A New Second Order Side Channel Attack Based on Linear Regression

Guillaume Dabosville, Julien Doget, and Emmanuel Prouff

Abstract—Embedded implementations of cryptographic primitives need protection against Side Channel Analysis. Stochastic attacks, introduced by Schindler *et al.* at CHES 2005, are an example of such an analysis. They offer a pertinent alternative to template attacks which efficiency is optimal, and they can theoretically defeat any kind of countermeasure including masking. In both template and stochastic attacks, the adversary needs to be able to carry out a profiling stage on a perfect copy of the target device. This makes them interesting tools to study the resistance of implementations against such a powerful adversary, but it limits their pertinency in practice. It is indeed difficult to have an open access to a copy of the device under attack and, even when it is possible, it remains difficult to exploit templates acquired on one device to attack another one.
In this paper, we propose a new attack technique which shares many similarities with stochastic attacks but does not require any profiling stage. As a consequence, no copy of the device is needed anymore. We conduct an in-depth analysis of this new attack to highlight its core foundations. Then, we apply it to widely used masking schemes and we illustrate its interest by a series of experiments on simulated and real curves.

1 Introduction

Side Channel Analysis (SCA for short) exploits information that leaks from physical implementations of cryptographic algorithms. This leakage (*e.g.* the power consumption or the electro-magnetic emanations) may indeed reveal information on the secret data manipulated during the execution. Among SCA attacks, two classes may be distinguished. The set of so-called *profiling SCA* [1], [8], [26] corresponds to a powerful adversary who controls a copy of the attacked device and uses it to evaluate the distribution of the leakage according to the processed values. Once such an evaluation is obtained, a maximum likelihood approach is carried out to recover the secret data manipulated by the attacked device. The second set of attacks is the set of so-called *non-profiling SCA*. It corresponds to a weaker adversary who is only able to observe the device behaviour and has no *a priori* knowledge about the implementation details. In those attacks, the physical leakage is compared to some simulated leakage obtained from a key-dependent model. Since the seminal work of Kocher *et al.* in the late nineties [16], a large variety of statistical tests, also called *distinguishers*, have been introduced for this purpose [3], [5], [13], [18]. Their goal was to better take advantage of the available information, *e.g.*, by allowing the adversary to incorporate more precise leakage models in the statistics. This paper pays particular attention to the non-profiling SCA threats, since it relates to the classical kind of adversary encountered *e.g.*, by the smartcard industry.

A SCA targeting the manipulation of a single variable is said to be *univariate*. To avoid instantaneous information leakage and thus to thwart univariate SCA, the classical strategy is to protect the implementation of a given algorithm by using *secret sharing* (*a.k.a. masking*) techniques [4], [27]. In such schemes, the internal state of the processing is usually randomly split into two shares. When this strategy is followed, a so-called *second order* SCA can still be performed by combining the leakages resulting from the manipulation of the shares. This enables the construction of a new signal that statistically depends on the secret that was shared. Those attacks are said to be *multivariate*. In view of the analyses in [12], [21], [28] and the experiments reported in [11], [22], it seems that multivariate SCA with Pearson correlation coefficient as distinguisher and with pre-processing as proposed in [21] is always the most efficient non-profiled attack when the noise is non-negligible.

Recently, a paper [10] has argued that the linear regression attack (*a.k.a. stochastic attack*) was a sound alternative to classical univariate non-profiled SCA. Continuing the seminal analysis in [26], the authors of [10] show that, whereas attacks like CPA or MIA require a sound model for the leakage (and the attack efficiency strongly depends on this choice), a linear regression attack needs a much weaker assumption. It indeed only requires that the deterministic part of the leakage can be expressed as a linear combination of functions chosen according to the nature of the device and the algorithm under attack. Actually, [10] shows that linear regression attacks encompass the classical CPA as a particular case and, thanks to their generic nature, they can succeed in situations where CPA fails. In view of this result, it seems natural to investigate whether linear regression attacks can be extended in multivariate contexts, and if yes, whether they remain a good alternative to the classical multivariate non-

profiled SCA.

In the particular context where a leakage profiling step is allowed, stochastic methods against masked implementations have already been studied by Lemke-Rust and Paar in [17]. However, nothing is said in this paper about how to apply the techniques when profiling is impossible (which is the case in practice). A first step toward this line has been done by Schindler in [25]. However, no details about the attack itself is given and several issues regarding the way how to implement it concretely are let open. This paper starts from the same observations as [25] and aims at fully specifying a second-order attack based on linear regression techniques. It moreover studies the relationship between this new attack and classical second-order SCA (with and without profiling step). In the second part of the paper, the attack is carried out against the *Boolean masking* [7], [15] and the *arithmetic masking* [9].

Thanks to those simulations and experimentations, we show that it is a valuable alternative to the classical second-order CPA, especially when the adversary has no precise knowledge about how the targeted device leaks information on the manipulated data.

2 Preliminaries and Notations

2.1 Statistics and Probability

In the sequel random variables are denoted by large letters. A realization of a random variable X is denoted by the corresponding lowercase letter x. A *sample* of several observations of X is denoted by (x) or by (x_i) if an indexation is needed. It will sometimes be viewed as a vector defined over the definition set of X. The notation $(x) \hookleftarrow X$ denotes the instantiation of the set of observations (x) from X.

The *mean* of X is denoted $\mathbb{E}[X]$, its *variance* by $\mathsf{var}(X)$. The later equals $\mathbb{E}\left[(X - \mathbb{E}[X])^2\right]$. The *covariance* of random variables X and Y is denoted by $\mathrm{cov}(X, Y)$ and satisfies $\mathrm{cov}(X, Y) = \mathbb{E}[(X - \mathbb{E}[X])(Y - \mathbb{E}[Y])]$. The estimator of the mean of X based on a sample of observations is denoted by $\widehat{\mathbb{E}}(\cdot)$.

A continuous random variable X is associated with a *probability density function* (pdf for short). In our context, a particular pdf called *Gaussian* pdf plays an important role. The Gaussian pdf of dimension d is defined w.r.t. a *mean vector* $\vec{m} \in \mathbb{R}^d$ and a *covariance matrix* $\mathbf{\Sigma} \in \mathcal{M}_{d,d}(\mathbb{R})$ such that, for every $\vec{u} \in \mathbb{R}^d$ we have:

$$\Phi_{\vec{m},\mathbf{\Sigma}}(\vec{u}) = \frac{1}{\sqrt{(2\pi)^d |\mathbf{\Sigma}|}} e^{-\frac{1}{2}(\vec{u}-\vec{m})\mathbf{\Sigma}^{-1}(\vec{u}-\vec{m})'}. \quad (1)$$

2.2 Linear Algebra

Let $\mathcal{F}$ be a $\mathbb{R}$-vector space of functions defined over a field E (*e.g.* $E = \mathbb{F}_2^n$ for some n). For a set of d functions g_1, ..., g_d in $\mathcal{F}$, we shall denote by $< \mathrm{g}_1, ..., \mathrm{g}_d >$ the vector space spanned by all the linear combinations of the g_i with coefficients in $\mathbb{R}$. For two functions f and g in $\mathcal{F}$, we call *distance between f and g* and we denote by $\mathrm{d}(f, g)$ the real value defined by:

$$\mathrm{d}(f, g) = \sum_{x \in E} (f(x) - g(x))^2 \ . \quad (2)$$

It corresponds to the *Euclidean distance* between the vectorial representations of f and g. For a function f and a set $\mathcal{G}$, we call *distance between f and $\mathcal{G}$* the real value $\mathrm{d}(f, \mathcal{G})$ defined by:

$$\mathrm{d}(f, \mathcal{G}) = \min_{g \in \mathcal{G}} \mathrm{d}(f, g) \ . \quad (3)$$

If $\mathcal{G}$ is the space $< \mathrm{g}_1, ..., \mathrm{g}_d >$, then (3) can be rewritten:

$$\mathrm{d}(f, \mathcal{G}) = \min_{(a_1, \cdots, a_d) \in \mathbb{R}^d} \mathrm{d}(f, \sum_{i=1}^{d} a_i \mathrm{g}_i) \ . \quad (4)$$

3 New Attack Description and Analysis

3.1 Attack Context

In this paper, we consider an adversary who has access to a physical implementation of a cryptographic algorithm and observes the side-channel leakage of successive processings over known inputs. During those computations, it is assumed that an intermediate variable $Z = F_k(X)$ is manipulated. It depends on both a known variable X and a secret k, called *key* in the rest of the paper. Variables X and k are assumed to be defined over $\mathbb{F}_2^n$ for some integer value n (*e.g.* $n = 8$) and the function $F : X, k \mapsto F_k(X)$ is a known function from $\mathbb{F}_2^{2n}$ into $\mathbb{F}_2^m$ with m such that $m \leq n$ (*e.g.* F is an s-box and $F_k(X) = F(X \oplus k)$). We denote by F_k^{-1} a reciprocal function of F_k which maps each image of F_k to one of its pre-image.

It is moreover assumed that the cryptosystem is protected by first-order masking. This implies that Z is never accessed directly but is randomly split into two shares that are manipulated at different times. The manipulation of each share results in two observable physical leakages L_1 and L_2. The analyses conducted in this paper are done under the assumption that the leakages satisfy:

$$L_1 = \delta(Z \star V) + B_1 \qquad \text{and} \qquad L_2 = \delta(V) + B_2 \quad (5)$$

where $\star$ is an operation law such that $(\mathbb{F}_2^n, \star)$ is a group[1], $\delta(\cdot)$ is a deterministic *unknown* function and B_1 and B_2 are independent but identical unidimensional Gaussian variables. Random variables Z and V are also assumed to be independent from B_1 and B_2.

1. for instance $\star$ may be the bitwise addition $\oplus$ or the addition $+$ modulo 2^n where Z and V are viewed as elements of $\mathbb{Z}/n\mathbb{Z}$

3.2 Attack Description

In what follows, a new second-order attack is introduced, extending to a masked context the strategy proposed in [10], [25]. The core idea is to discriminate the key-candidates by processing a *linear regression* on a key-dependent variable, denoted Y hereafter, which combines the two leakages defined in (5). To apply such a linear regression, the adversary must have chosen a basis $(\mathrm{g}_i)_{i=1,\cdots,d}$ of functions beforehand (see Sect. 3.4 on the basis choice). With this basis on hand, he then computes for each key-candidate a discriminating value and finally outputs the key-candidate which gave rise to the smallest value. For the sake of explanations, the linear regression at Step 5 of the attack below, is expressed in terms of distance from a function to a subspace of functions as introducted in Sect. 2.2 (see Appendix A for details). More precisely, the new attack is composed of the following six steps:

[Basis choice] Choose a family of functions $(\mathrm{g}_i)_{i=1,\cdots,d}$ defined from $\mathbb{F}_2^m$ into $\mathbb{R}$. The set spanned by the functions g_i is denoted by $\mathcal{H}$.

[Measurement step] For N plaintexts, collect measurements together with the corresponding plaintexts sub-parts: $(\ell_1^i, \ell_2^i, x_i)_i \hookleftarrow (L_1, L_2, X)$.

[Partitioning step] Partition the pair of leakage measurements into sets $\mathcal{L}_x$ defined for every x such that $\mathcal{L}_x = \{(\ell_1^i, \ell_2^i); x_i = x\}$.

[Combining step] For every x, compute:

$$y_N(x) = \frac{1}{|\mathcal{L}_x|} \sum_{(\ell_1,\ell_2)\in\mathcal{L}_x} (\ell_1 - \mu_1)(\ell_2 - \mu_2) \quad , \qquad (6)$$

where y_N is a function of x parametrized by the number of collected measurements, and μ_1 and μ_2 respectively denote $\mathbb{E}[L_1]$ and $\mathbb{E}[L_2]$.

[Linear regression] For every key hypothesis $\hat{k}$, compute:

$$\Delta_{\hat{k}}(N) = \mathrm{d}\left(y_N, \mathcal{G}_{\hat{k}}\right) \quad , \qquad (7)$$

where $\mathcal{G}_{\hat{k}}$ denotes the space $< \mathrm{g}_1 \circ F_{\hat{k}}, \ldots, \mathrm{g}_d \circ F_{\hat{k}} >$.

[Key candidate decision] Select the key hypothesis for which $\Delta_{\hat{k}}(N)$ is minimal.

A detailed discussion of the new attack soundness will be conducted in the next section. We can however sum-up the main steps in the following way. First, and due to the univariate aspect of the linear regression, the leakages L_1 and L_2 are combined to form an univariate random variable Y. This is the purpose of the fourth step, which can be viewed as the computation of a noisy observation $y_N(x)$ of the covariance between L_1 and L_2 knowing $X = x$. In the rest of the paper, we shall associate the value $y_N(x)$ to the random variable $Y|X = x$, with Y being defined by:

$$Y = \mathrm{cov}\,(L_1\,,\, L_2) \quad . \qquad (8)$$

The pertinence of this definition of Y (and hence of the construction of $y_N(x)$ in the attack) is discussed in Sect. 3.5.

The computation of the minimum distance at Step 5 involves a linear regression to model the functional relationship between Y and X. The function is searched into a set which basis is constructed by composing the functions g_i with the key-hypothesis dependent function $F_{\hat{k}}$ defined in Sect. 3.1. This point is detailed in the next section while the way how to choose the family of functions $(\mathrm{g}_i)_i$ is discussed in Sect. 3.4. The linear regression technique itself together with its link with the distance $\mathrm{d}(\cdot)$ between functions can be found in Appendix A.

3.3 Attack Soundness

Since L_1 and L_2 satisfy (5), and random variables B_1, B_2 and V are independent, (8) can be rewritten

$$Y = \varphi[F_k(X)] \quad , \qquad (9)$$

where φ denotes the function

$$z \mapsto \mathrm{cov}\left(\delta(z \star V)\,,\, \delta(V)\right) \quad .$$

By construction, the function y_N defined in Step 4 tends towards Y as the number of measurements increases. Therefore from (9) and some terms rearrangement, one deduces the following limit of $\Delta_{\hat{k}}(N)$, where Y is considered as a function of X:

$$\begin{aligned}\lim_{N\to\infty} \Delta_{\hat{k}}(N) &= \lim_{N\to\infty} \mathrm{d}\left(y_N, \mathcal{G}_{\hat{k}}\right) = \mathrm{d}(Y, \mathcal{G}_{\hat{k}}) \\ &= \min_{h\in\mathcal{H}} \mathrm{d}\left(\varphi \circ F_k \circ F_{\hat{k}}^{-1} \circ F_{\hat{k}}, h \circ F_{\hat{k}}\right) \quad . \qquad (10)\end{aligned}$$

Assuming that $F_{\hat{k}}$ is balanced, (10) simplifies to

$$\lim_{N\to\infty} \Delta_{\hat{k}}(N) = 2^{n-m} \cdot \mathrm{d}\left(\varphi \circ F_k \circ F_{\hat{k}}^{-1}, \mathcal{H}\right) \quad . \qquad (11)$$

Now, depending on whether $\hat{k}$ equals k or not, we have the two following situations:

Good hypothesis ($\hat{k} = k$): Equation (11) becomes $\lim_{N\to\infty} \Delta_{\hat{k}}(N) = 2^{n-m} \cdot \mathrm{d}(\varphi, \mathcal{H})$.

Wrong hypothesis ($\hat{k} \neq k$): Equation (11) cannot be simplified.

From those two situations, we deduce that the new attack outputs the correct key if the distance between $\varphi \circ F_k \circ F_{\hat{k}}^{-1}$ and $\mathcal{H}$ is minimized when $\hat{k} = k$ (*e.g.* when $\varphi \circ F_k \circ F_{\hat{k}}^{-1}$ equals φ). It highlights the importance of the choice of the basis $(\mathrm{g}_i)_i$. This choice is discussed in the next section.

3.4 Basis Choice

As pointed out in previous section, the basis choice is essential since it directly impacts the attack efficiency. Ideally, the basis should guarantee the adversary that $\mathrm{d}\left(\varphi \circ F_k \circ F_{\hat{k}}^{-1}, \mathcal{H}\right)$ is minimal when $\hat{k} = k$. In this section, we propose a strategy for the adversary to choose it.

By definition, the function φ to be approximated belongs to the space $\mathcal{F}$ of all the functions from $\mathbb{F}_2^m$ into $\mathbb{R}$. We recall that any function in $\mathcal{F}$ can be represented by a multivariate polynomial in $\mathbb{R}[z_1,\ldots,z_m]/({z_1}^2 - z_1,\ldots,{z_m}^2 - z_m)$ (*i.e.* the degree of every z_i in every monomial is at most 1). Consequently, there exists a unique set of real coefficients $(\alpha_u)_{u\in\mathbb{F}_2^m}$ such that for every $z \in \mathbb{F}_2^m$ we have:

$$\varphi(z) = \sum_{u=(u_1,\ldots,u_m)\in\mathbb{F}_2^m} \alpha_u \cdot z^u \ , \qquad (12)$$

where each term z^u denotes the *monomial* (function) $z \longmapsto z_1^{u_1} z_2^{u_2} \cdots z_m^{u_m}$ with values in $\mathbb{F}_2$ [6]. The *degree* of such a monomial is defined as the Hamming weight of u. It can moreover be checked that the family of functions $(z^u)_{u\in\mathbb{F}_2^m}$ spans $\mathcal{F}$. In the following, we denote by $\mathcal{F}_d$ the subset of $\mathcal{F}$ that contains all the functions of degree lower than or equal to d. This set is spanned by the basis $(z^u)_{u\in\mathbb{F}_2^m,\mathrm{HW}(u)\leq d}$.

Let us now come back to the attack described in Sect. 3.2 and analysed in Sect. 3.3. If the set $\mathcal{H}$ spanned by the functions $(\mathrm{g}_i)_i$ equals $\mathcal{F}$ (*i.e.* $(\mathrm{g}_i)_i$ is also a basis of $\mathcal{F}$), then for any $F_{\hat{k}}$ and F_k it is obvious that $\varphi \circ F_k \circ F_{\hat{k}}^{-1}$ is in $\mathcal{H}$. As a consequence, the distance $\mathrm{d}\left(\varphi \circ F_k \circ F_{\hat{k}}^{-1}, \mathcal{H}\right)$ is always null, the key hypothesis $\hat{k}$ being equal to k or not. This implies that choosing the basis $(\mathrm{g}_i)_i$ as large as possible is not a sound approach in the context of our attack. Let us now denote by $\mathcal{J}$ the set of functions $\{F_k \circ F_{\hat{k}}^{-1}; k \neq \hat{k}\}$. The ideal strategy an adversary can follow is to look at a subspace $\mathcal{H}$ such that $\varphi \in \mathcal{H}$ (*i.e.* the distance between φ and $\mathcal{H}$ is null) while the distance between the two sets $\mathcal{H}$ and $\mathcal{H}\circ\mathcal{J}$ is as high as possible (Fig. 1 illustrates it). For such a purpose, we propose here to make an assumption on the degree d of φ and to set $\mathcal{H} = \mathcal{F}_d$. This amounts to choose the basis such that $(\mathrm{g}_i)_i = (z^u)_{u\in\mathbb{F}_2^m,\mathrm{HW}(u)\leq d}$. Since the composition of functions $F_k \circ F_{\hat{k}}^{-1}$ is very likely to have a high degree (close to m) due to the cryptographic properties[2] of F, if d is small enough none of the functions $\varphi \circ F_k \circ F_{\hat{k}\neq k}^{-1}$ is in $\mathcal{F}_d$ whereas $\varphi \circ F_k \circ F_{\hat{k}=k}^{-1} = \varphi$ does (by hypothesis).

To conclude this section, we give hereafter an example of our strategy in a realistic context of attack.

Example 1: Let us assume that F_k is an AES s-box. Then the set $\mathcal{J}$ contains all the functions composed of two AES s-boxes parameterized by two different keys. By property of the AES s-box, every function in $\mathcal{J}$ will be at a large distance to the set of linear functions (this relates to the high *non-linearity* of the s-box). Hence, a good strategy is to assume that φ belongs to the set of linear functions $\mathcal{F}_1$ (*i.e.* $(\mathrm{g}_i)_i = (z^u)_{u\in\mathbb{F}_2^m,\mathrm{HW}(u)\leq 1}$). Indeed, in this case the linear regression will compute a good approximation of φ in $\mathcal{F}_1$, while by definition of $\mathcal{J}$, it will not be able to compute a good approximation (in term of distance) of $\varphi \circ j$ for any $j \in \mathcal{J}$.

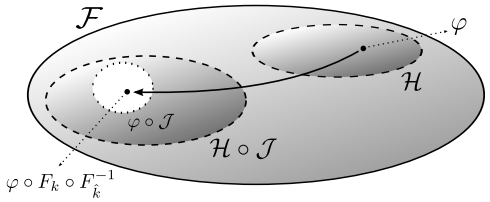


Fig. 1: Relationship between the different spaces.

Remark 1: In our strategy, we assumed that the attacker targets the result of a non-linear transformation (*e.g.* an s-box) and thus that the function F is likely to have a high degree. Nevertheless, one can choose to target the result of a linear transformation (typically the manipulation of the sensitive variable just *before* the non-linear transformation). In this case, the choice of the basis is less obvious and will be very dependent on the algebraic properties of φ. Therefore the choice of a basis must be adapted to the knowledge or assumptions on both φ and F (*i.e.* it depends on the nature of the leakage and the nature of the targeted sensitive variable).

3.5 Relationship with Other Attacks

3.5.1 Relationship with Second-Order CPA

A second-order CPA using the *centered product combining* function has been introduced in [21] and compared favorably to other attacks based on the correlation coefficient. In fact, this CPA may be viewed as a particular case of our attack where the space spanned by the basis (g_i) is reduced to a single function $\hat{\varphi}$ that is assumed to approximate the function φ defined in (9) (*e.g.* the Hamming weight function is chosen for $\hat{\varphi}$). Indeed, in such a particular case, the distance computation (7) can be rewritten:

$$\Delta_{\hat{k}}(N) = \mathrm{d}(y_N, \mathcal{H} \circ F_{\hat{k}}) = \mathrm{d}(y_N, \hat{\varphi} \circ F_{\hat{k}}) \ , \qquad (13)$$

since we have $\mathcal{H} = \{\hat{\varphi}\}$.

Now asymptotically (13) becomes:

$$\lim_{N\to\infty} \Delta_{\hat{k}}(N) = \mathrm{d}(Y, \hat{\varphi} \circ F_{\hat{k}}) = \mathrm{d}(Y, \hat{Y}) \ ,$$

where we have denoted $\hat{\varphi} \circ F_{\hat{k}}$ by $\hat{Y}$ and where we recall that Y denotes $\varphi \circ F_k$.

As a consequence, if $\rho(Y, \hat{Y})$ denotes the coefficient of correlation between Y and $\hat{Y}$, we get that (see Appendix C for the development details):

$$\lim_{N\to\infty} \Delta_{\hat{k}}(N) \quad = \quad a \cdot \rho(Y, \hat{Y}) + b \ , \qquad (14)$$

2. This property relates to the fact that, by construction, functions F_k and $F_{\hat{k}}$ must be as independent as possible when parameterized by different keys. Moreover, the family of functions F_k must have a high algebraic degree (close to m) to defeat linear and differential cryptanalyses. As a consequence, the composition of functions F_k and $F_{\hat{k}}$, with $k \neq \hat{k}$, must act as a random composition of functions with high algebraic degrees. With very high probability, such a composition results in a function with high degree. If required, this hypothesis may be tested for a target function F by computing the minimim degree of the functions in $\mathcal{J}$.

where a and b are independent of the key hypothesis provided σ_Y, $\sigma_{\hat{Y}}$, $\mathbb{E}\left[Y^2\right]$, $\mathbb{E}\left[\hat{Y}^2\right]$, $\mathbb{E}\left[Y\right]$ and $\mathbb{E}\left[\hat{Y}\right]$ are also independent of the key hypothesis. This is clearly the case with typical first-order masking schemes involving an addition, like Boolean and arithmetic masking schemes.

Equation (14) above shows that our new attack with space $\mathcal{H}$ reduced to a function $\hat{\varphi}$ is asymptotically equivalent to a second-order CPA involving the centered product as combining function and $\hat{\varphi}$ as prediction function.

3.5.2 Relationship with Maximum Likelihood Approach

In a second-order attack based on a maximum likelihood approach [8], [14], [19], [26], the adversary knows for every z a good estimation of the pdf f_z of the random variable $(L_1, L_2)|Z = z$. With such a knowledge and a sample $(\ell_1^i, \ell_2^i, x_i)_i \hookleftarrow (L_1, L_2, X)$ measured on the targeted device, the adversary then computes for each key candidate $\hat{k}$, a set of predictions $(\hat{z}_i)_i = (F_{\hat{k}}(x_i))_i$ and selects the key that maximizes the product $\prod_i f_{\hat{z}_i}(\ell_1^i, \ell_2^i)$. This class of attack, which has first been introduced in [8] under the name of *template attacks*, is very powerful. However, as previously observed in many papers, the assumption about the *a priori* knowledge of the f_z strongly limits the attack practicability and raises the need for alternative approaches. To some extent, the attack presented in Sect. 3.2 can be viewed as such an alternative. More precisely, it may be viewed as an application of the template attacks principle in a context where the adversary has no *a priori* knowledge of the f_z but tries to reconstruct them on-the-fly. To further discuss on this statement, the pdfs f_z must be developed.

When the leakage is defined as in (5), the f_z are *mixture* of *elliptic normal distributions* [20], namely they are defined such that:

$$f_z = \frac{1}{2^n} \sum_{v \in \mathbb{F}_2^n} \Phi_{\vec{m}_{z,v}, \mathbf{\Sigma}} \ , \tag{15}$$

where $\vec{m}_{z,v}$ and $\mathbf{\Sigma}$ satisfy:

$$\vec{m}_{z,v} = (\delta(z \star v), \delta(v)) \text{ and } \mathbf{\Sigma} = \begin{pmatrix} \sigma^2 & 0 \\ 0 & \sigma^2 \end{pmatrix} \ .$$

Our attack implicitly tries to approximate the distribution f_z by a bivariate Gaussian pdf and this is actually the main difference between it and template attacks. The use of such an approximation is known in the literature as the technique of *merging* the mixture components [24] with a limited and fixed number of components (here 2). It leads us to make the following approximation:

$$f_z \sim \Phi_{\vec{m}, \mathbf{\Sigma}_z} \ , \tag{16}$$

where $\vec{m} = (\mathbb{E}\left[\delta(z \star V)\right], \mathbb{E}\left[\delta(V)\right])$ and[3]

$$\mathbf{\Sigma}_z = \begin{pmatrix} \sigma^2 & Y \mid Z = z \\ Y \mid Z = z & \sigma^2 \end{pmatrix} \ .$$

where we recall that Y equals $\mathrm{cov}\,(L_1\,,\,L_2)$.

In view of the definitions of $\vec{m}$ and $\mathbf{\Sigma}_z$ it is clear that the only key-dependent parameter of the pdf approximation (16) is $Y|Z = z$. Thus, testing whether an observation (ℓ_1, ℓ_2) comes from a distribution $\Phi_{\vec{m}, \mathbf{\Sigma}_z}$ reduces to test whether (ℓ_1, ℓ_2) comes from a bivariate distribution with covariance $Y|Z = z$. As explained in Sect. 3.3, our new attack computes an estimation of this variable, the estimation being parametrized by a key hypothesis. Then, to validate the hypothesis (or equivalently the quality of the approximation of $Y|Z = z$ for every z), a mean-of-square test is computed. It is well known that this test is equivalent to a maximum likelihood computation under the Gaussian Assumption. Some simulations can be found in Appendix D.

Remark 2: Another more precise way of approximating the distributions may be to look for approximations by mixtures of Gaussian distributions. This approach has already been suggested in [17] but its soundness is still under discussion since it involves a class of algorithms, called *expectation-maximization* (EM) algorithms, which are difficult to deal with.

4 Application on Masking schemes

In previous sections, we exhibited a way to attack a masked implementation by using linear regression techniques. In the following we aim at confronting our analyses with simulations in realistic scenarios (Sect. 4.1 and 4.2) and experiments (Sect. 4.3). To ease the comparison, several attack parameters are considered: the underlying masking scheme that is used to protect the sensitive variable, the distinguisher on which the key discrimination is based, some distinguisher-related parameters to customize the attack, the origin of the leakage (simulation or real curves) and the efficiency of the attack (number of messages, *etc.*).

Remark 3: Our main purpose is to compare the new attack with the CPA techniques which are the most widely used in practice. However, in order to have an analysis as exhaustive as possible, we also implemented second-order MIA attacks. Among the different techniques which have been proposed to process the MIA, we chose to implement the one which is based on histogram approximation methods since it seems to be the most efficient in practice [2]. Further works may consist in deeper comparing the new attack with all the various MIA techniques [30] and also with the recently introduced attacks based on

3. Note that $\vec{m}$ exactly corresponds to the development of the mean vector $(\mathbb{E}\left[L_1|Z = z\right], \mathbb{E}\left[L_2\right])$ when using the linearity of the expectation and the fact that the noise is assumed to have zero mean

Kolmogorov-Smirnov distance estimator [31]. Those attacks indeed also aim to target masked implementations when the leakage has unpredictible behavior.

Attack Target. The attacks exploit the leakage related to the manipulation of two shares that jointly depend on a sensitive variable Z satisfying

$$Z = F_k(X) = F(X \oplus k) \ , \tag{17}$$

where X corresponds to an 8-bit uniformly distributed random value known by the adversary and F denotes the AES s-box. Depending on the underlying masking scheme, the definition of the two shares differ. The following masking schemes are considered in our attacks:

1) 1st-order Boolean masking: the operation $\star$ in (5) is the bitwise addition over $\mathbb{F}_2^8$. The two shares are $Z \oplus V$ and V, with V a uniformly distributed random variable independent of Z.
2) 1st-order arithmetic masking: the operation $\star$ in (5) is the modular addition over $\mathbb{Z}/256\mathbb{Z}$. The two shares are $Z + V \mod 256$ and V, with V a uniformly distributed random variable independent of Z.

Leakage Simulations. Leakages have been simulated in accordance with (5) for different definitions of $\delta(\cdot)$, leading to the three following scenarios:

Scenario 1 *(Hamming Weight Leakage):* Equation (5) becomes:

$$L_1 = \underbrace{\mathrm{HW}\left(Z \star V\right)}_{\delta(Z \star V)} + B_1 \quad \text{and} \quad L_2 = \underbrace{\mathrm{HW}\left(V\right)}_{\delta(V)} + B_2 \ . \tag{18}$$

In our attack settings, this first scenario is ideally suited for CPA since the model used by the adversary exactly corresponds to the deterministic function $\delta(\cdot)$.

Scenario 2 *(Linear Leakage):* Equation (5) becomes:

$$\begin{aligned} L_1 &= \underbrace{\alpha_0 + \sum_{i=1}^{8} \alpha_i \cdot \left(Z \star V\right)[i]}_{\delta(Z \star V)} + B_1 \quad \text{and} \\ L_2 &= \underbrace{\alpha_0 + \sum_{i=1}^{8} \alpha_i \cdot V[i]}_{\delta(V)} + B_2 \ , \end{aligned} \tag{19}$$

with coefficients $(\alpha_i)_{0 \leqslant i \leqslant 8}$ uniformly picked from $[-1, 1]$. This scenario is used to observe the distinguishers behaviour when the deterministic part of the leakage differs from the model used by the adversary. We restricted ourselves to functions $\delta(\cdot)$ that are linear combinations in $\mathbb{R}$ of the bit-coordinates of the shared values.

Scenario 3 *(Quadratic Leakage):* Equation (5) becomes:

$$\begin{aligned} L_1 &= \delta(Z \star V) + B_1 \\ &= \alpha_0 + \sum_{i=1}^{8} \alpha_i \cdot \left(Z \star V\right)[i] \\ &\quad + \sum_{\substack{i_1, i_2 = 1 \\ i_1 < i_2}}^{8} \alpha_{i_1, i_2} \cdot \left(Z \star V\right)[i_1] \cdot \left(Z \star V\right)[i_2] + B_1 \\ L_2 &= \delta(V) + B_2 \\ &= \alpha_0 + \sum_{i=1}^{8} \alpha_i \cdot V[i] \\ &\quad + \sum_{\substack{i_1, i_2 = 1 \\ i_1 < i_2}}^{8} \alpha_{i_1, i_2} \cdot V[i_1] \cdot V[i_2] + B_2 \ , \end{aligned} \tag{20}$$

with coefficients $(\alpha_i)_{0 \leqslant i \leqslant 36}$ uniformly picked from $[-1, 1]$. This scenario is used to observe the distinguishers behaviour when the deterministic part of the leakage differs in degree from the model used by the adversary. We restricted ourselves to functions $\delta(\cdot)$ that are quadratic combinations in $\mathbb{R}$ of the bit-coordinates of the shared values.

Leakage Measurements. The details about the leakage used in experiments have been confined to a dedicated section (see Sect.4.3).

Attack Distinguisher.

1) Correlation Power Analysis (CPA). Those attacks approximate $\rho(\mathcal{C}(L_1, L_2), \tau(F_{\hat{k}}(X)))$ to discriminate the key candidates, where $\mathcal{C}(\cdot)$ is a combining function from $\mathbb{R}^2$ to $\mathbb{R}$ and τ is a model function deduced from $\mathcal{C}(\cdot)$ and an hypothesis on $\delta(\cdot)$. A second-order CPA with model τ is denoted by CPA_τ
2) Linear Regression (LR) is used as described in this paper (see Sect. 3.2).
3) Mutual Information Analysis (MIA) with histogram estimation (the choice of the bin-width is done using the rule proposed in [13]) and Hamming weight model.

Model and Basis Choice. Albeit $Z \star V$ and V jointly depend on Z, each masking scheme induces a different dependency relationship which implies to adapt the attack strategy accordingly. Namely, for each of the attacks above, the choice of the consumption model (in CPA) or the choice of the basis (in LR attacks) requires a careful attention.

To perform the second-order CPA, we chose the centered product combining of the leakages and defined the optimal model function[4] τ as described in [23] under the assumption $\delta(\cdot) = \mathrm{HW}\left(\cdot\right)$. This kind of CPA is denoted $\mathrm{CPA}_{\mathrm{Opt}}$ in the sequel.

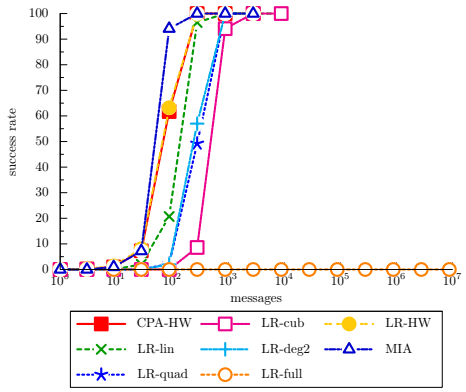
4. Notice that the optimal model function τ differs from one masking scheme to another and must therefore be computed for each different masking scheme.

As argued in Sect. 3.4, linear regression requires similarly a set of well-chosen basis functions to perform efficiently. To approximate the function $\varphi : z \mapsto \mathrm{cov}\left(\delta(z \star V), \delta(V)\right)$, we have analysed different choices of basis[5]:

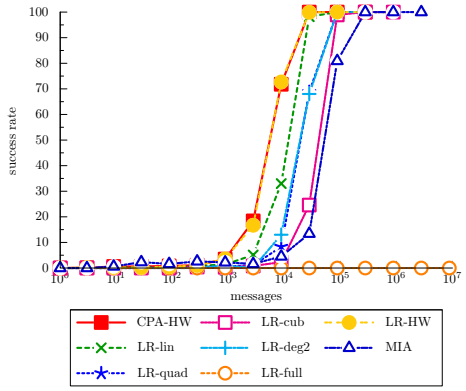
- *lin* where the g_i are the degree-1 monomials $z \mapsto z^u$ with $\mathrm{HW}(u) \leqslant 1$.
- *quad* where the g_i are the monomials $z \mapsto z^u$ with $\mathrm{HW}(u) \leqslant 2$.
- *cub* where the g_i are the monomials $z \mapsto z^u$ with $\mathrm{HW}(u) \leqslant 3$.
- *full* where the g_i are the monomials $z \mapsto z^u$ with $\mathrm{HW}(u) \leqslant 8$.
- *deg*2 where the g_i are the degree-2 monomials $z \mapsto z^u$ with $\mathrm{HW}(u) = 2$.
- *Opt* where the basis is reduced to the constant function $z \mapsto 1$ and a function g defined as the optimal (prediction) function as defined in [23]. In Sect. 4.1 (*i.e.* Boolean case), the basis *Opt* is denoted by HW to emphasis the affine equivalence between the optimal function and the Hamming weight when the optimal function is designed under the assumption $\delta(\cdot) = \mathrm{HW}$ and $\star = \oplus$.

In the sequel, an attack using the linear regression with basis **basis** will be denoted by LR-**basis**, where **basis** is chosen among *lin*, *quad*, *cub*, *deg*2, *full* and *Opt*.

Remark 4: It has been shown in Sect. 3.5 that $\mathrm{CPA}_{\mathrm{Opt}}$ is equivalent to LR-Opt, nevertheless we have conducted both attacks to confront this result to experimentations.



(a) No noise



(b) $\sigma = 4$

Fig. 2: Attacks against Boolean masking in Scenario 1

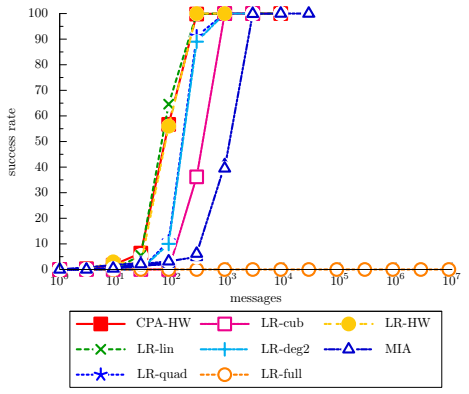
Attack Efficiency. In the following, an attack is said to be *successful* if the good key is output by the attack. An attack is said to be *more efficient than* another if it needs less messages to achieve the same success rate. Success rate is measured over $1,000$ tries.

We report and analyse in next sections our attack simulations results for Scenarios 1, 2 and 3 in case of Boolean (Sect. 4.1) and arithmetic masking schemes (Sect. 4.2). We inform the reader that we have plotted only attacks which are relevant. In other terms, some attacks never succeed and thus have not been plotted to ensure readability of figures.

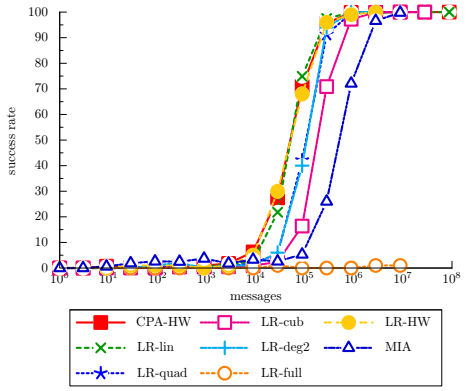
4.1 Simulation with Boolean Masking Scheme

In this section we assume that L_1 and L_2 satisfy (18) (Scenario 1), or (19) (Scenario 2), or (20) (Scenario 3). For each attack listed in the previous section, we have plotted in Fig(s). 2–4 the success rate as a function of the number of messages. We did this in two different contexts: a non-noisy one (B_1 and B_2 are null) and a noisy one (B_1 and B_2 have mean 0 and standard deviation 4). In Scenario 1 the most efficient attack is LR-HW except without noise where MIA is more efficient. As expected $\mathrm{CPA}_{\mathrm{HW}}$ is as efficient as LR-HW while the LR-lin attack is ranked second. This is due to the fact that the hypothesis made over $\delta(\cdot)$ induces a model that exactly corresponds to the leakage function. Nevertheless, LR-HW and $\mathrm{CPA}_{\mathrm{HW}}$ stop to be the most efficient attacks in Scenarios 2 and 3. This must be a consequence of the fact that, in those cases the model τ is built under the incorrect hypothesis $\delta(\cdot) = \mathrm{HW}(\cdot)$. In Scenario 2, LR-lin is the most efficient attack. The efficiency of the linear regression with basis *lin* is explained by the fact that $y_N(\cdot)$ in (6) is linear when $\delta(\cdot)$ does (this is a straightforward extension from the Hamming weight case shown in [23] to any linear function of the bit-coordinates) and it is thus well approximated in the linear basis. In Scenario 3, the results are rather the same than in Scenario 2 since LR-lin is still the most efficient attack. At first glance, this may appear as a surprising result since we could expect the LR-quad attack to be more efficient. Indeed, in this scenario y_N can be exactly approximated given the basis *quad* but cannot with basis *lin*. So the estimation of y_N returned by the linear regression is better in the quadratic case than in the linear one. Despite this difference, the attack with linear basis discriminates faster. This

5. Every basis contains the constant function, $\mathsf{g}_1 : z \mapsto 1$

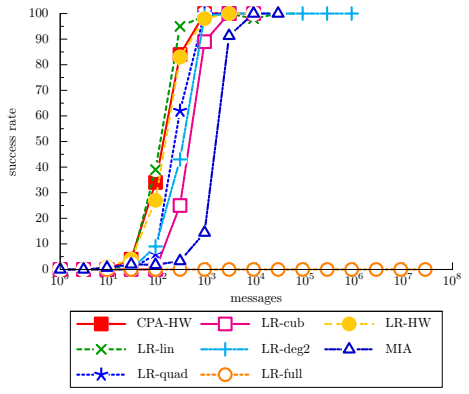


(a) No noise

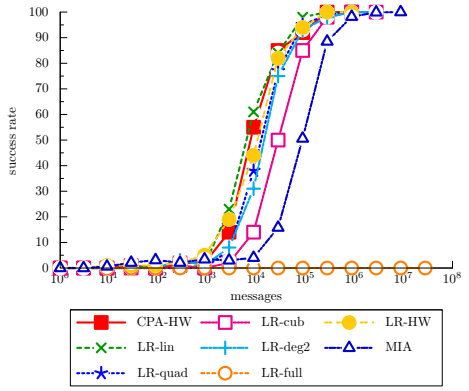


(b) $\sigma = 4$

Fig. 3: Attacks against Boolean masking in Scenario 2



(a) No noise



(b) $\sigma = 4$

Fig. 4: Attacks against Boolean masking in Scenario 3

shows that in some circumstances, it may be sufficient to approximate only the linear part of the leakage and that the computation overhead that a quadratic (or higher) basis brings on, does not worth.

Eventually, it seems that for each attack in each scenario, the presence of noise makes the curves to be closer from each other. Namely, attacks which reach a 100% success rate seem to become asymptotically equivalent when noise increases. It is explained by the fact that the number of messages needed to annihilate the noise is largely sufficient to have a great approximation with linear regression whatever the size of the basis.

Remark 5: As expected, MIA is always the less efficient attack except in a perfect condition (*i.e.* without noise and with the leakage deterministic part equal to the attack model – here Hamming weight –).

4.2 Simulation with Arithmetic Masking Scheme

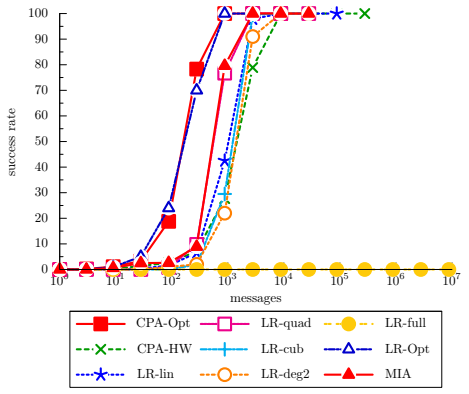
In this section, L_1 and L_2 satisfy either (18) (Scenario 1), or (19) (Scenario 2), or (20) (Scenario 3). For each attack listed before, we have performed the same attack simulations as in Sect. 4.1. The results are plotted in Fig(s). 5–7.

In the arithmetic case, all attacks based on the optimal model are the most efficient ones, even in Scenarios 2 and 3. The LR-quad attack is ranked second for each scenario and its efficiency is close to that of LR-Opt and $\mathrm{CPA}_{\mathrm{Opt}}$. In particular, it is always better than $\mathrm{CPA}_{\mathrm{HW}}$ and LR-lin which actually do not achieve a success rate greater than 85%. This situation can be explained by the fact that the quadratic terms of the function y_N defined in (6) have an important influence on the leakage when the masking is arithmetic and not Boolean. To illustrate this, focusing on LR-deg2 attack, it can be checked that its efficiency is close to that of LR-quad (namely the attack performs almost equivalently with and without the linear terms in y_N). The LR-cub attack is ranked third, behind the LR-quad. Therefore, due to the computation overhead induced by the use of a basis with cubic terms, the adversary will benefit from applying the LR-quad attack instead of the LR-cub one.

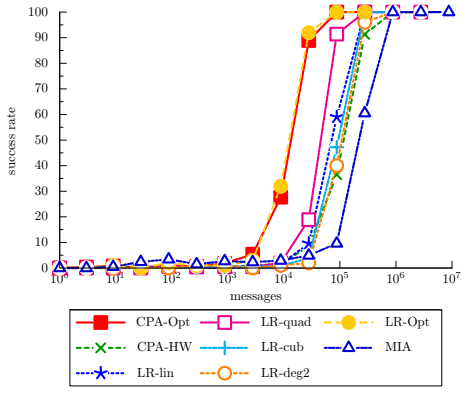
Remark 6: In the arithmetic case, MIA is often the less efficient attack. Only in the first scenario (leakage modeled as Hamming weight), MIA overpasses LR-deg2, LR-lin and CPA-HW.

4.3 Attacks Experiments in Real Life

In previous sections, we have confronted our theoretical analyses with simulations in realistic scenarios. In the following, we aim at confronting our results

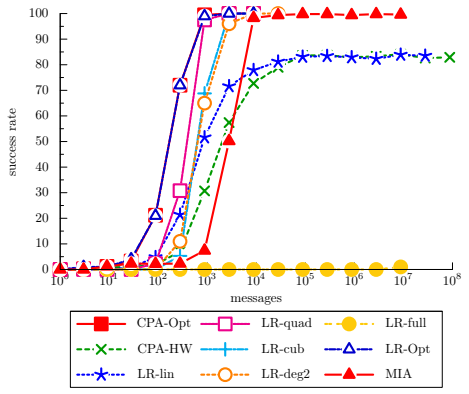


(a) No noise

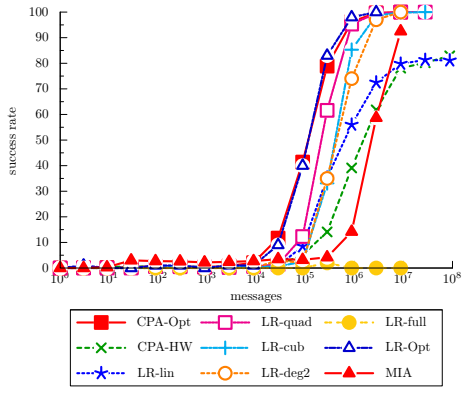


(b) $\sigma = 4$

Fig. 5: Attacks against arithmetic masking in Scenario 1



(a) No noise



(b) $\sigma = 4$

Fig. 6: Attacks against arithmetic masking in Scenario 2

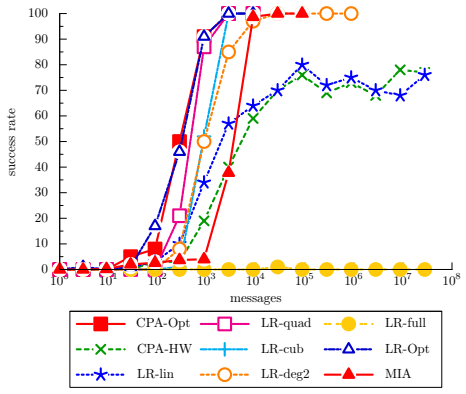
against real measurements. Attack parameters like the attack target, the masking scheme and the attack distinguisher remain the same as previously defined while the leakage now comes from real power consumption curves.

4.3.1 Leakage Measurements

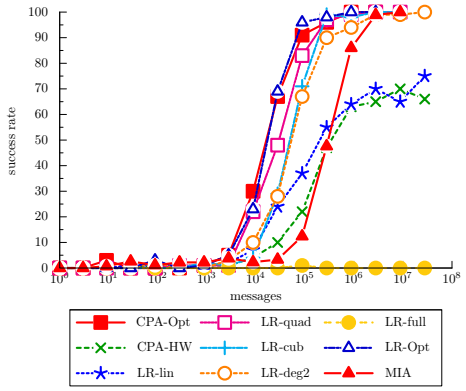
Power consumption leakages have been measured on a 8051 8-bit micro-controller. In each measurement curve the parts related to the manipulation of $Z \star V$ and V are composed each of $100,000$ points. We assume the curves to be synchronized (a glitch is used to synchronize at the beginning of the manipulation processing). Before mounting each attack, a pre-processing step has been performed on the curves to determine the two most pertinent *points of interest* (the first point corresponding to $Z \star V$ and the second one corresponding to V). By definition, this pair of points is the one that optimizes the attack efficiency among the $100,000^2$ possible pair of points. This more or less corresponds to the definition given in [29]. For the CPA, the pair corresponds to the pair of points for which the error resulting from the approximation of the leakage by the attack model is minimal. For the regression-based attacks, the points of interest are those for which the error resulting from the approximation of the leakage in the basis is minimal. During the pre-processing, we have used the fact that we knew the values $Z \star V$ and V manipulated by the device. Even if this does not correspond to a real life adversary, this pre-processing allows us to perform each attack with the optimal choice of points of interest, which is a fair context to compare them together.

For the attack comparisons, only the pair of points of interest resulting in the maximal distinguishing value has been considered for each attack.

Remark 7: This preprocessing step is not a prerequisite to the attacks thus it must not be assimilate to a profiling step. In fact, in this section we adopt the point of view of a defender which want to resist to the most powerful attacker. In this case, it is sound to assume that the defender (*e.g.* a chip designer) knows exactly the points of interest (*i.e.* the most favorable case for an attacker). In other words, from an attacker point of view, even if the points of interest are given, the attack must be inefficient. We notice that usually an attacker does not have access to such an information.



(a) No noise



(b) $\sigma = 4$

Fig. 7: Attacks against arithmetic masking in Scenario 3

4.3.2 *Experiments Results*

For each attack, the distinguishing coefficient has been computed for each key candidate and for a given (increasing) number of power traces up to $460,000$. We recorded the minimal number of messages needed to have the real key ranked first (*i.e.* emerging from others). Results are recorded in Tab. 1.

Attack \ Masking	Boolean	Arithmetic
CPA_{HW}	933	42,330
CPA_{Opt}		2,039
LR-HW	832	42,320
LR-Opt		2,043
LR-lin	976	6,384
LR-quad	3,907	5,907
LR-cub	15,737	6,620
LR-deg2	4,884	14,705

TABLE 1: Experimental results

Globally, the experiments confirm our simulations results. That is the attacks are ranked in the same order with the same difference magnitude between them. It confirms the soundness of our attack.

4.4 Conclusion on the Attack Simulations and Experiments

The theoretical analysis led in Sect. 3.5 is confirmed by the experimental results. At first, it corroborates the analysis which explains why the linear regression is effective. More than validating the effectiveness of LR attacks, experiments show that they are at least as efficient as the CPA and therefore appears as a real alternative to it. Secondly, it validates the great importance of the choice of the basis. Although attacks based on the optimal model in Scenario 1 (for both masking schemes) are always at the first place, this is no longer the case when the optimal model is built from a wrong hypothesis on $\delta(\cdot)$. For instance with Boolean masking, choosing a linear basis is sufficient to make LR more efficient than LR-Opt whereas with arithmetic masking a quadratic basis is needed. Finally, as predicted in Sect. 3.4, the LR-full (*i.e.* $\mathcal{H} = \mathcal{F}$) attacks always fail.

Remark 8: The presence of noise makes the curves to be closer each one to another. Moreover, whereas the maximal success rate of each attack is unchanged, the higher the noise, the higher the measurements number to achieve the same success rate. In fact the number of messages needed to annihilate the noise is largely sufficient to have a great approximation with linear regression even with a large basis.

5 Conclusion

In this paper we have introduced a second-order stochastic attack which does not assume any profiling capability on the adversary side. The attack was successfully applied to the two major first-order masking schemes, namely the Boolean and arithmetic ones. A theoretical analysis of the approach explains the core foundations of the attack, giving the reasons of its effectiveness together with its intrinsic limits. The effectiveness of the attack is confirmed by the experiments which show that it is a good alternative to existing solutions like the second-order CPA with a combining function. Both theoretical analysis and experiments highlight the importance of the choice of the basis involved in the attack. This point should be investigated in future works to take into account other masking schemes like the multiplicative or the affine ones.

REFERENCES

[1] Cédric Archambeau, Eric Peeters, François-Xavier Standaert, and Jean-Jacques Quisquater. Template Attacks in Principal Subspaces. In L. Goubin and M. Matsui, editors, *Cryptographic Hardware and Embedded Systems – CHES 2006*, volume 4249 of *Lecture Notes in Computer Science*, pages 1–14. Springer, 2006.

[2] L. Batina, B. Gierlichs, E. Prouff, M. Rivain, F.-X. Standaert, and N. Veyrat-Charvillon. Mutual information analysis: a comprehensive study. *to appear in the Journal of Cryptology*, 24(2):269–291, April 2011.

[3] R. Bévan and E. Knudsen. Ways to Enhance Power Analysis. In P.J. Lee and C.H. Lim, editors, *Information Security and Cryptology – ICISC 2002*, volume 2587 of *Lecture Notes in Computer Science*, pages 327–342. Springer, 2002.

[4] G.R. Blakely. Safeguarding cryptographic keys. In *National Comp. Conf.*, volume 48, pages 313–317, New York, June 1979. AFIPS Press.

[5] É. Brier, C. Clavier, and F. Olivier. Correlation Power Analysis with a Leakage Model. In M. Joye and J.-J. Quisquater, editors, *Cryptographic Hardware and Embedded Systems – CHES 2004*, volume 3156 of *Lecture Notes in Computer Science*, pages 16–29. Springer, 2004.

[6] Claude Carlet. Boolean functions for cryptography and error correcting codes. *Boolean Methods and Models*, page 257, 2010.

[7] S. Chari, C.S. Jutla, J.R. Rao, and P. Rohatgi. Towards Sound Approaches to Counteract Power-Analysis Attacks. In M.J. Wiener, editor, *Advances in Cryptology – CRYPTO '99*, volume 1666 of *Lecture Notes in Computer Science*, pages 398–412. Springer, 1999.

[8] S. Chari, J.R. Rao, and P. Rohatgi. Template Attacks. In B.S. Kaliski Jr., Ç.K. Koç, and C. Paar, editors, *Cryptographic Hardware and Embedded Systems – CHES 2002*, volume 2523 of *Lecture Notes in Computer Science*, pages 13–29. Springer, 2002.

[9] J.-S. Coron and L. Goubin. On Boolean and Arithmetic Masking against Differential Power Analysis. In Ç.K. Koç and C. Paar, editors, *Cryptographic Hardware and Embedded Systems – CHES 2000*, volume 1965 of *Lecture Notes in Computer Science*, pages 231–237. Springer, 2000.

[10] Julien Doget, Emmanuel Prouff, Matthieu Rivain, and François-Xavier Standaert. Univariate Side Channel Attacks and Leakage Modeling. *Journal of Cryptographic Engineering*, 1(2):123–144, 2011.

[11] Guillaume Fumaroli, Ange Martinelli, Emmanuel Prouff, and Matthieu Rivain. Affine Masking against Higher-Order Side Channel Analysis. In Alex Biryukov, Guang Gong, and Douglas R. Stinson, editors, *Selected Areas in Cryptography*, volume 6544 of *Lecture Notes in Computer Science*, pages 262–280. Springer, 2010.

[12] Benedikt Gierlichs, Lejla Batina, Bart Preneel, and Ingrid Verbauwhede. Revisiting Higher-Order DPA Attacks: Multivariate Mutual Information Analysis. Cryptology ePrint Archive, Report 2009/228, 2009. http://eprint.iacr.org/.

[13] Benedikt Gierlichs, Lejla Batina, Pim Tuyls, and Bart Preneel. Mutual information analysis. In Elisabeth Oswald and Pankaj Rohatgi, editors, *CHES*, volume 5154 of *Lecture Notes in Computer Science*, pages 426–442. Springer, 2008.

[14] Benedikt Gierlichs, Kerstin Lemke-Rust, and Christof Paar. Templates vs. Stochastic Methods. In L. Goubin and M. Matsui, editors, *Cryptographic Hardware and Embedded Systems – CHES 2006*, volume 4249 of *Lecture Notes in Computer Science*, pages 15–29. Springer, 2006.

[15] L. Goubin and J. Patarin. DES and Differential Power Analysis – The Duplication Method. In Ç.K. Koç and C. Paar, editors, *Cryptographic Hardware and Embedded Systems – CHES '99*, volume 1717 of *Lecture Notes in Computer Science*, pages 158–172. Springer, 1999.

[16] P. Kocher, J. Jaffe, and B. Jun. Differential Power Analysis. In M.J. Wiener, editor, *Advances in Cryptology – CRYPTO '99*, volume 1666 of *Lecture Notes in Computer Science*, pages 388–397. Springer, 1999.

[17] Kerstin Lemke-Rust and Christof Paar. Gaussian mixture models for higher-order side channel analysis. In Pascal Paillier and Ingrid Verbauwhede, editors, *Cryptographic Hardware and Embedded Systems – CHES 2007*, volume 4727 of *Lecture Notes in Computer Science*, pages 14–27. Springer, 2007.

[18] T.S. Messerges. Using Second-order Power Analysis to Attack DPA Resistant software. In Ç.K. Koç and C. Paar, editors, *Cryptographic Hardware and Embedded Systems – CHES 2000*, volume 1965 of *Lecture Notes in Computer Science*, pages 238–251. Springer, 2000.

[19] Elisabeth Oswald and Stefan Mangard. Template Attacks on Masking—Resistance is Futile. In Masayuki Abe, editor, *Topics in Cryptology – CT-RSA 2007*, volume 4377 of *Lecture Notes in Computer Science*, pages 243–256. Springer, 2007.

[20] J.K. Patel and C.B. Read. *Handbook of the Normal Distribution*. Statistics, textbooks and monographs. Marcel Dekker, 1996.

[21] Emmanuel Prouff and Matthieu Rivain. Theoretical and Practical Aspects of Mutual Information Based Side Channel Analysis. In Michel Abdalla, David Pointcheval, Pierre-Alain Fouque, and Damien Vergnaud, editors, *Applied Cryptography and Network Security – ANCS 2009*, volume 5536 of *Lecture Notes in Computer Science*, pages 499–518. Springer, 2009.

[22] Emmanuel Prouff and Matthieu Rivain. Theoretical and Practical Aspects of Mutual Information-Based Side Channel Analysis. *IJACT*, 2(2):121–138, 2010.

[23] Emmanuel Prouff, Matthieu Rivain, and Régis Bévan. Statistical Analysis of Second Order Differential Power Analysis. *IEEE Transactions on Computers*, 58(6):799–811, 2009.

[24] Andrew R. Runnalls. Kullback-Leibler Approach to Gaussian Mixture Reduction. *IEEE Transactions of Aerospace and Electronic Systems*, 43(3):989–999, July 2007.

[25] Werner Schindler. Advanced Stochastic Methods in Side Channel Analysis on Block Ciphers in the Presence of Masking. *Journal of Mathematical Cryptology*, 2:291–310, 2008.

[26] Werner Schindler, Kerstin Lemke, and Christof Paar. A Stochastic Model for Differential Side Channel Cryptanalysis. In J.R. Rao and B. Sunar, editors, *Cryptographic Hardware and Embedded Systems – CHES 2005*, volume 3659 of *Lecture Notes in Computer Science*. Springer, 2005.

[27] Adi Shamir. How to Share a Secret. *Communications of the ACM*, 22(11):612–613, November 1979.

[28] François-Xavier Standaert, Nicolas Veyrat-Charvillon, Elisabeth Oswald, Benedikt Gierlichs, Marcel Medwed, Markus Kasper, and Stefan Mangard. The World is not Enough: Another Look on Second-Order DPA. In Masayuki Abe, editor, *ASIACRYPT*, volume 6477 of *Lecture Notes in Computer Science*, pages 112–129. Springer, 2010.

[29] J. Waddle and D. Wagner. Toward Efficient Second-order Power Analysis. In M. Joye and J.-J. Quisquater, editors, *Cryptographic Hardware and Embedded Systems – CHES 2004*, volume 3156 of *Lecture Notes in Computer Science*, pages 1–15. Springer, 2004.

[30] Carolyn Whitnall and Elisabeth Oswald. A Comprehensive Evaluation of Mutual Information Analysis Using a Fair Evaluation Framework. In Phillip Rogaway, editor, *CRYPTO*, volume 6841 of *Lecture Notes in Computer Science*, pages 316–334. Springer, 2011.

[31] Carolyn Whitnall, Elisabeth Oswald, and Luke Mather. An Exploration of the Kolmogorov-Smirnov Test as Competitor to Mutual Information Analysis. In Vincent Rijmen and Emmanuel Prouff, editors, *CARDIS*, Lecture Notes in Computer Science, pages 316–334. Springer, 2011.

APPENDIX A
LINEAR REGRESSION

In this section we describe the linear regression technique when applied to our context. For a basis of functions $(\mathrm{g}_i)_{1\leq i\leq d}$, a set of noisy observations $(y_N(x))_{x\in\mathbb{F}_n}$ as defined in (6) and a key candidate $\hat{k}$, the goal is to estimate:

$$\begin{aligned}\Delta_{\hat{k}} &= \min_{(a_1,\cdots,a_d)\in\mathbb{R}^d} \mathrm{d}\left(y_N, \left(\sum_i a_i \mathrm{g}_i\right)\circ F_{\hat{k}}\right) \qquad (21)\\ &= \min \sum_{x\in\mathbb{F}_2^n}\left(y_N(x) - \left[\left(\sum_i a_i \mathrm{g}_i\right)\circ F_{\hat{k}}\right](x)\right)^2 .\end{aligned}$$

The linear regression technique involved in this paper starts by building the following *regression matrix*:

$$\mathbf{M} = \begin{pmatrix} \mathrm{g}_1(F_{\hat{k}}(0)) & \cdots & \mathrm{g}_d(F_{\hat{k}}(0)) \\ \mathrm{g}_1(F_{\hat{k}}(1)) & \cdots & \mathrm{g}_d(F_{\hat{k}}(1)) \\ \vdots & \ddots & \vdots \\ \mathrm{g}_1(F_{\hat{k}}(x)) & \cdots & \mathrm{g}_d(F_{\hat{k}}(x)) \\ \vdots & \ddots & \vdots \\ \mathrm{g}_1(F_{\hat{k}}(2^n-1)) & \cdots & \mathrm{g}_d(F_{\hat{k}}(2^n-1)) \end{pmatrix} ,$$

where the value x in $F_{\hat{k}}(x)$ is represented as an integer corresponding to the binary representation of $x \in \mathbb{F}_2^n$.

From the vector $\vec{y_N} = (y_N(0), \ldots, y_N(2^n-1))$ and $\mathbf{M}$, the following column vector $\vec{\alpha}$ is computed:

$$\vec{\alpha} = {}^t(\alpha_1, \ldots, \alpha_d) = \left({}^t\mathbf{M} \cdot \mathbf{M}\right)^{-1} \cdot {}^t\mathbf{M} \cdot {}^t\vec{y} \ .$$

Under the Gaussian assumption, the function $(\mathrm{g}_1, \ldots, \mathrm{g}_d) \cdot \vec{\alpha}$ is the function in $< \mathrm{g}_i >_{1 \le i \le d}$ that is the closest one to $x \mapsto y_N(x)$ for the Euclidean distance. In other terms, we have:

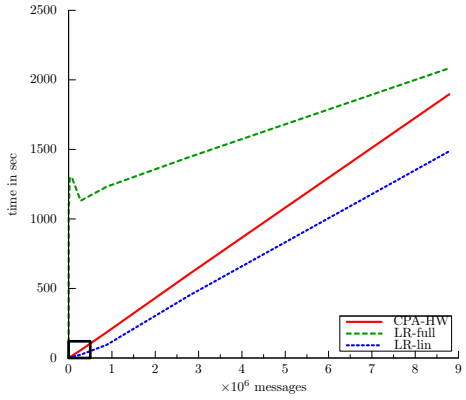
$$\Delta_{\hat{k}}(N) = \mathrm{d}(y_N, (\mathrm{g}_1, \ldots, \mathrm{g}_d) \cdot \vec{\alpha}) + \varepsilon \ ,$$

and the error term ε tends towards 0 asymptotically.

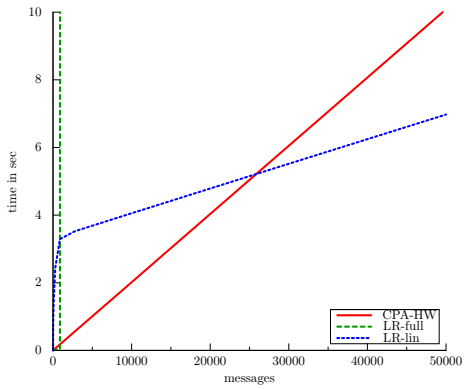
Remark 9: We assumed that the function y_N is defined for every value in $\mathbb{F}_2^n$. Nevertheless in some cases (*e.g.* for a small N) it may happen that y_N is defined only on a strict subset E of $\mathbb{F}_2^n$. In this case, the linear regression processing remains the same, except that lines corresponding to the values in $\mathbb{F}_2^n/E$ are discarded from the matrix $\mathbf{M}$.

Appendix B
Linear Regression vs CPA: a Timing Point of View

As demonstrated in [7], the efficiency of an attack decreases exponentially with the order of masking. In other terms, a successful attack will need a number of messages N growing exponentially w.r.t. the masking order. This implies that higher-order attacks must be able to efficiently deal with a huge number of observations. In particular, the time spent on processing the observations may become a bottleneck. Although the linear regression processing proposed in Appendix A is based on matrix operations, the regression matrix has a constant size w.r.t. to N (thanks to an initial averaging step – see (6) –). More precisely, the linear regression complexity can be split into two parts: the matrix operation which is constant w.r.t. to N and only depends on the basis size; and the least-square computation (a mean of square) which depends on N. Concerning CPA, its complexity relies on the computation of a mean of product, a product of means and two standard deviations that all depend on N. We can thus expect to have a faster attack when using a linear regression (when N is sufficiently large to neglect the matrix operation). To quantify the timing complexity of linear regression, we did several timing measurements and we compared them with those for CPA attacks. We have first processed linear regression with a linear model as a common use case and with a full basis model as the worst possible case (for $n = 8$), that is with the largest regression matrix (*i.e.* the slowest matrix computation). We remind the reader that in the latter case, the attack always failed (*cf.* Sect. 3.4) but here, we are interested in timings in the worst case. The results are plotted in Fig. 8a with a zoom on the small numbers of messages in Fig. 8b. The timings represented in Fig. 8 are measured over



(a)



(b)

Fig. 8: Timing comparison for CPA-HW, LR-lin and LR-full attacks.

100 attacks in an univariate setting. Since CPA and linear regression attacks are both univariate and, in this paper, feed with the same preprocessed vector of observations (a centered product combination of two leakage vectors), only the core computation differs from one to the other.

Results. First and as expected, it can be noticed that the performances of all the attacks are in the same order of magnitude (and thus are computationally viable). Nevertheless, with a linear model, the linear regression becomes noticeably faster than CPA attack (*i.e.* the constant matrix operation cost is small and

can be quickly neglected) for $N > 25,000$ (Fig. 8b). If we focus on linear regression with the full basis, the cost of the matrix operation is not negligible and thus a large number of messages ($N > 10^7$ messages) is needed to counterbalance it. In both cases, when the number of messages is sufficiently large to pass the timing offset due to the matrix operation, linear regression is faster than CPA as expected.

Conclusion. This brief analysis pinpointed the soundness of our attack also in terms of computability. That is in all cases the linear regression encompasses and outmatches CPA.

APPENDIX C
RELATIONSHIP WITH SECOND-ORDER CPA - THE DETAILS

The distance computation (7) can be rewritten:

$$\Delta_{\hat{k}}(N) = \mathrm{d}(y_N, \mathcal{H} \circ F_{\hat{k}}) = \mathrm{d}(y_N, \hat{\varphi} \circ F_{\hat{k}}) \ , \qquad (22)$$

since $\mathcal{H} = \{\hat{\varphi}\}$. Now asymptotically (22) becomes:

$$\lim_{N\to\infty} \Delta_{\hat{k}}(N) = \mathrm{d}(Y, \hat{\varphi} \circ F_{\hat{k}}) = \mathrm{d}(Y, \hat{Y}) \ , \qquad (23)$$

where we have denoted $\hat{\varphi} \circ F_{\hat{k}}$ by $\hat{Y}$ and where we recall that Y denotes $\varphi \circ F_k$.
Equation (23) can be rewritten:

$$\begin{aligned} \lim_{N\to\infty} \Delta_{\hat{k}}(N) &= \sum_{x\in\mathbb{F}_2^n} \left([\varphi \circ F_k](x) - [\hat{\varphi} \circ F_{\hat{k}}](x)\right)^2 \\ &= 2^n \cdot \mathbb{E}\left[\left(Y - \hat{Y}\right)^2\right] \ . \qquad (24) \end{aligned}$$

After developing (24), we get:

$$\begin{aligned} &\lim_{N\to\infty} \Delta_{\hat{k}}(N) \\ &= 2^n \cdot \left(\mathbb{E}\left[Y^2\right] + \mathbb{E}\left[\hat{Y}^2\right] - 2 \cdot \mathbb{E}\left[Y \cdot \hat{Y}\right]\right) \ . \qquad (25) \end{aligned}$$

We recall that the coefficient of correlation $\rho(Y, \hat{Y})$ satisfies:

$$\begin{aligned} \rho(Y, \hat{Y}) &= \frac{\mathrm{cov}\left(Y\ ,\ \hat{Y}\right)}{\sigma_Y \cdot \sigma_{\hat{Y}}} \qquad (26) \\ &= \frac{1}{\sigma_Y \cdot \sigma_{\hat{Y}}} \cdot \left(\mathbb{E}\left[Y \cdot \hat{Y}\right] - \mathbb{E}\left[Y\right] \cdot \mathbb{E}\left[\hat{Y}\right]\right) \ , \end{aligned}$$

From (25) and (26), we deduce:

$$\lim_{N\to\infty} \Delta_{\hat{k}}(N) = a \cdot \rho + b \ , \qquad (27)$$

where

$$\begin{aligned} a &= -2^{n+1} \cdot \sigma_Y \cdot \sigma_{\hat{Y}} \text{ and} \\ b &= 2^n \cdot \left(\mathbb{E}\left[Y^2\right] + \mathbb{E}\left[\hat{Y}^2\right] - 2 \cdot (\mathbb{E}\left[Y\right] \cdot \mathbb{E}\left[\hat{Y}\right])\right) \end{aligned}$$

are independent of the key hypothesis provided σ_Y, $\sigma_{\hat{Y}}$, $\mathbb{E}\left[Y^2\right]$, $\mathbb{E}\left[\hat{Y}^2\right]$, $\mathbb{E}\left[Y\right]$ and $\mathbb{E}\left[\hat{Y}\right]$ are also independent of the key hypothesis. This is clearly the case with typical first-order masking schemes involving an addition, like Boolean and arithmetic masking schemes.

APPENDIX D
A WORD ABOUT MAXIMUM LIKELIHOOD APPROACH

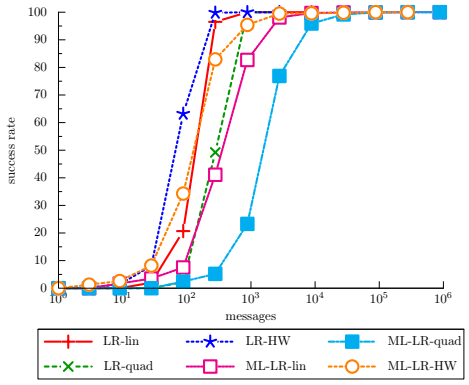
In Sect. 3.5.2, we have exhibited the link between our attack and the Maximum Likelihood approach with a *merge* of the mixture components. We propose here to go a step further by using a Maximum Likelihood test as the distinguisher of Step 6 instead of the mean-of-square.

We recall that the Maximum Likelihood test simply consists in computing the product $\prod_i f_{\hat{z}_i}(\ell_1^i, \ell_2^i)$ as already mentioned in Sect. 3.5. To be able to compute this latter, the adversary must have on hand the pdf f_z for every z. In view of the approximation that is made in (16), the only parameter of the pdf that he has to guess is $Y|Z = z$. This latter is already available, as an approximation, at Step 5 of the attack described in Sect. 3.3. With this pdf approximation on hand, the adversary replaces the mean-of-square distinguisher used in Step 6 by the Maximum Likelihood test and then outputs the key-candidate which gave rise to the highest discriminating value.

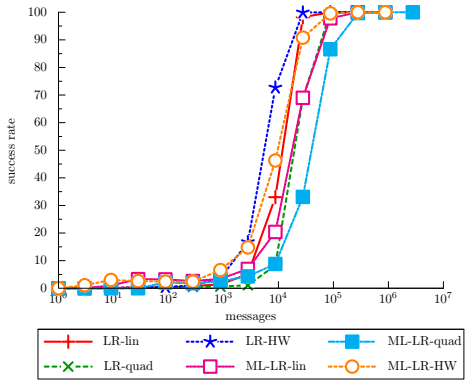
In Sect. 3.5.2, we have already shown that this maximum likelihood approach cannot be more efficient than the mean-of-square approach. To confirm and strengthen this fact experimentally, we have conducted some simulations in the Boolean case and scenario 1.

The simulations parameters are the same as in Sect. 4 and the results are plotted in Fig. 9.

As expected, for the same basis, the maximum likelihood approach is never more efficient than the corresponding linear regression approach. More interestingly, the maximum likelihood efficiency is largely lower than the linear regression (by a factor of 3). The reason is that, the approximation of Y returned by the linear regression is chosen w.r.t. the distance defined in (7). In other words, the approximation of Y itself is the result of a discriminating process. Then applying another discriminating test such as the maximum likelihood can only bring more noise.



(a) No noise



(b) $\sigma = 4$

Fig. 9: Comparison between mean-of-square and Maximum Likelihood approach against Boolean masking in Scenario 1