A Comparative Study of Mutual Information Analysis under a Gaussian Assumption

Amir Moradi[1,⋆], Nima Mousavi[2], Christof Paar[1], Mahmoud Salmasizadeh[2]

[1] Horst Görtz Institute for IT Security, Ruhr University Bochum, Germany
[2] Electronics Research Center, Sharif University of Technology, Tehran, Iran
{moradi, cpaar}@crypto.rub.de, nm@ee.sharif.edu, salmasi@sharif.edu

Abstract. In CHES 2008 a generic side-channel distinguisher, Mutual Information, has been introduced to be independent of the relation between measurements and leakages as well as between leakages and data processed. Assuming a Gaussian model for the side-channel leakages, correlation power analysis (CPA) is capable of revealing the secrets efficiently. The goal of this paper is to compare mutual information analysis (MIA) and CPA when leakage of the target device fits into a Gaussian assumption. We first theoretically examine why MIA can reveal the correct key guess amongst other hypotheses, and then compare it with CPA proofs. As our theoretical comparison confirms and shown recently in ACNS 2009 and CHES 2009, the MIA is less effective than the CPA when there is a linear relation between leakages and predictions. Later, we show detailed practical comparison results of MIA and CPA, by means of several alternative parameters, under the same condition using leakage of a smart card as well as of an FPGA.

1 Introduction

1.1 History

In 2000s side-channel attacks made a challenge on cryptographers' point of view that not only mathematical security of a cryptographic algorithm but also physical security of its implementation should be justified to call a system "secure". Since a side-channel adversary needs physical access to the target device, pervasive devices such as smart cards and RFIDs which can operate in an uncontrolled environment are at risk of such powerful implementation attacks, e.g., differential power analysis (DPA) [8] and electromagnetic analysis (EMA) [6, 14] which extract key materials by monitoring respectively the power consumption and electromagnetic emanation of the attacked device.

In a DPA attack, measurements are categorized into usually two (but can more, e.g., [9]) sets using a partition function which relies on one (respectively can on more) bit of intermediate value depending on the secret and a known input/output. Then, clear peaks in difference of means of two sets indicate the

⋆ Amir Moradi performed most of the work described in this contribution when he has been with Electronics Research Center of Sharif University of Technology.

correct key guess. Afterwards, a more general scheme namely correlation power analysis (CPA) [2] has been introduced that uses a hypothetical power consumption model ideally close to the actual leakage function of the target device. Depending on the hypothetical power model, known input/output, and each key guess, hypothetical power values are constructed and compared with measurements by means of Pearson correlation coefficient. Similarly to DPA, clear peaks in correlation coefficients identify the correct key hypothesis. It should be noted that improved side-channel key recovery attacks such as template attacks [4] and higher order attacks [12] have been designed to make the key recovery process more efficient or to defeat a range of countermeasures.

On the other hand, recently Gierlichs et al. introduced Mutual Information Analysis (MIA) [7] in which Mutual Information of guessed intermediate values and measurements is used as a side-channel distinguisher. In contrary to CPA, MIA has been designed to be effective without any knowledge about the particular dependencies between the processed data and leakages as well as between leakages and measurements. Also, CPA works efficiently under a Gaussian assumption when means and variances are estimated. However, MIA still can reveal the secrets if this Gaussian assumption does not hold.

Recently two articles [13] and [15] on mutual information analysis and its application have been published which shows that the topic is of highly interest for the relevant research community. In [13] the authors exposed theoretical foundation of the side-channel distinguishers, including MIA, and assessed their limitations and assets. They have answered a number of questions regarding the efficiency of MIA and the condition where it is better than the other distinguishers. Moreover, they generalized MIA to higher orders. As a short result, they showed that the MIA is less efficient than the CPA when the leakage is a linear function of the predictions, e.g., Hamming weight model. Further, it has been shown that a proposed extension of MIA is more efficient than existing higher-order attacks on masked implementations. Both articles argued that in addition to histograms, which has been used in the original description of MIA [7] to estimate the probability density functions, using other methods such as Kernel ones and parametric ones allows to improve the efficiency of the MIA attacks. The authors of [15] also discussed on the conditions in which the MIA can be more effective than the CPA, and proposed to use alternative probability-distance measure tools (which allows deciding which subkey is the most likely to be the correct one) with different impacts on the attack efficiency.

1.2 Motivation

Although it has been showed that the CPA is more effective than the MIA under a certain assumption, they have not been compared precisely with each other to make it clear how much the CPA is better than the other. What exactly we want to discuss in this paper is to compare the CPA and the MIA when the target device is a so-called general-purpose microcontroller or an FPGA whose leakages fit into a Gaussian assumption. In fact, we are going to answer several questions that arise by comparing MIA and CPA. These questions can be categorized as

follows. Note that in all of our comparisons, the histogram method which has been proposed in the original description of MIA, is used.

- How much CPA can reveal the secrets better than the MIA, i.e., the revealed secret reported by CPA is more distinguishable amongst others than that reported by MIA?
- How is the threat of MIA in the presence of noise? Does it still work when CPA can not reveal the secrets, or vice versa?
- As mentioned in [7], it is not essential to use a hypothetical leakage model perfectly matching with the actual leakage function of the attacked device, and it is sufficient to use a leakage function proportional to the particular actual one. What is the result if this inaccurate leakage function is used in a CPA? Does CPA works when MIA does not?
- Does CPA need less number of traces than a corresponding MIA attack, under the same condition?
- How much the hypothetical leakage model is independent of the actual leakage of the attacked device? Does it work without any knowledge about the particular dependencies? Is it the same for CPA? or MIA works more efficiently?
- To the best of our knowledge, success rate of a CPA targeting leakage of a function having linear relation with secrets is not perfectly 100%, so does MIA work better in this situation?

1.3 Organization

This paper is organized as follows. In Section 2 we give an overview of the side-channel model used in our theoretical comparison. Section 3 and Section 4 recall the basic notations of CPA and MIA respectively, and compare the cases where CPA and MIA are not capable of revealing the secret. Later, in Section 5 we present the practical results of MIA and CPA under the same condition for several different situations to give an insight on answer of the aforementioned questions. Finally, conclusions are given by Section 6.

2 Side-Channel Model

In this section, we restate the theoretical model for side-channel leakage of a cryptographic device which has been introduced in [7]. Later, we will use the notations given here to theoretically discuss on CPA and MIA.

As shown by Fig. 1, a cryptographic device which performs a cryptographic algorithm, E, using a secret key, k, on input values, x, and generates outputs y as $\mathrm{E}_k(x)$ is taken into account. During the computation of the device, intermediate values depending on the input x and unknown key k are changed and lead to some bit flips modeled by w. Changing the internal states of the target device leaks through a side-channel leakage function $\mathrm{L}(w) = l$. However, what a side-channel adversary can observe is a noisy measurement, o, (usually acquired by

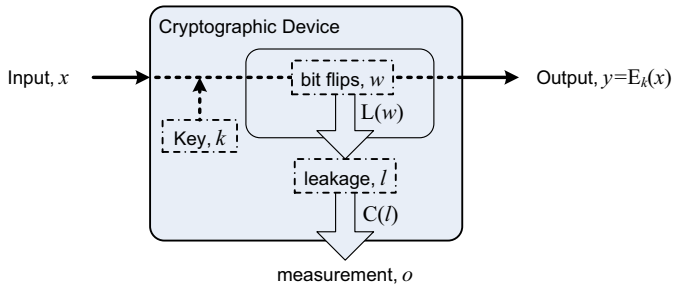

Fig. 1. Side-channel leakage model

a digital oscilloscope) of the leakage l, $o = \mathrm{C}(l)$. In short, a noisy function of bit flips of internal state would be observed by a side-channel adversary,

$$o = \mathrm{C}(\mathrm{L}(w)).$$

To mount a side-channel key recovery attack, an adversary collects q queries of measurements, $o_i,\ 0 < i \leq q$, knowing corresponding inputs x_i and/or outputs y_i. Supposing that instants of time when the cryptographic device operates on the target secret are known causes each measurement o_i to contain single value[3]. During the attack, the adversary using "divide and conquer" scheme guesses parts of the secret key k and estimates the bit flips w, then using the measurements o and a side-channel distinguisher tries to find the most probable hypothesis close to the correct one.

3 Correlation Power Analysis

If we model the leakage for each key hypothesis $k \in$ key space K as l_k and similarly to [2] suppose a linear relation between adversary's observation o and leakages, we can write the following equation for the correct key guess, k_0:

$$o = a \cdot l_{k_0} + b, \tag{1}$$

where a is a constant scaling factor of the channel C, and b is a random variable concerning the noise model (Gaussian assumption). We also assume that the random variable b is independent of l_{k_0}. An example for the leakage function satisfying the linear relation of Eq. (1) would be Hamming weight (HW) of the intermediate values as well as HW of a part of the intermediate values since similar to Eq. (1) there is a linear relation between HW and partial HW of a value with an additive and independent random variable, i.e.,

$$\mathrm{HW}(x) \;=\; \mathrm{HW}(x_L) \;+\; \mathrm{HW}(x_R)\ ;\quad x = x_L \| x_R.$$

[3] Otherwise the adversary needs to acquire longer measurements and takes each instant of time separately into account.

Considering linear relation of Eq. (1), correlation coefficient between two random variables of adversary's observation, o, and leakage for a wrong key, $l_{k'};\ k' \in K,\ k' \neq k_0$ can be written as follows:

$$\begin{aligned}\rho_{ol_{k'}} &= \frac{\mathrm{Cov}(o,l_{k'})}{\delta_o \cdot \delta_{l_{k'}}} = \frac{\mathrm{Cov}(a \cdot l_{k_0}+b,l_{k'})}{\delta_o \cdot \delta_{l_{k'}}} = \frac{a \cdot \mathrm{Cov}(l_{k_0},l_{k'})}{\delta_o \cdot \delta_{l_{k'}}} \\ &= \frac{a \cdot \mathrm{Cov}(l_{k_0},l_{k_0})}{\delta_o \cdot \delta_{l_{k_0}}} \cdot \frac{\mathrm{Cov}(l_{k_0},l_{k'})}{\delta_{l_{k_0}} \cdot \delta_{l_{k'}}} \\ &= \rho_{ol_{k_0}} \cdot \rho_{l_{k_0}l_{k'}}.\end{aligned} \tag{2}$$

Since $|\rho_{l_{k_0}l_{k'}}| \leq 1$, the leakage random variable has the maximum correlation with observations for the correct key guess. Thus, a CPA reveals the correct key as long as $|\rho_{l_{k_0}l_{k'}}| < 1$ for $\forall k' \in K;\ k' \neq k_0$. In other words, CPA is not capable of revealing the secret when $\exists k' \in K,\ k' \neq k_0$ for which $l_{k'}$ has a linear relation with l_{k_0}. Note that we discuss on this issue more in Section 4.

4 Mutual Information Analysis

By classifying a hypothetical leakage l_k for a key hypothesis $k \in K$ and making histograms of the measurements o, we can estimate $\mathrm{P}(l_k)$ and $\mathrm{P}(o)$ as the probability distributions of the random variables hypothetical leakage and measurements respectively. Further, we can estimate conditional probability distribution of measurements given a hypothetical leakage as $\mathrm{P}(o|l_k)$. As illustrated in [7], we can now estimate conditional entropy $\mathrm{H}(o|l_k)$ and hence have an estimation for mutual information of measurements and each key hypothesis as

$$\mathrm{I}(o;l_k) = \mathrm{H}(o) - \mathrm{H}(o|l_k). \tag{3}$$

First we should point out a markov chain as $l_{k'} \rightarrow l_{k_0} \rightarrow o$ for $\forall k' \neq k_0$ and equivalently the equality of $\mathrm{P}(o|l_{k_0}l_{k'}) = \mathrm{P}(o|l_{k_0})$. It can be easily verified by considering Eq. (1), the probability distribution of o given l_{k_0} is similar to the distribution of b (with a constant difference) which is also independent of $l_{k'}$. Note that o is not generally independent of $l_{k'}$. Now, writing $\mathrm{I}(o;l_{k_0}l_{k'})$ in two ways gives:

$$\mathrm{I}(o;l_{k_0}l_{k'}) = \mathrm{I}(o;l_{k_0}) + \mathrm{I}(o;l_{k'}|l_{k_0}) \overset{(a)}{=} \mathrm{I}(o;L_{k_0}) \tag{4}$$

$$\mathrm{I}(o;l_{k_0}l_{k'}) = \mathrm{I}(o;l_{k'}) + \mathrm{I}(o;l_{k_0}|l_{k'}), \tag{5}$$

where (a) follows from the aforementioned markov chain. Finally, we can write

$$\mathrm{I}(o;l_{k'}) = \mathrm{I}(o;l_{k_0}) - \mathrm{I}(o;l_{k_0}|l_{k'}). \tag{6}$$

Since mutual information is a positive function, $\mathrm{I}(o;l_{k'}) \leq \mathrm{I}(o;l_{k_0})$. Therefore, the leakage random variable for the correct key, l_{k_0}, has the maximum mutual information with random variable observations, and hence MIA gives the correct

key as long as $\mathrm{I}(o; l_{k_0}|l_{k'}) \neq 0$. In other words, MIA is not capable of revealing the secret when $\exists k' \neq k_0$ for which mutual information of o and l_{k_0} given $l_{k'}$ is zero. Further, a special case is when $\exists k' \neq k_0$ for which there is a one-to-one relation between l_{k_0} and $l_{k'}$. In such a case, $\mathrm{H}(l_{k_0}|l_{k'}) = \mathrm{H}(l_{k_0}|l_{k'}o) = 0$, so $\mathrm{I}(o; l_{k_0}|l_{k'}) = \mathrm{H}(l_{k_0}|l_{k'}) - \mathrm{H}(l_{k_0}|l_{k'}o) = 0$.

Comparison with CPA As mentioned in Section 3, CPA does not work if $\exists l_{k'},\ k' \neq k_0$ for which $\rho_{l_{k_0} l_{k'}} = 1$. This implies a linear relation between l_{k_0} and $l_{k'}$. Since a linear relation is also a one-to-one relation, MIA does not work in this case too. In general, MIA is unsuccessful if $\exists l_{k'},\ k' \neq k_0$ for which $\mathrm{I}(o; l_{k_0}|l_{k'}) = 0$ whereas CPA is successful in this case unless for the aforementioned linear relation. As a consequence, the situations in which CPA is unsuccessful are fewer than that of MIA.

Generally we can say that correlation coefficient captures the **linear** relation between two random variables while mutual information captures **any** relation between two random variables. Therefore, CPA is not successful when there is a **linear** relation between observations and leakage variable for a wrong key guess as much as a **linear** relation between observations and leakage variable for the correct key. However, MIA is not successful when there is **any** relation between observations and leakage variable for a wrong key guess as much as a **linear** relation between observations and leakage variable for the correct key. It seems that in the later case, capturing **any** relation between leakage variable for a wrong key and observations, when a **linear** model holds for the correct key, leads to a weaker attack.

5 Practical Comparison

5.1 Target Devices and Measurement Setup

We developed two experimental platforms. One is a programmable smart card embedded by an Atmel AVR ATmage163 microcontroller [1] in which we have developed an implementation of the AES Rijnael encryption algorithm. Since the microcontroller has an 8-bit architecture, each subbyte transformation is executed separately using pre-computed look-up tables. Further, to obtain the power consumption traces the voltage drop over a 100Ω resistor placed in GND pin of the microcontroller has been measured by a digital oscilloscope with the sampling rate of 500 MS/s. Note that a standard smart card reader, e.g., [5], has been used to communicate with a PC. The second one is a XC2S150 Spartan-II FPGA [16] running again the AES-128 encryption function. Only one combinational Sbox based on the netlist presented in [3] is implemented in the device and is shared in subbytes operation. Side-channel observations are collected by measuring the differential voltage of a 3.3Ω resistor in the $\mathrm{V}_{\mathrm{CCINT}}$ line with the same sampling rate of the smart card case. The only difference between the two devices is due to the clock signal which is supplied by the smart card reader in the first one and an external signal generator with a frequency of 1 MHz in the later one.

5.2 Results

Unmatched Power Models - Because of a pre-charged (or pre-discharged) architecture of microcontrollers' data bus usually HW of data transfered by the bus leaks through power traces. Thus, an efficient CPA attack using HW model can be mounted on microcontroller-based devices. However, in MIA it is not needed to apply a power model matching with the particular leakage function, e.g., HW [7]. For instance we can use a part, e.g., 7 bits, of intermediate values instead of their HW. The first question here is due to what happens if this model is used in a CPA. To examine this, we have performed MIA and CPA on 256 traces measured from our microcontroller due to 256 chosen plaintexts to cover all possible plaintexts when the target secret is a key byte and the leakage of the Sbox output is the target leakage. Note that in MIA we need to set a parameter due to the number of bins in which measurements are classified to build the histograms. In the rest of this paper, we denote this parameter as "number of bins". In contrary to [7] we have used an automated way to make histograms, i.e, we have just divided the range of measurement values (of course independently for each instant of time) by the number of bins to equal intervals. Also, we have used the same power model, 7 bits of Sbox output, in a CPA attack. Fig. 2 shows the maximum mutual information and correlation coefficient for each key hypothesis when two bins used for measurement classifications in the MIA. We have repeated this task for all possible power models as a part of the intermediate value on the same power traces, i.e., $\sum_{i=1}^{7} \binom{8}{i} = 254$ power models for MIA and CPA. Interestingly, in all cases both MIA and CPA are capable of revealing clearly the correct key byte amongst other hypotheses[4]. It is because of a high correlation between decimal values and their HWs. Suppose a random variable R containing decimal values of a n-bit binary data. Correlation coefficient of R and $\mathrm{HW}(R)$ is $\{1, 0.95, 0.88, \ldots, 0.65, 0.61\}$ for $n = 1, 2, 3, \ldots, 7, 8$. Thus, still there is a high correlation between the actual leakage of the target device, i.e., HW, and the decimal values as the power model in a CPA.

Further, we have done this procedure for all possible HW of a part of the target leakage, i.e., $\sum_{i=1}^{8} \binom{8}{i} = 255$ power models, which led to the same result as expected. As a result, in this case when the target device is a microcontroller with HW leakage, CPA like MIA can recover the secret for all possible leakage functions.

Distinguishability - One important criterion in comparison of MIA and CPA is how much a secret revealed by MIA is more distinguishable amongst other hypotheses than that by CPA (or vice versa). In order to have a parameter to compare the distinguishabilities we have computed the normalized difference between two most probable hypotheses. Suppose that $V = \{v_1, v_2, \ldots, v_n\}$ is a

[4] Note that we have computed absolute of correlation coefficient in CPAs; for this reason all coefficients seen in Fig. 2 are positive.

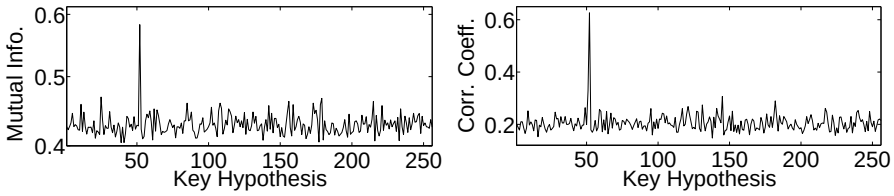

Fig. 2. Mutual information (left) and correlation coefficient (right) values over a key byte hypotheses using 7 bits of an Sbox output as the power model.

set of correlation coefficient or mutual information values reported by a CPA or MIA respectively, and suppose that v_i values are sorted from the biggest to the smallest, i.e., $v_i \geq v_j$; $i > j$. We have used $(v_1 - v_2)/v_1$ as the normalized difference to compare the distinguishabilities. Note that according to the previous results we know that always v_1 relates to the correct hypothesis.

Since the number of bins is a new parameter which does not exist in CPA, we examined its effect on distinguishability too. Although it is suggested in [7] "to use as many bins as there are distinct values in the domain covered by the sample set to ensure that no information is lost", we could not obtain good results by choosing big number of bins, we hence limited the number of bins to $\{2, 3, \ldots, 16\}$. Indeed, by selecting more number of bins the correct hypothesis was not distinguishable clearly amongst other hypotheses (of course using a particular number of traces). First we have considered all possible HW models (similarly to the previous experiments) and all possible number of bins, i.e., 255 power models for CPA and 255×15 power models for MIA. Then, we got average of distinguishability values over number of bits contributed in the power model leading to the diagram shown by Fig. 3(a). Further, we have repeated this scenario for a part of the target leakage, i.e., some bits of the Sbox output, as the power model, i.e., 254 power models for CPA and 254×15 power models for MIA. The comparative diagram is shown by Fig. 3(b). Obviously when a HW model is used, there is not a big difference between the normalized differences in MIA and CPA. Also, by increasing the number of bits contributed in HW model, the correct secret would be more distinguishable. However, better results are achieved by CPA when a part of the target leakage is used as the power model. For instance, when a high number of bits, e.g., 6 and 7, construct the power model, CPAs reveal the secret more distinguishably than MIA especially than those with high number of bins.

Minimum Number of Traces - Another parameter investigated is the minimum number of traces needed to have a successful attack in MIA and CPA. Since the role of thumb illustrated in [11] still is not examined for MIA, we had to repeat the previous attacks using different number of traces. To do so, because of the high computation overload, we have limited our examinations to eight power models for each number of bits contributed in the power model, i.e.,

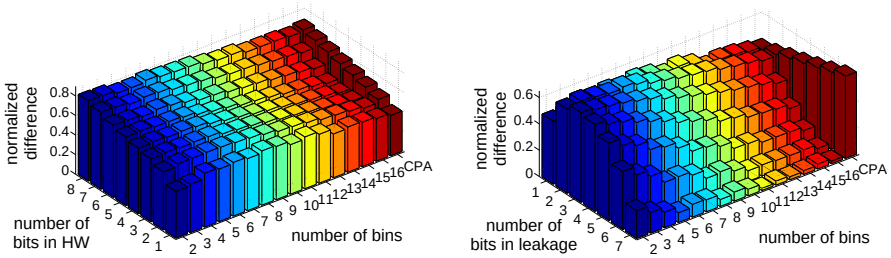

Fig. 3. Average of the normalized difference of two most probable hypotheses in CPA and MIA over the number of bins and over the number of bits contributed in (a) HW model and (b) a part of the target leakage.

$8 \times 7 = 56$ models when a part of the target leakage constructs the power model and $56 + 1$ models when HW power model is used. Moreover, we have limited the number of bins to $\{2, 4, 8, 16\}$. Getting the average on the minimum number of traces over the number of bits in power models led to diagrams shown by Fig. 4. As expected when a HW model is used in CPA as well as in MIA, the more bits contributed in the model, the less traces are needed. However, an unpredicted treatment from MIA is seen when a part of the target leakage is used as the power model. For instance, when the number of bins is 4, contributing 1 or 7 bits of the Sbox output leads to higher number of traces than other cases. More interestingly, in almost all cases CPA clearly needs less traces to reveal the secret.

Noise Effect - The next criterion by which MIA and CPA are compared is their capability of revealing the secret in the presence of noise. A parameter which we have used to compare them is the success rate. To compute the success rate for a certain condition, measurements are collected separately for all possible values for the key (in this case one byte). After performing a certain attack to recover the secret key separately for each key value using the corresponding power traces, the success rate would be computed as a ratio of the number of cases where the secret is recovered over the number of all cases. Indeed, based on the idea presented in [10] we are going to find a threshold of the noise standard deviation for successful MIA and CPA attacks. We first computed the standard deviation of electronic noise in our measurement setup by measuring 2 000 traces when our microcontroller runs the same function on a fixed plaintext. Then, for each possible key byte we collected 256 traces due to all possible values of a plaintext byte. In order to compute the success rate of an attack for a given noise standard deviation, we have added Gaussian distributed random noise with a zero mean value and the given standard deviation deducted from the estimated electronic noise standard deviation to each point of the measured power traces independently,

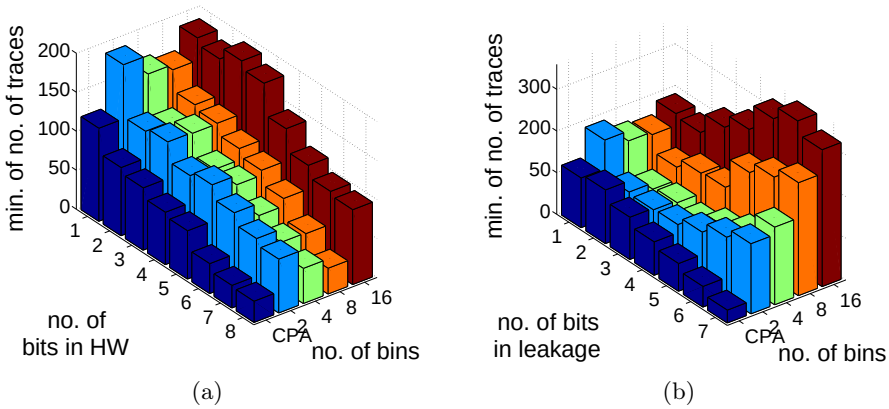

Fig. 4. Average of the minimum number of traces in CPA and MIA over the number of bins and over the number of bits contributed in (a) HW model and (b) a part of the target leakage.

then the attacks are performed for each key byte separately. Further, since the noise values are chosen randomly from a distribution, we have repeated each of the above procedure ten times to improve the accuracy of our estimation about the success rate. For instance to obtain the aforementioned threshold for a CPA using HW of 8 bits of the Sbox output, we have performed the attack 256×10 for each of 25 given noise standard deviations, i.e., in sum 64 000 times. This procedure has been repeated for MIA with HW model as well as a model which uses a part of the target leakage. However, we limited ourselves to HW model of 8 bits, a model using 3 bits of the Sbox output, and four different number of bins as $\{2, 4, 8, 16\}$. As shown by Fig. 5, the CPA which uses HW of all 8 bits has the highest threshold, and the noise standard deviation threshold decreases by increasing the number of bins in MIAs. Further, for each power model the best result is achieved by using 4 bins in measurements classification. Note that since the variance of signal is the same for all attacks, $0.095\,(\mathrm{mA})^2$, one can compare the SNR thresholds for the plotted diagrams in Fig. 5.

Linear Functions - It is clear that the success rate of an attack targeting the leakage of a linear function, e.g., an XOR, is not perfect if the leakage of the attacked device fits into a HW or HD model. In order to examine the success rate practically we have performed CPAs as well as MIAs on the traces we had collected for the noise effect, i.e., 256 traces for each possible key byte. Fig. 6 shows the success rate of CPA and MIA for different number of bins when the HW of an AddRoundKey output byte is considered as the target leakage. Moreover, we have repeated this task with different leakage models as a part of the AddRoundKey output byte, but neither CPA nor MIA (for any number of

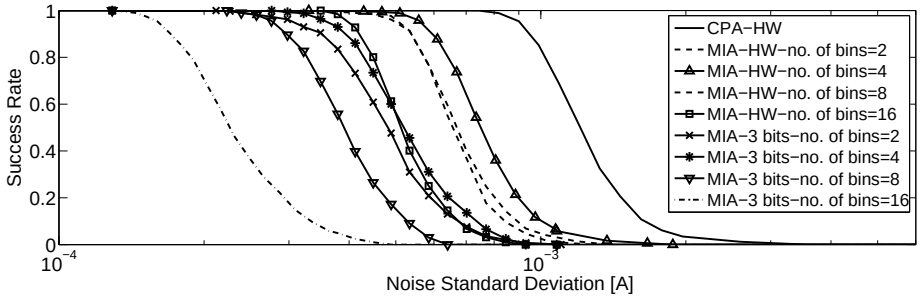

Fig. 5. Success rate of CPA and MIA for different conditions over noise standard deviation.

bins) led to a success rate over 0.05. From our point of view, in this case MIA does not have an advantage in comparison with CPA.

Independency on the Particular Leakage Function - MIA has been introduced to be independent of the particular relation between measurements and processed data [7]. In order to evaluate this we have measured a set of traces of our FPGA board and performed MIAs and CPAs using a set of different leakage models. Considering a HD model for an Sbox output, of course, MIA as well as CPA are successful. However, when a HW or a part of the Sbox output is used as the leakage model, neither MIA nor CPA can reveal the secret. We applied all power models used before on 10 000 traces of a random plaintext byte that measured separately for each possible key byte, but none of the attacks reported a success rate more than 0. In contrast, using a part of the XOR result of two consecutive Sbox outputs, i.e., bit flips, both MIA and CPA work similarly to the results presented so far. To our knowledge we can not say that MIA or any power analysis attack can work efficiently without any knowledge about the

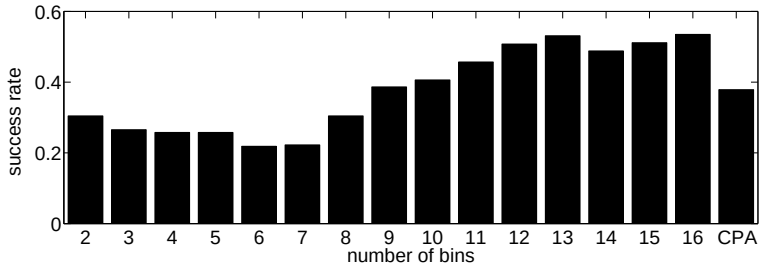

Fig. 6. Success rate of CPA and MIA for different number of bins targeting the leakage of an XOR operation using a HW model.

leakage function of the target device. At least we should know how intermediate values affect the side-channel leakage. For instance, we use HW model (or a part of the Sbox output) when attacking a microcontroller, because we know there is a pre-charged bus inside. Generally, one can not perform a universal attack on some traces measured from an unknown unprotected device, of course knowing algorithm and inputs/outputs.

6 Conclusions

We studied MIA in comparison with CPA from theoretical as well as practical points of view. First we have examined the theoretical reason of why MIA can recover the secrets, then expressed a situation in which in contrary to MIA, CPA is capable of revealing the secret. Further, we have compared them using the results of practical attacks on a microcontroller-based smart card and an FPGA board. According to the results, and as expected from the results of [13] and [15] there is no advantage for MIA over CPA when the leakage of the target device fits into a Gaussian assumption. Indeed, MIA works roughly the same as CPA, but with more parameters that affect the efficiency of the attack, i.e., number of bins. Further, MIA has more computational overhead in comparison with CPA especially for high number of bins. For instance, performing all of the attacks illustrated in Section 5.2 took weeks of computation on a 3GHz Intel Core2 PC.

In short, in comparison with CPA, MIA works worse in the presence of noise; it distinguishes the correct guess amongst other hypotheses weaker than CPA considering a part of the target leakage as the power model; it is not perfectly independent of the particular leakage function of the attacked device (similar to CPA); number of the traces which is needed to have a successful MIA attack is more than a corresponding CPA attack; and similarly to CPA, it is not 100% successful when the leakage of a function that has a linear relation with the secret is the target leakage.

Note that the results expressed here are only correct when the target device is an unprotected implementation which has side-channel leakage like HW or HD model. However, the MIA especially its multi-dimensional extension is much more effective than the CPA and respectively than classical higher-order CPA attacks when the leakage of the target device is not linearly proportional to the predictions, e.g., masked implementations.

References

1. Atmel. AVR ATmega163 Data Sheet. `http://www.atmel.com/dyn/resources/prod_documents/doc1142.pdf`.
2. E. Brier, C. Clavier, and F. Olivier. Correlation Power Analysis with a Leakage Model. In *Cryptographic Hardware and Embedded Systems - CHES 2004*, volume 3156 of *Lecture Notes in Computer Science*, pages 16–29. Springer, 2004.

3. D. Canright. A Very Compact S-Box for AES. In *Cryptographic Hardware and Embedded Systems - CHES 2005*, volume 3659 of *Lecture Notes in Computer Science*, pages 441–455. Springer, 2005.
4. S. Chari, J. Rao, and P. Rohatgi. Template Attacks. *Cryptographic Hardware and Embedded Systems-Ches 2002: 4th International Workshop, Redwood Shores, CA, USA, August 13-15, 2002: Revised Papers*, 2002.
5. CHIPDRIVE. Smart Card Reader. `http://www.chipdrive.de/cgi-bin/edcstore.cgi?category=Einkaufen;01_Chipkartenleser&user_action=detail&catalogno=P208199`.
6. K. Gandolfi, C. Mourtel, and F. Olivier. Electromagnetic Analysis: Concrete Results. In *Cryptographic Hardware and Embedded Systems - CHES 2001*, volume 2162 of *Lecture Notes in Computer Science*, pages 251–261. Springer, 2001.
7. B. Gierlichs, L. Batina, P. Tuyls, and B. Preneel. Mutual Information Analysis. In *Cryptographic Hardware and Embedded Systems - CHES 2008*, volume 5154 of *Lecture Notes in Computer Science*, pages 426–442. Springer, 2008.
8. P. C. Kocher, J. Jaffe, and B. Jun. Differential Power Analysis. In *CRYPTO 1999: Proceedings of the 19th Annual International Cryptology Conference on Advances in Cryptology*, pages 388–397, London, UK, 1999. Springer-Verlag.
9. T.-H. Le, J. Clédière, C. Canovas, B. Robisson, C. Servière, and J.-L. Lacoume. A Proposition for Correlation Power Analysis Enhancement. In *Cryptographic Hardware and Embedded Systems - CHES 2006*, volume 4249 of *Lecture Notes in Computer Science*, pages 174–186. Springer, 2006.
10. F. Macé, F.-X. Standaert, and J.-J. Quisquater. Information Theoretic Evaluation of Side-Channel Resistant Logic Styles. In *Cryptographic Hardware and Embedded Systems - CHES 2007*, volume 4727 of *Lecture Notes in Computer Science*, pages 427–442. Springer, 2007.
11. S. Mangard, E. Oswald, and T. Popp. *Power Analysis Attacks: Revealing the Secrets of Smart Cards*. Springer, 2007.
12. T. S. Messerges. Using Second-Order Power Analysis to Attack DPA Resistant Software. In *Cryptographic Hardware and Embedded Systems - CHES 2000*, volume 1965 of *Lecture Notes in Computer Science*, pages 238–251. Springer, 2000.
13. E. Prouff and M. Rivain. Theoretical and Practical Aspects of Mutual Information Based Side Channel Analysis. In *Applied Cryptography and Network Security - ACNS 2009*, volume 5536 of *Lecture Notes in Computer Science*, pages 499–518. Springer, 2009.
14. J.-J. Quisquater and D. Samyde. ElectroMagnetic Analysis (EMA): Measures and Counter-Measures for Smart Cards. In *Smart Card Programming and Security - E-smart 2001*, volume 2140 of *Lecture Notes in Computer Science*, pages 200–210. Springer, 2001.
15. N. Veyrat-Charvillon and F.-X. Standaert. Mutual Information Analysis: How, When and Why? In *Cryptographic Hardware and Embedded Systems - CHES 2009*, Lecture Notes in Computer Science. Springer, 2009. to appear. Also available at `http://www.dice.ucl.ac.be/~fstandae/PUBLIS/67.pdf`.
16. XILINX. Spartan-II FPGA Family Data Sheet. `http://www.xilinx.com/support/documentation/data_sheets/ds001.pdf`.