A Statistics-based Fundamental Model for Side-channel Attack Analysis*

Yunsi Fei, A. Adam Ding, Jian Lao, and Liwei Zhang

[1] Yunsi Fei Department of Electrical and Computer Engineering
Northeastern University, Boston, USA
Telephone: (617) 373-2039, Fax: (617) 373-8970
yfei@ece.neu.edu
[2] A. Adam Ding Department of Mathematics
Northeastern University, Boston, USA
a.ding@neu.edu
[3] Jian Lao Department of Electrical and Computer Engineering
Northeastern University, Boston, USA
jlao@ece.neu.edu
[4] Liwei Zhang Department of Mathematics
Northeastern University, Boston, USA
a.ding@neu.edu

Abstract. Side-channel attacks (SCAs) exploit leakage from the physical implementation of cryptographic algorithms to recover the otherwise secret information. In the last decade, popular SCAs like differential power analysis (DPA) and correlation power analysis (CPA) have been invented and demonstrated to be realistic threats to many critical embedded systems. However, there is still no sound and provable theoretical model that illustrates precisely what the success of these attacks depends on and how. Based on the maximum likelihood estimation (MLE) theory, this paper proposes a general statistical model for side-channel attack analysis that takes characteristics of both the physical implementation and cryptographic algorithm into consideration. The model establishes analytical relations between the success rate of attacks and the cryptographic system. For power analysis attacks, the side-channel characteristic of the physical implementation is modeled as signal-to-noise ratio (SNR), which is the ratio between the single-bit unit power consumption and the standard deviation of power distribution. The side-channel property of the cryptographic algorithm is extracted by a novel algorithmic confusion analysis. Experimental results of DPA and CPA on both DES and AES verify this model with high accuracy and demonstrate effectiveness of the algorithmic confusion analysis and SNR extraction. We expect the model to be extendable to other SCAs, like timing attacks, and would provide valuable guidelines for truly SCA-resilient system design and implementation.

Keywords: Side-channel attack, maximum likelihood estimation, success rate, DPA, CPA

* This work was supported in part by National Science Foundation under CAREER award CNS-0845871 and grant CNS-1314655.

1 Introduction

Nowadays, cryptographic primitives have been employed widely in various computer systems as the security engine. Despite the mathematical security strength of algorithms, it was found a decade ago that cryptosystems can be broken through exploiting system information leakage of their physical implementations, such as power consumption and timing information. These side channel attacks (SCAs) utilizing various leakage have posed serious realistic threats to many critical embedded systems. The most widely adopted SCAs are Differential Power Analysis (DPA) [1] and Correlation Power Analysis (CPA) [2]. They exploit the correlation between the intermediate data in algorithms and the power consumption of implementations to reveal sensitive information. Other variants of power analysis attacks presented include Mutual Information Attack (MIA) [3], Partitioning Power Analysis (PPA) [4], etc. Besides power consumption, leakage information like electromagnetic emanations [5, 6] and timing information [7] have also been exploited. Meanwhile, effective countermeasures at different design levels have been proposed [8–10].

Along with the research on side-channel attacks and countermeasures, common security metrics and standard evaluation methodology are another important line of research. Several generic metrics are proposed to evaluate the SCA resilience of a cryptosystem, including *number of measurements*, *success rate* [11, 12], *guessing entropy* [13] and *information theoretic metric* [13, 14]. Among them success rate is the ultimate metric that incorporates the effects of all factors including algorithms, implementations, and attacks. It is defined as the probability that a specific SCA succeeds under a certain leakage complexity. A low success rate for a SCA indicates the cryptosystem's high resilience against such SCA.

Intuitively, both the *cryptographic algorithm* and the *physical implementation* would affect the SCA resilience of a cryptosystem. Intrinsic features instilled in a cryptographic algorithm determines mathematically whether there exists SCA-related properties in the algorithm and to what extent. Physical implementation leaks a certain amount of SCA-related information, and the leaky quantity depends on how secure the system is designed and implemented. It is a challenging issue to accurately evaluate the influence of both the cryptographic algorithm and physical implementation on the system's SCA resilience.

Related Work: Although there has been some research efforts attempting to address the above issues, the effects of the algorithm and implementation on an side-channel attack were not clearly revealed, and a better quantitative model is needed to fully understand the interactions among algorithms, implementations, and attacks. Lacking of common metrics and standard evaluation methodologies has started to hinder the further development of side-channel attack research and practices. An approach is presented in [15] to model the signal-to-noise ratio (SNR) of DPA of a cryptographic system, without showing how the SNR determines the SCA resilience. In [16], the DPA efficiency is improved by analyzing the relation between the difference-of-means power consumption and key hypotheses, without considering the characteristics of the algorithm. In [17] and [18], a statistical model for CPA is presented, which does not take the correlation

between different keys into account and thus is inaccurate (see analysis in Section 4). Rivain [19] derived the success rate formula that takes into account the correlation among keys. However, his formula does not specify the relationship between SCA characteristics of the implementation and the cryptographic algorithm. A unified framework for SCA security evaluation based on information theory is presented in [13] with a security metric, mutual information, proposed. However, the framework lacks quantitative analysis between the security metric and success rate. Our study has found that the mutual information is just one factor affecting the success rate, and we have found other contributing factors explicitly. Work in [20] discusses the DPA-related behavior of SBox at algorithm level and introduces a new notion of *transparency order of an SBox*, without considering the implementation aspect.

Our contributions: This paper proposes a general statistical model for side-channel attack analysis, giving an explicit success rate formula based on maximum likelihood (ML) estimation. The model will provide better understanding of side-channel attacks on cryptosystems, and therefore more effective and efficient evaluation methods. The success rate formula is the first one to explicitly decouple contributions from physical implementation and cryptographic algorithm on the leakage. The SCA characteristic of physical implementation is represented by *signal-to-noise ratio* (SNR), which is the ratio between the single-bit unit power consumption and the standard deviation of power leakage. The SCA-related property of a cryptographic algorithm is characterized by confusion coefficients. *Algorithmic confusion analysis* was first introduced by us in [21] to obtain confusion coefficients for DPA model, and then used in [22] to derive the relation between the success rate of DPA and the confusion coefficients. This paper extends the definition of confusion coefficients for a general Gaussian leakage model, with the popular DPA and CPA models just as special cases. Confusion matrices are generated to measure the statistical correlation between different key candidates. The success rate formula provides a bound on the effectiveness of a side channel attack on a cryptosystem under a given leakage model. The DPA and CPA attacks are shown to be equivalent to ML-attacks with unknown system parameters. The explicit success rate formula facilitates application of multi-stage procedures combining SCAs on subkeys to recover the full key [23], and can also be useful for security analysis of leakage-resilience schemes where the security bounds on the subkeys are needed to derive the overall security metric for the total system.

The rest of the paper is organized as follows. Section 2.2 first reviews the typical SCA procedure, then presents our basic algorithmic confusion analysis for DPA, and extends the confusion analysis for general leakage models. Section 3 proposes our statistical model for success rate, and its application to DPA and CPA. The model is verified with experimental results on DES and AES in Section 4. Section 5 discusses more implications of the model and its possible applications. Finally conclusions are drawn in Section 6.

2 Preliminaries

This section will present background on SCAs first and then algorithmic confusion analysis for DPA and more general leakage models.

2.1 SCA Procedure

All SCAs undergo a common hypothesis test procedure as shown in Fig. 1. We next exemplify the procedure with the earliest discovered and fundamental DPA practice.

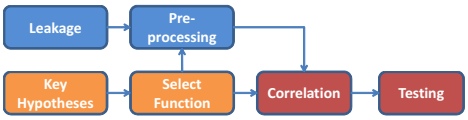

Fig. 1. Hypothesis test of SCA

- *Leakage* refers to the physical side-channel measurements L, e.g., waveforms of power consumption collected from the target device. Denote the *leakage population* as $\mathcal{L} = \{l_1, \ldots, l_n\}$, where l_m (m=1, 2, …, n) is a leakage trace measurement with a certain input, and n is the number of measurements. Denote N_M as the size of the input space. Successful SCA is a sampling process with $n << N_M$. Each l_m is a time series with number of p points, $l_m = \{l_{m,1}, \ldots, l_{m,p}\}$.
- *Key hypotheses* enumerate all possible values of the subkey k under attack, i.e., N_k candidates. Many symmetric block ciphers feature parallel computation over subkeys and plaintext blocks, and therefore SCAs on such algorithms can take the divide-and-conquer method to recover the subkeys one by one. Throughout the paper we assume that SCAs recover subkeys, and leave full key recovery for complexity analysis later.
- *Select function* V is a function of *intermediate data* d which is dependent on both the input (known plaintext or ciphertext), x, and the key, k, denoted as $V = \psi(d) = \psi(x, k)$. For example, the intermediate data d can be the output of a selected SBox. In DPA, V is a single bit of d, b_d, at value 1 or 0. In CPA, V can be a h-bit subset of d, with possible value in the range of $\{0, \ldots, h\}$. The leakage population has to go through a pre-processing stage to align the traces, reduce noises, and select the time points of interest (PoIs) on the trace that correspond to the select function.
- *Correlation* between the leakage $\mathcal{L}$ and the select function V under each key hypothesis is computed for a specific attack. The correlation for DPA is the difference-of-means (DOM) δ, i.e., the difference between the average power

consumption of the two waveform groups partitioned with $V = 1$ and 0 under a key guess. DOM is defined as:

$$\delta = \frac{\sum \mathcal{L}_{V=1}}{N_{V=1}} - \frac{\sum \mathcal{L}_{V=0}}{N_{V=0}} \tag{1}$$

where $N_{V=1}$ is the number of leakage measurements with $V = 1$, and $N_{V=0}$ the number of measurements with $V = 0$, $N_{V=1} + N_{V=0} = n$ and n is the total number of measurements. If the pre-processing keeps the whole or part of the power trace, the DOM for a key guess is also a time series. The correlation for CPA is the Pearson correlation [2], which can be written as:

$$\rho = \frac{E\{[V - E(V)][\mathcal{L} - E(\mathcal{L})]\}}{\sqrt{D(V)}\sqrt{D(\mathcal{L})}} \tag{2}$$

where $D(V)$ and $D(\mathcal{L})$ are the variance of the select function and measurements, respectively, and $E\{[V - E(V)][\mathcal{L} - E(\mathcal{L})]\}$ is the covariance between them.

- *Testing* with the maximum likelihood method chooses the key hypothesis with the maximum correlation as the correct key. In DPA, given sufficient number of measurements, the peak DOM δ_c for the correct key k_c converges to the unit physical power consumption ε related to the bit b_d under attack, written as $\lim_{n\to\infty} \delta_c = \varepsilon$, while the DOMs for incorrect keys are all much smaller, and therefore the correct key is distinguished. In CPA, the correlation used in testing is the Pearson correlation factor ρ. The correct key guess should yield the largest ρ, approaching 1.

2.2 DPA Algorithmic Confusion Analysis

As described above, a SCA utilizes the leakage related to the select function $V = \psi(x, k)$. Two key hypotheses k_i and k_j have two corresponding $V|k_i$ and $V|k_j$. The behavior of $V = \psi(x, k)$ under different keys k_i and k_j affects how difficult it is for SCA to distinguish the keys using the leakage measurements. In DPA, the select function is a single bit and V has only two possible outcomes 0 and 1. The probability that $V|k_i$ is different or the same with $V|k_j$ reveals DPA-related property of the cryptographic algorithm.

Our previous work [21] defines the *confusion coefficient* κ for DPA over two keys (k_i, k_j) as:

$$\kappa = \kappa(k_i, k_j) = \Pr\left[(V|k_i) \neq (V|k_j)\right] = \frac{N_{(V|k_i)\neq(V|k_j)}}{N_t} \tag{3}$$

where N_t is the total number of values for the input x, and $N_{(V|k_i)\neq(V|k_j)}$ is the number of occurrences (inputs) for which key hypotheses k_i and k_j result in different V values. For three keys k_h, k_i and k_j, we further define a *three-way confusion coefficient*:

$$\begin{aligned} \tilde{\kappa} &= \tilde{\kappa}(k_h, k_i, k_j) \\ &= \Pr\left[(V|k_i) = (V|k_j), (V|k_h) \neq (V|k_i)\right]. \end{aligned} \tag{4}$$

The three-way confusion coefficients are related to the two-way confusion coefficients by the following Lemma which is proven in Appendix A (note the proof is for the generalized three-way and two-way confusion coefficients introduced in the following section, and also applies to the specialized DPA confusion coefficients).

Lemma 1

$$\tilde{\kappa}(k_h, k_i, k_j) = \frac{1}{2}[\kappa(k_h, k_i) + \kappa(k_h, k_j) - \kappa(k_i, k_j)].$$

2.3 General Algorithmic Confusion Analysis

Here we first extend the confusion coefficients to more general settings. In Section 3, we then show that under a Gaussian leakage model, how the success rate of the strongest attack (ML-attack) is decided by the confusion coefficients $\kappa(k_i, k_j)$ and the system side-channel signal-to-noise-ratio (SNR). The popular DPA and CPA models become two special cases in this general approach.

In the DPA model above, the V has only two possible outcomes 0 and 1, and therefore the probability that V differs under two different keys k_i and k_j captures the confusion property. In general, V can take more than two values. We measurer the difference between the V values under the two keys by the expectation of their squared distance. That is, we define a general two-way confusion coefficient as:

$$\kappa(k_i, k_j) = E[(V|k_i - V|k_j)^2]. \tag{5}$$

Under the DPA model, $E[(V|k_i - V|k_j)^2]$ becomes $\Pr[(V|k_i) \neq (V|k_j)]$, showing that the generalized definition (5) agrees with the special definition (3).

Similarly, we define two generalized three-way confusion coefficients as:

$$\tilde{\kappa}(k_h, k_i, k_j) = E[(V|k_h - V|k_i)(V|k_h - V|k_j)] \tag{6}$$

$$\tilde{\kappa}^*(k_h, k_i, k_j) = E[(V|k_h - V|k_i)^2 (V|k_h - V|k_j)^2]. \tag{7}$$

It is easy to prove that definitions (6) and (7) both reduce to (4) under the DPA model.

For the nonlinear SBoxes in commonly used block ciphers such as DES and AES, with each key, the output of the SBox follows the same uniform distribution for uniformly distributed plaintext input x. That is, the select function V distribution is uniform and key-independent as stated below.

Assumption 1 *(Symmetric Key Assumption) For randomly uniformly distributed plaintext x, the intermediate variable V has the same distribution under all keys. That is, $V|k_c \stackrel{d}{=} V|k_{gi}$, $i = 1, ..., N_k - 1$, where $\stackrel{d}{=}$ denotes that the two random variables follow the same probability distribution.*

Under this assumption, our general three-way confusion coefficients $\tilde{\kappa}(k_h, k_i, k_j)$ are related to two-way coefficients: $\tilde{\kappa}(k_h, k_i, k_j) = \frac{1}{2}[\kappa(k_h, k_i) + \kappa(k_h, k_j) - \kappa(k_i, k_j)]$, in the same way as in DPA model. The other three-way confusion coefficients

$\tilde{\kappa}^*(k_h, k_i, k_j)$, however, cannot be explicitly related to two-way coefficients due to its higher-order definition. We will see the usage of the two different three-way confusion coefficients in Section 3.

We now apply the algorithmic confusion analysis to check the SCA-related property of SBoxes. We first take DES as an example. The DES SBox has a 6-bit input and 4-bit output, and the subkey used is 6-bit. Therefore, there are a total of $2^6 \times (2^6 - 1)/2 = 2016$ confusion coefficients $\kappa(k_i, k_j)$ for a select function on an SBox. For DPA on the first bit of the first DES SBox, the confusion coefficients fall into following nine values:

$$\{0.25, 0.3125, 0.375, 0.4375, 0.5, 0.5625, 0.625, 0.6875, 0.75\}.$$

We define these values as *characteristic confusion values* of a DES SBox. We believe they manifest some important SCA-related properties of the SBoxes. The distribution of confusion coefficients is shown in Fig. 2.

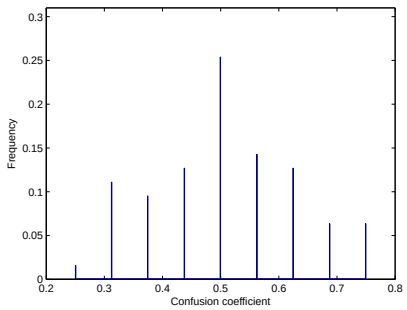

Fig. 2. Distribution of confusion coefficients $\kappa(k_i, k_j)$ of DPA on DES SBox.

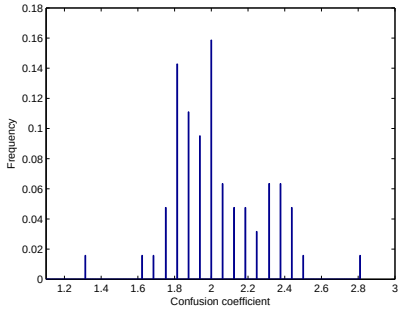

Fig. 3. Distribution of confusion coefficients $\kappa(k_i, k_j)$ of CPA on DES SBox.

Large confusion coefficients $\kappa(k_i, k_j)$ indicates that under keys k_i and k_j, the V values are different for a large portion of the inputs. Therefore, it is easier to distinguish keys k_i and k_j from the side-channel leakage measurements. Smaller $\kappa(k_i, k_j)$ value would make the two keys to be more resilient to side-channel attacks. However, small $\kappa(k_i, k_j)$ would mean that $\psi(x, k_i) = \psi(x, k_j)$ for most of the time. Therefore, $\psi(x, k)$ loses the encryption (diffusion) value for the two keys. For that reason, an ideal encryption algorithm should have $\kappa(k_i, k_j) = 0.5$ for DPA. We see that, for our select function, not all $\kappa(k_i, k_j) are 0.5$. So DPA would be more effective for distinguish some pairs of keys k_i and k_j with large $\kappa(k_i, k_j)$ than other pairs. The overall mean of all the confusion coefficients $\kappa(k_i, k_j)$ is 0.5.

Similarly, the distribution of confusion coefficients for a CPA on the first DES SBox is shown in Fig. 3, which is over 17 distinct values. In CPA on DES, the select function is the 4 bits of an SBox output and h=4. The overall mean of

$\kappa(k_i, k_j)$'s is 2, which is the value corresponding to the case when $\psi(x, k_i)$ and $\psi(x, k_j)$ are statistically uncorrelated Hamming weights. Again, certain pairs of keys k_i and k_j are easier to distinguish by the CPA than other pairs.

We also apply the confusion analysis to AES. Fig. 4 and Fig. 5 show the distribution of confusion coefficients for DPA and CPA, respectively, on a targeted AES SBox. The confusion coefficients for DPA on AES are also distributed over 9 values. We can see that these confusion coefficients are concentrated much closer to their mean values compared to DES. For example, for CPA on AES, the deviation of confusion coefficients is about 25% while that for CPA on DES is about 40%. This means the key candidates behave more similarly and randomly in AES than in DES, and therefore AES is harder to attack.

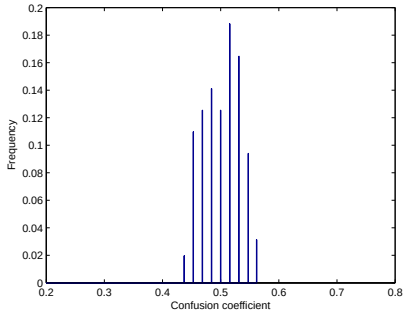

Fig. 4. Distribution of confusion coefficients $\kappa(k_i, k_j)$ of DPA on AES SBox.

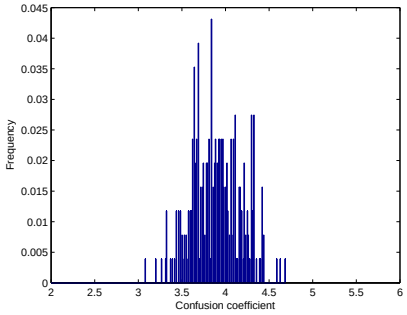

Fig. 5. Distribution of confusion coefficients $\kappa(k_i, k_j)$ of CPA on AES SBox.

3 Statistical Model for SCAs Using Maximum Likelihood Estimation

We first present a general statistical model for the maximum likelihood attack and introduce notations for the success rate formula. We then reveal its constituents of confusion coefficients and SNR under the Gaussian leakage model.

3.1 A General Statistic Model for Maximum Likelihood Attack

The target of side-channel attack is to distinguish the correct key k_c from all possible key hypotheses $k_g \in S$ based on n independent realizations of noisy physical leakage $l_1, l_2, \ldots, l_n \in L$. According to the Neyman-Pearson lemma [24], the most powerful distinguisher between two keys is the *maximum likelihood* (ML) attack. The general ML-attack maximizes the log-likelihood as the test statistic T:

$$\hat{k} = \arg\max_{k_g \in S} T_{k_g} = \arg\max_{k_g \in S} \frac{1}{n} \sum_{m=1}^{n} \log f_{L|k_g}(l_m) \tag{8}$$

where $f_{L|k}$ is the probability density function of L under a key guess k_g. MLE takes k_g as the estimated parameter if it yields the maximum probability of l_m under the probability density function $f_{L|k}$.

Dependent on the system implementation and attack, there are different power consumption models that correlate the leakage l with the select function v. In general, the power consumption contains both deterministic v-dependent components and random noise components. We will consider DPA and CPA models specifically in the following Sections 3.2 and 3.3. Here we first derive the general formula for the success rate of the ML-attack.

For the ML-attack to successfully distinguish the correct key k_c from other key hypotheses, it requires the log-likelihood of k_c to be larger than all other keys, written as

$$T_{k_c} > \{T_{\langle \overline{k_c} \rangle})\}$$

where $\langle \overline{k_c} \rangle$ denotes all the incorrect keys, i.e., $\{k_0, ..., k_{N_k-1}\}$ excluding k_c, and $\{T_{\langle \overline{k_c} \rangle}\}$ denotes the test statistics for other incorrect keys, i.e., $\{T_{k_0}, \ldots, T_{k_{N_k-1}}\}$ excluding T_{k_c}. The success rate to recover the correct key, SR, is defined as the probability that the test statistic for the correct key k_c, T_{k_c}, is larger than all $\{T_{\langle \overline{k_c} \rangle}\}$, i.e.,:

$$\mathrm{SR} = \mathrm{SR}\left[k_c, \langle \overline{k_c} \rangle\right] = \Pr\left[T_{k_c} > \{T_{\langle \overline{k_c} \rangle}\}\right] \tag{9}$$

The success rate is $(N_k - 1)$-dimensional. We next show the derivation of the success rate starting from the simple one-dimension success rate.

1-dimension success rate. We first consider the 1-dimension success rate, i.e., the success rate of k_c over an incorrect key k_g chosen out of $\langle \overline{k_c} \rangle$, written as:

$$\mathrm{SR}_1 = \mathrm{SR}\left[k_c, k_g\right] = \Pr\left[T_{k_c} > T_{k_g}\right] = \Pr\left[\Delta(k_c, k_g) > 0\right]$$

Here

$$\begin{aligned} \Delta(k_c, k_g) &= T_{k_c} - T_{k_g} \\ &= \tfrac{1}{n} \sum_{m=1}^{n} [\log f_{L|k_c}(l_m) - \log f_{L|k_g}(l_m)]. \end{aligned} \tag{10}$$

We denote $\Delta_1(k_c, k_g)$ for $\Delta(k_c, k_g)$ with only one leakage observation l_1, and the mean and variance of $\Delta_1(k_c, k_g)$ are given by:

$$\mu_{\Delta_1(k_c,k_g)} = E[\log f_{L|k_c}(l_1) - \log f_{L|k_g}(l_1)], \tag{11}$$

$$\sigma^2_{\Delta_1(k_c,k_g)} = Var[\log f_{L|k_c}(l_1) - \log f_{L|k_g}(l_1)]. \tag{12}$$

With n independently and identically selected power measurements, by the Central Limit Theorem [25], $[\Delta(k_c, k_g) - \mu_{\Delta(k_c,k_g)}]/\sigma_{\Delta(k_c,k_g)}$ converges in law to the standard Gaussian distribution $\mathcal{N}(0, 1)$, with $\mu_{\Delta(k_c,k_g)} = \mu_{\Delta_1(k_c,k_g)}$, $\sigma^2_{\Delta(k_c,k_g)} = \frac{1}{n}\sigma^2_{\Delta_1(k_c,k_g)}$. Let $\Phi(x) = \frac{1}{2}[1 + erf(\frac{x}{\sqrt{2}})]$ denote the cumulative distribution function (cdf) of the standard normal distribution, where erf(x) is the error function $erf(x) = \frac{2}{\sqrt{\pi}} \int_{-\infty}^{x} e^{-t^2/2} dt$. Then,

$$\begin{aligned} \mathrm{SR}_1 &= \Pr\left[\Delta(k_c, k_g) > 0\right] \\ &= 1 - \Phi(-\tfrac{\mu_{\Delta(k_c,k_g)}}{\sigma_{\Delta(k_c,k_g)}}) = \Phi(\tfrac{\mu_{\Delta(k_c,k_g)}}{\sigma_{\Delta(k_c,k_g)}}) \end{aligned} \tag{13}$$

Note that the Gaussian distribution here is the asymptotic limit of ML attack statistics coming from the Central Limit Theorem, and is independent of the noise distribution in the system leakage.

From equation (13), the asymptotic success rate of the ML-attack is always determined by the two quantities $\mu_{\Delta(k_c,k_g)}$ and $\sigma_{\Delta(k_c,k_g)}$. Under a linear power model with Gaussian noises, we will show how these two quantities are decided by the confusion coefficients and the SNR in the following subsections 3.2 and 3.3.

2-dimension success rate. Next we consider the 2-dimension success rate, i.e., the success rate of k_c over any two chosen keys k_{g1} and k_{g2} out of $\langle\overline{k_c}\rangle$, written as

$$\begin{aligned}\mathrm{SR}_2 = \mathrm{SR}\left[k_c, \{k_{g1}, k_{g2}\}\right] &= \Pr\left[T_{k_c} > T_{k_{g1}}, T_{k_c} > T_{k_{g2}}\right] \\ &= \Pr\left[y_1 > 0, y_2 > 0\right]\end{aligned}$$

where

$$\begin{aligned}y_1 &= \Delta(k_c, k_{g1}) = (T_{k_c} - T_{k_{g1}}), \\ y_2 &= \Delta(k_c, k_{g2}) = (T_{k_c} - T_{k_{g2}}).\end{aligned}$$

By the multivariate Central Limit Theorem, the random vector $Y_2 = [y_1, y_2]^T$ converges in law to the two-dimension normal distribution $\mathcal{N}(\mu_2, \Sigma_2)$, with

$$\boldsymbol{\mu}_2 = \begin{bmatrix}\mu_{y_1} \\ \mu_{y_2}\end{bmatrix}, \qquad \boldsymbol{\Sigma}_2 = \begin{bmatrix}\mathrm{Cov}(y_1, y_1)\ \mathrm{Cov}(y_1, y_2) \\ \mathrm{Cov}(y_1, y_2)\ \mathrm{Cov}(y_2, y_2)\end{bmatrix}.$$

Here and below, ^{T} denotes the transpose of the vector.

Let $\Phi_2(\boldsymbol{x})$ denote the cdf of the 2-dimensional standard normal distribution. Then we have

$$\mathrm{SR}_2 = \Phi_2({\boldsymbol{\Sigma}_2}^{-1/2}\boldsymbol{\mu}_2). \tag{14}$$

When $\mathrm{Cov}(y_1, y_2) = 0$, the 2-dimension success rate is simply the product of the two 1-dimension success rates $SR_1(k_c, k_{g1})SR_1(k_c, k_{g2})$. This mistaken assumption has been commonly used in prior work, for example, in [17] and [18]. However, generally the tests statistics $\Delta(k_c, k_{g1})$ and $\Delta(k_c, k_{g2})$ are correlated and $\mathrm{Cov}(y_1, y_2) \neq 0$.

($\mathbf{N_k - 1}$)-dimension success rate. The overall success rate is the success rate of k_c over all other $(N_k - 1)$ keys $\langle\overline{k_c}\rangle$,

$$\mathrm{SR} = \mathrm{SR}_{N_k-1} = \mathrm{SR}\left[k_c, \langle\overline{k_c}\rangle\right] = \Pr\left[T_{k_c} > \{T_{\langle\overline{k_c}\rangle}\}\right] = \Pr\left[Y > 0\right]$$

where Y is the $(N_k - 1)$-dimension vector

$$Y = \boldsymbol{\Delta} = [\Delta(k_c, k_0), \ldots, \Delta(k_c, k_c - 1), \Delta(k_c, k_c + 1), \ldots, \Delta(k_c, k_{N_k-1})] \tag{15}$$

with elements $\Delta(k_c, k_g)$ defined in (10). We denote $\boldsymbol{\Delta}_1$ as $\boldsymbol{\Delta}$ with only one leakage observation l_1, and the mean and variance of $\boldsymbol{\Delta}_1$ are a $1 \times (N_k - 1)$ vector,

$\boldsymbol{\mu}$, and a $(N_k-1)\times(N_k-1)$ matrix, $\boldsymbol{\Sigma}$, respectively. With n independently and identically selected power measurements, $\boldsymbol{\Delta}$ converges in law to the (N_k-1)-dimensional Gaussian distribution, $N(\boldsymbol{\mu}, \boldsymbol{\Sigma}/n)$. So the overall success rate of the ML-attack can be defined as the probability that every element in $Y = \boldsymbol{\Delta}$ is non-negative with given n, which can be expressed as:

$$SR = \Phi_{N_k-1}(\sqrt{n}\boldsymbol{\Sigma}^{-1/2}\boldsymbol{\mu}) \tag{16}$$

where $\Phi_{N_k-1}(\boldsymbol{x})$ is the cumulative distribution function of the (N_k-1)-dimensional standard Gaussian distribution. Note that this multivariate Gaussian distribution is the asymptotic limit of ML attack statistics coming from the Central Limit Theorem, and is independent of the noise distribution in the system leakage. Equation (16) holds generally for any SCA, while the mean vector $\boldsymbol{\mu}$ and variance matrix $\boldsymbol{\Sigma}$ would be different for different power leakage models.

Formula (16) provides a general security metric against an SCA. With it, SCA security evaluation is reduced to calculation of the mean vector $\boldsymbol{\mu}$ and variance matrix $\boldsymbol{\Sigma}$. The element of $\boldsymbol{\mu}$, $\mu_{\Delta(k_c,k_g)} = E_{L|k_c}\{\log[f_{L|k_c}(l_1)] - \log[f_{L|k_g}(l_1)]\}$, is the relative entropy (also called Kullback-Leibler divergence [26]) of the leakage distribution under the correct key k_c to the leakage distribution under a guessed key k_g. This is similar to the conditional entropy defined in [13]. The mutual information analysis in [13] solely depends on the conditional entrioy, while our security evaluation against SCA also includes the effect of the variance matrix $\boldsymbol{\Sigma}$. For side-channel attack analysis under a general leakage model, the conditional density function $f_{L|k_i}(\cdot)$ has to be estimated for all keys k_i. Next we show that $\boldsymbol{\mu}$ and $\boldsymbol{\Sigma}$ can be expressed in a closed form consisting of confusion coefficients and system SNR under a Gaussian leakage model.

3.2 Statistical Model for DPA

We focus on a widely used power consumption model with additive Gaussian noises for both DPA and CPA,

$$l_m = \epsilon v_m + c + r_m, \qquad m = 1, ..., n. \tag{17}$$

where l_m is the power leakage measurement, c and ϵ are unknown constants, $v_m = \psi(x_m, k_c)$ is the select function, and r_m is the random noise coming from circuitry and measurement, following a Gaussian distribution $N(0, \sigma^2)$. Under this power model, the $f_{L|k_g}(l_m)$ in (8) is the probability density function for $N(c+\epsilon\psi(x_m, k_g), \sigma^2)$. Hence, the signal-noise-ratio (SNR) of the implementation is defined as ϵ/σ.

For the DPA model, the select function v_m in (17) is one single bit. Hence ϵ is the differential power value of this bit, as discussed in Section 2.2. We will first quantify the success rate of the ML-attack (16) in terms of SNR and confusion coefficients. Then we show that the commonly used DPA, difference-of-means (DOM) attack, is in fact a ML-attack with unknown SNR parameter values. Furthermore, the DPA asymptotically achieves the same success rate as ML-attack with known SNR value. This confirms that DPA is also asymptotically the strongest attack under the DPA leakage model.

For DPA, the entry of vector $\boldsymbol{\mu}$ (the mean of $\boldsymbol{\Delta_1}$) is:

$$\mu_{k_g} = E[\Delta_1(k_c, k_g)] = \frac{\epsilon^2}{2\sigma^2} E[(v_c - v_g)^2] = \frac{\kappa(k_c, k_g)}{2} (\frac{\epsilon}{\sigma})^2. \tag{18}$$

The ij-th element in the $(N_k - 1) \times (N_k - 1)$ dimensional variance matrix, $\boldsymbol{\Sigma}$, is (See Appendix B for proof):

$$\begin{aligned} \sigma^2_{k_{gi}, k_{gj}} &= \tilde{\kappa}(k_c, k_{gi}, k_{gj})(\tfrac{\epsilon}{\sigma})^2 \\ &\quad + \tfrac{1}{4}[\tilde{\kappa}(k_c, k_{gi}, k_{gj}) - \kappa(k_c, k_{gi})\kappa(k_c, k_{gj})](\tfrac{\epsilon}{\sigma})^4 \end{aligned} \tag{19}$$

where $\tilde{\kappa}(k_c, k_{gi}, k_{gj}) = Pr(V|k_{gi} = V|k_{gj}, V|k_c \neq V|k_{gi})$ is the three-way confusion coefficient defined in Equation (4).

We can formalize the above results in matrix terms. Let $\boldsymbol{\kappa}$ denote a $(N_k - 1)$-dimension *confusion vector* for the correct key k_c with entries $\kappa(k_c, k_{gi})$, $i = 1, ..., N_k - 1$; $\boldsymbol{\kappa}^T$ denotes the transpose of $\boldsymbol{\kappa}$, and $\boldsymbol{K}$ is the $(N_k - 1) \times (N_k - 1)$ *confusion matrix* of the cryptographic algorithm for k_c, with elements $\{\varkappa_{ij}\}$ as $\varkappa_{ij} = \tilde{\kappa}(k_c, k_{gi}, k_{gj})$. When $i = j$, $\varkappa_{ii} = \tilde{\kappa}(k_c, k_{gi}, k_{gi}) = \kappa(k_c, k_{gi})$.

The confusion matrix $\boldsymbol{K}$ fully depicts the relation between all the key candidates (i.e., the algorithm) and how they affect the success rate. Summarizing (18) and (19) in matrix form, we arrive at the following theorem.

Theorem 1 *Under the DPA model,*

$$\boldsymbol{\mu} = \frac{1}{2}(\frac{\epsilon}{\sigma})^2 \boldsymbol{\kappa}; \qquad \boldsymbol{\Sigma} = (\frac{\epsilon}{\sigma})^2 \mathbf{K} + \frac{1}{4}(\frac{\epsilon}{\sigma})^4 (\mathbf{K} - \boldsymbol{\kappa}\boldsymbol{\kappa}^T). \tag{20}$$

The success rate of the ML-attack under DPA model is given by

$$SR = \Phi_{N_k - 1}\{\sqrt{n}\frac{\epsilon}{2\sigma}[\mathbf{K} + (\frac{\epsilon}{2\sigma})^2(\mathbf{K} - \boldsymbol{\kappa}\boldsymbol{\kappa}^T)]^{-1/2}\boldsymbol{\kappa}\}. \tag{21}$$

The detailed proof of the Theorem 1 is given in Appendix B.

The general ML-attack (8) under the DPA model uses the test-statistic T_k assuming the parameters (c, ε, σ) known. In practice, the attacker does not know the value of (c, ε, σ), and the common DPA instead uses the distance-of-means (DOM) statistics δ_k in (1). That is, DPA selects the key guess maximizing the DOM δ_k as the correct key.

For DPA to succeed, $\delta_{k_c} > \delta_{k_g}$ holds for all $k_g \neq k_c$. We can define a $(N_k - 1)$-dimension vector for DPA similar to (15):

$$Y_{DOM} = [\delta_{k_c} - \delta_{k_{g0}}, \delta_{k_c} - \delta_{k_{g1}}, \ldots, \delta_{k_c} - \delta_{k_{gN_k - 1}}]. \tag{22}$$

It is easy to see that the element of Y_{DOM} has mean that can be expressed by the confusion coefficients:

$$\tilde{\mu}_{k_g} = E[\delta_{k_c} - \delta_{k_g}] = 2\epsilon\kappa(k_c, k_g). \tag{23}$$

The variance of Y_{DOM} can also be expressed in terms of confusion coefficients. Using the Central Limit Theorem, the asymptotic success rate of DPA can be calculated from the mean and variance of Y_{DOM}. We summarize the relation between the DPA attack and the ML attack in the following Theorem.

Theorem 2 *With the DPA model,*

- **(A)** *The DPA is asymptotically equivalent with the ML-attack with parameters* (c, ε, σ) *values unknown.*
- **(B)** *The asymptotic success rate of DPA also follows (21)*

$$SR = \Phi_{N_k-1}\{\sqrt{n}\frac{\epsilon}{2\sigma}[\mathbf{K} + (\frac{\epsilon}{2\sigma})^2(\mathbf{K} - \boldsymbol{\kappa}\boldsymbol{\kappa}^T)]^{-1/2}\boldsymbol{\kappa}\}.$$

Our prior work in CHES 2012 [22] proved the success rate formula (21) for DPA specifically. Instead of repeating the proof, we take an alternative general approach here. Theorem 2 can be considered as a special case of Theorem 4 for CPA in the next subsection, and we will give the proof of Theorem 4 in Appendix C. Since DPA (DOM attack) achieves the same success rate as the ML-attack with known SNR value, it is the strongest attack under the DPA leakage model (17).

The DPA success rate (21) is determined by two components. One is $\kappa(k_c, k_g)$, which is only related to the algorithm (and the select function in the attack); and the other one is ϵ/σ, which is defined as the *signal-to-noise ratio* (SNR) of the side-channel leakage and is only determined by the physical implementation.

3.3 Statistical Model for CPA

We now consider the Gaussian leakage model for CPA, $l_m = \epsilon v_m + c + r_m$, where the intermediate value $V = \psi(x, k)$ is the Hamming distance (or Hamming weight) of multiple SBox output bits in contrast to a single bit in the DPA model, and c, ϵ, r_m are the same parameters as in the leakage model for DPA.

Similar to the derivation of the statistical model for DPA above, we shall first show that the ML-attack success rate under CPA model has a similar expression as Equation (21) with generalized confusion coefficients. We then study the relationship between the ML-attack under the CPA model and the common CPA attack.

We also define a confusion vector $\boldsymbol{\kappa}$ for CPA model, a $(N_k - 1)$-dimensional vector with element $\kappa(k_c, k_g)$ in (5) where $k_i = k_c$ and $k_j = k_g$. We define two $(N_k-1)\times(N_k-1)$ dimensional confusion matrices, $\mathbf{K}$ and $\mathbf{K}^*$ with their elements $\{\varkappa_{ij}\}$ and $\{\varkappa^*_{ij}\}$ as the three-way confusion coefficients in (6) and (7):

$$\varkappa_{ij} = \tilde{\kappa}(k_c, k_{gi}, k_{gj}) \tag{24}$$

$$\varkappa^*_{ij} = \tilde{\kappa}^*(k_c, k_{gi}, k_{gj}) \tag{25}$$

Theorem 3 *Under the CPA model,*

$$\boldsymbol{\mu} = \frac{1}{2}(\frac{\epsilon}{\sigma})^2\boldsymbol{\kappa}; \qquad \boldsymbol{\Sigma} = (\frac{\epsilon}{\sigma})^2\mathbf{K} + \frac{1}{4}(\frac{\epsilon}{\sigma})^4(\mathbf{K}^* - \boldsymbol{\kappa}\boldsymbol{\kappa}^T). \tag{26}$$

The success rate of ML-attack is

$$SR = \Phi_{N_k-1}\{\sqrt{n}\frac{\epsilon}{2\sigma}[\mathbf{K} + (\frac{\epsilon}{2\sigma})^2(\mathbf{K}^* - \boldsymbol{\kappa}\boldsymbol{\kappa}^T)]^{-1/2}\boldsymbol{\kappa}\}. \tag{27}$$

The proof of Theorem 3 is provided in Appendix B.

The success rate of the ML-attack above provides an asymptotic upper bound for the leakage under the CPA power model (17). In practice, the parameters (c, ε, σ) are not known. The realistic CPA attack chooses the key that maximizes the Pearson correlation ρ [2]. This is in contrast to DPA which maximizes the difference of means (DOM).

We then summarize the property of the common CPA attack in the following Theorem.

Theorem 4 *With the CPA power leakage model, we have:*

- **(A)** *The CPA is equivalent to the ML-attack with parameters (c, ε, σ) values unknown.*
- **(B)** *Under the Symmetric Key Assumption, the asymptotic success rate of CPA is given by*

$$SR = \Phi_{N_k-1}\{\sqrt{n}\frac{\epsilon}{2\sigma}[\mathbf{K} + (\frac{\epsilon}{2\sigma})^2(\mathbf{K}^{**} - \boldsymbol{\kappa}\boldsymbol{\kappa}^T)]^{-1/2}\boldsymbol{\kappa}\}. \tag{28}$$

Here $\mathbf{K}^{**}$ is another $(N_k - 1) \times (N_k - 1)$ dimensional confusion matrix with elements:

$$\begin{aligned} \varkappa_{ij}^{**} &= \kappa^{**}(k_c, k_{gi}, k_{gj}) \\ &= E[4(V|k_c - E(V|k_c))^2(V|k_c - V|k_{gi})(V|k_c - V|k_{gj})]. \end{aligned} \tag{29}$$

The proof of Theorem 4 is provided in Appendix C.

Rivain et al. [19] showed that the CPA success rate also follows the general formula $\Phi_{N_k-1}(\sqrt{n}\boldsymbol{\Sigma}^{-1/2}\boldsymbol{\mu})$ in (16). However, there was no explicit analytic formula for $\boldsymbol{\Sigma}$ and $\boldsymbol{\mu}$ given. With our algorithmic confusion analysis, formula (28) analytically specifies these quantities asymptotically.

While the DPA and the ML-attack under the DPA model achieve the same success rate (21), the CPA's success rate (28) is slightly different from the ML-attack's success rate under the CPA model (27). Note that for DPA model, $h = 1$, $E(V|k_c) = 1/2$. Therefore, $4[V|k_c - E(V|k_c)]^2 = 1$ always. This implies that $\mathbf{K}^{**} = \mathbf{K}^{*} = \mathbf{K}$ under the DPA model.

However, when the SNR $\frac{\epsilon}{\sigma}$ is small (less than one), both success rate formulas (27) and (28) can be simplified to:

$$\Phi_{N_k-1}\{\sqrt{n}\frac{\epsilon}{2\sigma}\mathbf{K}^{-1/2}\boldsymbol{\kappa}\}. \tag{30}$$

That is, for small SNR, CPA achieves the same success rate asymptotically as the ML-attack with known (c, ϵ, σ). When the SNR is big, formula (28) for CPA is different from (27) for ML-attack of the CPA power model. However, both formulas are asymptotic with Central Limit Theorem and only hold for large sample size n. With a large SNR, the success rates (27) and (28) both converge to 1 rapidly as n increases. For small n, neither formula is meaningful for CPA with the Gaussian distribution of the test statistic not holding.

4 Experimental Results

We now evaluate the statistical models for DPA and CPA on both DES and AES algorithms.

4.1 DPA and CPA on DES

With the DES data set from DPAcontest [27], secmatv1, we performed both DPA and CPA on it. The select function for DPA involves the first output bit of the first SBox in the last round, while CPA involves all the 4 bits of the first SBox output. We take the maximum DOM value obtained from the DPA as ϵ, and the corresponding key is the correct key k_c, which is k_{60}. Note that all the attacks are on a single time point. Discussions on multi-point leakage are beyond this paper, and will be investigated in future work.

The empirical success rate are generated with 1000 trials for both the DPA and the CPA as in [11, 12]. 5 groups of key guesses are chosen to verify our model under different dimensions, which are $\mathrm{SR}_1 = \mathrm{SR}(k_c, k_0)$, $\mathrm{SR}_2 = \mathrm{SR}(k_c, \{k_0, k_1\})$, $\mathrm{SR}_8 = \mathrm{SR}(k_c, \{k_0, \ldots, k_7\})$, and the overall $\mathrm{SR}_{63} = \mathrm{SR}(k_c, \langle \overline{k_c} \rangle)$. For each group in DPA, a successful trial will be recorded only when the DOM value ϵ of k_c is larger than the DOM values of all other key guesses; while for CPA, a successful trial occurs when the Pearson correlation factor of k_c is the largest one.

To compute the theoretical success rate, we first calculate the physical implementation parameter SNR=ϵ/σ. σ is the standard deviation of the noise in the power leakage, which is $(l_m - \epsilon v_m - c)$ part in power measurements. Both DPA and CPA share the same ϵ value here since ϵ is the power difference of one bit transition of a real DES implementation. For DES, the attack is conducted on the 15750-th time point of the power trace. For DPA, $\epsilon = 0.0016$ and $\sigma = 0.0046$ so that SNR=0.347; while for CPA on DES $\sigma = 0.0048$ and SNR=0.333.

Second we find the confusion vector $\boldsymbol{\kappa}$ and confusion matrices $\mathbf{K}$, $\mathbf{K}^*$ defined earlier. With the two-way confusion coefficients $\kappa(k_c, k_g)$ and three-way $\tilde{\kappa}(k_c, k_{gi}, k_{gj})$ and $\tilde{\kappa}^*(k_c, k_{gi}, k_{gj})$ all algorithm-dependent only, they are easily calculated according to Equations (5), (6) and (7).

The confusion matrix $\mathbf{K}$ for DPA on the first bit of the first DES SBox, given in Equation (20), is shown in Fig. 6; and $\mathbf{K}$ of CPA given in Equation (26) in Fig. 8. By definition, the matrix $\mathbf{K}$ is a symmetric square matrix. Its diagonal elements are confusion coefficients $\kappa(k_c, k_{gi})$, i.e., the confusion vector $\boldsymbol{\kappa}$, whose mean value is 0.5 and 2 for DPA and CPA, respectively. We also plot the diagonal confusion vector in Fig. 7 and 9. The off-diagonal elements of the matrix $\mathbf{K}$ are the three-way confusion coefficients $\kappa(k_c, k_{gi}, k_{gj})$ whose mean value is 0.25 and 1, for DPA and CPA respectively.

Fig. 10 and Fig. 11 plot the empirical success rates (the solid curves) and theoretical success rates (the dashed curves) of our model for DPA (21) and CPA (27), respectively. We show the different dimensional success rates for $k_c = k_{60}$. From top down, they are: $\mathrm{SR}_1, \mathrm{SR}_2, \mathrm{SR}_8$, and SR_{63}. We can see that the two curves for SR_{63} track each other very well, showing the accuracy of our theoretical model. In this implementation, SNR is very small for DPA and CPA,

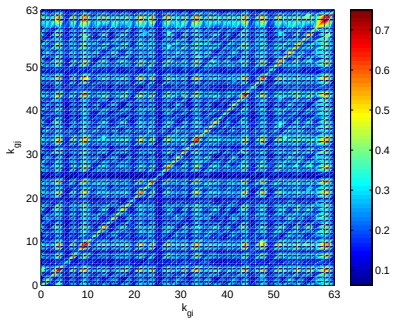

Fig. 6. The confusion matrix $\mathbf{K}$ of DPA on first bit of the first DES SBox.

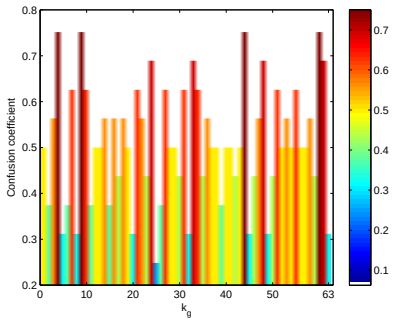

Fig. 7. The confusion vector $\boldsymbol{\kappa}$ of DPA on first bit of the first DES SBox.

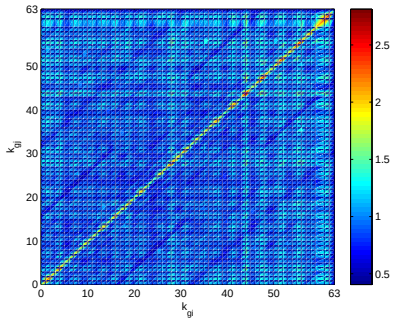

Fig. 8. The confusion matrix $\mathbf{K}$ of CPA on the first DES SBox.

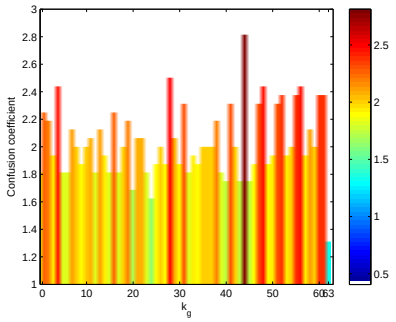

Fig. 9. The confusion vector $\boldsymbol{\kappa}$ of CPA on the first DES SBox.

and therefore the asymptotic success rate for the ML attacks with the DPA and CPA models agree with the empirical success rates.

We also plot the other existing explicit success rate formula in [17] and [18] as the dotted curves:

$$\mathrm{SR} = \left(\int_0^{\infty} \frac{1}{\frac{1}{\sqrt{N_m - 3}}\sqrt{2\pi}} \exp\left\{ -\frac{(x-r)^2}{\frac{2}{N_m - 3}} \right\} \mathrm{d}x \right)^{N_k - 1} \tag{31}$$

where r is the Pearson correlation of CPA for the correct key, N_k is the number of key guesses in CPA, and N_m is the number of measurements. This formula gives the correct 1-dimensional success rate, but the accuracy deteriorates for higher dimensional success rates. That is due to the fact that formula (31) does not account for correlations between attack statistics under different keys as shown before.

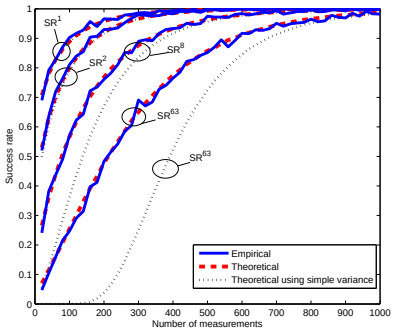

Fig. 10. Empirical and theoretical success rates of DPA on DES.

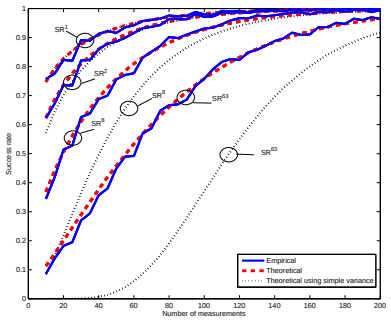

Fig. 11. Empirical and theoretical success rates of CPA on DES.

4.2 DPA and CPA on AES

We next perform DPA and CPA on an AES implementation. The select function for DPA is the Hamming distance of the third bit of the first state byte in the last round, and for CPA it is the entire state byte (8 bits). We have measured the power consumption data using the SASEBO GII board with AES implementation designated by DPAcontest [28]. The total number of measurements in the data set is $100,000$.

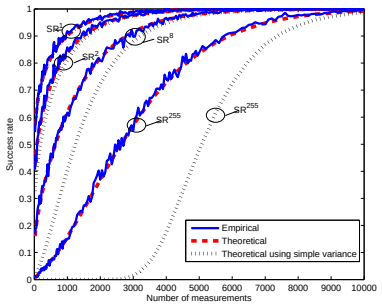

Fig. 12. Empirical and theoretical success rates of DPA on AES.

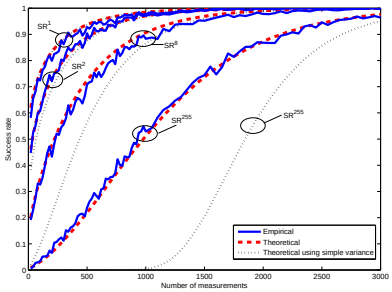

Fig. 13. Empirical and theoretical success rates of CPA on AES.

For AES, we conduct attacks on the 594-th time point of the power traces. Fig. 12 and Fig. 13 show the empirical success rates (solid curves) and theoretical success rates (dashed curves) of DPA and CPA on AES. The two 255-keys success rate curves of empirical and theoretical track each other very well, demonstrating that the model is also very accurate for AES.

5 Discussions

We believe our statistical analysis is the first of its kind to build a quantitative model for side-channel attack analysis on a cryptographic system, which is composed by system-inherent parameters, including ϵ, σ and κ. Next we present more SCA-related insights on implementation and algorithm from the model, and discuss how to use the model to evaluate countermeasures and algorithms.

5.1 Signal and Noise of the Side Channel

Cryptographic algorithms can be implemented on different hardware systems, including micro controller, Application Specific Integrated Circuit (ASIC), Field Programmable Gate Array (FPGA), and general purpose microprocessors. For any platform, we can always use the *signal-to-noise ratio* (SNR) defined as ϵ/σ to uniformly represent the side channel leakage. SNR is an essential parameter that affect the effectiveness of the ML attack, and Equations (21) and (27) show how the SNR determines the success rate. It can be used as a metric to measure the SCA resilience of the implementation of a cryptographic system. Our SNR definition is similar to that in [15, 29], however with more explicit quantitative implications in our model.

Common countermeasures against side-channel attacks include *random masking* [30–32], *power-balanced logic and algorithm* [33–35], and *hiding* (random delay) [36–39]. The effect of power balance logic/algorithm and random delay is straightforward with our model – reducing the implementation SNR, and the effect of random masking is reducing the algorithmic confusion coefficients.

5.2 Confusion Property of Cryptographic Algorithms

Our algorithmic confusion analysis reveals the inherent side-channel property of a cryptographic algorithm. Confusion coefficients are determined by both the cryptographic algorithm and select function ψ, and they indicate how differently the key candidates behave. Confusion coefficients have a direct effect on the success rate, as been illustrated in Equation (21) and (27) that larger confusion coefficients lead to higher success rates. For DPA, the select function in only one bit, and the mean value of the confusion coefficients is 0.5; while for CPA, the select function is 4 bits for DES and 8 bits for AES, so the mean values for the confusion coefficients are 2 and 4, respectively. The larger confusion coefficients coming from more bits in select function explain why CPA is more effective than DPA for a common algorithm. However, for different algorithms, even though CPA on AES has larger confusion coefficients than that of DES, the dimension of key space dominates over κ on the overall success rate. AES has 256 key candidates and the overall success rate is from the 255-dimension cumulative Gaussian distribution, making it more resilient than the 63-dimension success rate of DES.

Actually, the overall success rate can be evaluated in a pretty simple but faster way under certain conditions. We take CPA on AES as example here. If

the SNR is much smaller than 1, which is the case in the real SASEBO implementation, the $\mathbf{\Sigma}$ in Equation(20) and (26) is approximately $\mathbf{\Sigma} = (\frac{\epsilon}{\sigma})^2\mathbf{K}$. The success rate is then given by (30), $\Phi_{N_k-1}\{\sqrt{n}\frac{\epsilon}{2\sigma}\mathbf{K}^{-1/2}\boldsymbol{\kappa}\}$. Since the overall success rate will involve all the key candidates; even though they will yield different $\kappa(k_c, k_g)$, we can expect that the overall effect of κ in the success rate approaches its mean value. We next replace all the κ for CPA on AES with its mean value 4. For matrix $\mathbf{K}$, its diagonal values are the two-way coefficients with their means at 4, and the off-diagonal values are three-way coefficients with their means at 2, according to Lemma 1. This simplified success rate curve without calculating confusion coefficients (solid line) and the theoretical formula (30) (dash line) are both shown in Fig. 14. They track each other closely, demonstrating that statistical properties in AES algorithm have made its resilience to side-channel attack pretty key-independent.

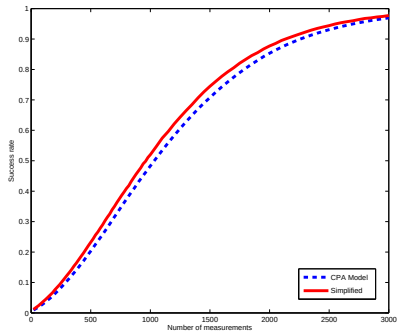

Fig. 14. Simplified and theoretical success rates of CPA on AES.

The experiments in Section 4.2 define the select function ψ for AES as the Hamming distance of two intermediate data due to the characteristics of ASIC implementation. In micro-controller implementation, the select function is defined directly as the Hamming weight of one intermediate data. A good select function for attacks gives larger confusion coefficient $\kappa(k_c, k_g)$ and therefore larger success rate. The algorithmic confusion analysis can also serve as a methodology to evaluate how good select functions are at distinguishing keys.

Our statistical model is built for linear power leakage model with Gaussian noises, which holds generally in most systems. Under this assumption, confusion coefficients are second moments of the distance between V values under a pair of keys, as defined in (5). A Gaussian distribution is totally determined by its first two moments. Therefore confusion coefficients contain all relevant information for the leakage analysis. If the noises are not Gaussian, then the algorithmic confusion analysis needs to take into the account of the joint distribution of

V values under a pair of keys, which is much harder to formulate. We will investigate this in the future work.

6 Conclusions

In this paper, a general theoretical model based on maximum likelihood estimation (MLE) is presented to evaluate the success rate for side-channel attacks on cryptographic systems. The model establishes the relation between the success rate and an algorithm and its implementation over a multivariate Gaussian distribution, with algorithmic confusion analysis illustrating the SCA-related inherent properties of the algorithm, and signal-to-noise ratio (SNR) indicating how resilient the physical implementation is. Our experimental results from DPA and CPA on DES and AES have verified this model. We believe that this model is innovative, provides valuable insights on side-channel characteristics of cryptosystems, and could significantly facilitate SCA-resilient design and implementations. The explicit formula is also useful for analyzing full-key recovery that combines attacks on multiple subkeys, as been adopted by the nascent CHES 2013 work [23].

References

1. P. C. Kocher, J. Jaffe, and B. Jun, "Differential power analysis," in *Proc. Int. Cryptology Conf. on Advances in Cryptology*, 1999, pp. 388–397.
2. E. Brier, C. Clavier, and F. Olivier, "Correlation power analysis with a leakage model," in *Int. Workshop on Cryptographic Hardware & Embedded Systems*, 2004, pp. 135–152.
3. B. Gierlichs, L. Batina, P. Tuyls, and B. Preneel, "Mutual information analysis," in *Int. Workshop Cryptographic Hardware & Embedded System*, 2008, pp. 426—442.
4. T.-H. Le, J. Clédière, C. Canovas, B. Robisson, C. Servière, and J.-L. Lacume, "A proposition for correlation power analysis enhancement," in *Int. Workshop on Cryptographic Hardware & Embedded Systems*, 2006, pp. 174–186.
5. J.-J. Quisquater and D. Samyde, "Electromagnetic analysis (EMA): Measures and counter-measures for smart cards," in *Smart Card Programming & Security*, 2001, pp. 200–210.
6. K. Gandolfi, C. Mourtel, and F. Olivier, "Electromagnetic analysis: Concrete results," in *Int. Workshop on Cryptographic Hardware & Embedded Systems*, 2001, pp. 251–261.
7. P. C. Kocher, "Timing attacks on implementations of Diffie-Hellman, RSA, DSS, and other systems," in *Proc. Int. Cryptology Conf. on Advances in Cryptology*, 1996, pp. 104–113.
8. S. Chari, C. S. Jutla, J. R. Rao, and P. Rohatgi, "Towards sound approaches to counter power analysis attacks," in *Proc. Crypto*, Aug. 1999, pp. 398–412.
9. K. Tiri and I. Verbauwhede, "A VLSI design flow for secure side-channel attack resistant ICs," in *Proc. Design, Automation & Test in Europe*, 2005, pp. 58–63.
10. C. Clavier, J.-S. Coron, and N. Dabbous, "Differential power analysis in the presence of hardware countermeasures," in *Int. Workshop on Cryptographic Hardware & Embedded Systems*, 2000, pp. 252–263.

11. B. Gierlichs, K. Lemke-Rust, and C. Paar, "Templates vs. stochastic methods: A performance analysis for side channel cryptanalysis," in *Int. Workshop on Cryptographic Hardware & Embedded Systems*, 2006, pp. 15–29.
12. F.-X. Standaert, P. Bulens, G. de Meulenaer, and N. Veyrat-Charvillon, "Improving the rules of the DPA contest," in *Cryptology ePrint Archive, Report 2008/517*, 2008, http://eprint.iacr.org/2008/517.
13. F.-X. Standaert, T. Malkin, and M. Yung, "A unified framework for the analysis of side-channel key recovery attacks," in *Advances in Cryptology – EUROCRYPT 2009*, 2009, pp. 443–461.
14. N. Veyrat-Charvillon and F.-X. Standaert, "Mutual information analysis: How, when and why?" in *Int. Workshop on Cryptographic Hardware & Embedded Systems*, 2008, pp. 429–443.
15. T. S. Messerges, E. A. Dabbish, and R. H. Sloan, "Examining smart-card security under the threat of power analysis attacks," *IEEE Trans. on Computers*, vol. 51, no. 5, pp. 541–552, 2002.
16. R. Bevan and E. Knudsen, "Ways to enhance differential power analysis," in *Int. Conf. on Information Security & Cryptology*, 2003, pp. 327–342.
17. S. Mangard, "Hardware countermeasures against DPA: A statistical analysis of their effectiveness," in *CT-RSA*, 2004, pp. 1988–1998.
18. F.-X. Standaert, E. Peeters, G. Rouvroy, and J. Quisquater, "An overview of power analysis attacks against field programmable gate arrays," *Proc. IEEE*, vol. 94, no. 2, 2006.
19. M. Rivain, "On the exact success rate of side channel analysis in the gaussian model," in *Selected Areas in Cryptography*, ser. Lecture Notes in Computer Science, R. Avanzi, L. Keliher, and F. Sica, Eds. Springer Berlin Heidelberg, 2009, vol. 5381, pp. 165–183.
20. E. Prouff, "DPA attacks and S-Boxes," in *Int. Workshop on Fast Software Encryption*, 2005, pp. 1–8.
21. Q. Luo and Y. Fei, "Algorithmic collision analysis for evaluating cryptographic systems and side-channel attacks," in *IEEE Int. Symp. Hardware Oriented Security & Trust*, 2011, pp. 75–80.
22. Y. Fei, Q. Luo, and A. A. Ding, "A statistical model for DPA with novel algorithmic confusion analysis," in *Int. WkShp on Cryptographic Hardware and Embedded Systems*, Sept. 2012, pp. 233–250.
23. E. P. A. Thillard and T. Roche, "Success through confidence: Evaluating the effectiveness of a side-channel attack," in *Int. WkShp on Cryptographic Hardware and Embedded Systems*, Sept. 2013, pp. 21–36.
24. J. Neyman and E. S. Pearson, "On the Problem of the Most Efficient Tests of Statistical Hypotheses," *Royal Society of London Philosophical Transactions Series A*, vol. 231, pp. 289–337, 1933.
25. O. T. Johnson, *Information Theory and the Central Limit Theorem*. Imperial College Press, 2004.
26. S. Kullback and R. A. Leibler, "On information and sufficiency," *Annals of Mathematical Statistics*, vol. 22, pp. 49–86, 1951.
27. DPA Contest. {http://www.dpacontest.org/}.
28. "Side-channel attack standard evaluation board (SASEBO)," Research Center for Information Security (RCIS).{http://www.rcis.aist.go.jp/special/SASEBO/index-en.html}.
29. S. Mangard, E. Oswald, and T. Popp, *Power Analysis Attacks: Revealing the Secrets of Smart Cards (Advances in Information Security)*. Springer-Verlag New York, 2007.

30. E. Oswald, S. Mangard, N. Pramstaller, and V. Rijmen, "A side-channel analysis resistant description of the AES S-box," in *Fast Software Encryption*, 2005, pp. 413–423.
31. D. Canright and L. Batina, "A very compact perfectly masked S-box for AES," in *Applied Cryptography & Network Security*, 2008, pp. 446–459.
32. K. Tiri and P. Schaumont, "Changing the odds against masked logic," in *Selected Areas in Cryptography*, 2007, pp. 134–146.
33. Z. Chen, A. Sinha, and P. Schaumont, "Implementing virtual secure circuit using a custom-instruction approach," in *Proc. Int. Conf. on Compilers, Architectures & Synthesis for Embedded Systems*, 2010, pp. 57–66.
34. K. Tiri, M. Akmal, and I. Verbauwhede, "A dynamic and differential CMOS logic with signal independent power consumption to withstand differential power analysis on smart cards," in *Proc. European Solid-State Circuits Conf.*, 2002, pp. 403–406.
35. K. Tiri and I. Verbauwhede, "A logic level design methodology for a secure DPA resistant ASIC or FPGA implementation," in *Proc. Int. Conf. on Design, Automation & Test in Europe*, 2004, pp. 246–251.
36. S. Yang, W. Wolf, N. Vijaykrishnan, D. Serpanos, and Y. Xie, "Power attack resistant cryptosystem design: A dynamic voltage and frequency switching approach," in *Proc. Int. Conf. Design Automation & Test in Europe*, 2005.
37. J. Coron and I. Kizhvatov, "An efficient method for random delay generation in embedded software," in *Int. Workshop on Cryptographic Hardware & Embedded Systems*, 2009, pp. 156–170.
38. ——, "Analysis and improvement of the random delay countermeasure of CHES 2009," in *Int. Workshop on Cryptographic Hardware & Embedded Systems*, 2011, pp. 95–109.
39. M. Bucci, R. Luzzi, M. Guglielmo, and A. Trifiletti, "A countermeasure against differential power analysis based on random delay insertion," in *Proc. IEEE Int. Symp. Circuits & Systems*, 2005, pp. 3547–3550.

Appendix

A Proof for Lemma 1

We prove the Lemma for the general confusion coefficients defined in (5) and (6). Then, of course, it also holds for the DPA confusion coefficients as a special case.

$$\begin{aligned}
&\kappa(k_h,k_i)+\kappa(k_h,k_j)-\kappa(k_i,k_j)\\
=&E[(V|k_h-V|k_i)^2+(V|k_h-V|k_j)^2-(V|k_i-V|k_j)^2]\\
=&E[2(V|k_h)^2-2(V|k_h)(V|k_i)-2(V|k_h)(V|k_j)+2(V|k_i)(V|k_j)]\\
=&2E[(V|k_h-V|k_i)(V|k_h-V|k_j)]\\
=&2\tilde{\kappa}(k_h,k_i,k_j).
\end{aligned}$$

Therefore: $\tilde{\kappa}(k_h,k_i,k_j)=\frac{1}{2}[\kappa(k_h,k_i)+\kappa(k_h,k_j)-\kappa(k_i,k_j)]$.

B Proof of Theorems 1 and 3

We shall make the derivation for the general leakage model (17) with Gaussian noise. This would express the general success rate formula in confusion coefficients as in Theorem 3. Then Theorem 1 can be further derived as a special case.

Since we already know the general success rate formula (16) for ML-attack, the proof only needs to verify the formula (26)for the mean $\boldsymbol{\mu}$ and variance $\boldsymbol{\Sigma}$. We shall do this by direction calculation.

We first find a simplified expression of $\Delta(k_c, k_g)$, the difference between ML-attack statistic for the correct key and a guessed key. From model (17), we have the likelihoods

$$f_{L|V}(l|v_c) = \frac{1}{\sqrt{2\pi}\sigma} e^{-\frac{(l-\epsilon v_c - c)^2}{2\sigma^2}}$$
$$f_{L|V}(l|v_g) = \frac{1}{\sqrt{2\pi}\sigma} e^{-\frac{(l-\epsilon v_g - c)^2}{2\sigma^2}}.$$

Therefore, using equation (10),

$$\begin{aligned}\Delta(k_c, k_g) &= \sum_{m=1}^{n} \frac{(l_m - c - \epsilon v_{m,g})^2 - (l_m - c - \epsilon v_{m,c})^2}{2n\sigma^2} \\ &= \frac{1}{2n\sigma^2} \sum_{m=1}^{n} \{[r_m + \epsilon(v_{m,c} - v_{m,g})]^2 - (r_m)^2\} \\ &= \frac{\epsilon^2}{2n\sigma^2} \sum_{m=1}^{n} [(v_{m,c} - v_{m,g})^2 + \frac{2}{\epsilon}(v_{m,c} - v_{m,g}) r_m].\end{aligned} \tag{32}$$

We now calculate the mean and variance of the vector $\boldsymbol{\Delta}$ from this expression to verify (26).

Since r_m has mean zero and is independent of $(v_{m,c} - v_{m,g})$, $E[(v_{m,c} - v_{m,g})r_m] = 0$. Hence the entry of vector $\boldsymbol{\mu}$ (the mean of $\boldsymbol{\Delta}$) is:

$$\mu_{k_g} = E[\Delta(k_c, k_g)] = \frac{\epsilon^2}{2n\sigma^2} n E[(v_{1,c} - v_{1,g})^2] = \frac{\kappa(k_c, k_g)}{2} (\frac{\epsilon}{\sigma})^2 \tag{33}$$

with $\kappa(k_c, k_g)$ defined as in (5). This verifies the first half of (26).

The entries in the $(N_k - 1) \times (N_k - 1)$ dimensional variance matrix, $\boldsymbol{\Sigma}$, are:

$$\begin{aligned}&Cov[\Delta(k_c, k_{gi}), \Delta(k_c, k_{gj})] \\ &= (\frac{\epsilon^2}{2n\sigma^2})^2 \sum_{m=1}^{n} \sum_{m^*=1}^{n} E\{[(v_{m,c} - v_{m,gi})^2 + \frac{2}{\epsilon}(v_{m,c} - v_{m,gi}) r_m] \\ &\qquad [(v_{m^*,c} - v_{m^*,gj})^2 + \frac{2}{\epsilon}(v_{m^*,c} - v_{m^*,gj}) r_{m^*}]\} \\ &\quad - E[\Delta(k_c, k_{gi})] E[\Delta(k_c, k_{gj})].\end{aligned}$$

Since $E(r_m) = E(r_{m^*}) = E(r_m r_{m^*}) = 0$ for $r_m \neq r_{m^*}$, and $E[(r_m)^2] = \sigma^2$, the above expression becomes

$$\begin{aligned}
&Cov[\Delta(k_c, k_{gi}), \Delta(k_c, k_{gj})] \\
&= (\tfrac{\epsilon^2}{2n\sigma^2})^2 \{ \sum_{m=1}^{n} \sum_{m^*=1}^{n} E[(v_{m,c} - v_{m,gi})^2 (v_{m^*,c} - v_{m^*,gj})^2] \\
&\qquad + \sum_{m=1}^{n} (\tfrac{2}{\epsilon})^2 \sigma^2 E[(v_{m,c} - v_{m,gi})(v_{m,c} - v_{m,gj})] \} \\
&\quad - \mu_{k_{gi}} \mu_{k_{gj}} \\
&= (\tfrac{\epsilon^2}{2n\sigma^2})^2 \{ \sum_{m=1}^{n} E[(v_{m,c} - v_{m,gi})^2 (v_{m,c} - v_{m,gj})^2] \\
&\qquad + \sum_{m \neq m^*}^{n} E[(v_{m,c} - v_{m,gi})^2] E[(v_{m^*,c} - v_{m^*,gj})^2] \\
&\qquad + \sum_{m=1}^{n} (\tfrac{2\sigma}{\epsilon})^2 E[(v_{m,c} - v_{m,gi})(v_{m,c} - v_{m,gj})] \} \\
&\quad - \tfrac{\kappa(k_c, k_{gi}) \kappa(k_c, k_{gj})}{4} (\tfrac{\epsilon}{\sigma})^4.
\end{aligned}$$

By the definition of the confusion coefficients $\kappa(k_c, k_{gi}, k_{gj}) = E[(V|k_c - V|k_{gi})(V|k_c - V|k_{gj})]$ in (6), and $\kappa^*(k_c, k_{gi}, k_{gj}) = E[(V|k_c - V|k_{gi})^2 (V|k_c - V|k_{gj})^2]$ in(7), we have

$$\begin{aligned}
&Cov[\Delta(k_c, k_{gi}), \Delta(k_c, k_{gj})] \\
&= \tfrac{1}{4n^2} (\tfrac{\epsilon}{\sigma})^4 \{ n\kappa^*(k_c, k_{gi}, k_{gj}) + n(n-1)\kappa(k_c, k_{gi})\kappa(k_c, k_{gj}) \\
&\qquad + n(\tfrac{2\sigma}{\epsilon})^2 \kappa(k_c, k_{gi}, k_{gj}) \} - \tfrac{\kappa(k_c, k_{gi})\kappa(k_c, k_{gj})}{4} (\tfrac{\epsilon}{\sigma})^4 \\
&= \tfrac{1}{4n} (\tfrac{\epsilon}{\sigma})^4 \kappa^*(k_c, k_{gi}, k_{gj}) - \tfrac{1}{4n} (\tfrac{\epsilon}{\sigma})^4 \kappa(k_c, k_{gi})\kappa(k_c, k_{gj}) \\
&\qquad + \tfrac{1}{n} (\tfrac{\epsilon}{\sigma})^2 \kappa(k_c, k_{gi}, k_{gj}) \\
&= \tfrac{1}{n} \{ (\tfrac{\epsilon}{\sigma})^2 \kappa(k_c, k_{gi}, k_{gj}) \\
&\qquad + \tfrac{1}{4} (\tfrac{\epsilon}{\sigma})^4 [\kappa^*(k_c, k_{gi}, k_{gj}) - \kappa(k_c, k_{gi})\kappa(k_c, k_{gj})] \}.
\end{aligned} \tag{34}$$

This verifies the second half of (26). The formula (26) is exactly the expression (33) and (34) in vector and matrix forms. Plug these expressions of $\boldsymbol{\mu}$ and $\boldsymbol{\Sigma}$ into the success rate formula (16) for ML-attack, we arrives at the explicit formula (27). This finishes the proof of Theorem 3.

Notice that for the DPA model, the V value is either 1 or 0, so that $(V|k_c - V|k_g)^2$ is always either 1 or 0. Hence, as explained after equations (5), (6) and (7), the general confusion coefficients specialize to the confusion coefficients definition for DPA. Hence all formulas (20) and (21) in Theorem 1 holds as the special cases of the corresponding formulas in Theorem 3. Therefore, the Theorem 1 follows.

C Proof of Theorem 4

(A) Here we wish to show the ML-attack with unknown (c, ϵ, σ) parameters value is equivalent to CPA. We will use direct calculation to find the ML-attack test statistic with unknown (c, ϵ, σ). That is, we maximize $T_g = \frac{1}{n} \sum_{m=1}^{n} \log f_{L|k_g}(l_m)$ over (c, ϵ, σ). Under model (17), this becomes maximizing $T_g = -\frac{1}{n} \sum_{m=1}^{n} \frac{(l_m - \epsilon v_{m,g} - c)^2}{2\sigma^2} -$

$\log(\sqrt{2\pi}\sigma)$ over (c, ϵ, σ). This is the same problem as finding maximization likelihood estimation under the linear regression model, and the solution is

$$\begin{aligned} &\hat{\sigma}_g^2 = \frac{1}{n}\sum_{m=1}^{n}(l_m - \hat{\epsilon}_g v_{m,g} - \hat{c}_g)^2, \qquad \hat{c}_g = \bar{l} - \hat{\epsilon}_g \bar{v}_g, \\ &\hat{\epsilon}_g = \frac{\sum_{m=1}^{n}(l_m - \bar{l})(v_{m,g} - \bar{v}_g)}{\sum_{m=1}^{n}(v_{m,g} - \bar{v}_g)^2}, \end{aligned} \tag{35}$$

with $\bar{l} = \frac{1}{n}\sum_{m=1}^{n} l_m$ and $\bar{v}_g = \frac{1}{n}\sum_{m=1}^{n} v_{m,g}$. Plug the solution of $\hat{\sigma}_g^2$, $\hat{c}_g$ and $\hat{\epsilon}_g$ back into the test statistics T_g, we get

$$T_g = -\log(\hat{\sigma}_g) + constant.$$

Hence the ML-attack with unknown (c, ϵ, σ) will select key k_g to minimize $\hat{\sigma}_g^2$. From (35),

$$\begin{aligned} \hat{\sigma}_g^2 &= \frac{1}{n}\Big[\sum_{m=1}^{n}(l_m - \bar{l})^2 - \frac{[\sum_{m=1}^{n}(l_m - \bar{l})(v_{m,g} - \bar{v}_g)]^2}{\sum_{m=1}^{n}(v_{m,g} - \bar{v}_g)^2}\Big] \\ &= \frac{1}{n}\sum_{m=1}^{n}(l_m - \bar{l})^2(1 - \hat{\rho}_g^2), \end{aligned}$$

where $\hat{\rho}_g$ is the Pearson Correlation

$$\hat{\rho}_g = \frac{\sum_{m=1}^{n}(l_m - \bar{l})(v_{m,g} - \bar{v}_g)}{\sqrt{\sum_{m=1}^{n}(l_m - \bar{l})^2 \sum_{m=1}^{n}(v_{m,g} - \bar{v}_g)^2}}.$$

Since $\sum_{m=1}^{n}(l_m - \bar{l})^2$ value does not change under different keys, $\hat{\sigma}_g^2$ is minimized when $\hat{\rho}_g^2$ is maximized. Hence the attack select the same key as CPA.

(B). Rivain [19] has shown that CPA also have a success rate described by the multivariate Gaussian distribution. Our task here is to express the success rate in terms of SNR and the confusion coefficients. To achieve this, we first find an asymptotically equivalent statistic, and then compute the mean and variance of it.

First, we define some notations to simplify the calculations later. Under the (Symmetric Keys) Assumption 1, $V|k_c$ has the same distribution as $V|k_g$ for all k_g. Hence the j-th moment of V is the same under all keys. That is, we can denote $c_j = E(V^j|k_c) = E(V^j|k_g)$, $j = 1, 2, ...$ W.l.o.g (without loss of generality), let $c_1 = 0$. This holds for CPA by subtracting $h/2$ from the Hamming Weight/Distance.

Also, w.l.o.g., we assume that $\epsilon > 0$ so that asymptotically the CPA succeeds when $\hat{\rho}_c > \hat{\rho}_g$ for all k_g. To calculate the probability that $\hat{\rho}_c > \hat{\rho}_g$ for all k_g, let

us denote

$$\tilde{b}_c = \frac{\sum_{m=1}^{n}(l_m-\bar{l})(v_{m,c}-\bar{v}_c)}{\sqrt{n\sum_{m=1}^{n}(v_{m,c}-\bar{v}_c)^2}}$$
$$\tilde{b}_g = \frac{\sum_{m=1}^{n}(l_m-\bar{l})(v_{m,g}-\bar{v}_g)}{\sqrt{n\sum_{m=1}^{n}(v_{m,g}-\bar{v}_g)^2}}.$$

Then $\hat{\rho}_c > \hat{\rho}_g$ is equivalent to $\tilde{b}_c > \tilde{b}_g$. Since $\sum_{m=1}^{n}(v_{m,c} - \bar{v}_c) = 0$, we have

$$\tilde{b}_c = \frac{\sum_{m=1}^{n}(r_m+\epsilon v_{m,c})(v_{m,c}-\bar{v}_c)}{\sqrt{n\sum_{m=1}^{n}(v_{m,c}-\bar{v}_c)^2}},$$
$$\tilde{b}_g = \frac{\sum_{m=1}^{n}(r_m+\epsilon v_{m,c})(v_{m,g}-\bar{v}_g)}{\sqrt{n\sum_{m=1}^{n}(v_{m,g}-\bar{v}_g)^2}}.$$

Using Central Limit Theorem, $\bar{v}_c = O_p(1/\sqrt{n})$, $\bar{v}_g = O_p(1/\sqrt{n})$, $\sum_{m=1}^{n}(v_{m,c} - \bar{v}_c)^2 = nc_2 + O_p(\sqrt{n})$ and $\sum_{m=1}^{n}(v_{m,g} - \bar{v}_g)^2 = nc_2 + O_p(\sqrt{n})$. We denote

$$b_c = \frac{1}{n\sqrt{c_2}}\sum_{m=1}^{n}(r_m + \epsilon v_{m,c})v_{m,c},$$
$$b_g = \frac{1}{n\sqrt{c_2}}\sum_{m=1}^{n}(r_m + \epsilon v_{m,c})v_{m,g},$$

so that $\tilde{b}_c = b_c + O_p(1/\sqrt{n})$ and $\tilde{b}_g = b_g + O_p(1/\sqrt{n})$. We shall calculate the asymptotic success rate of CPA by finding the asymptotic probability that $b_c > b_g$ for all k_g.

$$b_c - b_g = \frac{1}{n\sqrt{c_2}}\sum_{m=1}^{n}[r_m(v_{m,c} - v_{m,g}) + \epsilon v_{m,c}(v_{m,c} - v_{m,g})].$$

What remains is to calculate the mean and variance of the vector with elements as $b_c - b_g$ similar to the proof of Theorem 3.

The mean vector has elements

$$E(b_c - b_g) = \frac{1}{n\sqrt{c_2}} n\epsilon E[v_{1,c}(v_{1,c} - v_{1,g})] = \frac{\epsilon\kappa(k_c, k_g)}{2\sqrt{c_2}}. \tag{36}$$

Here the second equality comes from the following Lemma whose proof is provided at the end.

Lemma 2 *Under Assumption 1,* $E[(V|k_c)(V|k_c - V|k_g)] = \kappa(k_c, k_g)/2$.

Now, the elements in the variance matrix are

$$\begin{aligned}
&Cov(b_c - b_{g1}, b_c - b_{g2}) \\
&= E[(b_c - b_{g1})(b_c - b_{g2})] - E[(b_c - b_{g1})]E[(b_c - b_{g2})] \\
&= (\tfrac{1}{n\sqrt{c_2}})^2 \sum_{m_1=1}^{n} \sum_{m_2=1}^{n} E[(r_{m_1} + \epsilon v_{m_1,c})(r_{m_2} + \epsilon v_{m_2,c}) \\
&\qquad (v_{m_1,c} - v_{m_1,g1})(v_{m_2,c} - v_{m_2,g2})] \\
&\quad -(\tfrac{\epsilon}{2\sqrt{c_2}})^2 \kappa(k_c, k_{g1})\kappa(k_c, k_{g2}) \\
&= \tfrac{1}{n^2 c_2} \sum_{m=1}^{n} E[(r_m + \epsilon v_{m,c})^2 (v_{m,c} - v_{m,g1})(v_{m,c} - v_{m,g2})] \\
&\quad + \tfrac{1}{n^2 c_2} \sum_{m_1 \neq m_2} E[(r_{m_1} + \epsilon v_{m_1,c})(r_{m_2} + \epsilon v_{m_2,c}) \\
&\qquad (v_{m_1,c} - v_{m_1,g1})(v_{m_2,c} - v_{m_2,g2})] \\
&\quad - \tfrac{\epsilon^2}{4c_2} \kappa(k_c, k_{g1})\kappa(k_c, k_{g2}).
\end{aligned} \tag{37}$$

For $m_1 \neq m_2$, $E(r_{m_1}) = E(r_{m_2}) = E(r_{m_1} r_{m_2}) = 0$, using the independence of noises r_{m_1} and r_{m_2} from $v_{m_1,c}$, $v_{m_1,g1}$, $v_{m_2,c}$, and $v_{m_2,g2}$, we have

$$\begin{aligned}
&E[(r_{m_1} + \epsilon v_{m_1,c})(r_{m_2} + \epsilon v_{m_2,c})(v_{m_1,c} - v_{m_1,g1}) \\
&\qquad (v_{m_2,c} - v_{m_2,g2})] \\
&= E[\epsilon v_{m_1,c} \epsilon v_{m_2,c} (v_{m_1,c} - v_{m_1,g1})(v_{m_2,c} - v_{m_2,g2})] \\
&= \tfrac{\epsilon^2 \kappa(k_c, k_{g1})\kappa(k_c, k_{g2})}{4}.
\end{aligned} \tag{38}$$

The last step used the fact that $v_{m_1,c}$ and $v_{m_1,g1}$ are independent from $v_{m_2,c}$ and $v_{m_2,g2}$, and Lemma 2.

For $m_1 = m_2 = m$, since $E(r_m) = 0$, $E(r_m^2) = \sigma^2$, we have

$$\begin{aligned}
&E[(r_m + \epsilon v_{m,c})^2 (v_{m,c} - v_{m,g1})(v_{m,c} - v_{m,g2})] \\
&= \sigma^2 E[(v_{m,c} - v_{m,g1})(v_{m,c} - v_{m,g2})] \\
&\quad + \epsilon^2 E[(v_{m,c})^2 (v_{m,c} - v_{m,g1})(v_{m,c} - v_{m,g2})] \\
&= \sigma^2 \kappa(k_c, k_{g1}, k_{g2}) + \epsilon^2 \kappa^{**}(k_c, k_{g1}, k_{g2}).
\end{aligned} \tag{39}$$

Hence using (38) and (39), (37) becomes

$$\begin{aligned}
&Cov(b_c - b_{g1}, b_c - b_{g2}) \\
&= \tfrac{1}{n^2 c_2} n[\sigma^2 \kappa(k_c, k_{g1}, k_{g2}) + \epsilon^2 \kappa^{**}(k_c, k_{g1}, k_{g2})] \\
&\quad + \tfrac{1}{n^2 c_2} n(n-1) \tfrac{\epsilon^2 \kappa(k_c, k_{g1})\kappa(k_c, k_{g2})}{4} \\
&\quad - \tfrac{\epsilon^2}{4c_2} \kappa(k_c, k_{g1})\kappa(k_c, k_{g2}) \\
&= \tfrac{\sigma^2}{nc_2} \{\kappa(k_c, k_{g1}, k_{g2}) \\
&\qquad + (\tfrac{\epsilon}{\sigma})^2 [\kappa^{**}(k_c, k_{g1}, k_{g2}) - \tfrac{1}{4}\kappa(k_c, k_{g1})\kappa(k_c, k_{g2})]\}.
\end{aligned} \tag{40}$$

Put (36) and (40) into matrix forms, the multivariate Central Limit Theorem results in the success rate formula (28). This finishes the proof of Theorem 4.

Proof of Lemma 2.

$$\begin{aligned}
&\kappa(k_c, k_g) = E[(V|k_c - V|k_g)^2] \\
&= E[(V|k_c)^2] - 2E[(V|k_c)(V|k_g)] + E[(V|k_g)^2].
\end{aligned}$$

By the Symmetric Keys Assumption, $E[(V|k_c)^2] = E[(V|k_g)^2]$. So this becomes

$$\begin{aligned}\kappa(k_c, k_g) &= 2E[(V|k_c)^2] - 2E[(V|k_c)(V|k_g)] \\ &= 2E[(V|k_c)(V|k_c - V|k_g)].\end{aligned}$$

That is, $E[(V|k_c)(V|k_c - V|k_g)] = \kappa(k_c, k_g)/2$.