A Unified Metric for Quantifying Information Leakage of Cryptographic Devices under Power Analysis Attacks

Liwei Zhang, A. Adam Ding, Yunsi Fei, and Pei Luo

[1] Department of Mathematics, Northeastern University, Boston, MA 02115
[2] Department of Electrical and Computer Engineering
Northeastern University, Boston, MA 02115
zhang.liw@husky.neu.edu,a.ding@neu.edu,
yfei@ece.neu.edu,silenceluo@coe.neu.edu

Abstract. To design effective countermeasures for cryptosystems against side-channel power analysis attacks, the evaluation of the system leakage has to be lightweight and often times at the early stage like on cryptographic algorithm or source code. When real implementations and power leakage measurements are not available, security evaluation has to be through metrics for the information leakage of algorithms. In this work, we propose such a general and unified metric, information leakage amount - ILA. ILA has several distinct advantages over existing metrics. It unifies the measure of information leakage to various attacks: first-order and higher-order DPA and CPA attacks. It works on algorithms with no mask protection or perfect/imperfect masking countermeasure. It is explicitly connected to the success rates of attacks, the ultimate security metric on physical implementations. Therefore, we believe ILA is an accurate indicator of the side-channel security level of the physical system, and can be used during the countermeasure design stage effectively and efficiently for choosing the best countermeasure.

Keywords: Information leakage amount, side-channel security, power analysis attack

1 Introduction

In the past decade, various side channel attacks (SCAs) utilizing the system power consumption information, such as differential power analysis (DPA) [16], correlation power analysis (CPA) [5], mutual information (MI) attacks [14] and template attacks [6], have been presented to exploit the weakness in cryptographic implementations to recover the secret key. Masking is one of the most popular SCA countermeasures used to randomize sensitive variables [7]. When applying masking at a higher level, e.g., algorithmic or source code level, every key-sensitive intermediate variable is masked with at least one random value M by a carefully designed masking function f, e.g., normally exclusive OR or

multiplication. Therefore, during the cryptographic execution, any intermediate variable Z is substituted by its masked counterpart, $f(Z, M)$, to prevent side-channel leakage. Perfectly masked devices with appropriate masking functions and unbiased random masks can eliminate first-order leakage, e.g., it is not feasible to break the system by exploiting only one time point of the power leakage traces which corresponds to one intermediate variable. However, they are still susceptible to second-order and higher-order attacks which combine two or more time points of power leakage to retrieve the secret key. Some practical masking schemes with limited implementation resources are not perfect and may still have some first-order leakage.

How to evaluate a system's SCA vulnerability/resilience comprehensively and accurately under different attacks is an important research issue. Sound quantitative metrics will be used to guide the implementation of countermeasures and fairly compare the overall strength of countermeasures. One widely used metric is success rate, the probability that an attack succeeds given a number of side-channel leakage measurements [21]. This is indeed the ultimate practical measure of a system's SCA vulnerability/resilience, which depends on the cryptographic algorithm, the specific implementation (with power measurement data available), and the attack model (whether it is DPA, CPA, MIA, etc.) as illustrated in [18, 12]. We classify this metric as one for measuring the system *physical leakage*. In recent years, there are research interests in using other physical leakage metrics on instructions of cryptographic software and therefore pinpoint the location of leakage to guide automatic implementation of countermeasures. Bayrak et al. [2] introduced a methodology for detecting power leakage, using an information theoretic metric - mutual information, $\mathrm{MI_L}$, between the key and leakage measurements. Although not explicitly related to the success rate, the metric $\mathrm{MI_L}$ can be used to bound the success rate [21, 10] in some models. However, it requires power consumption data. There are also other efforts in evaluating the cryptosystem *information leakage* at an early stage, i.e., on source code of cipher software or even algorithms and with no need of power measurement data. The automatic software verification tools for SCA vulnerabilities [3, 8] employ mutual information between the secret key and intermediate variables, denoted as $\mathrm{MI_A}$. The metric of quantitative masking strength, QMS, is defined by [11] to quantify the software leakage amount under imperfect masking, and a verification process is formulated to find the QMS value of cryptographic software source code. However, none of the prior work has shown the relationship between these system information leakage metrics and the success rate. It is not easy to translate the bound on these information leakage ($\mathrm{MI_A}$ and QMS) to the final security measure of the implemented physical system, the success rate.

In this work, we propose a new unified metric, information leakage amount (ILA), to quantify the *system information leakage* under various power analysis attacks at the early cryptographic algorithm or software code level, whether the cipher is unprotected or protected with masking. What is more, we also relate this metric to the success rate of DPA/CPA attacks in analytic models. Note that in this work we choose DPA/CPA because it has been shown both theoretically

and empirically that the first-order and second-order CPA attacks are equivalent to the strongest maximum-likelihood attacks under Gaussian noise models [13, 9, 15]. Our metric is unified, in the sense that it works on original algorithms with no masking, perfect masking, or imperfect masking under first-order DPA/CPA or second-order CPA. The success rate formulas are more general and simpler than the formulas in [12, 13, 9], which are only for first-order DPA/CPA on unmasked devices and for higher-order attacks on perfectly masked devices. Our explicit success rate formulas in terms of ILA bridge the gap between the system information leakage measure and the physical leakage measure. The metric ILA, as a great indicator of the ultimate side-channel security level of the physical system, can therefore be used during the countermeasure design stage (without real implementations and power measurements) effectively and efficiently for choosing the best countermeasure.

Table 1 summarizes the properties of our metric and compares it with other three metrics, QMS, $\mathrm{MI_A}$, and $\mathrm{MI_L}$. A question mark means that the metric on the column may be able to achieve the objective on the row, but it has not been demonstrated in literature. For example, work in [21] shows that the mutual information $\mathrm{MI_L}$ has a monotonic relationship with the success rate of an attack with only two candidate keys, but generally the $\mathrm{MI_L}$ may not be converted to the success rate explicitly.

Table 1. Comparison among ILA, QMS and MI as leakage evaluation metrics

		ILA	QMS	$\mathrm{MI_A}$	$\mathrm{MI_L}$
1.	First-order DPA/CPA Metric on Software Code/Algorithm	√	√	√	×
2.	Relate to First-order DPA Success Rate	√	√	√	√
3.	Relate to First-order CPA Success Rate	√	×	?	?
4.	Second-order DPA/CPA Metric on Software Code/Algorithm	√	×	?	×
5.	Relate to Second-order DPA/CPA Success Rate	√	×	?	?

The rest of the paper is organized as follows. Section 2 gives an overview of the existing leakage metrics and defines our proposed metric. Section 3 establishes the success rate formula for CPAs in terms of our metrics. Section 4 presents experimental results to evaluate the metrics and compare them with others. Section 5 concludes the paper.

2 Leakage Metrics for Cryptosystems with Masking Countermeasure

In this section, we first introduce the notations used and existing metrics, and propose our unified metric ILA for first-order and second-order attacks on cryptographic algorithm with imperfect/perfect masking. We then analyze these metrics in the case of Boolean masking.

2.1 Notations and Existing First-Order Metrics

We denote sets by calligraphic letters (e.g., $\mathcal{X}$), denote random variables by capital letters (e.g., X) which take values on the set ($\mathcal{X}$), and denote observations of the random variables by lowercase letters (e.g., x). We let $X_{(i)}$ denote the ith bit of X. $\mathbb{P}_X$ and $\mathbb{E}_X$ are the notations for the probability and the expectation with respect to X, respectively. For a cryptographic system with masking protection, K, X, M denote the random variables for the key, the plaintext, and the mask, respectively, and each takes values in sets $\mathcal{X}$, $\mathcal{K}$, $\mathcal{M}$. Let $F = f(X, K, M)$ denote the algorithmic intermediate variable that possibly leaks, which is an algorithmic function of the known input X, unknown key K and the random mask M. For a second-order attack on masked devices, there are two select functions. One is $V_0(X, K, M) = g(F)$, which works on a key-sensitive intermediate variable and therefore is also a function of the input X, the key K and the mask M. Note the select function for an attack is determined by the system's power model, and $g(\cdot)$ is usually Hamming weight or Hamming distance. Without loss of generality, the other select function is $V_1 = g(M)$ which depends on the mask M only. The mask may be biased, i.e., not following the uniform distribution. If the mask is unbiased and the masking operation is appropriate, we call it perfect masking. Let k_c be the secret key, $k_g \in \mathcal{K}\backslash\{k_c\}$ be any other possible key hypothesis, and $N_k = |\mathcal{K}|$ be the dimension of the key set.

A first-order attack uses only one select function V_0 that corresponds to one time point on power traces. Therefore a *first-order leakage metric* measures the leakage of one select function that can be sensitive to both key and mask. Given a plaintext x, the secret key k_c and a random number m, the target select function is $v_0^c = V_0(x, k_c, m)$. The information leakage is measured by the dependency of v_0^c on k_c. Under perfect masking, the distribution of v_0^c is independent of k_c, and hence the secret key could not be recovered from the leakage measurements of v_0^c. Otherwise, v_0^c is still vulnerable to first-order power analysis attacks. There are mainly two existing first-order information leakage metrics.

Eldib et al. [11] proposed to quantify the masking strength under DPA by

$$\text{QMS} = (1 - \Delta_{qms}), \quad \text{with } \Delta_{qms} = \max_{x,x' \in \mathcal{X}, k,k' \in \mathcal{K}} |D_{x,k}(F) - D_{x',k'}(F)|, \quad (1)$$

where $D_{x,k}$ denotes the distribution of F given (x, k), and Δ_{qms} is the maximum distribution difference. For perfect masking, QMS is maximum and reaches one, which indicates that the key K and the intermediate variable F are statistically independent. Without masking, QMS=0. For imperfect masking schemes, QMS is in the range of (0,1).

The other metric uses the mutual information, an information theoretic quantity commonly used for leakage evaluation. The mutual information between two discrete random variables X and Y is defined as:

$$\text{MI}(X, Y) = \sum_{x \in \mathcal{X}, y \in \mathcal{Y}} p(x, y) \log \left(\frac{p(x, y)}{p(x)p(y)} \right), \quad (2)$$

where $p(x, y)$ is the joint probability distribution function of X and Y, with $p(x)$ and $p(y)$ as the corresponding marginal functions. For continuous random

variables, the summation in definition (2) is replaced by integrations. Work in [3, 8] uses the mutual information between K and F to measure the information leakage at the software code level. This mutual information only depends on the algorithm and we denote it by $\mathrm{MI_A} = \mathrm{MI}(K, F)$. In contrast, work in [2] uses the mutual information between K and the leakage measurements L. We denote it by $\mathrm{MI_L} = \mathrm{MI}(K, L)$, which is a physical leakage measure.

Note that there is no second-order system information leakage metric based on QMS or MI shown in literature. In this work, we propose a general and unified metric on the selection functions (V_0 for first-order attacks, V_0 and V_1 for second-order attacks), which reflects the system susceptibility to attacks.

2.2 Our Proposed Information Leakage Metric

Eldib et al. empirically [11] showed that there is a relationship between QMS and the number of traces needed in DPA. However, there is no theoretical proof for such relation, and how QMS relates with multi-bit CPA or higher-order attacks is unknown. We are seeking a new unified metric to reflect the information leakage at the algorithm level, similar to QMS and $\mathrm{MI_A}$, and meanwhile can explicitly relate to the success rate of different attacks, including DPA, CPA, and high-order attacks.

Fei et al. [13] defined the confusion coefficient, for unmasked algorithm, as $\kappa(k_c, k_g) = \mathbb{E}_X\{[V(X, k_c) - V(X, k_g)]^2\}$ for the selection function V and the expectation being taken over X. Each confusion coefficient is defined between two key values. They showed that the confusion coefficients and the implementation signal-noise-ratio (SNR) together explicitly determine the success rates for DPA and CPA. However, these confusion coefficients do not reflect the masking strength as they are defined for unmasked algorithms only. The confusion coefficients are also used to model the success rates for higher-order attacks with perfect masking in [9].

We propose to generalize the confusion coefficient definition for masked algorithms (possibly imperfect). We then propose the new metric ILA based on the generalized confusion coefficients. The ILA measures the information leakage of V_0 (and V_1) under the protection of any masking countermeasure.

Definition 1. *We define the new first-order confusion coefficient $\kappa_{1O}(k_c, k_g)$ of masked algorithm as*

$$\kappa_{1O}(k_c, k_g) = \mathbb{E}_X\{[\mathbb{E}_M(V_0|(X, k_c)) - \mathbb{E}_M(V_0|(X, k_g))]^2\}, \tag{3}$$

where $\mathbb{E}_M(V_0|(X, k))$ is the conditional expectation of V_0 given (X, k) over $\mathcal{M}$, and $\mathbb{E}_X$ is the expectation over $\mathcal{X}$.

Definition 2. *The first-order information leakage amount $\mathrm{ILA_{1O}}$ is defined as*

$$\mathrm{ILA_{1O}} = \mathbb{E}_{\mathcal{K}\backslash\{k_c\}}[\kappa_{1O}(k_c, k_g)], \tag{4}$$

where $\mathbb{E}_{\mathcal{K}\backslash\{k_c\}}$ is the expectation over all possible key hypothesis k_g in $\mathcal{K}\backslash\{k_c\}$.

$\mathrm{ILA}_{1\mathrm{O}}$, MI_{A} and QMS are all metrics for sensitivity evaluation at the algorithm level that do not require leakage measurements. QMS focuses on the extreme value among the differences of distributions of any pair $(x,k),(x',k') \in (\mathcal{X},\mathcal{K})$, but ignores the other differences. The extreme value indicates the probability distance between the secret key to the one guessed key which is easiest to distinguish. However, the SCA succeeds only if the secret key is distinguished from all other guessed keys, not just one. Hence the expectation would be a better measure for information leakage than the extreme value. We can see that $\mathrm{ILA}_{1\mathrm{O}}$ is an expectation of squared distances:

$$\begin{aligned}\mathrm{ILA}_{1\mathrm{O}} &= \sum_{k_g \in \mathcal{K}\backslash\{k_c\}} p(k_g)\kappa_{1O}(k_c,k_g) \\ &= \sum_{k_g \in \mathcal{K}\backslash\{k_c\}} p(k_g) \sum_{x\in\mathcal{X}} p(x)\cdot\{\mathbb{E}_M[V_0|(x,k_c)] - \mathbb{E}_M[V_0|(x,k_g)]\}^2. \end{aligned} \tag{5}$$

The calculation of $\mathrm{ILA}_{1\mathrm{O}}$ through equation (5) involves iterations over $k_g \in \mathcal{K}\backslash k_c$ and $x \in \mathcal{X}$, which can be time-consuming for large sets of $\mathcal{K}$ and $\mathcal{X}$. These same iterations appear in MI_{A} and QMS definitions too. As recommended for MI calculations by [2, 8], the exhaustive iterations in calculating $\mathrm{ILA}_{1\mathrm{O}}$ can be replaced by averaging over a random subset of sufficiently large size. Thus the computational complexity is similar for the three metrics $\mathrm{ILA}_{1\mathrm{O}}$, MI_{A} and QMS.

Different from MI_{A} and QMS, we find that $\mathrm{ILA}_{1\mathrm{O}}$ can be related to the success rates of DPA and CPA in explicit formulas, similar to the work in [13]. In addition, $\mathrm{ILA}_{1\mathrm{O}}$ can be extended to a second-order metric $\mathrm{ILA}_{2\mathrm{O}}$ as well, while there is no such work on MI_{A} and QMS yet.

A second-order attack retrieves the secret key by combining the information leakage at two leakage points, $V_0(X,K,M)$ and $V_1(M)$. A second-order metric measures the leakage under second-order CPA attacks.

Definition 3. *For a key hypothesis $k_g \in \mathcal{K}\backslash\{k_c\}$, we define the second-order confusion coefficient of masked algorithm as*

$$\kappa_{2O}(k_c,k_g) = \mathbb{E}_X\{[\mathbb{E}_M(\widetilde{V_0}\widetilde{V_1}|(X,k_c)) - \mathbb{E}_M(\widetilde{V_0}\widetilde{V_1}|(X,k_g))]^2\}, \tag{6}$$

where $\widetilde{V_i} = V_i - \mathbb{E}_{X,M}[V_i], i = 0,1$, are the centered select function values.

Definition 4. *The second-order information leakage amount* $\mathrm{ILA}_{2\mathrm{O}}$ *is defined as*

$$\mathrm{ILA}_{2\mathrm{O}} = \mathbb{E}_{\mathcal{K}\backslash\{k_c\}}[\kappa_{2O}(k_c,k_g)]. \tag{7}$$

Comment: Although the definitions (4) and (7) of ILA depend on the correct key k_c, in many practical situations ILA is key-independent. The leaked intermediate values often depend on key k_c only through $X \oplus k_c$. In that case, for uniformly distributed plaintext X, the ILA is in fact independent of k_c since $k_g \oplus k_c$ iterates over the same values for all k_c.

2.3 Analysis of the Metrics Under Boolean Masking

To better understand the metrics ILA, $\mathrm{MI_A}$ and QMS, we compare them in detail for a specific setting of biased Boolean masking $F = Z(X,k) \oplus M$ as in [11], under several commonly used assumptions on the distribution of unmasked $Z(X,k)$ and keys. Here $Z(X,k)$ denotes an unmasked intermediate variable with X being the random plaintext. Hence $V_0 = g(Z(X,k) \oplus M)$.

Assumption 1 *(Uniform Intermediate Variable) Given a key $k \in \mathcal{K}$, for random plaintext X, the unmasked intermediate variable $Z(X,k)$ is uniformly distributed. That is, $Z(X,k) \sim U(0, 2^b-1)$, for all $k \in \mathcal{K}$, where $U(0, 2^b-1)$ denotes the discrete uniform distribution on $\{0, 1, ..., 2^b - 1\}$ with b being the number of bits for $Z(X,k)$.*

Let $V_0^*(X,k) = g(Z(X,k))$ denote the unmasked select function. Under Assumption 1, $\mathbb{E}_X[V_0^*(X,k)]$ is a constant independent of keys k. In general, we would like the unmasked select function values under two different keys to be uncorrelated.

Assumption 2 *(Uncorrelated Keys) For any pair of keys $k_1, k_2 \in \mathcal{K}$, and random plaintext X, the select functions $V_0^*(X,k_1)$ and $V_0^*(X,k_2)$ are uncorrelated so that $\mathbb{E}_X[V_0^*(X,k_1)V_0^*(X,k_2)] = \mathbb{E}_X[V_0^*(X,k_1)]\mathbb{E}_X[V_0^*(X,k_2)]$.*

Under Assumptions 1 and 2, $\mathbb{E}_X[V_0^*(X,k_1)V_0^*(X,k_2)] = \{\mathbb{E}_X[V_0^*(X,k_1)]\}^2$ will also be a constant independent of keys k_1 and k_2. Unfortunately, many select functions (e.g., the Hamming weights of an AES S-Box output) do not satisfy Assumption 2. However, for a random key k_2, a weaker assumption often holds.

Assumption 3 *(Weak Uncorrelated Keys) For any fixed key k_1, let k_2 be a random key $\in \mathcal{K}\backslash\{k_1\}$. For a random plaintext X, the intermediate variables $Z(X,k_1)$ and $Z(X,k_2)$ are uncorrelated so that $\mathbb{E}_{X,k_2}[V_0^*(X,k_1)V_0^*(X,k_2)] = \{\mathbb{E}_X[V_0^*(X,k_1)]\}^2$.*

Under Assumptions 1 and 3, $\mathbb{E}_{X,k_2}[V_0^*(X,k_1)V_0^*(X,k_2)]$ is a constant, which helps us to derive simple explicit formulas of ILA in this section. Assumption 3 makes the calculation of the metrics easier here, as it removes ILA's dependence on many aspects of the algorithm including k_c value. The leakage metrics ILA under these assumptions reflect the masking strength only. In the next section, Assumption 3 will not be assumed for DPA/CPA success rates derivations though.

We first consider the DPA attack, where V_0 is on a single bit. Since $\oplus$ is taken bit by bit, we can take both $Z(X,k_c)$ and M as variables with one single bit, and $V_0 = Z \oplus M$. Let the distribution of the mask bit be $\mathbb{P}(M=1) = p$ and $\mathbb{P}(M=0) = 1-p$, we have the following property.

Property 1 *For the DPA model under Assumptions 1 and 3, if $\mathbb{P}(M=1) = p$, then*

- $\mathrm{ILA_{1O}} = (1-2p)^2/2$,

- $\mathrm{ILA}_{2\mathrm{O}} = 2p^2(1-p)^2$,
- $\mathrm{MI}_{\mathrm{A}} = 1 + (1-p)\log_2(1-p) + p\log_2(p)$,
- $QMS = 1 - |1-2p|$.

The detailed calculations are given in Appendix A. Note that although the generalized confusion coefficients $\kappa_{1O}(k_c, k_g)$ (Equation 3) and $\kappa_{2O}(k_c, k_g)$ (Equation 6) are determined by the algorithm, their average terms $\mathrm{ILA}_{1\mathrm{O}}$ and $\mathrm{ILA}_{2\mathrm{O}}$ become algorithm-independent and are only determined by the bias of the mask distribution, p, according to Assumption 3. For perfect masking, $p = 1/2$; unmasked, $p = 0$ or $p = 1$; imperfect masking, p takes other values. All metrics change with p and have one-to-one correspondence between each other. Particularly, $\mathrm{ILA}_{1\mathrm{O}} = (1 - \mathrm{QMS})^2/2$. Work in [11] empirically finds that the number of traces needed for DPA is approximately $\mathrm{N}_{\mathrm{trace}} = 1/(1-\mathrm{QMS})^{2.2}$. In Section 4.1, we will show that number of traces $\mathrm{N}_{\mathrm{trace}} \propto 1/\mathrm{ILA}_{1\mathrm{O}} \propto 1/(1-\mathrm{QMS})^2$ instead.

Fig. 1 shows the relationship between these metrics and the probability p. It is symmetric about the x-axis which implies the same effect of the mask bit being 0 and 1. From Fig. 1, we see that $\mathrm{ILA}_{1\mathrm{O}}$ and MI_{A} have the same pattern, but $\mathrm{ILA}_{1\mathrm{O}}$ increases from 0 to 1/2 and MI_{A} increases from 0 to 1 as p goes from 1/2 to 0 (or 1). When $p = 0$ or $p = 1$, the device is without any masking protection, $\mathrm{QMS} = 0$ while $\mathrm{ILA}_{1\mathrm{O}}$ and MI_{A} both reach their maximum. When $p = 1/2$, the devices is protected by perfect masking, $\mathrm{ILA}_{1\mathrm{O}} = \mathrm{MI}_{\mathrm{A}} = 0$ and $\mathrm{QMS} = 1$ which are consistent with no first-order information leakage. However, the second-order leakage still exists under perfect masking, and actually reaches its maximum (biggest leakage) 1/8. As the mask gets more biased, the first-order leakage increases while the second-order leakage decreases.

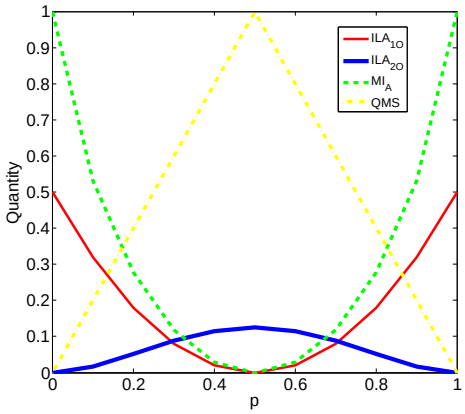

Fig. 1. The quantities of several metrics under the biased masking for DPA.

Next we consider CPA in this setting. For CPA, $V_0 = HW(Z \oplus M)$ is the Hamming weight function of a b-bit variable. We assume that the bits in the mask

are independent from the same distribution with $\mathbb{P}(M_{(i)} = 1) = p$, $i = 1, ..., b$. Here $M_{(i)}$ denotes the ith bit of the b-bit mask variable M.

Property 2 *For the CPA model under Assumptions 1 and 3,*

$$\mathrm{ILA}_{1\mathrm{O}} = b(1-2p)^2/2, \qquad \mathrm{ILA}_{2\mathrm{O}} = 2bp^2(1-p)^2. \tag{8}$$

The proof is provided in Appendix B.

For the CPA model, the $\mathrm{ILA}_{1\mathrm{O}}$ and $\mathrm{ILA}_{2\mathrm{O}}$ follow the similar pattern as in the DPA model, just differing by a factor of b, the number of bits. In fact, the DPA model is a special case of the CPA model with $b = 1$. The other two metrics MI_{A} and QMS are harder to derive for CPA. It is hard, if not impossible, to relate MI_{A} and QMS to the success rate of CPA.

3 Relating ILA to DPA and CPA Success Rates

As shown in [12, 13, 9], the success rates of first-order DPA and CPA on unmasked devices and second-order CPA on perfectly masked devices can all be expressed in terms of the confusion coefficients and the implementation signal-to-noise-ratio (SNR). Our metrics $\mathrm{ILA}_{1\mathrm{O}}$ and $\mathrm{ILA}_{2\mathrm{O}}$ are algorithmic properties like the confusion coefficients. We generalize the results of [12, 13, 9] to masked implementations (possibly with imperfect masking), and show that the success rates of CPA/DPA should be determined by the SNRs and our generalized confusion coefficients. The formulas are further simplified to consist of $\mathrm{ILA}_{1\mathrm{O}}$ and $\mathrm{ILA}_{2\mathrm{O}}$. We show derivations for the success rates of first-order and second-order DPA and CPA on masked devices in this section. We then use these metrics to compare the first-order leakage and second-order leakage.

3.1 First-Order Power Analysis Attack Model

We assume a commonly used linear power consumption model with additive noises for both DPA and CPA,

$$L_0 = c_0 + \epsilon_0 V_0 + \sigma_0 r_0, \tag{9}$$

where r_0 is the unit noise variable (the mean is 0 and the variance is 1) and ϵ_0 is the single-bit unit power consumption. Hence the physical system SNR is $\delta_0 = \epsilon_0/\sigma_0$. We derive the success rate formulas for first-order CPA in terms of SNR and $\mathrm{ILA}_{1\mathrm{O}}$, and consider DPA as a special case of CPA with $b = 1$. Notice that some other researchers defined SNR differently as $\mathrm{SNR}^* = \epsilon_0^2 Var(V_0)/\sigma_0^2$, which includes the variance of intermediate value V_0 also. We consider $Var(V_0)$ to be part of algorithmic leakage measured by $\mathrm{ILA}_{1\mathrm{O}}$, since it depends on V_0. Our SNR reflects purely the physical system property, since ϵ_0 reflects the power consumption differential caused by one-bit.

The leakage measurements of L_0 are denoted as $\mathcal{L} = \{l_{1,0}, l_{2,0}, ..., l_{n,0}\}$, where n is the number of traces. For unmasked devices, the CPA exploits the correlation between the leakage L and unmasked select function $V_0^* = g(Z(X, k))$

to discover the secret key. For masked devices, the attacker does not know M value, and therefore does not know the value of $V_0 = g(F(X, k, M))$. To conduct CPA, the attacker has to correlate L with $\mathbb{E}_M[V_0|(X, k, M)]$, the expectation of V_0 over all possible mask values. This value is $V_0^*(X, k)$ for unmasked devices, and is a constant (thus no leakage) for perfectly masked devices. Let $v_{m,i,0}^g$ denote $V_0(x_i, k_g, m_i)$ for the i-th power trace, the selection function value under plaintext x_i, guess key k_g and the mask m_i, $\mathbb{E}_M[v_{m,i,0}^g]$ denote the targeted expectation of $V_0(x_i, k_g, m)$ over all $m \in \mathcal{M}$, and $\mathbb{E}[V_0^g]$ denote the expectation of $V_0(x, k_g, m)$ over all $x \in \mathcal{X}$ and $m \in \mathcal{M}$. Under the power model (9) with imperfect masking, the first-order CPA distinguishes the key k_g by the Pearson's correlation:

$$\hat{\rho}^g = \frac{\sum_{i=1}^{n}(l_{i,0} - \bar{l}_{.,0})[\mathbb{E}_M(v_{m,i,0}^g) - \mathbb{E}(V_0^g)]}{\sqrt{\sum_{i=1}^{n}(l_{i,0} - \bar{l}_{.,0})^2 \sum_{i=1}^{n}[\mathbb{E}_M(v_{m,i,0}^g) - \mathbb{E}(V_0^g)]^2}}, \tag{10}$$

where $\bar{l}_{.,0} = \sum_{i=1}^{n} l_{i,0}/n$ is the mean of power leakage.

The CPA succeeds when $\hat{\rho}^c - \hat{\rho}^g > 0$ for all $k_g \in \mathcal{K}\backslash\{k_c\}$. For a random plaintext attack with a large number of traces, under Assumption 1, the denominator of (10) converges to the same limit for all k_g, since $\mathbb{E}[\mathbb{E}_M(v_{m,i,0}^g)] = \mathbb{E}(V_0^g) = \mathbb{E}(V_0^c)$ and $\mathbb{E}\{\mathbb{E}_M[(v_{m,i,0}^g)^2]\} = \mathbb{E}\{\mathbb{E}_M[(v_{m,i,0}^c)^2]\}$. Hence $\hat{\rho}^c - \hat{\rho}^g > 0$ is equivalent to that the difference in the numerators of (10) is positive. That is, $\hat{\rho}^c - \hat{\rho}^g > 0$ when $\Delta_n^{1O}(k_c, k_g) > 0$, where

$$\Delta_n^{1O}(k_c, k_g) = \sum_{i=1}^{n} \frac{(l_{i,0} - \bar{l}_{.,0})}{\sigma_0}[\mathbb{E}_M(v_{m,i,0}^c) - \mathbb{E}_M(v_{m,i,0}^g)]. \tag{11}$$

Let $\boldsymbol{\Delta}_n^{1O}$ denote the $(N_k - 1)$-dimension vector consisting of these $\Delta_n^{1O}(k_c, k_g)$ for all $k_g \in \mathcal{K}\backslash\{k_c\}$. Let $\boldsymbol{\mu}$ and $\boldsymbol{\Sigma}$ denote the mean and variance of $\boldsymbol{\Delta}_1^{1O}(k_c, k_g)$. Then following the work in [20, 13], the success rate can be described with a multivariate Gaussian distribution $N(\boldsymbol{\mu}, \boldsymbol{\Sigma}/n)$ using the Central Limit Theorem. That is,

$$SR = \Phi_{\boldsymbol{\Sigma}}(\sqrt{n}\boldsymbol{\mu}). \tag{12}$$

where $\Phi_{\boldsymbol{\Sigma}}$ is the cumulative distribution function (CDF) of the $N_k - 1$ dimensional Gaussian distribution with mean $\mathbf{0}$ and variance $\boldsymbol{\Sigma}$.

For unmasked devices, the mean vector $\boldsymbol{\mu}$ and the variance matrix $\boldsymbol{\Sigma}$ are expressed by Fei et al. [13] in terms of their confusion coefficients κ. With imperfect masking, we show (in Appendix C) that similar expressions hold with our generalized confusion coefficients κ_{1O}.

Theorem 1. *Under CPA leakage model (9), the success rate of the CPA is given by equation (12). Under Assumption 1, the element in the mean vector $\boldsymbol{\mu}$ corresponding to key k_{gi} is*

$$\mu_{gi} = \frac{\delta_0}{2}\kappa_{1O}(k_c, k_{gi}); \tag{13}$$

And the elements of covariance matrix $\boldsymbol{\Sigma}$ *are*

$$\sigma_{k_{gi},k_{gi}} = \kappa_{1O}(k_c, k_{gi}), \qquad \sigma_{k_{gi},k_{gj}} = \kappa_{1O}(k_c, k_{gi}, k_{gj}) \text{ for } k_{gi} \neq k_{gj}, \tag{14}$$

where $\kappa_{1O}(k_c, k_{gi}, k_{gj}) = \mathbb{E}_X\{[\mathbb{E}_M(v^c_{m,1,0}) - \mathbb{E}_M(v^{gi}_{m,1,0})][\mathbb{E}_M(v^c_{m,1,0}) - \mathbb{E}_M(v^{gj}_{m,1,0})]\}$.

Similar to [13], we can get the above three-way generalized confusion coefficients $\kappa_{1O}(k_1, k_2, k_3)$ from two-way generalized confusion coefficients $\kappa_{1O}(k_1, k_2)$ (see more details in Appendix D).

Lemma 1. *Given* $k_c, k_{gi}, k_{gj} \in \mathcal{K}$,

$$\kappa_{1O}(k_c, k_{gi}, k_{gj}) = \frac{1}{2}[\kappa_{1O}(k_c, k_{gi}) + \kappa_{1O}(k_c, k_{gj}) - \kappa_{1O}(k_{gi}, k_{gj})]. \tag{15}$$

The average of $\kappa_{1O}(k_c, k_{gi})$ over all k_{gi} is $\mathrm{ILA}_{1\mathrm{O}}$. By Lemma 1, the average of $\kappa_{1O}(k_c, k_{gi}, k_{gj})$ over all $k_{gi} \neq k_{gj}$ is $\mathrm{ILA}_{1\mathrm{O}}/2$. Replacing all the confusion coefficient terms in equations (13) and (14) by their averages, we get an approximate asymptotic success rate for first-order CPA on masked devices:

$$SR = \Phi_{\frac{1}{2}[\boldsymbol{I}_{N_k-1} + \boldsymbol{J}_{N_k-1}]}\left(\frac{\delta_0\sqrt{n}\sqrt{\mathrm{ILA}_{1\mathrm{O}}}}{2}\mathbf{1}_{N_k-1}\right), \tag{16}$$

where $\boldsymbol{I}_{N_k-1}$ is the $(N_k-1)\times(N_k-1)$ identity matrix with diagonal entries of ones and off-diagonal entries of zeros, $\boldsymbol{J}_{N_k-1}$ is the $(N_k-1)\times(N_k-1)$ matrix with all entries of ones, and $\mathbf{1}_{N_k-1}$ is the (N_k-1) dimensional vector of ones.

The approximation SR formula (16) is very close to the SR formula (12) for small SNR δ_0. We will examine the approximation in Section 3.3.

3.2 Second-Order Power Analysis Attack Model

Second-order power analysis attack combines the two leakage measurements of V_0 and V_1 at two different positions involving the same mask M to break the masking protection. Similar to (9), we assume linear leakage for V_1

$$L_1 = c_1 + \epsilon_1 V_1 + \sigma_1 r_1, \tag{17}$$

where r_1 is the unit noise.

Second-order CPA uses n pairs of independent realizations of noisy physical leakage $(l_{1,0}, l_{1,1}), (l_{2,0}, l_{2,1}), ..., (l_{n,0}, l_{n,1})$ for (L_0, L_1). Here $l_{i,j} = c_j + \epsilon_j v_{i,j} + \sigma_j r_{i,j}, i = 1, ..., n, j = 0, 1$. Denote the centered version of L_j and V_j by $\widetilde{L}_j = L_j - \mathbb{E}(L_j)$ and $\widetilde{V}_j = V_j - \mathbb{E}(V_j)$, for $j = 0, 1$. While the first-order CPA exploits the correlation between $\widetilde{L}_0$ and $\widetilde{V}_0$, the second-order CPA exploits the correlation between $\widetilde{L}_0\widetilde{L}_1$ and $\widetilde{V}_0\widetilde{V}_1$. That is, it uses the centered product statistic:

$$\frac{1}{n}\sum_{i=1}^{n} \widetilde{l}_{i0}\widetilde{l}_{i1}\mathbb{E}_M[\widetilde{v}^g_{m,i,0}\widetilde{v}_{m,1}], \tag{18}$$

where $\widetilde{l}_{ij} = (l_{i,j} - \bar{l}_{.,j})/\sigma_j, j = 0, 1$, is the centered leakage, $\widetilde{v}^g_{m,i,0} = v^g_{m,i,0} - \mathbb{E}[V^g_0]$ and $\widetilde{v}_{m,1} = v_{m,1} - \mathbb{E}[V_1]$ are the centered select functions values under guessed key k_g given mask m, and $v_{m,1} = V_1(m)$.

We denote the difference between the centered product statistics under secret key k_c and guessed key k_g as

$$\Delta^{2O}_n(k_c, k_g) = \frac{1}{n}\sum_{i=1}^{n} \widetilde{l}_{i0}\widetilde{l}_{i1}[\mathbb{E}_M(\widetilde{v}^c_{m,i,0}\widetilde{v}_{m,1}) - \mathbb{E}_M(\widetilde{v}^g_{m,i,0}\widetilde{v}_{m,1})]. \tag{19}$$

The second-order CPA succeeds when $\Delta^{2O}_n(k_c, k_g) > 0$ for all $k_g \in \mathcal{K}\backslash\{k_c\}$. Using derivations in [19, 9, 17], the success rate of second-order CPA also follows equation (12): $SR = \Phi_{\boldsymbol{\Sigma}}(\sqrt{n}\boldsymbol{\mu})$.

Ding et al. [9] expressed $\boldsymbol{\mu}$ and $\boldsymbol{\Sigma}$ in terms of confusion coefficients κ under perfect masking. With possibly imperfect masking, we generalize the formula in terms of our generalized confusion coefficients κ_{2O} (see details in Appendix E).

Theorem 2. *Under CPA leakage model (9) and (24), the success rate of the second-order CPA is given by equation (12). Under Assumption 1, the element in $\boldsymbol{\mu}$ corresponding to key k_{gi} is*

$$\mu_{gi} = \frac{\delta_0\delta_1}{2}\kappa_{2O}(k_c, k_{gi}); \tag{20}$$

And the elements of covariance $\boldsymbol{\Sigma}$ are

$$\sigma_{k_{gi},k_{gi}} = \kappa_{2O}(k_c, k_{gi}), \qquad \sigma_{k_{gi},k_{gj}} = \kappa_{2O}(k_c, k_{gi}, k_{gj}) \text{ for } k_{gi} \neq k_{gj}, \tag{21}$$

where $\kappa_{2O}(k_c, k_{gi}, k_{gj}) = \mathbb{E}_X\{[\mathbb{E}_M(\widetilde{v}^c_{m,1,0}\widetilde{v}_{m,1}) - \mathbb{E}_M(\widetilde{v}^{gi}_{m,1,0}\widetilde{v}_{m,1})][\mathbb{E}_M(\widetilde{v}^c_{m,1,0}\widetilde{v}_{m,1}) - \mathbb{E}_M(\widetilde{v}^{gj}_{m,1,0}\widetilde{v}_{m,1})]\}$.

Similar with Lemma 1, for $k_c, k_{gi}, k_{gj} \in \mathcal{K}$,

$$\kappa_{2O}(k_c, k_{gi}, k_{gj}) = \frac{1}{2}[\kappa_{2O}(k_c, k_{gi}) + \kappa_{2O}(k_c, k_{gj}) - \kappa_{2O}(k_{gi}, k_{gj})]. \tag{22}$$

As in the first-order analysis, replacing the generalized confusion coefficients κ_{2O} by ILA_{2O}, we get the approximate asymptotic success rate:

$$SR = \Phi_{\frac{1}{2}[\boldsymbol{I}_{N_k-1} + \boldsymbol{J}_{N_k-1}]}\left(\frac{\delta_0\delta_1\sqrt{n}\sqrt{\mathrm{ILA}_{2O}}}{2}\mathbf{1}_{N_k-1}\right). \tag{23}$$

Next we evaluate the above approximations.

3.3 Approximation Errors in the Simple Success Rate Formulas

Work in [13, 9] gives the explicit theoretical success rate formulas for two cases: the first-order CPA on unmasked devices and the second-order CPA on perfectly masked devices, respectively. By plugging ILA_{1O} when $p = 0$ in (16) and ILA_{2O} when $p = 1/2$ in (23), we get the two corresponding simple success rate formulas.

Compared to the formulas in [13, 9], our simple formulas ignore some higher order terms and replace the confusion coefficients by ILA. Here we study the effect of the simplification for CPA on unmasked and perfect masked AES.

We show the difference between our simplified success rate formulas and the explicit success rate formulas of [13, 9] in Fig. 2. The average error-ratio is defined as: $\mathbb{E}_{SR}[\,|\,\mathrm{N}_{\mathrm{Explicit,SR}} - \mathrm{N}_{\mathrm{Simple,SR}}\,|\,/\,\mathrm{N}_{\mathrm{Explicit,SR}}]$, where $\mathrm{N}_{\mathrm{Explicit,SR}}$ and $\mathrm{N}_{\mathrm{Simple,SR}}$ are numbers of traces needed to achieve a fixed SR value by the explicit and simplified theoretical success rate formulas respectively, and $\mathbb{E}_{\mathrm{SR}}$ is the expectation over all success rate values SR ranging from 0 to 1. Here, we take the expectation over discrete success rate values $\mathrm{SR} = [0.1, 0.2, 0.3, ..., 0.9]$.

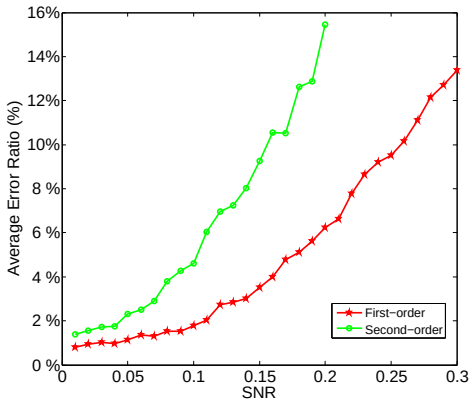

Fig. 2. The average error-ratio of number of measurements between explicit and simplified success rate formulas on AES S-Box

Fig. 2 shows that as the SNR grows, both error ratios increase. The error-ratio$\leq$ 10% when SNR ≤ 0.26 for the first-order attack, and when SNR ≤ 0.16 for the second-order attack. Hence the simplified success rate formulas in Equations (16) and (23) work well for small SNR values. For practical physical implementations, devices with large SNR values are very leaky and not considered secure. The success rate analysis is only meaningful when the SNR is small.

3.4 Comparing Effectiveness of the First-Order Attack and the Second-Order Attack

For unmasked devices, first-order leakage is sufficient to discover the secret key. With perfect masking, only second-order leakage can be used to discover the secret key. However, for imperfect masking implementations, both first-order and second-order leakage exist. Which leakage is more effective to exploit? We can compare them using the proposed metrics through formulas (16) and (23).

Property 3 *For a masked implementation*

- *The first-order attack is more effective when* $\delta_1 < \sqrt{\frac{\mathrm{ILA_{1O}}}{\mathrm{ILA_{2O}}}}$;
- *The second-order attack is better when* $\delta_1 > \sqrt{\frac{\mathrm{ILA_{1O}}}{\mathrm{ILA_{2O}}}}$.

For very small SNR, the first order leakage will dominate. The threshold SNR value to determine dominance by the first-order or the second-order leakage is given by the square root of the ratio between the two information leakages: $\sqrt{\frac{\mathrm{ILA_{1O}}}{\mathrm{ILA_{2O}}}}$. If the typical SNR value is known for certain physical devices, we can predict which type of leakage dominates and therefore guide the software designer in effective leakage reduction.

3.5 Extension to Higher-Order Power Analysis Attack Model

We now consider a cryptography algorithm protected by J-th order masking, with mask shares $M_1, M_2, ..., M_J$. A J-th order attack combines the information leakage of $V_0(X, K, M_1, ..., M_J)$ and the leakage of $V_1(M_1), ..., V_J(M_J)$ to retrieve the secret key. J-th order power analysis attack combines the $J+1$ leakage measurements of $V_0, V_1, ..., V_J$ at $J+1$ different positions to break the masking protection. The leakage vector is $\boldsymbol{l}_i = (l_{i,0}, ..., l_{i,J})$. Similar to (9) and (24), the leakage model is now:

$$L_j = c_j + \epsilon_1 V_j + \sigma_j r_j, \qquad j = 0, ..., J. \tag{24}$$

where r_j is the unit noise.

For a key hypothesis $k_g \in \mathcal{K}\backslash\{k_c\}$, we define the J-th order confusion coefficient of masked algorithm as

$$\kappa_{JO}(k_c, k_g) = \mathbb{E}_X\{[\mathbb{E}_M(\widetilde{V_0}\widetilde{V_1}...\widetilde{V_J}|(X, k_c)) - \mathbb{E}_M(\widetilde{V_0}\widetilde{V_1}...\widetilde{V_J}|(X, k_{gi}))]^2\}, \tag{25}$$

where $\widetilde{V_i} = V_i - \mathbb{E}_{X,M}[V_i], i = 0, 1, ..., J$, are the centered select function values.

Definition 5. *The J-th order information leakage amount* $\mathrm{ILA_{JO}}$ *is defined as*

$$\mathrm{ILA_{JO}} = \mathbb{E}_{\mathcal{K}\backslash\{k_c\}}[\kappa_{JO}(k_c, k_g)]. \tag{26}$$

As in [9] and in section 3.2, we can derive the approximate asymptotic success rate as:

$$SR = \Phi_{\frac{1}{2}[\boldsymbol{I}_{N_k-1}+\boldsymbol{J}_{N_k-1}]}\left(\frac{\sqrt{n}\sqrt{\mathrm{ILA_{JO}}}\prod_{j=0}^{J}\delta_j}{2}\mathbf{1}_{N_k-1}\right). \tag{27}$$

4 Numerical Results

In this section, we first numerically investigate the relationship between success rates of DPA/CPA and the metrics ILA, $\mathrm{MI_A}$ and QMS on synthetic data examples. We also evaluate our metrics and the simplified success rates of DPA/CPA on realistic measurement data.

4.1 Numerical Comparison of Metrics versus Success Rates

We first show, by numerical examples, that ILA_{1O} measures the leakage information amount under CPA, but $\mathrm{MI_A}$ and QMS do not. We consider synthetic data examples with biased masking on the outputs of an AES S-Box, where the masking bits are independent with $p_i = \mathbb{P}(M_{(i)} = 1), i = 1, 2, ..., 8$.

In the first example, the last 4-bits are perfectly masked with $p_5 = p_6 = p_7 = p_8 = 0.5$, and the information leakage is through the Hamming weights of the first 4-bits according to model (9). We consider two cases where $\overrightarrow{p}_4 = [p_1, p_2, p_3, p_4] = [0.5, 0.2, 0.2, 0.1]$ and $\overrightarrow{p}_4 = [0, 0.4, 0.4, 0.4]$ respectively. We calculate the values of the different metrics through definitions in equations (1), (2) and (5), rather than using specialized formulas in Properties 1 and 2 (which only apply to Boolean masking with equal p_i's for each bit). Detailed algorithms are provided in Appendix F. In both cases $\mathrm{MI_A} = 1.09$, but the information leakage amount differs with $\mathrm{ILA_{1O}} = 0.68$ and $\mathrm{ILA_{1O}} = 0.56$, respectively. Fig. 3 (a) shows the success rates of CPA in both cases on synthetic data generated from the power model (9) with SNR = 0.1. The empirical success rate for a fixed number of measurements $\mathrm{N_{trace}}$ is found by repeatedly randomly sampling $\mathrm{N_{trace}}$ traces for an attack, and calculating the proportion of attacks that retrieves the correct secret key. We see that the $\mathrm{ILA_{1O}}$ correctly predicts the two different CPA success rates curves (with difference about 10%), while by $\mathrm{MI_A}$ the information leakage should be the same in these two cases. Note that from Fig. 2, the error ratio of our simplified SR formula under first-order CPAs is only 1.5% when SNR = 0.1.

In the second example, the last 6-bits are perfectly masked. For two cases of $\overrightarrow{p}_2 = [p_1, p_2] = [0.3, 0.3]$ and $\overrightarrow{p}_2 = [0.1, 0.5]$, QMS = 0.4, but ILA = 0.16 and 0.32 respectively. Fig. 3 (b) shows that $\mathrm{ILA_{1O}}$ correctly predicts the different empirical CPA success rate curves, while QMS incorrectly labels the two cases as equally leaky. Therefore, only $\mathrm{ILA_{1O}}$ correctly measures the CPA leakage in these examples.

The formulas (16) and (23) give the CPA success rates using ILA and SNR. Fig. 4 plots the number of traces N_{trace} needed to achieve success rate of SR = 80%, when ILA and SNR vary. Fig. 4 (a) is for the first-order CPA attack (16) and (b) is for the second-order CPA attack (23). As ILA increases or SNR increases, less traces are needed to get SR = 80%. For a fixed SNR value, the number of traces $\mathrm{N_{trace}}$ is inverse proportional to ILA.

For the special case of single-bit DPA, all three metrics are monotonic functions of each other (Property 1). Thus, $\mathrm{MI_A}$ and QMS can predict the DPA success rate through their relationship with ILA. Particularly, for DPA, $\mathrm{ILA_{1O}} = (1 - \mathrm{QMS})^2/2$ and the $\mathrm{N_{trace}}$ traces needed for DPA is inverse proportional to $(1 - \mathrm{QMS})^2$.

4.2 Experimental Results on Physical Implementations

We next verify the prediction of success rates by ILA, and show that it also correctly predicts the dominance by first-order or second-order CPA leakage on

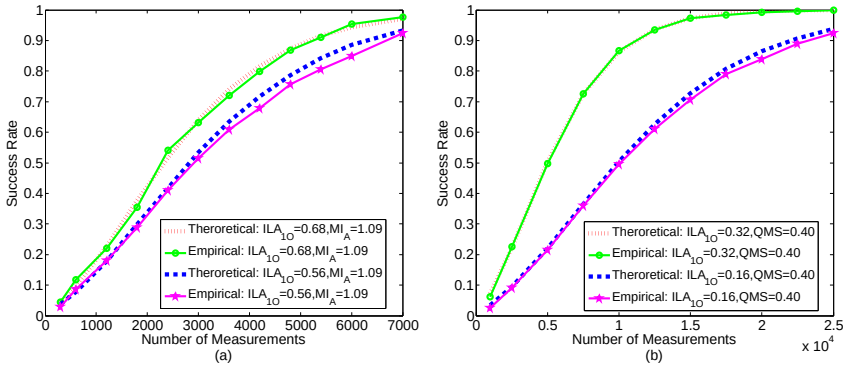

Fig. 3. First-order CPA attacks under two different biased masking schemes with SNR = 0.1 (a) with the same $\mathrm{MI_A}$ value but different $\mathrm{ILA_{1O}}$ values; (b) with the same QMS value but different $\mathrm{ILA_{1O}}$ values

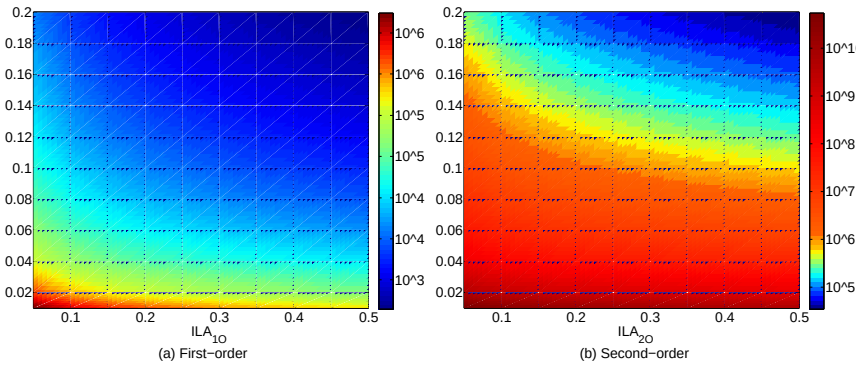

Fig. 4. The theoretical number of traces needed for SR = 80% under first-order and second-order CPA.

real physical systems. Two physical implementations of masked Keccak and AES algorithms are considered. The masked AES [1] is implemented on an SASEBO-GII board [22]. The protected Keccak implementation with secret sharing [4] is on the 32-bit Microblaze processor of the SASEBO-GII board. All the power traces are collected using a LeCroy WaveRunner 640Zi oscilloscope.

We get several power data sets with biased masking through choosing parts of the fully masking data set according to biased masks distributions. The first two data sets are on the same AES implementation with $\delta_0 = 0.10, \delta_1 = 0.12$ but with different biased masks. The leakage amount on the first data set is $\mathrm{ILA_{1O}} = 0.338, \mathrm{ILA_{2O}} = 13.8$, while the leakage amount on the second data set is $\mathrm{ILA_{1O}} = 0.174, \mathrm{ILA_{2O}} = 15.7$ for CPA attacks. For the third data set on Keccak, $\delta_0 = 0.10, \delta_1 = 0.10, \mathrm{ILA_{1O}} = 0.010, \mathrm{ILA_{2O}} = 0.006$ for DPA attacks.

For these three data sets, $\sqrt{\mathrm{ILA}_{1\mathrm{O}} \,/\, \mathrm{ILA}_{2\mathrm{O}}}/\delta_1 = 1.3, 0.88, 2.02$ respectively. By Property 3, the first-order attack is more effective in the first and third data sets, and the second-order attack is more effective in the second data set.

Fig. 5 shows the success rates of CPAs on the first two data sets for AES. Each figure plots four curves, the theoretical success rates for first-order CPA (16) and the second-order CPA (23), and two corresponding empirical success rate curves. The empirical success rates are close to the theoretical success rates. The first-order leakage and second-order leakage are ranked in the order predicted by Property 3.

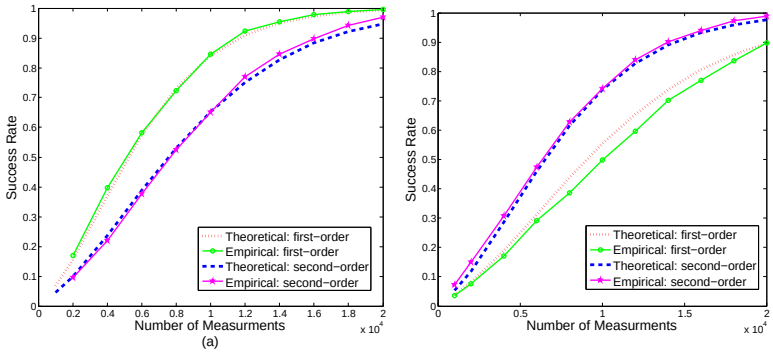

Fig. 5. The first-order CPA attack and second-order CPA attack on AES with different masking biases.

Fig. 6 shows the success rates of CPA on the Keccak data are also as predicted by Equations (16) and (23).

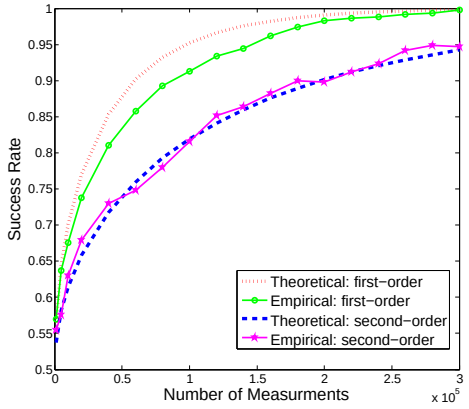

Fig. 6. The first-order CPA attack and the second-order CPA attacks on Keccak data subset.

5 Conclusion

In this work, we propose a new unified metric, ILA, to measure the information leakage at the early stage of cryptographic software under different power analysis attacks. It quantifies the leakage amount of algorithms with various masking strength to first-order or second-order power analysis attacks. Unlike existing metrics, ILA relates to the attack success rate on the physical implementations through a simple explicit formula. We demonstrate that it accurately quantifies the leakage amount comparing to existing metrics on both synthetic data and real physical implementation data. Therefore, it would be a reliable metric for system designers to predict the system leakage and develop better protections.

Acknowledgments. This work is supported in part by the National Science Foundation under grants CNS-1314655 and CNS-1337854.

References

1. Akkar, M.L., Giraud, C.: An implementation of des and aes, secure against some attacks. In: Int. Wksp on Cryptographic Hardware & Embedded Systems. pp. 309–318 (2001)
2. Bayrak, A.G., Regazzoni, F., Brisk, P., Standaert, F.X., Ienne, P.: A first step towards automatic application of power analysis countermeasures. In: Proceedings of the 48th Design Automation Conference. pp. 230–235 (2011)
3. Bayrak, A., Regazzoni, F., Novo, D., Ienne, P.: Sleuth: Automated verification of software power analysis countermeasures. In: Int. Wksp on Cryptographic Hardware & Embedded Systems, pp. 293–310 (2013)
4. Bertoni, G., Daemen, J., Peeters, M., Van Assche, G.: Building power analysis resistant implementations of Keccak. In: Second SHA-3 Candidate Conference (2010)
5. Brier, E., Clavier, C., Olivier, F.: Correlation power analysis with a leakage model. In: Int. Wksp on Cryptographic Hardware & Embedded Systems, pp. 135–152 (2004)
6. Chari, S., Rao, J., Rohatgi, P.: Template attacks. In: Int. Wksp on Cryptographic Hardware & Embedded Systems. pp. 51–62 (2003)
7. Chari, S., Jutla, C., Rao, J., Rohatgi, P.: Towards sound approaches to counteract power-analysis attacks. In: Advances in Cryptology CRYPTO, pp. 398–412 (1999)
8. Dan Walters, A.H., Kedaigle, E.: Sleak: A side -channel leakage evaluator and analysis kit. In: International Cryptographic Module Conference (2014)
9. Ding, A.A., Zhang, L., Fei, Y., Luo, P.: A statistical model for multivariate DPA on masked devices. In: Int. Wksp on Cryptographic Hardware & Embedded Systems. pp. 147–169 (2014)
10. Duc, A., Faust, S., Standaert, F.X.: Making masking security proofs concrete. In: Advances in Cryptology – EUROCRYPT, pp. 401–429 (2015)
11. Eldib, H., Wang, C., Taha, M., Schaumont, P.: Qms: Evaluating the side-channel resistance of masked software from source code. In: Proceedings of the 51st Annual Design Automation Conference. pp. 209:1–209:6 (2014)
12. Fei, Y., Luo, Q., Ding, A.A.: A statistical model for DPA with novel algorithmic confusion analysis. In: Int. Wksp on Cryptographic Hardware & Embedded Systems. pp. 233–250 (2012)

13. Fei, Y., Ding, A.A., Lao, J., Zhang, L.: A statistics-based fundamental model for side-channel attack analysis. Cryptology ePrint Archive (2014), http://eprint.iacr.org/
14. Gierlichs, B., Batina, L., Tuyls, P., Preneel, B.: Mutual information analysis. In: Int. Wksp on Cryptographic Hardware & Embedded Systems. pp. 426–442 (2008)
15. Heuser, A., Rioul, O., Guilley, S.: Good is not good enough: Deriving optimal distinguishers from communication theory. In: Int. Wksp on Cryptographic Hardware & Embedded Systems. pp. 55–74 (2014)
16. Kocher, P.C., Jaffe, J., Jun, B.: Differential power analysis. In: Proc. Int. Cryptology Conf. on Advances in Cryptology, pp. 388–397 (1999)
17. Lomné, V., Prouff, E., Rivain, M., Roche, T., Thillard, A.: How to estimate the success rate of higher-order side-channel attacks. In: Int. Wksp on Cryptographic Hardware & Embedded Systems, pp. 35–54 (2014)
18. Luo, Q., Fei, Y.: Algorithmic collision analysis for evaluating cryptographic systems and side-channel attacks. In: IEEE Int. Symp. Hardware Oriented Security & Trust. pp. 75–80 (2011)
19. Prouff, E., Rivain, M., Bevan, R.: Statistical analysis of second order differential power analysis. pp. 799 – 811 (2009)
20. Rivain, M.: On the exact success rate of side channel analysis in the gaussian model. In: Selected Areas in Cryptography, pp. 165–183 (2009)
21. Standaert, F.X., Malkin, T., Yung, M.: A unified framework for the analysis of side-channel key recovery attacks. In: Advances in Cryptology - EUROCRYPT. pp. 443–461 (2009)
22. Evaluation environment for side-channel attacks. http://www.risec.aist.go.jp/project/sasebo/

Appendices

A Derivation of ILA, QMS and $\mathrm{MI_A}$ for DPA Model

For the DPA model, Z is one single bit, as well as M. Under Assumption 1, $\mathbb{P}(Z=0)=\mathbb{P}(Z=1)=1/2$. For the Boolean masking, $V_0=F=Z\oplus M$. Hence $\mathbb{P}(Z\oplus M=0)=\mathbb{P}(Z\oplus M=1)=1/2$,

$$\mathbb{P}(Z\oplus M=1|Z)=(1-2p)Z+p=p \text{ or } 1-\mathrm{p}. \tag{28}$$

Using equation (28), $D_{x,k}(F)=p$ or $1-p$, which implies $\max\{|D_{x,k}(F)-D_{x',k'}(F)|\}=|1-2p|$. Hence QMS $=1-|1-2p|$.

For $\mathrm{MI_A}$, we calculate the entropies first.

$$H(K)=-\sum_{k\in\mathcal{K}}p(k)\log_2 p(k)=-\sum_{k\in\mathcal{K}}\frac{1}{N_k}\log_2\frac{1}{N_k}=\log_2 N_k.$$

$$\begin{aligned}H(K|V_0)&=-\sum_{k\in\mathcal{K}}p(k).\sum_{x\in\{0,1\}}p(x).\sum_{v_0\in\{0,1\}}p(v_0|k,x).\log_2 p(k|v_0,x)\\&=-\sum_{k\in\mathcal{K}}\tfrac{1}{N_k}.\sum_{x\in\{0,1\}}\tfrac{1}{2}.[p\log_2\tfrac{\frac{p}{2N_k}}{\frac{1}{4}}+(1-p)\log_2\tfrac{\frac{1-p}{2N_k}}{\frac{1}{4}}]\\&=\log_2 N_k-[1+(1-p)\log_2(1-p)+p\log_2 p].\end{aligned}$$

Therefore,

$$\mathrm{MI_A} = H(K) - H(V_0|K) = 1 + (1-p)\log_2(1-p) + p\log_2 p. \tag{29}$$

We will derive the $\mathrm{ILA_{1O}}$ and $\mathrm{ILA_{2O}}$ expressions in the CPA model in Appendix B. Plugging-in $b = 1$, we get their DPA model expressions.

B Derivation of $\mathrm{ILA_{1O}}$ and $\mathrm{ILA_{2O}}$ For CPA model

For the CPA model, the selection is Hamming weights $V_0 = H(Z \oplus M)$, $V_1 = H(M)$, and both M and Z are b-bit variables. Since $\mathbb{P}(M_{(i)} = 1) = p, i = 1, 2, ..., b$, we have:

$$\mathbb{E}_M[H(M)] = bp, \qquad \mathbb{E}_M[H(M)^2] = bp + b(b-1)p^2. \tag{30}$$

Under Assumption 1, Z has uniform distribution for any key k_g so that always

$$\mathbb{E}_X[H(Z)] = b/2, \qquad \mathbb{E}_X[H(Z)^2] = (b^2+b)/4. \tag{31}$$

Here $V_0^*(X,k) = H[Z(X,k)]$. Under Assumptions 1 and 3,

$$\begin{aligned}
&\mathbb{E}_{k_g}\kappa(k_c,k_g) = \mathbb{E}_{k_g}\mathbb{E}_X\{[V_0^*(X,k_c) - V_0^*(X,k_g)]^2\} \\
&= \mathbb{E}_{k_g}\mathbb{E}_X[V_0^*(X,k_c)^2] + \mathbb{E}_{k_g}\mathbb{E}_X[V_0^*(X,k_g)^2] - 2\mathbb{E}_{k_g}\mathbb{E}_X[V_0^*(X,k_c)V_0^*(X,k_g)] \\
&= 2\mathbb{E}_X[V_0^*(X,k_c)^2] - 2\{\mathbb{E}_X[V_0^*(X,k_c)]\}^2.
\end{aligned} \tag{32}$$

Using (31), this becomes

$$\mathbb{E}_{k_g}\kappa(k_c,k_g) = 2(\tfrac{b^2+b}{4}) - 2(\tfrac{b^2}{4}) = \tfrac{b}{2}. \tag{33}$$

By the property 2 in [19], with $\wedge$ denoting the bit-wise multiplication,

$$\begin{aligned}
\mathbb{E}_M[H(Z\oplus M)|(X,k_c)] &= \mathbb{E}_M[H(Z) + H(M) - 2H(Z\wedge M)|(X,k_c)] \\
&= (1-2p)H(Z) + bp.
\end{aligned} \tag{34}$$

Then for the first-order CPA, using equations (34) and (33)

$$\begin{aligned}
&\mathrm{ILA_{1O}} = \mathbb{E}_{k_g}[\kappa_{1O}(k_c,k_g)] \\
&= \mathbb{E}_{k_g}[\mathbb{E}_X\{[\mathbb{E}_M(H(Z\oplus M)|(X,k_c)) - \mathbb{E}_M(H(Z\oplus M)|(X,k_g))]^2\}] \\
&= \mathbb{E}_{k_g}[(1-2p)^2\kappa(k_c,k_g)] = \tfrac{b(1-2p)^2}{2}.
\end{aligned} \tag{35}$$

Similar to (34), using (30),

$$\begin{aligned}
&\mathbb{E}_M\{[H(Z\oplus M) - \tfrac{b}{2}][H(M) - bp]|(X,k_c)\} \\
&= \mathbb{E}_M\{[H(Z\oplus M)H(M) - bpH(Z\oplus M)]|(X,k_c)\} \\
&= \mathbb{E}_M\{[H(Z)H(M) + H(M)^2 - 2H(Z\wedge M)H(M) - bpH(Z\oplus M)]|Z\} \\
&= H(Z)bp + [bp + b(b-1)p^2] - 2[p + (b-1)p^2]H(Z) \\
&\qquad -bp[(1-2p)H(Z) + bp] \\
&= -2p(1-p)[H(Z) - \tfrac{b}{2}].
\end{aligned} \tag{36}$$

Hence for the second-order CPA, using equations (36) and (33)

$$\begin{aligned}
&\mathrm{ILA_{2O}} = \mathbb{E}_{k_g}[\kappa_{2O}(k_c,k_g)] \\
&= \mathbb{E}_{k_g}[\mathbb{E}_X\{[\mathbb{E}_M(\widetilde{V_0V_1}|(X,k_c)) - \mathbb{E}_M(\widetilde{V_0V_1}|(X,k_g))]^2\}] \\
&= \mathbb{E}_{k_g}[4p^2(1-p)^2\kappa(k_c,k_g)] = 2bp^2(1-p)^2.
\end{aligned} \tag{37}$$

C Theorem 1: $\boldsymbol{\mu}$ and $\boldsymbol{\Sigma}$ in the first-order CPA (12)

Denote $v^g_{m,1,0} = V_0(x_1, k_g, m)$ and $v_{1,0} = V_0(x_1, k_c, m_1)$. Recall that, under Assumption 1, $\mathbb{E}[v^g_{m,1,0}] = \mathbb{E}[V^g_0] = \mathbb{E}[V^c_0]$ and $\mathbb{E}_X\{[\mathbb{E}_M(v^g_{m,1,0})]^2\} = \mathbb{E}_X\{[\mathbb{E}_M(v^c_{m,1,0})]^2\}$ for any k_g. Hence we have an useful expression that will be used later,

$$\begin{array}{l} \mathbb{E}_X\{\mathbb{E}_M(v^c_{m,1,0})[\mathbb{E}_M(v^c_{m,1,0}) - \mathbb{E}_M(v^g_{m,1,0})]\} \\ = \frac{1}{2}\mathbb{E}_X\{[\mathbb{E}_M(v^c_{m,1,0})]^2 + [\mathbb{E}_M(v^g_{m,1,0})]^2 - 2\mathbb{E}_M(v^g_{m,1,0})\mathbb{E}_M(v^g_{m,1,0})]\} \\ = \frac{1}{2}\mathbb{E}_X\{[\mathbb{E}_M(v^c_{m,1,0}) - \mathbb{E}_M(v^g_{m,1,0})]^2\} = \frac{1}{2}\kappa_{1O}(k_c, k_g). \end{array} \tag{38}$$

For large n, $\bar{l}_{.,0} = c_0 + \epsilon_0\mathbb{E}(v_{1,0})$ and $l_{1,0} = c_0 + \epsilon_0 v_{1,0} + \sigma_0 r_{1,0}$, then equation (11) becomes

$$\Delta^{1O}_1(k_c, k_g) = \{\delta_0[v_{1,0} - \mathbb{E}(v_{1,0})] + r_{1,0}\}[\mathbb{E}_M(v^c_{m,1,0}) - \mathbb{E}_M(v^g_{m,1,0})]. \tag{39}$$

Since $\mathbb{E}[r_{1,0}] = 0$, we have:

$$\begin{array}{rl} \boldsymbol{\mu}_{k_g} &= \delta_0\mathbb{E}\{(v_{1,0} - \mathbb{E}[v_{1,0}])(\mathbb{E}_M[v^c_{m,1,0}] - \mathbb{E}_M[v^g_{m,1,0}])\} \\ &= \delta_0\mathbb{E}\{v_{1,0}(\mathbb{E}_M[v^c_{m,1,0}] - \mathbb{E}_M[v^g_{m,1,0}])\} = \frac{\delta_0}{2}\kappa_{1O}(k_c, k_g). \end{array} \tag{40}$$

The last equality uses the fact that $\mathbb{E}_M[v_{1,0}] = \mathbb{E}_M[v^c_{m,1,0}]$ and equation (38).

The element in covariance Σ corresponding to k_{gi} and k_{gj} is:

$$\sigma_{k_{gi},k_{gj}} = COV(\Delta^{1O}_1(k_c, k_{gi}), \Delta^{1O}_1(k_c, k_{gj})) = E[\Delta^{1O}_1(k_c, k_{gi})\Delta^{1O}_1(k_c, k_{gj})] - \mu_{k_{gi}}\mu_{k_{gj}}. \tag{41}$$

Since $\mathbb{E}[r^2_{1,0}] = 1$, keep the leading term (dropping the terms with δ_0), we have

$$\sigma_{k_{gi},k_{gj}} = \mathbb{E}_X\{(\mathbb{E}_M[v^c_{m,1,0}] - \mathbb{E}_M[v^{gi}_{m,1,0}])(\mathbb{E}_M[v^c_{m,1,0}] - \mathbb{E}_M[v^{gj}_{m,1,0}])\} = \kappa_{1O}(k_c, k_{gi}, k_{gj}). \tag{42}$$

D Proof of Lemma 1

Similar to the derivation of (38),

$$\begin{array}{l} \kappa_{1O}(k_c, k_{gi}, k_{gj}) \\ = \mathbb{E}_X\{(\mathbb{E}_M[v^c_{m,1,0}] - \mathbb{E}_M[v^{gi}_{m,1,0}])(\mathbb{E}_M[v^c_{m,1,0}] - \mathbb{E}_M[v^{gj}_{m,1,0}])\} \\ = \mathbb{E}_X\{(\mathbb{E}_M[v^c_{m,1,0}])^2 - \mathbb{E}_M[v^c_{m,1,0}]\mathbb{E}_M[v^{gi}_{m,1,0}] \\ \qquad -\mathbb{E}_M[v^c_{m,1,0}]\mathbb{E}_M[v^{gj}_{m,1,0}] + \mathbb{E}_M[v^{gi}_{m,1,0}]\mathbb{E}_M[v^{gj}_{m,1,0}]\} \\ = \frac{1}{2}\mathbb{E}_X\{(\mathbb{E}_M[v^c_{m,1,0}] - \mathbb{E}_M[v^{gi}_{m,1,0}])^2 + (\mathbb{E}_M[v^c_{m,1,0}] - \mathbb{E}_M[v^{gj}_{m,1,0}])^2 \\ \qquad -(\mathbb{E}_M[v^{gi}_{m,1,0}] - \mathbb{E}_M[v^{gj}_{m,1,0}])^2\} \\ = \frac{1}{2}[\kappa_{1O}(k_c, k_{gi}) + \kappa_{1O}(k_c, k_{gj}) - \kappa_{1O}(k_{gi}, k_{gj})]. \end{array} \tag{43}$$

E Theorem 2: $\boldsymbol{\mu}$ and $\boldsymbol{\Sigma}$ in the Second-order CPA (12)

For large sample n, $\bar{l}_{.,j} = c_j + \epsilon_j E[v_{1,j}]$, then $l_{1,j} = c_j + \epsilon_j \widetilde{v}_{1,j} + \sigma_j r_{1,j}, j = 0, 1$, where $\widetilde{v}_{1,j} = v_{1,j} - \mathbb{E}(v_{1,j})$ are the centered version of $v_{1,0} = V_0(x_1, k_c, m_1)$ and $v_{1,1} = V_1(m1)$. Similarly, let $\widetilde{v}_{m,1,0}$, $\widetilde{v}^g_{m,1,0}$, and $\widetilde{v}_{m,1}$ denote the centered versions of corresponding quantities $v_{m,1,0}$, $v^g_{m,1,0}$, and $v_{m,1}$. We have

$$\Delta_1^{2O}(k_c, k_g) = (\delta_0 \widetilde{v}_{1,0} + r_{1,0})(\delta_1 \widetilde{v}_{1,1} + r_{1,1})(\mathbb{E}_M[\widetilde{v}^c_{m,1,0} \widetilde{v}_{m,1}] - \mathbb{E}_M[\widetilde{v}^g_{m,1,0} \widetilde{v}_{m,1}]). \tag{44}$$

Since $\mathbb{E}[r_{1,0}] = \mathbb{E}[r_{1,1}] = 0$,

$$\begin{aligned} \boldsymbol{\mu}_{k_g} &= \delta_0 \delta_1 \mathbb{E}\{\widetilde{v}_{1,0} \widetilde{v}_{1,1} (\mathbb{E}_M[\widetilde{v}^c_{m,1,0} \widetilde{v}_{m,1}] - \mathbb{E}_M[\widetilde{v}^g_{m,1,0} \widetilde{v}_{m,1}])\} \\ &= \delta_0 \delta_1 \mathbb{E}_X\{\mathbb{E}_M\{\widetilde{v}_{1,0} \widetilde{v}_{1,1} (\mathbb{E}_M[\widetilde{v}^c_{m,1,0} \widetilde{v}_{m,1}] - \mathbb{E}_M[\widetilde{v}^g_{m,1,0} \widetilde{v}_{m,1}])\}\}. \end{aligned} \tag{45}$$

By assumption 1, $\mathbb{E}[\widetilde{v}_{1,0} \widetilde{v}_{1,1}] = \mathbb{E}[\widetilde{v}^c_{m,1,0} \widetilde{v}_{m,1}] = \mathbb{E}[\widetilde{v}^g_{m,1,0} \widetilde{v}_{m,1}]$. Similar to the derivation of (38),

$$\begin{aligned} \boldsymbol{\mu}_{k_g} &= \delta_0 \delta_1 \mathbb{E}_X\{\mathbb{E}_M[\widetilde{v}_{1,0} \widetilde{v}_{1,1}] (\mathbb{E}_M[\widetilde{v}^c_{m,1,0} \widetilde{v}_{m,1}] - \mathbb{E}_M[\widetilde{v}^g_{m,1,0} \widetilde{v}_{m,1}])\} \\ &= \tfrac{\delta_0 \delta_1}{2} \mathbb{E}_X\{\{\mathbb{E}_M[\widetilde{v}^c_{m,1,0} \widetilde{v}_{m,1}] - \mathbb{E}_M[\widetilde{v}^g_{m,1,0} \widetilde{v}_{m,1}]\}^2\} \\ &= \tfrac{\delta_0 \delta_1}{2} \kappa_{2O}(k_c, k_g). \end{aligned} \tag{46}$$

The element in covariance Σ corresponding to k_{gi} and k_{gj} is:

$$\sigma_{k_{gi}, k_{gj}} = COV(\Delta_1(k_c, k_{gi}), \Delta_1(k_c, k_{gj})) = E[\Delta_1(k_c, k_{gi}) \Delta_1(k_c, k_{gj})] - \boldsymbol{\mu}_{k_{gi}} \boldsymbol{\mu}_{k_{gj}}. \tag{47}$$

Since $\mathbb{E}[r^2_{1,0}] = \mathbb{E}[r^2_{1,1}] = 1$, the leading term (dropping terms with δ_0 or δ_1) is ,

$$\begin{aligned} &\sigma_{k_{gi}, k_{gj}} \\ &= \mathbb{E}_X\{(\mathbb{E}_M[\widetilde{v}^c_{m,1,0} \widetilde{v}_{m,1}] - \mathbb{E}_M[\widetilde{v}^{gi}_{m,1,0} \widetilde{v}_{m,1}])(\mathbb{E}_M[\widetilde{v}^c_{m,1,0} \widetilde{v}_{m,1}] - \mathbb{E}_M[\widetilde{v}^{gj}_{m,1,0} \widetilde{v}_{m,1}])\} \\ &= \kappa_{2O}(k_c, k_{gi}, k_{gj}). \end{aligned} \tag{48}$$

F Algorithms for Calculating $\mathbf{ILA_{1O}}$

Here, we describe the algorithm of computing $\mathrm{ILA}_{1\mathrm{O}}$ knowing the mask distribution. Algorithm 1 assigns the probability distribution of mask with the known probability for each masking bit. Algorithm 2 calculates the first-order information leakage amount based on this probability distribution. These algorithms are used to calculate the $\mathrm{ILA}_{1\mathrm{O}}$ values in Section 4.1.

Algorithm 1 Probability Distribution of Mask

Input: Probability distribution of masking bits $\overrightarrow{p}$
Output: Probability distribution of mask f_M

$N_m \leftarrow$ size of key space $|\mathcal{M}|$
$N_{bit} \leftarrow$ size of byte $|\overrightarrow{p}|$
for $m = 0 \rightarrow N_m - 1$ **do**
    $f_M[m] = 1$
    for $i = 0 \rightarrow N_{bit} - 1$ **do**
        if $m_{(i)} = 1$ **then** ▷ $m_{(i)}$ the $(i+1)$th bit of m
            $f_M[m] \leftarrow f_M[m] * p_i$ ▷ p_i the $(i+1)$th value of $\overrightarrow{p}$
        end if
        if $m_{(i)} = 0$ **then**
            $f_M[m] \leftarrow f_M[m] * (1 - p_i)$
        end if
    end for
end for

Algorithm 2 Calculation of $\mathrm{ILA}_{1\mathrm{O}}$

Input: Correct Key k_c, probability distribution of mask f_M, intermediate value V (a $N_k \times N_x \times N_m$ dimension matrix)
Output: $\mathrm{ILA}_{1\mathrm{O}}$

$N_k \leftarrow$ size of key space $|\mathcal{K}|$
$N_x \leftarrow$ size of plaintext (ciphertext) $|\mathcal{X}|$
$N_m \leftarrow$ size of mask $|\mathcal{M}|$
$\mathrm{ILA}_{1\mathrm{O}} \leftarrow 0$
for $k_g = 0 \rightarrow N_k - 1$ **do**
    $E_2[k_g] \leftarrow 0$
    for $x = 0 \rightarrow N_x - 1$ **do**
        $E_1[k_g][x] \leftarrow 0$
        for $m = 0 \rightarrow N_m - 1$ **do**
            $E_1[k_g][x] \leftarrow E_1[k_g][x] + (V[k_c][x][m] * f_M[m] - V[k_g][x][m] * f_M[m])$
        end for
        $E_2[k_g] \leftarrow E_2[k_g] + E_1[k_g][x] * E_1[k_g][x] * \frac{1}{N_x}$ ▷ $E_2[k_g] = \kappa_{1O}(k_c, k_g)$
    end for
    $\mathrm{ILA}_{1\mathrm{O}} \leftarrow \mathrm{ILA}_{1\mathrm{O}} + E_2[k_g] * \frac{1}{N_k - 1}$
end for