2}{\pi} - \varepsilon - \mu \leq \widetilde{\mathbf{W}}^1[f] \leq \frac{2}{\pi} + O(\varepsilon) + \mu. \tag{6.3.1}$$

Proof. (Theorem 6.4)
If the conditions are met the degree-one weight $\mathbf{W}^1[f]$ is, according to Theorem 4.3, bounded by

$$\frac{2}{\pi} - \varepsilon \leq \mathbf{W}^1[f] \leq \frac{2}{\pi} + O(\varepsilon).$$

From Definition 6.1 it follows that the estimated degree-one weight $\widetilde{\mathbf{W}}^1[f]$ lies between

$$\mathbf{W}^1[f] - \mu \leq \widetilde{\mathbf{W}}^1 \leq \mathbf{W}^1[f] + \mu.$$

It is appropriate to do two worst case estimations to show that $\widetilde{\mathbf{W}}^1[f]$ fulfills (6.3.1).
Case $\widetilde{\mathbf{W}}^1[f] = \mathbf{W}^1[f] + \mu$:
From Theorem 4.3 it follows

$$\begin{aligned}
&\frac{2}{\pi} - \varepsilon \leq \mathbf{W}^1[f] \leq \frac{2}{\pi} + O(\varepsilon)\\
\Leftrightarrow\ &\frac{2}{\pi} - \varepsilon + \mu \leq \mathbf{W}^1[f] + \mu \leq \frac{2}{\pi} + O(\varepsilon) + \mu\\
\Leftrightarrow\ &\frac{2}{\pi} - \varepsilon + \mu \leq \widetilde{\mathbf{W}}^1[f] \leq \frac{2}{\pi} + O(\varepsilon) + \mu\\
\Rightarrow\ &\frac{2}{\pi} - \varepsilon - \mu \leq \widetilde{\mathbf{W}}^1[f] \leq \frac{2}{\pi} + O(\varepsilon) + \mu.
\end{aligned}$$

Case $\widetilde{\mathbf{W}}^1[f] = \mathbf{W}^1[f] - \mu$:
From Theorem 4.3 it follows

$$\begin{aligned}
&\frac{2}{\pi} - \varepsilon \leq \mathbf{W}^1[f] \leq \frac{2}{\pi} + O(\varepsilon)\\
\Leftrightarrow\ &\frac{2}{\pi} - \varepsilon - \mu \leq \mathbf{W}^1[f] - \mu \leq \frac{2}{\pi} + O(\varepsilon) - \mu\\
\Leftrightarrow\ &\frac{2}{\pi} - \varepsilon - \mu \leq \widetilde{\mathbf{W}}^1[f] \leq \frac{2}{\pi} + O(\varepsilon) - \mu\\
\Rightarrow\ &\frac{2}{\pi} - \varepsilon - \mu \leq \widetilde{\mathbf{W}}^1[f] \leq \frac{2}{\pi} + O(\varepsilon) + \mu.
\end{aligned}$$

The worst case estimations shows that (6.3.1) applies for the smallest and largest case of $\widetilde{\mathbf{W}}^1[f]$. Thus (6.3.1) applies also for all

$$\widetilde{\mathbf{W}}^1[f] \in \left[\mathbf{W}^1[f] - \mu, \mathbf{W}^1[f] + \mu\right].$$

□

From Theorem 6.4 it follows that the absolute error of the degree-one weight approximation of a Boolean function influences the interval from Theorem 4.3. Combining the distance hypotheses of Theorem 4.3 with Theorem 6.4 it is possible to approximate the distance bound introduced by Theorem 4.3.

Theorem 6.5. *Let $f : \{-1,1\}^n \to \{-1,1\}$ be an $(\varepsilon,1)$-regular Boolean function, c a universal constant and $\widetilde{\mathbf{W}}^1[f]$ a degree-one weight approximation with an absolute error μ, such that $\widetilde{\mathbf{W}}^1[f] - \mu \leq 2/\pi$. If f fulfills both hypotheses of Theorem 4.3, then*

$$0 \leq 3\sqrt{c+2}\sqrt{\frac{2}{\pi} - \widetilde{\mathbf{W}}^1[f] - \mu} \leq O\left(\sqrt{\varepsilon}\right)$$

describes a distance approximation to the linear threshold function $\operatorname{sgn}\left(f^{=1}\right)$.

Proof. Since f is an $(\varepsilon,1)$-regular Boolean function and f fulfills both hypotheses of Theorem 4.3, all requirements of Theorem 6.4 are fulfilled. From this it follows

$$\begin{aligned} \frac{2}{\pi} - \varepsilon - \mu &\leq \widetilde{\mathbf{W}}^1[f] \\ -\varepsilon &\leq \widetilde{\mathbf{W}}^1[f] - \frac{2}{\pi} + \mu \\ \varepsilon &\geq \frac{2}{\pi} - \widetilde{\mathbf{W}}^1[f] - \mu. \end{aligned} \tag{6.3.2}$$

If the conditions are met and utilizing the upper bound (4.1.21) for the distance estimation to an unbiased LTF from Theorem 4.3, then from (6.3.2) and $c = 0.5129$ (Korolev and Shevtsova [KS10]) it follows

$$\begin{aligned} 3\sqrt{c+2}\sqrt{\frac{2}{\pi} - \widetilde{\mathbf{W}}^1[f] - \mu} &\leq 3\sqrt{c+2}\sqrt{\varepsilon} \\ \Leftrightarrow 3\sqrt{c+2}\sqrt{\frac{2}{\pi} - \widetilde{\mathbf{W}}^1[f] - \mu} &\leq O\left(\sqrt{\varepsilon}\right). \end{aligned} \tag{6.3.3}$$

From $\widetilde{\mathbf{W}}^1[f] - \mu \leq 2/\pi$ and (6.3.3) it follows

$$0 \leq 3\sqrt{c+2}\sqrt{\frac{2}{\pi} - \widetilde{\mathbf{W}}^1[f] - \mu} \leq O\left(\sqrt{\varepsilon}\right).$$

□

Theorem 6.5 approximates the distance of a Boolean function to an unbiased LTF, which is introduced by Theorem 4.3. Further, when using Theorem 6.5, the distance $\operatorname{dist}\left(f, f^{=1}\right)$ according to Theorem 4.3 can be under- or overestimated. In the case, that Theorem 6.5 underestimates $\operatorname{dist}\left(f, f^{=1}\right)$ a lower bound for the distance is given which is a desirable circumstance when testing a functions closeness to an unbiased LTF. The other case, when Theorem 6.5 provides a lower bound for the distance estimation to an unbiased LTF, it can give a sharper bound for the distance to an unbiased LTF. Supposing the absolute approximation error μ and the regularity parameter ε are sufficiently small and $\widetilde{\mathbf{W}}^{1}[f] - \mu \leq 2/\pi$ then Theorem 6.5 can give evidence that an $(\varepsilon, 1)$-regular Boolean function f is close to an unbiased LTF. It is advisable to approximate the Fourier coefficients of an $(\varepsilon, 1)$-regular Boolean function when using Theorem 6.5 to be more confident about ε and therefore the distance bound. Furthermore, it is unclear how well Theorem 6.5 performs in practice. For the benefit of the empirical surveys of the degree-one weight approximation methods, a more in-depth analysis of Theorem 6.5 is rejected.

7 SURVEY OF DEGREE-ONE WEIGHT APPROXIMATION

Section 6 presents two ways of estimating the degree-one weight of a Boolean function with a probabilistic assurance to have a particular absolute error. Due to the use of the Hoeffding inequality 5.1, both attempts to approximate the degree-one weights need a lot of queries which is unpractical in some situations. For example, if the evaluation of a PUF is slow or the number of challenge-response pairs is limited by the recycling of data from another survey. This section presents two surveys with the intention to show that in practice a much lower number of queries is needed to get an indication of the degree-one weight for the considered types of functions. For this purpose, a survey design is used that can be applied to both methods.

The core of the surveys is a controlled experiment which should check the hypothesis

Claim 7.1. "The empirical number of queries needed to approximate the degree-one weight of a Boolean function f with a specific absolute error using method M is, in fact, lower than the theoretical number of queries needed for the estimation of the degree-one weight with the same absolute error and a certain confidence.".

The idea to check the Claim 7.1 is to approximate the degree-one weights for certain functions and a certain method M for different numbers of inputs. Based on this data, it is possible to recognize whether fewer queries are required for this particular function f than suggested in theory.

EXPERIMENT STRUCTURE

This chapter defines a controlled experiment which generates pairs of degree-one weight approximations and the number of queries used.

The independent variable is the number of queries needed to approximate the degree-one weight. Extraneous variables are the Boolean function f and the estimation method M.

In order to create clarity, Algorithm 7.1 describes which data points are generated.

Conducting the Experiment

To estimate degree-one weights with the in Section 6 described methods it is necessary to generate randomly distributed inputs.

Algorithm 7.1 Controlled experiment degree-one weight approximation

```
Extraneous variables
    • Approximation method M
    • Boolean function $f : \{-1,1\}^n \to \{-1,1\}$
Independent variable
    • Number of queries $N$
Execution
If M works according to Corollary 6.1:
    1. For each $i \in \{1, \dots, N\}$ calculate
       $\widetilde{\mathbf{W}}^1 [f]_i = \mathrm{M}(f)$ with $3 \cdot i$ queries
    2. result = $\left(\widetilde{\mathbf{W}}^1 [f]_1, 3\right), \dots, \left(\widetilde{\mathbf{W}}^1 [f]_N, N\right)$
else:
    1. For each $i \in \{1, \dots, N\}$ calculate
       $\widetilde{\mathbf{W}}^1 [f]_i = \mathrm{M}(f)$ with $i$ queries
    2. result = $\left(\widetilde{\mathbf{W}}^1 [f]_1, 1\right), \dots, \left(\widetilde{\mathbf{W}}^1 [f]_N, N\right)$
Output
    result
```

Besides, the evaluation of a function takes a certain amount of time. Define N as the number of inputs needed to agree with Claim 7.1. If every estimation of the degree-one weight from one to N queries needs a new set of randomly distributed inputs f evaluates $O(N^2)$ times. To decrease the experiment runtime, it is useful to use a set of N randomly distributed inputs which results in a set of responses such that f only evaluates $O(N)$ times. To enhance the validity of the study, 1000 experiments were carried out which results in 1000 different independently distributed inputs sets used.

Further, the experiments for a certain function f and method M are executed for 64 and 128 input bits, which are state of the art scales for analyzing Strong PUFs (for example see Ruehrmair et al. [RS14]).

The general validity of this study is limited by the specific characteristics of the Boolean functions used. Furthermore, only functions for which the degree-one weight is known are used for the experiments. The functions checked are dictator, bent and unbiased LTFs. Inner product modulo two from Definition 2.9 is used as a bent function. The dictator function according to Definition 2.10 has the dictator bit for all dictators at input bit two. Last but not least, the weights of the unbiased LTFs according to Definition 2.7 are normally distributed with 0 mean and standard deviation 1.

As shown in Algorithm 6.1, the method for approximation of Matulef et al. requires the absolute error as a parameter. This error is considered here as part of the extraneous variable M and is set to $\mu = 0.05$.

Used Tools

The experiments were implemented and carried out using Python 3.5.4. The PUF simulation and learning framework pypuf[1] was used here. The framework is developed by the ID-Management workgroup[2] from the Freie Universität Berlin and offers a fast and clear implementation of LTFs. Furthermore, 64-bit precision was used for the calculations of floating point numbers. The analysis, on the other hand, was performed with 32-bit floating point numbers due to insufficient memory capacity. In addition, the framework offers an interface for parallel execution of experiments, which writes the results to logfiles. To be able to use this feature, the experiments have been adapted to this interface. After carrying out the experiments, the data was written to a rudimentary SQLite[3] database and then further processed with Python. The approximation of the degree-one weights is based on randomly selected inputs. In addition, the weights of the examined LTFs are normally distributed. In order to ensure the reproducibility of the results, the RandomState[4] class with fixed seeds was used. The RandomState class is based on the pseudo-random number generator (PRNG) Mersenne Twister by M. Matsumoto and T. Nishimura [MN98]. The seed for the PRNGs are combinations of numbers of the mathematical constant π and other certain numbers, which can be looked up in the GitHub repository of pypuf[5].

Last but not least, the experiments were carried out on a Ubuntu Mate 17 system with an Intel Core i7-3770K and 10GB memory.

RESULTS EMPIRICAL DEGREE-ONE WEIGHT APPROXIMATION

This section presents the results and analysis of controlled experiments for the empirical degree-one weight approximation. There were 1000 experiments for the number of queries from 1 to 10^5. Figure

[1] https://github.com/nils-wisiol/pypuf (accessed March 5, 2018)

[2] http://www.mi.fu-berlin.de/inf/groups/ag-idm/index.html (accessed March 25, 2018)

[3] https://sqlite.org/index.html (accessed March 5, 2018)

[4] https://docs.scipy.org/doc/numpy-1.13.0/reference/generated/numpy.random.RandomState.html#numpy.random.RandomState (accessed March 5, 2018)

[5] https://github.com/nils-wisiol/pypuf/tree/c02749783d08b2d8bd1314c5f4cde091054bf35f (accessed March 11, 2018)

7.2.1 shows the common statistical characteristic numbers for the different Boolean functions. The reference degree-one weights, shown as a black horizontal line, can be found in Section 2.1. As a degree-one weight for the LTFs, $^2/_\pi$ is assumed to be ideal.

Here, the values for dictator and bent-function, look similar, with the median and arithmetic mean not being optically different and 50 percent of the degree-one weights distributed close to the mean value. In the case of LTFs, the median and the arithmetic mean also hardly differ from each other. In contrast, the dispersion around the mean value is much wider in comparison to the dictator and bent-function. Where in the dictator and bent-function an interval of width 0.025 is sufficient to show the statistical key figures, in the LTFs an interval with a width of 0.09 is needed. The higher variance of the degree-one weight approximations might come from a more complex degree-one Fourier coefficient distribution caused by normally distributed weights. However, the estimated degree-one weights of the LTFs are also distributed close to the area around the mean value. Furthermore, all functions show that the approximate degree-one weights, as described in the theory, probably have a smaller absolute error with increasing number of queries.

In addition, the number of inputs required for the experiments to maintain a certain absolute error with a high probability for $n = 64$ and $n = 128$ is much lower than the theoretical bound. To confirm this statement, the limits of queries are entered in Table 7.1, from which the absolute error for all approximate degree-one weights was less than 0.0011. Table 7.1 shows the theoretical numbers of queries needed to guarantee with confidence $0.99 = 1 - \delta$ particular absolute errors in contrast to the empirically calculated limits. To improve readability, the theoretical values calculated for arbitrary functions have been rounded up. The distance between the empirical results and the theoretical boundary is so high that no false statement should be made by the rounding. Recalling the experiment which empirically calculates degree-one weight with a certain number of queries. The bounds for the approximation in Table 7.1 specify that from this number of queries all approximated degree-one weights calculated with equal or more inputs fulfill a particular absolute error.

	Absolute Error				
Function	0.0011	0.0012	0.01	0.025	0.05
Theoretical n=64	$\approx 10^{11}$	$\approx 9 \cdot 10^{10}$	$\approx 2 \cdot 10^{9}$	$\approx 2 \cdot 10^{8}$	$\approx 5 \cdot 10^{7}$
Dictator n=64	-	90747	10783	4309	2066
IP mod 2 n=64	99516	92239	10461	4783	2058
LTF n=64	-	-	-	34290	7018
Theoretical n=128	$\approx 4 \cdot 10^{11}$	$\approx 4 \cdot 10^{11}$	$\approx 5 \cdot 10^{9}$	$\approx 8 \cdot 10^{8}$	$\approx 2 \cdot 10^{8}$
Dictator n=128	-	-	19499	8459	3584
IP mod 2 n=128	-	-	18510	7153	3660
LTF n=128	-	-	-	33356	9401

Table 7.1: The boundary of queries for the empirical degree-one weight approximation experiments, such that a certain absolute error was maintained. Further, the hyphen indicates that one or more experiments exceeded the absolute error.

The results show that for the considered functions and specific absolute errors, far fewer queries were required to maintain the absolute errors for a certain number of entries. Especially interesting is that the required number of inputs for the LTFs with $\mu = 0.025$ are much higher than for the other functions. Table 7.1 also shows that the $n = 64$ bit LTF requires more inputs than the $n = 128$. This indicates that the number of inputs for the LTF and an absolute error $\mu = 0.025$ is not accurate. However, the approximations seem to converge to the actual degree-one weight. Therefore, the Claim 7.1 for this particular setup and absolute errors $\mu \geq 0.025$ is true.

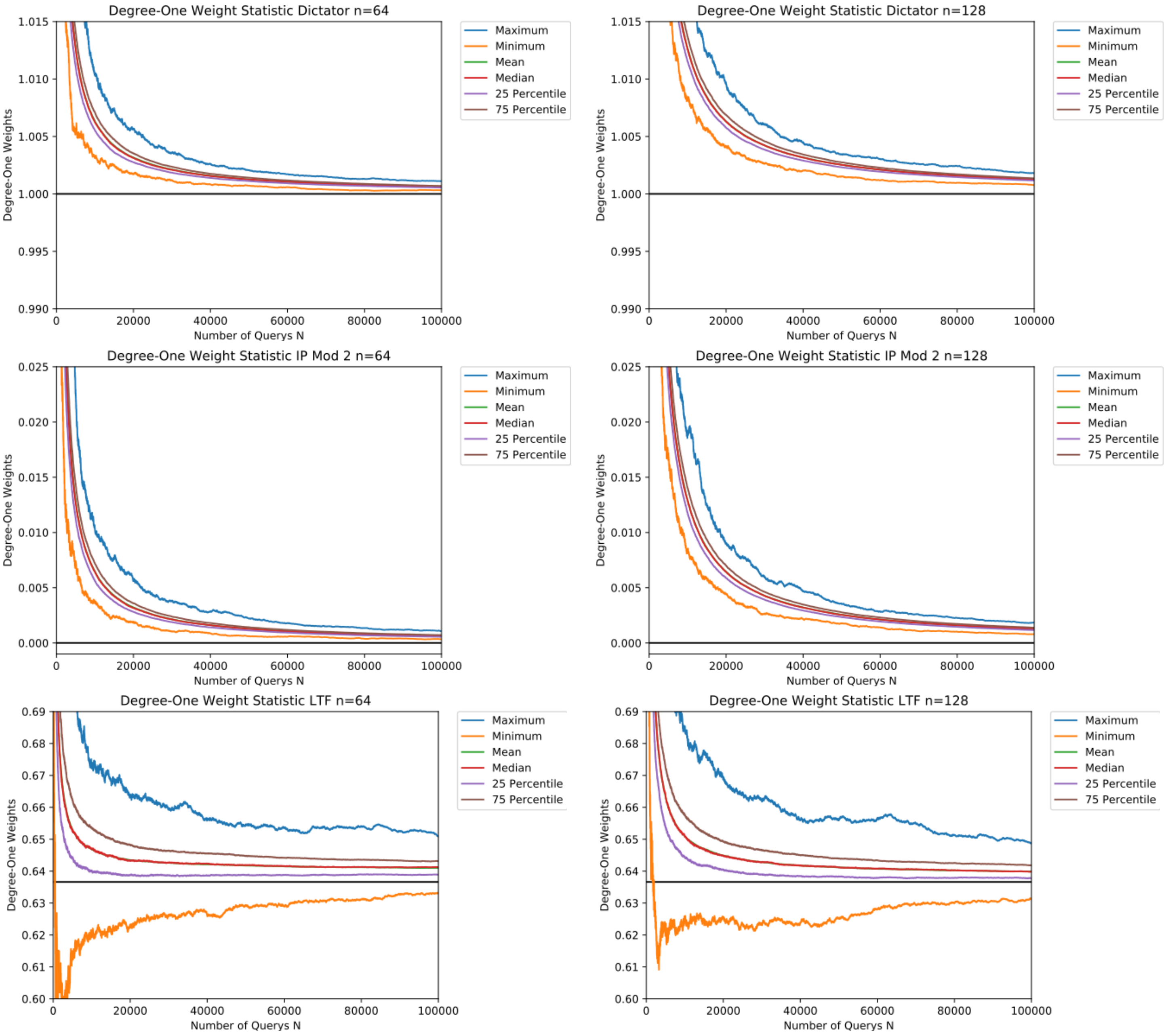

Figure 7.2.1: Statistical results of 1000 controlled experiments for the empirical degree-one weight approximation with 1 to 10^5 queries. The black horizontal line marks the ideal degree-one weight.

results approximate sums of powers of fourier coefficients

This section presents the results and analysis of the 1000 controlled experiments for the approximation of the degree-one weights using Algorithm 6.1, which approximates the sum of the powers of degree-one Fourier coefficients in respect to an absolute error. Here, the absolute error is fixed at $\mu = 0.05$ for all experiments.

The graphics in Figure 7.3.1 show the usual statistical indicators of the approximate degree-one weights such as minimum, maximum, median, arithmetic mean, 25 percentile and 75 percentile. In all functions, the median cannot be distinguished optically from the arithmetic mean.

It is noteworthy that for all considered functions the minimum and maximum, even with $12 \cdot 10^5$ queries, differ with an absolute error of circa 0.2 from the actual degree-one weight. It is also striking that the average approximate degree-one weight seems to converge against the actual degree-weight minus the absolute error $\mu = 0.05$. However, if you look at the theoretical bound in Theorem 6.3 again, this is not yet a sign of corrupted data, caused by an insufficient implementation or analysis, because an error of less than or equal to μ is theoretically valid. In order to verify Claim 7.1, the limits are listed in Table 7.2 such that with confidence $0.99 = 1 - \delta$ a certain absolute error was observed in the approximation.

	Absolute Error					
Function	0.05	0.1	0.15	0.25	0.5	0.75
Theoretical	≈ 5090000	≈ 318000	≈ 62800	≈ 8140	≈ 509	≈ 101
Dictator n=64	-	-	-	813439	153838	46264
Dictator n=128	-	-	-	1029397	132568	70903
IP mod 2 n=64	-	-	-	535513	137695	62260
IP mod 2 n=128	-	-	-	680395	115135	39814
LTF n=64	-	-	-	932482	179497	40324
LTF n=128	-	-	-	857044	127132	61234

Table 7.2: The boundary of queries for the sum of the powers of degree-one Fourier coefficients approximation experiments, such that a certain absolute error was maintained. To ensure that the theoretical number of queries is maintained in order to guarantee an absolute error with confidence $0.99 = 1 - \delta$, a round to the next higher whole number was performed. Further, the hyphen indicates that one or more experiments exceeded the absolute error.

The statistical overview from Figure 7.3.1 suggests that such a statement for the experiments carried out is not durable for small absolute errors.

The bounds for the approximation in Table 7.2 specify that from this number of queries all approximated degree-one weights calculated with equal or more inputs fulfill the absolute error. To be clear if only one of the 1000 experiment cannot hold the absolute error for a bound of inputs this bound is rejected. Due to the high variance of this method, the applied method to determine a bound of inputs is strict but necessary to be able to be comparable to the first survey.

Because of the high variance of the approximation in the empirical results, it is not possible to define the limits of number of queries for absolute errors smaller than or equal to 0.15. The next observed absolute error is 0.25, so it is unclear whether 0.25 is the smallest absolute error for Claim 7.1. Apart from this, the analysis of the data indicates that Claim 7.1 does not even apply to an absolute error of more than 0.25. Since sufficiently large and small degree-one weights have been calculated as examples, Claim 7.1 for absolute errors of less than or equal to 0.15 can be rejected with certainty.

At this point, it should be mentioned that the results in Table 7.2 are based on Algorithm 6.1 with $\mu = 0.05$ as parameter. Since the desired absolute error μ influences the random variables $\mathbf{Y} = \left(\mathbf{y^1}, \ldots, \mathbf{y^N}\right)$, for other absolute errors the Algorithm 6.1 could have done better. Furthermore, the number of random inputs used for the experiments was only about 1/5 of the bound, proven in theory. Whether claim 7.1 would also be rejected for more random inputs remains open.

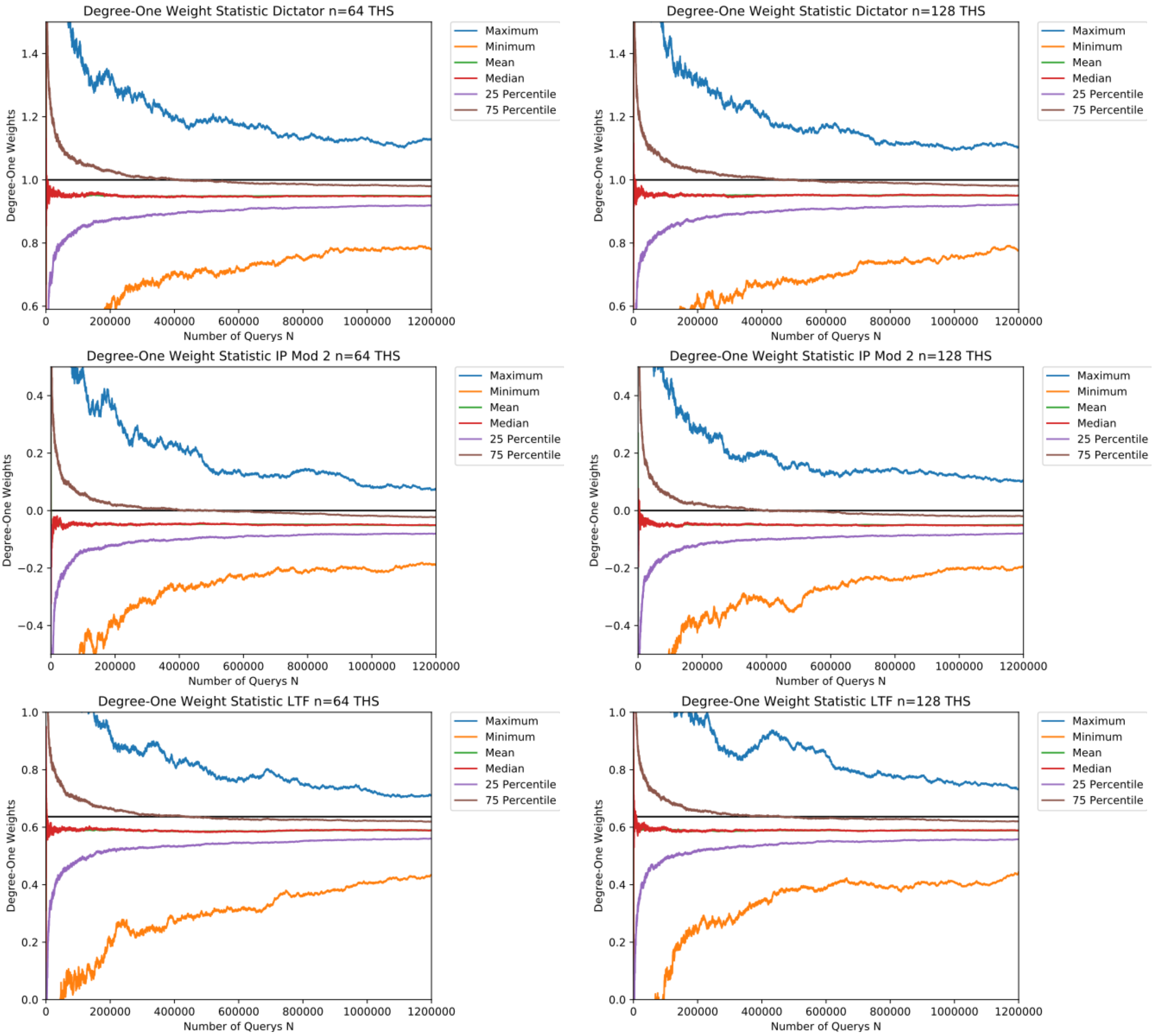

Figure 7.3.1: Statistical results of 1000 controlled experiments for the degree-one weight approximation through Algorithm 6.1 with $\mu = 0.05$ and 3 to $12 \cdot 10^5$ queries. The black horizontal line marks the ideal degree-one weight.

8 CONCLUSION

This thesis proposes the degree-one weight of Boolean functions as a metric, which serves as an indicator for the predictability of a PUF. The proposed metric can only reliably recognize certain cases of predictability as explained in Chapter 3.

An important point here is the distance of a Boolean function to an unbiased LTF, which can be determined using Theorem 4.3.

For the practical application of the degree-one weight it is essential to know how the degree-one weight of a Boolean function can be calculated. With this in mind, two probabilistic approaches were presented.

The empirical degree-one weight approximation from Section 6.1 utilizes approximated degree-one Fourier coefficients and then calculates the degree-one weight with them. The advantage of the utilized empirical degree-one Fourier coefficient approximation method compared to the Low-Degree algorithm is that no specific concentration of the degree-one weight is required to be applicable. The empirical degree-one weight approximation theoretical requires a lot of randomly selected samples to achieve a sufficiently small absolute error, which strongly depends on the number of input bits of the Boolean function. Therefore, another method for approximating the degree-one weight is considered.

The advantage of the degree-one weight approximation according to Matulef et al. described in Algorithm 6.1 over the empirical method is that it is independent of the input length of the Boolean function. It is shown in Figure 6.2.1 that the theoretical limit of randomly selected samples with respect to practical feasibility, is for certain absolute error much smaller than that of the empirical approximation.

Subsequently, Section 6.3 shows how the absolute error of a degree-one weight approximation affects the distance to an unbiased LTF presented in Theorem 4.3. With Theorem 6.5 it is possible to approximate the distance to an unbiased LTF.

Finally, two studies are carried out which examine the approximation behavior of the empirical and the method according to Matulef et al. for certain Boolean functions.

It becomes clear that for the empirical degree-one weight approximation, with much less randomly selected samples than the theoretical limit indicates, particular absolute errors can be observed.

Due to the high variance of the results in the worst case consideration for the studied absolute errors, the method of Matulef et al. has far exceeded the theoretical limit of the randomly selected samples.

Further, this thesis neglects the analysis of noisy inputs for the approximation methods. Thus the performance of the shown methods with real noised PUF instances is unclear.

In conclusion, if a provable approximated degree-one weight for a Boolean function with a large number of input bits is required, the procedure of Matulef et al. should be used.

Due to the simple implementation and based on the study carried out in this thesis, for the internal design process of PUFs, the empirical approximation can be used for a first impression of the design concerning the degree-one weight. Furthermore, the empirical approach uses the estimated degree-one Fourier coefficients, such that it is possible to detect possible influence imbalances of the individual input bits.

BIBLIOGRAPHY

[Aar16] Maurice Aarts. “Hardware Attacks Tamper Resistance, Tamper Response and Tamper Evidence.” In: *Date of retrieval* 23 (2016) (cit. on p. 1).

[And10] Jason H Anderson. “A PUF Design for Secure FPGA-based Embedded Systems.” In: *Proceedings of the 2010 Asia and South Pacific Design Automation Conference*. ASPDAC ’10. Piscataway, NJ, USA: IEEE Press, 2010, pp. 1–6 (cit. on p. 2).

[Arm+16] Frederik Armknecht, Daisuke Moriyama, Ahmad-Reza Sadeghi, and Moti Yung. “Towards a Unified Security Model for Physically Unclonable Functions.” In: *Topics in Cryptology - CT-RSA 2016*. Springer International Publishing, 2016, pp. 271–287 (cit. on pp. 14, 15).

[Bec15] Georg T Becker. “The Gap Between Promise and Reality: On the Insecurity of XOR Arbiter PUFs.” In: *Cryptographic Hardware and Embedded Systems – CHES 2015*. Springer Berlin Heidelberg, 2015, pp. 535–555 (cit. on p. 15).

[Ber41] Andrew C Berry. “The Accuracy of the Gaussian Approximation to the Sum of Independent Variates.” In: *Trans. Amer. Math. Soc.* 49.1 (1941), pp. 122–136 (cit. on p. 23).

[Bis06] Christopher M Bishop. *Pattern recognition and machine learning*. springer, 2006 (cit. on p. 22).

[Che+11] Q Chen, G Csaba, P Lugli, U Schlichtmann, and U Rührmair. “The Bistable Ring PUF: A new architecture for strong Physical Unclonable Functions.” In: *2011 IEEE International Symposium on Hardware-Oriented Security and Trust*. June 2011, pp. 134–141 (cit. on pp. 2, 16, 21).

[Che+12] Q Chen, G Csaba, P Lugli, U Schlichtmann, and U Rührmair. “Characterization of the bistable ring PUF.” In: *2012 Design, Automation Test in Europe Conference Exhibition (DATE)*. Mar. 2012, pp. 1459–1462 (cit. on p. 16).

[CV95] Corinna Cortes and Vladimir Vapnik. “Support-vector networks.” In: *Mach. Learn.* 20.3 (Sept. 1995), pp. 273–297 (cit. on p. 22).

[DR02] Joan Daemen and Vincent Rijmen. *The Design of Rijndael: AES - The Advanced Encryption Standard*. en. Springer-Verlag, 2002 (cit. on p. 1).

[Gas+02] Blaise Gassend, Dwaine Clarke, Marten van Dijk, and Srinivas Devadas. “Silicon Physical Random Functions.” In: *Proceedings of the 9th ACM Conference on Computer and Communications Security*. CCS ’02. New York, NY, USA: ACM, 2002, pp. 148–160 (cit. on p. 1).

[Gas+04] Blaise Gassend, Daihyun Lim, Dwaine Clarke, Marten Van Dijk, and Srinivas Devadas. “Identification and authentication of integrated circuits.” In: *Concurr. Comput.* 16.11 (2004), pp. 1077–1098 (cit. on p. 1).

[Gua+07] Jorge Guajardo, Sandeep S Kumar, Geert-Jan Schrijen, and Pim Tuyls. “FPGA Intrinsic PUFs and Their Use for IP Protection.” In: *Cryptographic Hardware and Embedded Systems - CHES 2007*. Springer Berlin Heidelberg, 2007, pp. 63–80 (cit. on p. 2).

[HHS17] Robert Hesselbarth, Johann Heyszl, and Georg Sigl. “Fast and reliable PUF response evaluation from unsettled bistable rings.” In: *Microprocess. Microsyst.* 52 (July 2017), pp. 325–332 (cit. on p. 14).

[Hoe63] Wassily Hoeffding. “Probability Inequalities for Sums of Bounded Random Variables.” In: *J. Am. Stat. Assoc.* 58.301 (Mar. 1963), pp. 13–30 (cit. on pp. 18, 36).

[Kho+07] Subhash Khot, Guy Kindler, Elchanan Mossel, and Ryan O’Donnell. “Optimal inapproximability results for MAX-CUT and other 2-variable CSPs?” In: *SIAM J. Comput.* 37.1 (2007), pp. 319–357 (cit. on pp. 23, 26).

[KS10] V Korolev and I Shevtsova. “On the Upper Bound for the Absolute Constant in the Berry–Esseen Inequality.” In: *Theory Probab. Appl.* 54.4 (Jan. 2010), pp. 638–658 (cit. on pp. 26, 31, 55).

[LMN89] N Linial, Y Mansour, and N Nisan. “Constant depth circuits, Fourier transform, and learnability.” In: *30th Annual Symposium on Foundations of Computer Science*. Oct. 1989, pp. 574–579 (cit. on p. 33).

[LMN93] Nathan Linial, Yishay Mansour, and Noam Nisan. “Constant Depth Circuits, Fourier Transform, and Learnability.” In: *J. ACM* 40.3 (July 1993), pp. 607–620 (cit. on pp. 33, 34).

[ML14] R Maes and V van der Leest. “Countering the effects of silicon aging on SRAM PUFs.” In: *2014 IEEE International Symposium on Hardware-Oriented Security and Trust (HOST)*. May 2014, pp. 148–153 (cit. on p. 16).

[Mae12] Roel Maes. “Physically Unclonable Functions: Constructions, Properties and Applications (Fysisch onkloonbare functies: constructies, eigenschappen en toepassingen).” PhD thesis. Arenberg Doctoral School of Science, Engineering & Technology, 2012 (cit. on p. 14).

[MS09] A Maiti and P Schaumont. “Improving the quality of a Physical Unclonable Function using configurable Ring Oscillators.” In: *2009 International Conference on Field Programmable Logic and Applications*. Aug. 2009, pp. 703–707 (cit. on p. 16).

[MKD10] M Majzoobi, F Koushanfar, and S Devadas. “FPGA PUF using programmable delay lines.” In: *2010 IEEE International Workshop on Information Forensics and Security*. Dec. 2010, pp. 1–6 (cit. on p. 16).

[MKP08] M Majzoobi, F Koushanfar, and M Potkonjak. “Lightweight secure PUFs.” In: *2008 IEEE/ACM International Conference on Computer-Aided Design*. Nov. 2008, pp. 670–673 (cit. on p. 2).

[MN98] Makoto Matsumoto and Takuji Nishimura. “Mersenne Twister: A 623-dimensionally Equidistributed Uniform Pseudorandom Number Generator.” In: *ACM Trans. Model. Comput. Simul.* 8.1 (Jan. 1998), pp. 3–30 (cit. on p. 59).

[Mat+10] K Matulef, R O’Donnell, R Rubinfeld, and R Servedio. “Testing Halfspaces.” In: *SIAM J. Comput.* 39.5 (Jan. 2010), pp. 2004–2047 (cit. on pp. 21, 23, 26, 41, 47, 49, 51).

[ODo14] Ryan O’Donnell. *Analysis of Boolean Functions*. en. Cambridge University Press, June 2014 (cit. on pp. 5–13, 23, 24, 26, 33, 36).

[Pap+02] Ravikanth Pappu, Ben Recht, Jason Taylor, and Neil Gershenfeld. “Physical one-way functions.” en. In: *Science* 297.5589 (Sept. 2002), pp. 2026–2030 (cit. on p. 1).

[Rah+14] Tauhidur Rahman, Domenic Forte, Jim Fahrny, and Mohammad Tehranipoor. “ARO-PUF: An Aging-resistant Ring Oscillator PUF Design.” In: *Proceedings of the Conference on Design, Automation & Test in Europe*. DATE ’14. 3001 Leuven, Belgium, Belgium: European Design and Automation Association, 2014, 69:1–69:6 (cit. on p. 16).

[Roj13] Raul Rojas. *Neural Networks: A Systematic Introduction*. en. Springer Science & Business Media, June 2013 (cit. on p. 21).

[Rot76] O S Rothaus. “On “bent” functions.” In: *J. Combin. Theory Ser. A* 20.3 (May 1976), pp. 300–305 (cit. on p. 11).

[RS14] U Rührmair and J Sölter. "PUF modeling attacks: An introduction and overview." In: *2014 Design, Automation Test in Europe Conference Exhibition (DATE)*. Mar. 2014, pp. 1–6 (cit. on pp. 15, 58).

[RBK10] Ulrich Rührmair, Heike Busch, and Stefan Katzenbeisser. "Strong PUFs: Models, Constructions, and Security Proofs." In: *Towards Hardware-Intrinsic Security: Foundations and Practice*. Ed. by Ahmad-Reza Sadeghi and David Naccache. Berlin, Heidelberg: Springer Berlin Heidelberg, 2010, pp. 79–96 (cit. on p. 2).

[Rüh+10] Ulrich Rührmair, Frank Sehnke, Jan Sölter, Gideon Dror, Srinivas Devadas, and Jürgen Schmidhuber. "Modeling attacks on physical unclonable functions." In: *Proceedings of the 17th ACM conference on Computer and communications security*. ACM, Oct. 2010, pp. 237–249 (cit. on pp. 2, 21).

[Sko05] Sergei Petrovich Skorobogatov. "Semi-invasive attacks: a new approach to hardware security analysis." PhD thesis. Citeseer, 2005 (cit. on p. 1).

[SD07] G Edward Suh and Srinivas Devadas. "Physical Unclonable Functions for Device Authentication and Secret Key Generation." In: *Proceedings of the 44th Annual Design Automation Conference*. DAC '07. New York, NY, USA: ACM, 2007, pp. 9–14 (cit. on p. 2).

[Taj+15] S Tajik, H Lohrke, F Ganji, J P Seifert, and C Boit. "Laser Fault Attack on Physically Unclonable Functions." In: *2015 Workshop on Fault Diagnosis and Tolerance in Cryptography (FDTC)*. 2015, pp. 85–96 (cit. on p. 15).

[Taj+17] Shahin Tajik, Enrico Dietz, Sven Frohmann, Helmar Dittrich, Dmitry Nedospasov, Clemens Helfmeier, Jean-Pierre Seifert, Christian Boit, and Heinz-Wilhelm Hübers. "Photonic Side-Channel Analysis of Arbiter PUFs." In: *J. Cryptology* 30.2 (Apr. 2017), pp. 550–571 (cit. on p. 15).

[Wis+] Nils Wisiol, Christoph Graebnitz, Marian Margraf, Manuel Oswald, Tudor Soroceanu, and Benjamin Zengin. "Why Attackers Lose: Design and Security Analysis of Arbitrarily Large XOR Arbiter PUFs." In: vol. 49. EPiC Series in Computing. EasyChair, pp. 68–51 (cit. on p. 14).

[Xu+15] Xiaolin Xu, Ulrich Rührmair, Daniel E Holcomb, and Wayne Burleson. "Security Evaluation and Enhancement of Bistable Ring PUFs." en. In: *Radio Frequency Identification*. Lecture Notes in Computer Science. Springer, Cham, June 2015, pp. 3–16 (cit. on p. 21).

[Yam+11] Dai Yamamoto, Kazuo Sakiyama, Mitsugu Iwamoto, Kazuo Ohta, Takao Ochiai, Masahiko Takenaka, and Kouichi Itoh. "Uniqueness Enhancement of PUF Responses Based on the Locations of Random Outputting RS Latches." In: *Cryptographic Hardware and Embedded Systems – CHES 2011*. Springer Berlin Heidelberg, 2011, pp. 390–406 (cit. on p. 2).

[Zen17] Benjamin Zengin. "Fourier Analysis of Arbiter Physical Unclonable Functions." MA thesis. Freie Universität Berlin, July 2017 (cit. on p. 36).

DECLARATION

Selbständigkeitserklärung

Hiermit erkläre ich, dass diese Masterarbeit von mir selbst und ohne unerlaubte Beihilfe, unter der Verwendung der Angegebenen Quellen und Hilfsmittel verfasst wurde. Zudem sind alle Bezüge auf fremde Quellen als solche gekennzeichnet und die Arbeit ist frei von Plagiaten. Desweiteren bestätige ich, dass diese Arbeit bei keiner anderen Universität in gleicher oder ähnlicher Form als Prüfungsleistung eingereicht wurde.

Berlin, April 9, 2018

Christoph Graebnitz