Information Entropy Based Leakage Certification

Changhai Ou[1], Xinping Zhou[2], and Siew Kei Lam[1]

[1] School of Computer Science and Engineering,
Nanyang Technological University, Singapore.
chou@ntu.edu.sg, ASSKLam@ntu.edu.sg
[2] Beijing Unionpay Card Technology Co., Ltd, China
zhouxinping@bctest.com

Abstract. Side-channel attacks and evaluations typically utilize leakage models to extract sensitive information from measurements of cryptographic implementations. Efforts to establish a true leakage model is still an active area of research since Kocher proposed Differential Power Analysis (DPA) in 1999. Leakage certification plays an important role in this aspect to address the following question: "*how good is my leakage model?*". However, existing leakage certification methods still need to tolerate assumption error and estimation error of unknown leakage models. There are many probability density distributions satisfying given moment constraints. As such, finding the most unbiased and most reasonable model still remains an unresolved problem. In this paper, we address a more fundamental question: "*what's the true leakage model of a chip?*". In particular, we propose Maximum Entropy Distribution (MED) to estimate the leakage model as MED is the most unbiased, objective and theoretically the most reasonable probability density distribution conditioned upon the available information. MED can theoretically use information on arbitrary higher-order moments to infinitely approximate the true leakage model. It well compensates the theory vacancy of model profiling and evaluation. Experimental results demonstrate the superiority of our proposed method for approximating the leakage model using MED estimation.

Keywords: information theory, maximum entropy, maximum entropy distribution, leakage model, leakage certification, side channel attack

1 Introduction

Side-channel attacks, which aim to extract secret information that are unintentionally leaked in a cryptographic implementation, have been regarded as one of the most important threats against the security of embedded devices [27]. Power attacks, the most classic one of this family, can be divided into two categories: profiled attacks and non-profiled attacks. Non-profiled attacks such as Differential Power Analysis (DPA) [17], classify measurements (i.e. power traces) according to the intermediate values, and then calculate the differences. The correct key corresponds to the most obvious differential value (i.e. peak). The advantage

of non-profiled attacks is that the attacker does not require prior knowledge of the leakage model.

Standard profiled attacks include Template Attacks (TA) [5, 26] and stochastic models [30] as stated in [37]. They include two stages: leakage profiling and exploiting. The attacker needs to profile a leakage model before exploiting the leakage to recover the key. The true leakage of the cryptographic hardware is unknown and difficult to derive, and normal distribution is often used as the hypothetical model. Actually, the true leakage model may not well follow it. Hypothetical models such as Hamming weight [4], Hamming distance [25] and Switch [24], can also be used to approximate the leakage in other attacks such as Correlation Power Analysis (CPA) [4] to improve the efficiency. Exploring a true leakage model continues to be an active area of research.

In this paper, we aim to investigate the most unbiased, most reasonable and realistic leakage model, in order to address the question: "*what's the true leakage model of a chip?*". Existing works in side-channel attacks and evaluations (e.g. leakage detections and assessments) have also attempted to propose such a model, but without thorough study and suitable answers. In the following section, we will discuss these existing works along with leakage certification methods, before describing the main contributions of our work.

1.1 Related Works

Side channel attacks. A good leakage model has a significant impact on the effectiveness of side channel attacks. Many recent works have been undertaken to accurately profile the leakage model. However, most of them only considered first- and second-order moments (i.e. mean and variance) when profiling probability density distribution. This typically happens in Template Attack [26], which takes advantage of an off-line learning phase in order to estimate the leakage model. Since the true leakage model is unknown, the profiling methods are typically based on some assumptions on the leakage distribution (e.g. Gaussian noise) as in [10], which is not representative of the true leakage. Flament et. al. discussed probability density function estimation for side-channel attacks in [11]. They compared parametric estimation and histogram estimation, but did not consider information on higher-order moments.

Side channel evaluations. For attackers, the accuracy of the leakage model affects the effectiveness of the attack. For evaluators, the accuracy of leakage model affects the reliability of the evaluations (e.g. Success Rate (SR) and Guessing Entropy (GE) [38]). Since model errors provide evaluators with a false security level. Leakage detections, which relate to the concrete security level of an implementation given a model, are very important tools for side-channel evaluation. Unlike the above-mentioned side-channel attacks that are based on an assumption model, leakage detection tests such as Welch's t-test [7, 28, 2], Normalized Inter-Class Variance (NICV) [3], correlation ρ-test [8] and χ^2 test [21], use a bounded moment model [16]. They try to quantify the security of an implementation, of which the model reflects the leakage of target device. Leakage detection and assessment have been performed before cryptographic algorithms

are implemented on devices in [28] and [6]. These tests aim to detect the presence of leakage, without regards to whether the leakage can be exploited. Leakage assessments seek a standard approach that enables a fast, reliable and robust evaluation of the side-channel vulnerability of the given devices [31]. They can be regarded as an extension of leakage detection, which also require a bounded moment model rather than a true leakage distribution model. The above-mentioned works usually consider moments that are less than 4^{th} order. Higher-order moments (larger than 4) leakage detection and assessments (e.g. [16, 23, 29] and [31]) are seldom studied.

Leakage certifications. The effectiveness of both side-channel attacks and evaluations rely heavily on the true leakage model. However, this model is usually unknown. The following question underpins all the efforts that range from assuming a good leakage model (e.g. Hamming weight model and Hamming distance model) to profiling a good leakage model (e.g. normal distribution model and higher-order moments model used in higher-order attacks [20, 22]): How good is the leakage model? The answer to this question can be traced back to the complete evaluation framework proposed by Standaert et al. in [38]. The authors used Mutual Information (MI) to quantify the leakage and encountered the notoriously difficult problem of designing an unbiased and non-parametric estimator. Renauld et al. improved the work by introducing Perceived Information (PI) to estimate the MI biased by side-channel adversary's model [27]. In this case, the accuracy of the model determines the closeness of PI and MI. To better answer the question above, Durvaux et al. in [10] proposed leakage certification, which attempted to solve the fundamental problem that all evaluations were potentially biased by both assumption and estimation errors. They also tried to quantify the leakage of a chip and certify that the amount of information extracted was close to the maximum value that would be obtained with a perfect model. This work was further improved in [9].

1.2 Our Contributions

Existing works on side-channel attacks and evaluations have incessantly pursued a true leakage model. While existing leakage certification methods can provide a reasonable leakage model, they do not alleviate the attacker or evaluator from having to deal with model assumption error and estimation error. Moreover, the probability density distribution model under higher-order moments has not been discussed in existing works. Finally, even though there are numerous probability density distributions satisfying given moment constraints, achieving the most unbiased, most reasonable and least hypothetical leakage model still remains an unresolved problem.

To address the shortcomings of existing work on leakage certification, we propose Maximum Entropy Distribution (MED) to estimate the true leakage model of a chip. MED is the most unbiased, random, uniform and theoretically the most reasonable probability density distribution conditioned upon the available information. Here MED presents the probability density function assigned by using principle of maximum entropy. MED can theoretically use information on

arbitrary higher-order moments to infinitely approximate the true distribution of leakage, rather than assume a leakage model. To the best of our knowledge, this is the first work that considers information on higher-order moments when estimating probability density distribution. Experimental results demonstrate the superiority of our proposed method for approximating the leakage model using maximum entropy distribution estimation.

1.3 Organization

The rest of the paper is organized as follows. Information entropy, maximum entropy and leakage certification are introduced in Section 2. In Section 3, MED, including its estimation, parameter determination and fitting performance between estimated Probability Density Function (PDF) model and true leakage, is given. Then, we use Newton-Raphson nonlinear programming optimization method to fit MED with true distribution in Section 4. The specific algorithm and the optimal choice of histogram bins are also given in this section. Experiments are performed on simulated traces and measurements of ATMega644P micro-controller provided in [9] in Sections 5 and 6 to demonstrate the efficiency of our MED. Finally, we conclude the paper in Section 7.

2 Preliminaries

2.1 Information Entropy

Information entropy is a very important concept in information theory. Let X be a discrete random variable consisting of n observations of $\boldsymbol{x} = (x_1, x_2, \ldots, x_n)$, and the corresponding probabilities are $\boldsymbol{p} = (p_1, p_2, \ldots, p_n)$. Shannon defined information entropy (or uncertainty) as

$$H(x) = -\sum_{i=1}^{n} p_i \mathbf{ln} p_i \tag{1}$$

in [35], which was also denoted as self-information. Here $0 \leq p_i \leq 1$, and $\mathbf{ln}$ denotes the logarithmic function. If X is a continuous random variable, then the Shannon entropy is

$$H(x) = -\int_a^b f(x) \mathbf{ln} f(x) dx. \tag{2}$$

Here $[a, b]$ is the integral interval, and $f(x)$ is the probability density function. Information entropy is widely used in side-channel analysis such as Mutual Information Analysis (MIA) [13]. Self information of measurements can be used to quantify the leakage model of a chip.

2.2 Maximum Entropy Principle

Information theory provides a constructive criterion for setting up probability distributions on the basis of partial knowledge and leads to a type of statistical inference which is called the maximum entropy estimate [15]. Maximum entropy estimation is the most unbiased or most uniform probability distribution conditioned upon the available information [33]. Maximum entropy here means maximizing information entropy in Eq. 1 or Eq. 2.

There is an implicit constraint in Eq. 1 where

$$\sum_{i=1}^{n} p_i = 1. \tag{3}$$

The direct problem is to determine $\boldsymbol{p}$ conditioned upon Eq. 3. As detailed by Munirathnam et al. in [36], maximum entropy solved this problem by the maximization of Shannon entropy (uncertainty measure) of probabilities given in Eq. 1. By considering Lagrange multipliers, in order to maximize the entropy, the probabilities $\boldsymbol{p} = (p_1, p_2, \ldots, p_n)$ should satisfy

$$\varphi\left(p_1, p_2, \ldots, p_n\right) = -\sum_{i=1}^{n} p_i \mathbf{ln} p_i + \lambda\left(\sum_{i=1}^{n} p_i - 1\right). \tag{4}$$

The purpose of this paper is to profile the true leakage model from the observed samples. So, we use observer to represent side-channel attackers and evaluators. By differentiating φ with respect to p_i, the observer gets

$$\frac{\partial \varphi}{\partial p_i} = -\left(\mathbf{ln} p_i + 1\right) + \lambda = 0. \tag{5}$$

That is, $(\mathbf{ln} p_i + 1) = \lambda$, we deduce that $p_i = e^{\lambda - 1}$. When combined with Eq. 3, we obtain $\sum_{i=1}^{n} e^{\lambda - 1} = 1$. So,

$$\lambda = \mathbf{ln}\left(\frac{1}{n}\right) + 1. \tag{6}$$

Finally, we obtain $p_i = \frac{1}{n}$. That is to say, if we don't make any further assumptions on p_i, we can maximize the entropy of probability density function. In this case, the maximum entropy distribution is the most reasonable choice, any other choice would mean that we add additional constraints or unreasonable assumptions that are not available based on the existing information. In other words, maximum entropy contains minimum spurious information.

2.3 Leakage Certification

Side-channel attacks and evaluations require a perfect model to extract all information from the leakage measurements. However, the leakage model is never

perfect with errors arising from assumption and estimation. Durvaux et al. proposed the pioneering leakage certification in [10] and improved it in [9]. Leakage certification aims to bound and reduce the assumption and estimation errors, thus providing a good enough leakage model for attacks and evaluations.

Assumption error. Since the true leakage model of devices is unknown, the observer has to establish an assumption leakage model before he performs attacks or evaluations. For example, Gaussian model including mean and variance are used in Template Attack [26], and Hamming weight model in CPA [4]. These models include subjective assumptions and can easily lead to assumption error. A good model should reflect the basic information of the leakage, but it is not the real leakage model of the chip. The goodness of fit of these two models can be quantified by hypothesis testing.

Estimation error. The estimation error is the difference between the estimated parameters and true parameters of the leakage model. The main cause of this error is that the number of measurements is insufficient, which makes the probability density distribution estimation deviate from the true distribution. Typical estimation error is shown in Fig. 1, where two models estimated from samples deviate from the true distribution. It can be observed that Model 1 deviates further from the true leakage model than Model 2. Estimation error can be made arbitrarily small through more measurements and using cross validation techniques [9].

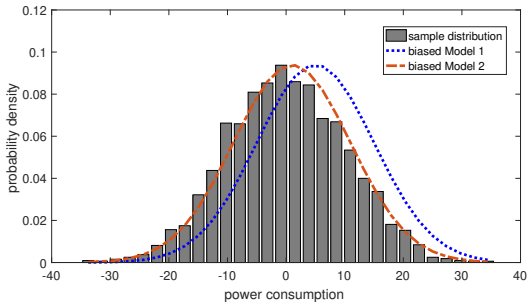

Fig. 1. Estimation errors in leakage model profiling.

Cross-validation. For each plaintext z, the observer randomly acquires samples and estimates the d^{th} order mean $\hat{\mu}^d$ using cross validation. Suppose that k-fold cross validation is used and n measurements are acquired. Measurements are divided into k non-overlapping folds of approximately the same size as introduced in [9]. The observer then selects the j^{th} $(1 \leq j \leq k)$ fold as the validation set and other $k-1$ folds as profiling set. The observer then randomly generates samples from the estimated leakage model. Each repetition generates a d^{th} order moment estimate $\tilde{m}_z^{d,(j)}$. The d^{th} order mean $\tilde{\mu}^d$ of real samples is processed in

the same way. Thus,

$$\begin{aligned}\hat{\mu}_z^d =& \hat{\mathbf{E}}_j\left(\hat{m}_z^{d,(j)}\right), \hat{\sigma}_z^d = \sqrt{\hat{\mathbf{var}}_j\left(\hat{m}_z^{d,(j)}\right)},\\ \tilde{\mu}_z^d =& \tilde{\mathbf{E}}_j\left(\tilde{m}_z^{d,(j)}\right), \tilde{\sigma}_z^d = \sqrt{\tilde{\mathbf{var}}_j\left(\tilde{m}_z^{d,(j)}\right)}.\end{aligned} \tag{7}$$

Here $\mathbf{E}(\cdot)$ and **var** denote the sample mean and variance operator. Then, Welch's t-test is performed as:

$$\Delta_z^d = \frac{\hat{\mu}_z^d - \tilde{\mu}_z^d}{\sqrt{\frac{(\hat{\sigma}_z^d)^2 + (\tilde{\sigma}_z^d)^2}{k}}}. \tag{8}$$

Let CDF_t denote the Cumulative Distribution Function (CDF) in t-test, d_f denote the number of freedom degrees (see Section 4.1 in [9]). The probability of observed difference coming from the effects of estimation is:

$$p = 2 \times \left(1 - \mathrm{CDF}_t\left(\left|\Delta_z^d, d_f\right|\right)\right). \tag{9}$$

The probability p only indicates that the difference between true samples and simulated samples has statistical significance. It doesn't reflect how large the difference is. The larger p is, the smaller the probability of estimation error. Leakage certification test uses information on higher-order moments to profile bounded moment leakage model [16], rather than making assumptions on the leakage distribution.

3 Maximum Entropy Distribution Estimation

A perfect leakage model can accurately reflect the leakage of devices and improve the effectiveness of side-channel attacks and security evaluations. However, such perfect models are generally unknown. Density estimation techniques, such as Maximum Entropy Distribution (MED) [41], have to be used to approximate the leakage distribution.

3.1 Maximum Entropy Distribution

Suppose that geometrical moments are used, the maximum entropy of the random variable X can be obtained by maximizing Shannon's entropy (see Eq. 1) subject to the constraints:

$$\int x^i f(x) dx = \mu_i, \quad i = 0, \ldots, N \tag{10}$$

where μ_i is the expectation value calculated from samples (e.g. $\mu_0 = 1$). N denotes that the first $N+1$ moment constraints $(\mu_0, \mu_1, \ldots, \mu_N)$ are used in our

side-channel attacks or evaluations. This can be expressed as the Lagrangian:

$$\begin{aligned}\mathcal{L} = &- \int f(x)\mathbf{ln}f(x)dx + (\lambda_0 + 1)\left[\int f(x)dx - 1\right] \\ &+ \sum_{i=1}^{m} \lambda_i \left[\int x^i f(x)dx - \mu_i\right].\end{aligned} \tag{11}$$

Here, $\boldsymbol{\lambda} = (\lambda_0, \lambda_1, \ldots, \lambda_m)$ are unknown Lagrange multipliers, and $\mathbf{ln}$ denotes natural logarithm. The setting of coefficient $(\lambda_0 + 1)$ is to facilitate the solution of $\boldsymbol{\lambda}$. Maximum entropy usually occurs at the extreme point of function $\boldsymbol{\lambda}$. By differentiating $\mathcal{L}$ with respect to $f(x)$, we have

$$\frac{\partial \mathcal{L}}{\partial f(x)} = -\int \left[\mathbf{ln}f(x) + 1\right] dx + (\lambda_0 + 1)\int dx + \sum_{i=1}^{m} \lambda_i \int x^i dx. \tag{12}$$

We set $\frac{\partial \mathcal{L}}{\partial f(x)} = 0$ and obtain

$$-\left[\mathbf{ln}f(x) + 1\right] + (\lambda_0 + 1) + \sum_{i=1}^{m} \lambda_i x^i = 0. \tag{13}$$

By transposition, we further get

$$\mathbf{ln}f(x) = \lambda_0 + \sum_{i=1}^{m} \lambda_i x^i. \tag{14}$$

Finally, we derive the maximum entropy probability density function (MED) as

$$f(x) = \mathbf{exp}\left(\lambda_0 + \sum_{i=1}^{m} \lambda_i x^i\right). \tag{15}$$

Maximum entropy accommodates information on higher-order moments and therefore facilitates a higher quality probability density function model. The observer does not make any assumptions on the leakage model except the moment information from the samples, which also shows the objectivity and rationality of $f(x)$.

3.2 Parameter Determination

We have derived the maximum entropy probability density function in Section 3.1. We can get the corresponding expression after solving the Lagrange Multipliers in $f(x)$. Since

$$\int f(x)dx = \int \mathbf{exp}\left(\lambda_0 + \sum_{i=1}^{m} \left(\lambda_i x^i\right)\right) dx = 1, \tag{16}$$

by multiplying both sides of the equality by $e^{-\lambda_0}$, we obtain

$$e^{-\lambda_0} = \int \mathbf{exp}\left(\sum_{i=1}^{m}\left(\lambda_i x^i\right)\right) dx. \tag{17}$$

The first unknown Lagrange multiplier can be expressed as:

$$\lambda_0 = -\mathbf{ln}\int \mathbf{exp}\left(\sum_{i=1}^{m}\left(\lambda_i x^i\right)\right) dx. \tag{18}$$

By differentiating λ_0 with respect to λ_i (see Eq. 17), we can also get

$$\frac{\partial \lambda_0}{\partial \lambda_i} = \int x^i \mathbf{exp}\left(\sum_{i=1}^{m}\lambda_i x^i\right) dx. \tag{19}$$

This means, $\frac{\partial \lambda_0}{\partial \lambda_i} = \mu_i$. Since $\int \mathbf{exp}\left(\sum_{i=1}^{m}\lambda_i x^i\right) dx = 1$, the Lagrange multipliers can be defined by the sum of residuals:

$$r_i = 1 - \frac{\int x^i \mathbf{exp}\left(\sum_{i=1}^{m}\lambda_i x^i\right) dx}{\mu_i \int \mathbf{exp}\left(\sum_{i=1}^{m}\lambda_i x^i\right) dx} \tag{20}$$

for $i = 1, 2, \ldots, m$. The minimum residual can be expressed as:

$$\mathbf{min} \quad R = \sum_{i=1}^{m} r_i. \tag{21}$$

Suppose that ϵ is the permissible error of the observer. If $R < \epsilon$, then R converges, he accepts the corresponding Lagrange multipliers $\boldsymbol{\lambda} = (\lambda_0, \lambda_1, \ldots, \lambda_m)$ and recovers the probability density function $f(x)$. The problem of Shannon entropy maximization is a convex minimization problem.

3.3 Fitting Performance Metrics

Maximum entropy is a monotonic decreasing function, which means that the observer obtains smaller maximum entropy when the algorithm iterates. The probability density function MED is obtained after $R < \epsilon$. This will be followed by testing whether the profiled model can accurately reflect the true leakage of device (i.e. test of goodness of fit).

According to the report "*Guide to Expression of Uncertainty in Measurement (GUM)*" (see [14]), standard uncertainty of the result of measurement corresponding to maximum entropy is expressed as a standard deviation. By performing maximum entropy estimation on the observations, the expectation and deviation are

$$\hat{\mu} = \int x\hat{f}(x)dx \tag{22}$$

and

$$\hat{\sigma} = \int \left[x - \hat{x}\left(x\right)\right]^2 \hat{f}(x)dx. \tag{23}$$

If $f(x)$ approximates the true distribution, $\hat{\mu} \to \mu_1$ and $\hat{\sigma} \to \mu_2$.

Actually, to test whether this model is consistent with the real leakage model, the leakage certification test of Durvaux et al. (see [9] and [10]) can be employed. This work performed hypothesis tests on samples generated from the estimated leakage model and real samples to determine if the model can be accepted based on the test results. Other tests such as Chi-square χ^2 [21] and Root Mean Square Error (RMSE) [34], can also be used to test maximum entropy probability density distribution. In principle, the more moments are used, the more accurate the model is, and the smaller the error.

In our paper, we combine GUM's test and Welch's t-test introduced by Durvaux et al. in [10] to detect estimation error. Specifically, referring to the leakage certification test of Durvaux et al., we divide the collected measurements into k-folds of approximately the same size. Each iteration selects a new validation fold and uses other $k-1$ folds as training set. We first find the interval of training set and calculate $\hat{\mu}$ and $\hat{\sigma}$ in GUM's test. We then randomly generate samples of the same size as the validation set from this model. Referring to Eq. 8 and Eq. 9, we carry out Welch's t-test to quantify the probability of the difference caused by the estimation error.

4 Nonlinear Programming Optimization

The minimum residual given by Eq. 21 can be solved using nonlinear programming optimization, which minimizes residual by calculating the least squares of error. If r_i is a linear function for all i, R can be solved by linear least square method. R here is a non-linear function that can be solved using nonlinear least square method. This is based on the basic principle of using a series of linear least squares to solve nonlinear least square problems.

4.1 Newton-Raphson Method

By combining Eq. 10 and Eq. 15, the i^{th} order moment can be expressed as:

$$G_i\left(\boldsymbol{\lambda}\right) = \int x^i \mathbf{exp}\left(\lambda_0 + \sum_{i=1}^{m} \lambda_i x^i\right) dx = \mu_i \tag{24}$$

if Eq. 10 is regarded as a function of $\boldsymbol{\lambda} = (\lambda_0, \lambda_1, \ldots, \lambda_m)$. According to [42, 19], one can expand $G_i\left(\boldsymbol{\lambda}\right)$ in Taylor's series around trial values of $\boldsymbol{\lambda}^0$, dropping the second and higher-order moments

$$\begin{aligned} \mu_i^{'} =& G_i\left(\boldsymbol{\lambda}\right) \\ \cong& G_i\left(\boldsymbol{\lambda}^0\right) + \left(\boldsymbol{\lambda} - \boldsymbol{\lambda}^0\right)^{\mathsf{T}} \left[\mathbf{grad}\ G_i\left(\boldsymbol{\lambda}\right)\right]_{\boldsymbol{\lambda}=\boldsymbol{\lambda}^0} \end{aligned} \tag{25}$$

and solving them iteratively. Here the symbol 'T' indicates vector or matrix transposition and symbol '**grad**' indicates gradient function. If the first five moments are taken into consideration, then $m = 4$. Mean, variance, skewness and kurtosis [22] are often used in side-channel attacks. However, very a few papers discussed very higher-order moments (e.g. [16]). Obviously, for nonlinear functions like R, the observer can solve them using higher-order Taylor expansions, of which $\mu_i^{'}$ is closer to μ_i.

The work in [19] defined two vectors

$$\boldsymbol{\delta} = \boldsymbol{\lambda} - \boldsymbol{\lambda}^0 \tag{26}$$

and

$$\boldsymbol{v} = \left[\mu_0^{'} - G_0\left(\boldsymbol{\lambda}^0\right), \ldots, \mu_N^{'} - G_N\left(\boldsymbol{\lambda}^0\right)\right]^{\mathsf{T}}. \tag{27}$$

Here the superscript 0 of $\boldsymbol{\lambda}^0$ represents the number of iterations. The authors then defined a matrix $\boldsymbol{G}$ by

$$\boldsymbol{G} = \left[g_{nk}\right] = \left[\frac{\partial G_n(\boldsymbol{\lambda})}{\partial \lambda_k}\right]_{(\boldsymbol{\lambda}-\boldsymbol{\lambda}^0)}. \tag{28}$$

$\boldsymbol{G}$ is a Hankel matrix, of which

$$\begin{aligned} g_{nk} &= \int x^n x^k \mathbf{exp}\left(\sum_{i=1}^{m} \lambda_i x^i\right) dx \\ &= \int x^{n+k} \mathbf{exp}\left(\sum_{i=1}^{m} \lambda_i x^i\right) dx \\ &= g_{n+k}. \end{aligned} \tag{29}$$

This means $g_{nk} = g_{kn}$. This also means that in order to calculate the first five moments ($m = 4$) of $f(x)$, we have to calculate $\boldsymbol{G}_0, \ldots, \boldsymbol{G}_8$. Solving Eq. 21 is equivalent to solving the linear system of equations:

$$\boldsymbol{G}\boldsymbol{\delta} = \boldsymbol{v}. \tag{30}$$

The above is the first iteration of $\boldsymbol{\lambda}^0$. The observer obtains the error $\boldsymbol{\delta}^0$ of $\boldsymbol{\lambda}^0$ in probability density function $f(x)$. Then $\boldsymbol{\lambda}^0$ is replaced by $\boldsymbol{\lambda}^1 = \boldsymbol{\lambda}^0 + \boldsymbol{\delta}^0$ and the next iteration is executed. The iteration continues until $\boldsymbol{\delta}$ becomes appropriately small (i.e. $R < \epsilon$). In principle, the smaller the ϵ, the better $f(x)$ fits the true leakage distribution.

4.2 Algorithm Implementation

We have described the principle of nonlinear programming optimized MED estimation in the previous sub-section. Here we provide the detailed algorithm

in Algorithm 1. In our algorithm, the samples and accuracy serve as inputs, $\boldsymbol{\lambda}$ and maximum entropy $MaxEnt$ are the outputs. There are no other parameters to set, which indicates that our algorithm is very simple and does not need to handle the complex parameter optimization problem. The algorithm first estimates the optimal number of bins h_n using function **BinsEstimation** (as detailed in Section 4.3). Then, it estimates the probability density function using histogram, where the corresponding inputs include the samples and h_n. The outputs of function **Histogram** include the probability density distribution $\boldsymbol{p}$ and the mid-points of all bins $\mathbf{x}$.

Algorithm 1: Nonlinear programming optimized MED estimate.

Input: samples $\boldsymbol{x}$ and ϵ
Output: estimated parameters $\boldsymbol{\lambda}$ and $MaxEnt$

the number of bins $h_n = \mathbf{BinsEstimation}(\boldsymbol{x})$;
estimate PDF $(\boldsymbol{p}, \mathbf{x}) = \mathbf{Histogram}(\boldsymbol{x}, h_n)$;
calculate moments $G_1, \ldots, G_N$ using $\boldsymbol{p}$ and $\mathbf{x}$;
$\lambda_0 = \mathbf{max}(\boldsymbol{x}) - \mathbf{min}(\boldsymbol{x})$;
while *1* **do**
    calculate $\boldsymbol{v}$;
    solve $\boldsymbol{G\delta} = \boldsymbol{v}$;
    update $\boldsymbol{\lambda} = \boldsymbol{\lambda} + \boldsymbol{\delta}$;
    update $f(\mathsf{x})$ and R;
    if $R < \epsilon$ **then**
        $MaxEnt = -\sum_{j=1}^{h_n} f(\mathsf{x}_j)\mathbf{ln}\ f(\mathsf{x}_j)$;
        break;
    end
    update $G_1, \ldots, G_N$ using $f(\mathsf{x})$ and $\mathbf{x}$;
end

The purpose of Algorithm 1 is to fit $\boldsymbol{p}$ and $f(x)$ and find the parameters in $\boldsymbol{\lambda}$ that satisfies the fitness condition. It is worth noting that our algorithm does not need to set $\boldsymbol{\lambda}$. We only initialize $\lambda_0 = \mathbf{max}(\boldsymbol{x}) - \mathbf{min}(\boldsymbol{x})$ as suggested in [19]. All λ-s will be adjusted in the following repetitions. Our algorithm initializes moments $G_1, \ldots, G_N$ using $\boldsymbol{p}$ and $\mathbf{x}$. It then calculates $\boldsymbol{v}$, solves $\boldsymbol{G\delta} = \boldsymbol{v}$ and updates $\boldsymbol{\lambda} = \boldsymbol{\lambda} + \boldsymbol{\delta}$. It then updates $f(x)$ using $\boldsymbol{\lambda}$. $\boldsymbol{\lambda}$ and R will gradually stabilize after a number of iterations. In this case, $f(x)$ approximates the distribution $\boldsymbol{p}$, and the measurement uncertainty in Eq. 22 and Eq. 23 approaches the mean and variance of true samples.

For Newton-Raphson method, one of the conditions for iterative convergence is that $\boldsymbol{G}$ is a non-singular matrix. If $\boldsymbol{G}$ is singular, then nonlinear programming optimizations such as damped least square (i.e. Levenberg-Marquardt) method, can also be taken into consideration. Unlike Newton-Raphson method, Levenberg-Marquardt method needs to set several parameters. To optimize these parameters, we need to consider the specific samples and model, which is not

easy. Moreover, the algorithm may return a local optimal solution during iteration. In order to find the global optimal solution, Algorithm 1 can be combined with Simulated Annealing (SA) [12] or Genetic Algorithm (GA) [41].

4.3 Optimal Bin Width in Histogram

Probability density estimation is a widely-used method for estimating the distribution model of samples. It can be broadly classified to parametric estimation and non-parametric estimation. Parametric estimation is utilized If we have already known what kind of probability density distribution the observed samples follow and only need to determine its parameters. The most commonly used parametric estimation methods are Maximum Likelihood Estimation (MLE) and Bayesian estimation. If we do not know the true distribution of the observed samples, we can only use the non-parametric estimation method to estimate its probability density distribution model. The non-parametric estimation methods mainly include histogram estimation and Kernel Density Estimation (KDE) [39]. Since the true leakage model of a chip is unknown, we consider the non-parametric estimation method in Algorithm 1. Specifically, we use histogram to estimate the probability density distribution in this paper.

Let κ denote the number of bins in histogram, and x denote the mid-points of bins. The mid-point x_j of each interval $(1 \leq j \leq \kappa)$ is often selected as representative value of this bin [36]. To derive the probability density distribution, the observer needs to calculate the frequency of each bin. Suppose that he obtains the frequency distribution of $\boldsymbol{x}$ as $(f_1, f_2, \ldots, f_\kappa)$ (see [36]). In this case, the expectation value of the i^{th} order moment of samples can be expressed as:

$$\mu_i = \mathbf{E}\left(\boldsymbol{x}^i\right) \cong \frac{1}{\kappa} \sum_{j=1}^{\kappa} f_j \mathsf{x}_j^i. \tag{31}$$

In order to avoid overflow, the domain of $\boldsymbol{x}$ can also be transformed into interval $[0, 1]$ using equation $x^{'} = (x - x_{min})/(x_{max} - x_{min})$. Here x_{min} and x_{max} denote the minimum and maximum values of $\boldsymbol{x}$.

It is difficult to determine the optimal bin width when constructing a histogram. To illustrate this, we simulate the normal distribution $\mathcal{N}\left(0, 5^2\right)$ and randomly generate 1000 measurements from this model. The probability density distributions when the number of bins is set to 5, 10, 20 and 200 are shown in Fig. 2. As can be observed, it will be unreasonable to set the number of bins to 5 and 200, as this will lead to the number of bins being either too small or too large. As a result, profiling will lose a lot of information of the distribution. On the other hand, determining whether 10 bins or 20 bins are reasonable is not straightforward. The authors in [40] suggested that the bin width should be chosen so that the histogram displays the essential structure of the data, without giving too much credence to the data set at hand.

Sample size is an important indicator for side-channel evaluations. The implicit prerequisite in profiling stage is that the observer can capture a sufficient number of measurements so that he can profile a sufficiently accurate leakage

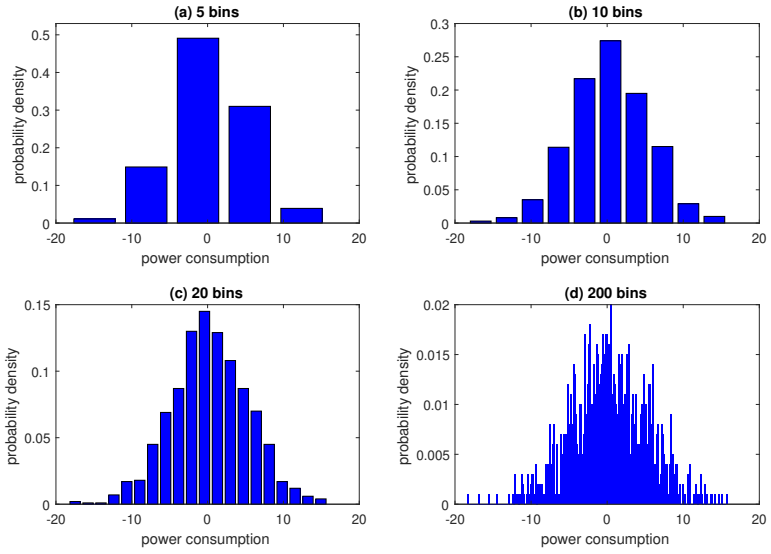

Fig. 2. Probability density distribution under different numbers of bins.

model. However, the fewer power traces he uses in the attack stage, the more powerful and higher efficiency is the scheme. It is desirable if he can profile an accurate leakage model under a small sample size. Therefore, deriving an optimal bin width is an important issue that must be considered.

The authors in [32] indicated that the formula for determining the optimal histogram bin width should asymptotically minimize the integrated mean squared error. They proposed the following to determine the bin width:

$$h_n = \frac{3.49s}{\sqrt[3]{n}}, \tag{32}$$

where s was an estimate of the standard deviation and n was the sample size. In side-channel attacks, we often assume that the leakage follows Gaussian distribution. However, this assumption may be incorrect, or at least inaccurate, since the true leakage model is unknown. Although Eq. 32 is established on the basis of Gaussian density, fortunately, it can be also used for non-Gaussian data. Thanks to Scott's solution, the problem of estimating the bin width in our side-channel attacks can be resolved.

5 Simulated Experiments

5.1 Leakage Function

Our first experiment is performed on simulated measurements. Let $\mathbf{HW}(\cdot)$ denote the Hamming weight function, $\mathbf{SBOX}(\cdot)$ denote the SubBytes operation of AES-128, z_i denote the i^{th} plaintext byte and k^* denote the encryption key byte. The leakage function is defined as:

$$l_i = \mathbf{HW}(\mathbf{SBOX}(\mathsf{z}_i \oplus k^*)) + \theta, \tag{33}$$

where $\oplus$ denotes the XOR operation, l_i denotes the corresponding leakage sample and θ denotes the noise component [18] that follows normal distribution $\mathcal{N}(0, 10^2)$.

5.2 Information on Higher-order Moments

Maximum entropy decreases with increase in moment constraints. Since each moment contains information, the uncertainty of the model is reduced if a new moment is added. However, this conclusion is not always established when measurements are limited (as shown in Table.1). Here 800 measurements are used, ϵ is set to 10^{-8} and C_i denotes the i^{th} order moment. The maximum entropy under the constraint of natural moment (C_0) is about 14.6171 and changes to 14.6460 after adding the first-order moment constraint. This means that the maximum uncertainty of distribution varies by 0.0289 after adding the first-order moment. The second-order moment makes the maximum uncertainty decrease the most followed by the first-order moment. $MaxEnt$ changes very little after reaching 9.8123. In this case, the fitting performance also gradually approaches the optimum.

Table 1. Maximum entropy under different constraint sets.

Constraint sets	$MaxEnt$
$\{\mathsf{N}\}$	14.6171
$\{\mathsf{N}, \mathsf{C}_1\}$	14.6460
$\{\mathsf{N}, \mathsf{C}_1, \mathsf{C}_2\}$	9.4689
$\{\mathsf{N}, \mathsf{C}_1, \mathsf{C}_2, \mathsf{C}_3\}$	9.8123
$\{\mathsf{N}, \mathsf{C}_1, \mathsf{C}_2, \mathsf{C}_3, \mathsf{C}_4\}$	9.8123
$\{\mathsf{N}, \mathsf{C}_1, \mathsf{C}_2, \mathsf{C}_3, \mathsf{C}_4, \mathsf{C}_5\}$	9.8123

Since we only initialize λ_0 in our MED, the fitting performance of $f(x)$ and true leakage distribution is not good at the initial iterations. As the number of iterations increases, the variables in $\boldsymbol{\lambda}$ are constantly updated, $f(x)$ also converges to the true distribution (as shown in Fig. 3(1)). Finally, the required accuracy is achieved after 10 iterations.

The information entropy on different moments is different, as with the fitting performance. We analyse the different moments on $f(x)$ of the above 800 traces

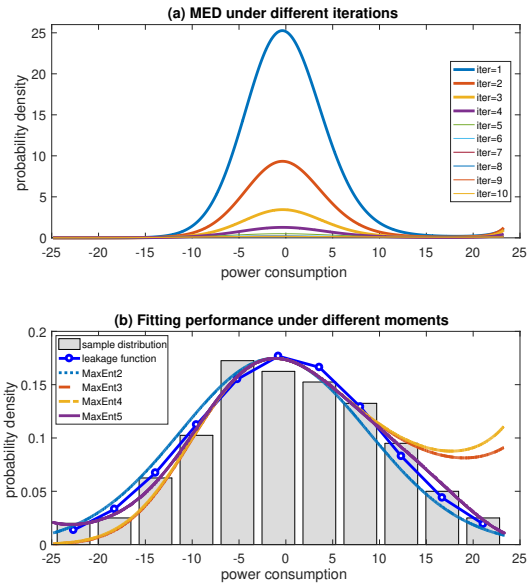

Fig. 3. MED under different iterations and fitting performance under different moments.

and consider the moments with orders higher than 3. Let MaxEnt2 denotes a set of moments including $\{\mathsf{N}, \mathsf{C}_1, \mathsf{C}_2\}$, and MaxEnt3 denotes a set of moments including $\{\mathsf{N}, \mathsf{C}_1, \mathsf{C}_2, \mathsf{C}_3\}$. Similar notations apply for MaxEnt4 and MaxEnt5. It can be observed from the experimental results in Fig. 3(2) that the higher the orders use, the better the fitting performance of $f(x)$ and the true distribution. When considering MaxEnt2 and MaxEnt3, $f(x)$ still deviates from the true distribution of the leakage, and the two have a good fit in MaxEnt4. The fitting performance in MaxEnt5 is better and $f(x)$ almost passes through the middle of all bins. On one hand, this indicates that to fit the real leakage distribution, we need to combine the information on all six moments in MaxEnt5. On the other hand, this indicates that the information on higher-order moments are limited. It is obvious that there is still a deviation between the estimated model $f(x)$ and the true leakage model. This is mainly due to the small number of samples we use and the large deviation between the sample distribution and real leakage distribution in our experiment. In order to better fit the real leakage distribution, we can further reduce ϵ or consider higher-order moments, or even use more measurements.

5.3 Fitting Performance

The evaluator can encrypt any number of plaintexts and collect their leakage to profile sufficiently accurate PDF model. Compared to evaluator, the number of measurements obtained by the attacker is limited, so it is important to make full use of the information on them. The number of measurements is also the most important factor in our MED estimation. So, estimating the most reasonable, most unbiased leakage model from the limited model is a very important issue that they needs to be taken into consideration. Here we also compare fitting performance of our MED estimation under different numbers of measurements. The experimental results corresponding to Hamming weight 0 are shown in Fig. 4, Fig. 5 and Table 2.

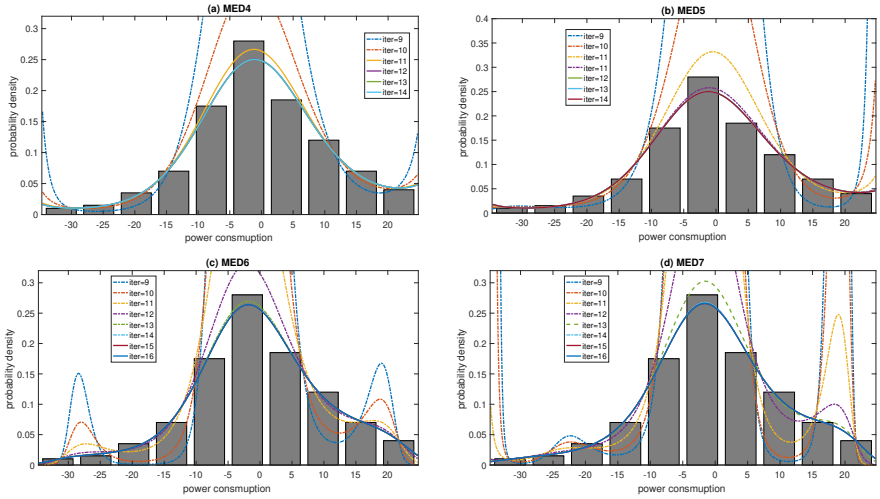

Fig. 4. MED under different iterations using 200 simulated measurements.

First, considering the fitness between MED and the real distribution under different moments, we simulate 200 power traces and fit the corresponding distribution with its first $5 \sim 8$ moments (the corresponding Maximum Entropy Distribution is expressed as MED4∼MED7). The experimental results are shown in Fig. 4. Since we only initialize λ_0, $f(x)$ deviates from the true distribution at the initial iterations. So, the MED corresponding to iterations less than 8 is not given. $f(x)$ appears to exhibit a complex distribution under different moments. It gradually fits to the real distribution as the number of iterations increases. However, the fitting performance under different moments is very different, MED6 and MED7 fit better than MED4 and MED5. Moreover, the number of iterations is closely related to the complexity of distribution of samples. The more complex this is, the more iterations are required to achieve a better fitness.

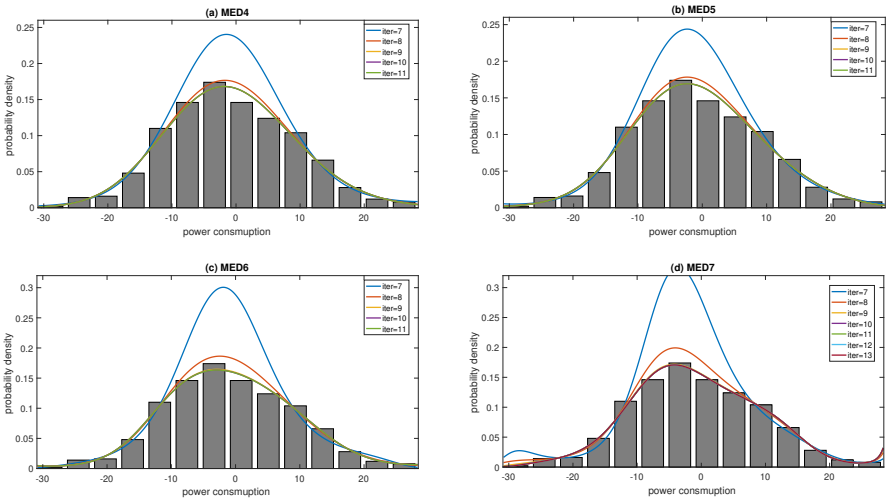

Fig. 5. MED under different iterations using 500 simulated measurements.

We also show the fitting performance of MED and true distribution under different moments when 500 simulated measurements are used. With increasing number of measurements, the maximum entropy decreases gradually. The leakage model becomes simpler and more definite, and the fitting performance between MED and the true leakage function becomes better (see Fig. 5). This indicates that the higher-order moments make full use of information on measurements. MED6 and MED7 better reflect the true distribution than MED4 and MED5, and pass through the middle of most of bins. The GUM test in Table 2 also illustrates this. Moreover, the distribution of samples reflects the leakage function better and is more conducive to the fitness of $f(x)$. As such, the number of iterations in Algorithm 1 also decreases.

Table 2. Parameters under different numbers of measurements.

measurements	iteration	MaxEnt	GUM	
			$\hat{\mu}$	$\hat{\sigma}$
200	14	12.1936	4.4432	10.7905
400	13	9.5870	5.3772	9.3680
800	12	8.8991	4.9424	9.8951
1600	11	7.7859	4.9751	9.9917
3200	9	6.5737	5.1266	9.9241
6400	9	5.8106	4.8918	10.1207

It is noteworthy that both $f(x)$ and true leakage function do not successfully pass through the middle of all the bins. This would have been unrealistic especially when there are many bins which are not well distributed. This is not

a concern as $f(x)$ has already approximated the true leakage model. Although the true leakage model of cryptographic devices is unknown. It is therefore not necessary to make $f(x)$ pass through the middle of all bins, as long as the fitness requirements in leakage certification test is met.

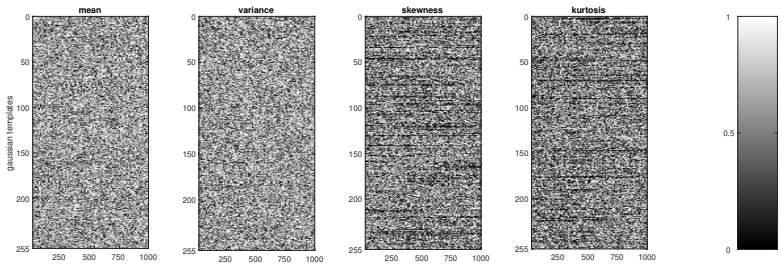

Fig. 6. Leakage certification tests on simulated measurements.

We use the MATLAB source code provided by Durvaux et. al. in [1] to perform our leakage certification test. Specifically, we randomly generate 1000 samples from the leakage model given in Eq. 33 for each possible intermediate value and train 256 leakage models independently. Each leakage model can be expressed as $\mathcal{N}\left(\hat{\mu}, \hat{\sigma}^2\right)$, the first six moments and cross validation are used. Samples with same size of validation set are randomly generated from this model. We then perform leakage certification test on them, of which the experimental results are shown in Fig. 6. The p-values output by our different t-tests are in grey-scale, for four statistical moments (i.e. the mean, variance, skewness and kurtosis). The results show that our MED model fits the measured leakages quite accurately. We only consider the first six order of moments MED5 in this experiment. In order to achieve better fitting performance, higher-order moments can also be taken into consideration. Actually, the results of leakage certification tests using MED5 are better than those using MED4 in our experiments.

6 Experiments on ATMega644P Microcontroller

6.1 Measurement Setup

Our second experiment is performed on the measurements provided by Durvaux et al. in their leakage certification code [1]. These measurements are leaked from an AES Furious algorithm implemented on an 8-bit Ateml AVR (ATMega644P) microcontroller. Let z and k denote the target input plaintext byte and subkey, and $y = z \oplus k$. For each possible value of y, 1000 encryptions and measurements are collected. Then, leakage certification tests are performed on them.

6.2 Low Discretization of Leakage Samples

Compared with real leakage, measurements sampled from simulated leakage model are more random. They also have higher discretization and better satisfy the given distribution. Moreover, we know the specific leakage function (i.e. real leakage model) in simulation experiments. In order to compare the fitting performance between MED and real model, we can simply compare MED with leakage function. However, the real leakage model of cryptographic devices is unknown and can only be measured by other methods such as hypothesis tests.

It is worth noting that the leakage samples of ATMega644P microcontroller provided by Durvaux et al. in [1] is with low discretization. We have tested a lot of y-s under all 1000 measurements and give the probability density functions corresponding to $y = 0, \ldots, 3$ in Fig. 7. The probability density values close to the middle of distribution are 0, but some others close to the edges are significantly high. We also carry out experiments on AT89S52 micro-controller and obtain similar conclusions. The randomness of leakage model reflected by these low discrete samples is also reduced. There could be three reasons for this phenomenon: (1) the leakage of the device is not normally distributed, (2) the size of measurements is too small, and (3) the measurement limitations of the oscilloscope.

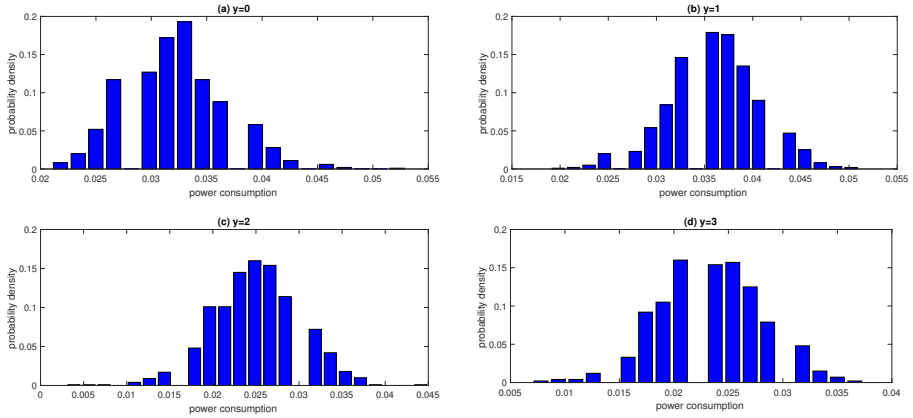

Fig. 7. Low data discretization of leakage from ATMega644P micro-controller.

The matrix $\boldsymbol{G}$ is close to singularity if we use Newton-Raphson method to fit MED and true leakage distribution under the condition that the leakage samples are with low discretization. We change the accuracy ϵ in our iteration to 10^{-6}. It is worth noting that, although we reduce the accuracy in our iteration in Table 3 (y=1), the number of iterations increase compared to Table 2. The algorithm

needs to iterate about 16 times. Moreover, the uncertainty of distribution decreases when we use more measurements. We also show the experimental results of GUM tests in Table 3, which indicates that the mean of these samples is about 0.0249 and the variance is about 0.0050.

Table 3. Parameters under different numbers of measurements.

measurements	iteration	MaxEnt	GUM	
			$\bar{\mu}$	$\bar{\sigma}$
200	15	5.7612	0.0247	0.0050
400	16	5.2089	0.0249	0.0051
600	16	4.6202	0.0257	0.0049
800	17	4.4929	0.0249	0.0052
1000	16	4.2925	0.0249	0.0050

6.3 Fitting Performance

We use the first six moments to analyse the measurements corresponding to $y = 1$. The number of bins in histogram varies with the size of measurements used according to Eq. 32. Considering the first 200 and the first 300 measurements, h_n is 9 and 11 respectively. Unlike Fig. 7, the new divisions do not exhibit the complex phenomenon that the probability density is almost 0 in the middle and high on both two sides, which is also amenable to MED fitness. However, the histogram shows another complex distribution when $n = 200$: the probability density is low in the middle and high on both two sides. Obviously, normal distribution considering skewness and kurtosis is not enough to fit this distribution. To solve this, the observer can increase or decrease the number of bins, or improve the algorithm so that $f(x)$ can still fit the complex distribution.

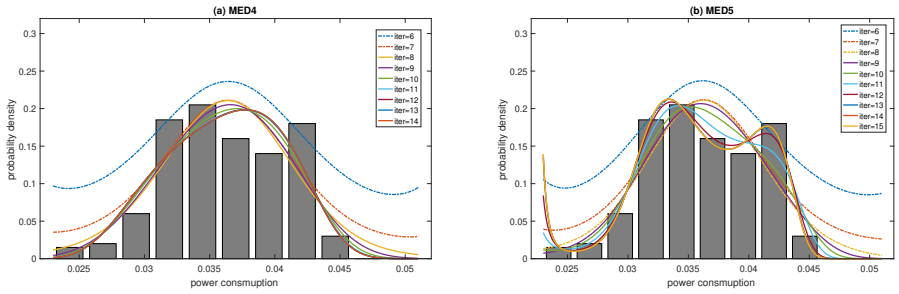

Fig. 8. MED4 and MED5 under different iterations.

Fortunately, one advantage of our MED is that it can theoretically fit complex distributions by making full use of information on arbitrary higher-order

moments. $f(x)$ gradually fits the sample distribution in iterations under first five and six moments (see MED4 and MED5 in Fig. 8). Specifically, the irregular probability density distribution of samples has been found after 11 iterations in our MED. $f(x)$ shows the same characteristics as the probability density distribution of samples in the twelfth iteration: low in the middle, high on the left and low on the right. Although the error ϵ reduces, MED-s almost coincide in the 13^{th}, 14^{th} and 15^{th} iterations. It is very difficult to distinguish them in Fig. 8(2). $f(x)$ passes through the middle of most of bins, which shows very good fitness performance with the distribution of measurements. Since the distribution is complex under these 200 measurements, we also consider MED6 and MED7, of which the experimental results are shown in Fig. 9. In order to fit bins, both two ends of MED6 and MED7 are higher than those of MED4 and MED5 at the initial iterations. Fortunately, they decrease rapidly and converge quickly as the number of iterations increases. In other words, the fitting performance converges to the optimum quickly. Compared to MED4 and MED5, the fitting curves of MED6 and MED7 are more complex and curved, which implies that the fitting performance is much better.

The number of iterations of MED6 and MED7 is also higher than that of MED4 and MED5 under the same accuracy. Moreover, the higher order moments fit better than the lower order moments under the same number of iterations. We also obtain similar conclusions in Section 5.3. $\boldsymbol{\lambda}$ in Fig. 9 and Fig. 10 changes slightly as the iteration reaches a certain point. Maximum entropy distributions $f(x)$ also change very little, and they eventually overlap in the last a few iterations.

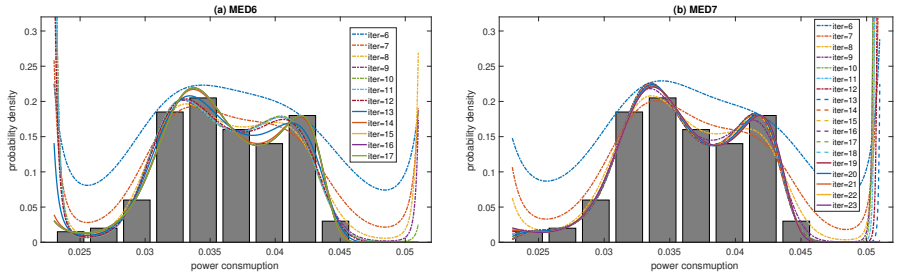

Fig. 9. MED6 and MED7 under different iterations.

It is noteworthy that the observer is likely to obtain different $f(x)$ when using varying numbers of measurements, or different measurement sets of the sample size. However, $f(x)$ can well reflect the true distribution of current measurements. MED represents the most unbiased, most objective and most reasonable distribution estimation of the observed measurements. When the number of power traces increases, the observer gets a better sample distribution and

a decreasing maximum entropy (as shown in Table 3). We also test our MED under more measurements. For example, the MED of first 300 measurements corresponding to $y = 1$, of which the probability follows a distribution with left side up but the right side sloping down smoothly. Therefore, $f(x)$ first ascends at both ends and finally the left-end ascends to fit the high probability density while the right-end gradually descends to fit the low probability density on the right. Finally, $f(x)$ fits well to the true distribution of measurements. Similar conclusions and fitting process can also be obtained from Fig. 4, Fig. 5, Fig. 8 and Fig. 9.

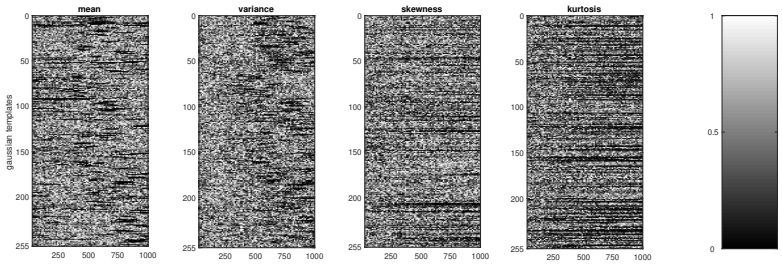

Fig. 10. Leakage certification of leakage from ATMega644P Microcontroller.

Fig. 8 and Fig. 9 fully embody the super fitness ability of our MED. We use the first six moments (MED5) in our leakage certification test and obtain very good fitting performance in our leakage certification tests (as shown in Fig. 10). However, due to the pre-mentioned low data dispersion of the leakage of ATMega644P microcontroller, many MED models cannot fit the distribution of measurements on the model $\mathcal{N}\left(\hat{\mu}, \hat{\sigma}^2\right)$ and pass the leakage certification tests (see horizontal blank lines in Fig. 10). Moreover, p value in Welch's t-test is in function of the number of measurements used for certification as stated by Durvaux et al. in [9]. This also indicates that the information on the first six moments (MED5) is insufficient to ensure that $f(x)$ accurately fits the distribution of measurements. Therefore, in order to fit the distribution, we need to take information on higher-order moments into consideration. For example, the first seven or eight moments (MED6 and MED7) in Fig. 9, or even using the information on moments with orders larger than 7. We also carry out leakage certification tests on MED4, of which the results show that p values on MED5 look 'whiter'. This also shows that the fitting performance of the higher-order moments is better.

7 Conclusion and Future Works

The accuracy of a leakage model plays a very important role in side-channel attacks and evaluations. In this paper, we aim to determine the true leakage model of a chip. To achieve this, we performed Maximum Entropy Distribution (MED) estimation on higher-order moments of measurements to approximate the true leakage model of devices rather than assume a leakage model. Then, non-linear programming is used to solve the Lagrange multipliers. The MED is the most unbiased, objective and reasonable probability density distribution estimation that is built on known moment information. It does not include the profiler's subjective knowledge of the model. MED can well approximate the true distribution of the leakage of devices, thus reducing the model assumption error and estimation error. It can also well approximate the complex distribution (e.g. non-gaussian distribution). Both theoretical analysis and experimental results verify the feasibility of our proposed MED.

MED can theoretically use information on arbitrary higher-order moments to infinitely approximate the true distribution of leakage. In this case, more moments mean more information. However, more moments also necessitate more computation. In our future work, we will explore methods to accurately measure the amount of information on each moment and make MED choose the right moments for each iteration. We also plan to improve our MED to make it converge faster, thereby reducing the number of iterations and computation time.

References

1. `http://perso.uclouvain.be/fstandae/PUBLIS/171.zip`.
2. F. Bache, C. Plump, and T. Güneysu. Confident leakage assessment - A side-channel evaluation framework based on confidence intervals. In *2018 Design, Automation & Test in Europe Conference & Exhibition, DATE 2018, Dresden, Germany, March 19-23, 2018*, pages 1117–1122, 2018.
3. S. Bhasin, J. Danger, S. Guilley, and Z. Najm. Side-channel leakage and trace compression using normalized inter-class variance. In *HASP 2014, Hardware and Architectural Support for Security and Privacy, Minneapolis, MN, USA, June 15, 2014*, pages 7:1–7:9, 2014.
4. E. Brier, C. Clavier, and F. Olivier. Correlation power analysis with a leakage model. In *Cryptographic Hardware and Embedded Systems - CHES 2004: 6th International Workshop Cambridge, MA, USA, August 11-13, 2004. Proceedings*, pages 16–29, 2004.
5. S. Chari, J. R. Rao, and P. Rohatgi. Template attacks. In *Cryptographic Hardware and Embedded Systems - CHES 2002, 4th International Workshop, Redwood Shores, CA, USA, August 13-15, 2002, Revised Papers*, pages 13–28, 2002.
6. Y. L. Corre, J. Großschädl, and D. Dinu. Micro-architectural power simulator for leakage assessment of cryptographic software on ARM cortex-m3 processors. In *Constructive Side-Channel Analysis and Secure Design - 9th International Workshop, COSADE 2018, Singapore, April 23-24, 2018, Proceedings*, pages 82–98, 2018.

7. A. A. Ding, C. Chen, and T. Eisenbarth. Simpler, faster, and more robust t-test based leakage detection. In *Constructive Side-Channel Analysis and Secure Design - 7th International Workshop, COSADE 2016, Graz, Austria, April 14-15, 2016, Revised Selected Papers*, pages 163–183, 2016.
8. F. Durvaux and F. Standaert. From improved leakage detection to the detection of points of interests in leakage traces. In *Advances in Cryptology - EUROCRYPT 2016 - 35th Annual International Conference on the Theory and Applications of Cryptographic Techniques, Vienna, Austria, May 8-12, 2016, Proceedings, Part I*, pages 240–262, 2016.
9. F. Durvaux, F. Standaert, and S. M. D. Pozo. Towards easy leakage certification. In *Cryptographic Hardware and Embedded Systems - CHES 2016 - 18th International Conference, Santa Barbara, CA, USA, August 17-19, 2016, Proceedings*, pages 40–60, 2016.
10. F. Durvaux, F. Standaert, and N. Veyrat-Charvillon. How to certify the leakage of a chip? In *Advances in Cryptology - EUROCRYPT 2014 - 33rd Annual International Conference on the Theory and Applications of Cryptographic Techniques, Copenhagen, Denmark, May 11-15, 2014. Proceedings*, pages 459–476, 2014.
11. F. Florent, G. Sylvain, D. Jean-Luc, E. M, M. Houssem, and S. Laurent. About probability density function estimation for side channel analysis. In *Constructive Side-Channel Analysis and Secure Design - 1st International Workshop, COSADE 2010, 2010*, pages 15–23, 2010.
12. S. Geman and D. Geman. Stochastic relaxation, gibbs distributions, and the bayesian restoration of images. *IEEE Trans. Pattern Anal. Mach. Intell.*, 6(6):721–741, 1984.
13. B. Gierlichs, L. Batina, P. Tuyls, and B. Preneel. Mutual information analysis. In *Cryptographic Hardware and Embedded Systems - CHES 2008, 10th International Workshop, Washington, D.C., USA, August 10-13, 2008. Proceedings*, pages 426–442, 2008.
14. I. ISO and B. OIML. Guide to the expression of uncertainty in measurement. *Geneva, Switzerland*, 1995.
15. E. T. Jaynes. Information theory and statistical mechanics. *Physical review*, 106(4):620, 1957.
16. A. Journault and F. Standaert. Very high order masking: Efficient implementation and security evaluation. In *Cryptographic Hardware and Embedded Systems - CHES 2017 - 19th International Conference, Taipei, Taiwan, September 25-28, 2017, Proceedings*, pages 623–643, 2017.
17. P. C. Kocher, J. Jaffe, and B. Jun. Differential power analysis. In *Advances in Cryptology - CRYPTO '99, 19th Annual International Cryptology Conference, Santa Barbara, California, USA, August 15-19, 1999, Proceedings*, pages 388–397, 1999.
18. S. Mangard, E. Oswald, and T. Popp. *Power analysis attacks - revealing the secrets of smart cards*. Springer, 2007.
19. A. Mohammad-Djafari. A matlab program to calculate the maximum entropy distributions. In *Maximum Entropy and Bayesian Methods*, pages 221–233. Springer, 1992.
20. A. Moradi. Statistical tools flavor side-channel collision attacks. In *Advances in Cryptology - EUROCRYPT 2012 - 31st Annual International Conference on the Theory and Applications of Cryptographic Techniques, Cambridge, UK, April 15-19, 2012. Proceedings*, pages 428–445, 2012.

21. A. Moradi, B. Richter, T. Schneider, and F.-X. Standaert. Leakage detection with the χ^2-test. In *Cryptographic Hardware and Embedded Systems - CHES 2018 - 20th International Conference, Taipei, Taiwan, September 25-28, 2018, Proceedings*, pages 209–237, 2018.
22. A. Moradi and F. Standaert. Moments-correlating DPA. In *Proceedings of the ACM Workshop on Theory of Implementation Security, TIS@CCS 2016 Vienna, Austria, October, 2016*, pages 5–15, 2016.
23. A. Moradi and A. Wild. Assessment of hiding the higher-order leakages in hardware - what are the achievements versus overheads? In *Cryptographic Hardware and Embedded Systems - CHES 2015 - 17th International Workshop, Saint-Malo, France, September 13-16, 2015, Proceedings*, pages 453–474, 2015.
24. E. Peeters. *Advanced DPA Theory and Practice*. Springer New York, 2013.
25. E. Peeters, F. Standaert, N. Donckers, and J. Quisquater. Improved higher-order side-channel attacks with FPGA experiments. In *Cryptographic Hardware and Embedded Systems - CHES 2005, 7th International Workshop, Edinburgh, UK, August 29 - September 1, 2005, Proceedings*, pages 309–323, 2005.
26. C. Rechberger and E. Oswald. Practical template attacks. In *Information Security Applications, 5th International Workshop, WISA 2004, Jeju Island, Korea, August 23-25, 2004, Revised Selected Papers*, pages 440–456, 2004.
27. M. Renauld, F. Standaert, N. Veyrat-Charvillon, D. Kamel, and D. Flandre. A formal study of power variability issues and side-channel attacks for nanoscale devices. In *Advances in Cryptology - EUROCRYPT 2011 - 30th Annual International Conference on the Theory and Applications of Cryptographic Techniques, Tallinn, Estonia, May 15-19, 2011. Proceedings*, pages 109–128, 2011.
28. O. Reparaz. Detecting flawed masking schemes with leakage detection tests. In *Fast Software Encryption - 23rd International Conference, FSE 2016, Bochum, Germany, March 20-23, 2016, Revised Selected Papers*, pages 204–222, 2016.
29. O. Reparaz, B. Gierlichs, and I. Verbauwhede. Fast leakage assessment. In *Cryptographic Hardware and Embedded Systems - CHES 2017 - 19th International Conference, Taipei, Taiwan, September 25-28, 2017, Proceedings*, pages 387–399, 2017.
30. W. Schindler, K. Lemke, and C. Paar. A stochastic model for differential side channel cryptanalysis. In *Cryptographic Hardware and Embedded Systems - CHES 2005, 7th International Workshop, Edinburgh, UK, August 29 - September 1, 2005, Proceedings*, pages 30–46, 2005.
31. T. Schneider and A. Moradi. Leakage assessment methodology - A clear roadmap for side-channel evaluations. In *Cryptographic Hardware and Embedded Systems - CHES 2015 - 17th International Workshop, Saint-Malo, France, September 13-16, 2015, Proceedings*, pages 495–513, 2015.
32. D. W. Scott. On optimal and data-based histograms. *Biometrika*, 66(3):605–610, 1979.
33. A. Shamilov, C. Giriftinoglu, I. Usta, and Y. M. Kantar. A new concept of relative suitability of moment function sets. *Applied Mathematics and Computation*, 206(2):521–529, 2008.
34. A. Shamilov, I. Usta, and Y. M. Kantar. Performance of maximum entropy probability density in the case of data which are not well distributed. In *Proceedings of the 6th WSEAS International Conference on Simulation, Modelling and Optimization, Lisbon, Portugal, September 22 - 24, 2006, Proceedings*, pages 361–364, 2006.
35. Shannon. The mathematical theory of communication. *Bell System Technical Journal*, 27, 1948.

36. M. Srikanth, H. K. Kesavan, and P. H. Roe. Probability density function estimation using the minmax measure. *IEEE Trans. Systems, Man, and Cybernetics, Part C*, 30(1):77–83, 2000.
37. F. Standaert, F. Koeune, and W. Schindler. How to compare profiled side-channel attacks? In *Applied Cryptography and Network Security, 7th International Conference, ACNS 2009, Paris-Rocquencourt, France, June 2-5, 2009. Proceedings*, pages 485–498, 2009.
38. F. Standaert, T. Malkin, and M. Yung. A unified framework for the analysis of side-channel key recovery attacks. In *Advances in Cryptology - EUROCRYPT 2009, 28th Annual International Conference on the Theory and Applications of Cryptographic Techniques, Cologne, Germany, April 26-30, 2009. Proceedings*, pages 443–461, 2009.
39. A. Venelli. Efficient entropy estimation for mutual information analysis using b-splines. In *Information Security Theory and Practices. Security and Privacy of Pervasive Systems and Smart Devices, 4th IFIP WG 11.2 International Workshop, WISTP 2010, Passau, Germany, April 12-14, 2010. Proceedings*, pages 17–30, 2010.
40. M. Wand. Data-based choice of histogram bin width. *The American Statistician*, 51(1):59–64, 1997.
41. F. Xinghua and S. Mingshun. Estimation of maximum-entropy distribution based on genetic algorithms in evaluation of the measurement uncertainty. In *Intelligent Systems (GCIS), 2010 Second WRI Global Congress on*, volume 1, pages 292–297. IEEE, 2010.
42. A. Zellner and R. A. Highfield. Calculation of maximum entropy distributions and approximation of marginalposterior distributions. *Journal of Econometrics*, 37(2):195–209, 1988.