A proposition for Correlation Power Analysis enhancement

Thanh-Ha Le[◇], Jessy Clédière[◇], Cécile Canovas[◇], Bruno Robisson[◇], Christine Servière[⋆], Jean-Louis Lacoume[⋆]

[◇]CEA-LETI
17 avenue des Martyrs, 38 054 Grenoble Cedex 9, France,
[⋆]Laboratoire des Images et des Signaux
961 rue de la Houille Blanche, 38 402 Saint Martin d'Hères Cedex
{thanhha.le, jessy.clediere, cecile.canovas, bruno.robisson}@cea.fr,
{christine.serviere, jean-louis.lacoume}@inpg.fr

Abstract. Cryptographic devices are vulnerable to the nowadays well known side channel leakage analysis. Secret data can be revealed by power analysis attacks such as Simple Power Analysis (SPA), Differential Power Analysis (DPA) and Correlation Power Analysis (CPA). First, we give an overview of DPA in mono-bit and multi-bit cases. Next, the existing multi-bit DPA methods are generalized into the proposed Partitioning Power Analysis (PPA) method. Finally, we focus on the CPA technique, showing that this attack is a case of PPA with special coefficients and a normalization factor. We also propose a method that allows us to improve the performance of CPA by restricting the normalization factor.

Keywords. side channel, power analysis, DPA, multi-bit DPA, PPA, CPA, correlation, DES, AES.

1 Introduction

Differential analysis on side channel signals were set up by Kocher et al. [10, 11] on DES algorithm. Power consumption signals of CMOS chips were used, giving good results to retrieve key values by difference of mean curves selected on a defined criteria. Electromagnetic radiation signals, acquired by dedicated sensors, were then successfully used by several authors [17, 20, 21]. Hereafter, the terms DPA and CPA have been generalized for any side channel signal (i.e., power consumption and electromagnetic radiation signals). Since then, differential analysis has been applied on various cryptographic algorithms, including DES and AES, and several countermeasures have been proposed to secure those algorithms from first and high order differential attacks [9, 7, 1, 2]. Some authors [3, 14, 4, 22] have extended Kocher's et al. attack, introducing multi-bit DPA methods to improve differential analysis. Currently, there are different multi-bit DPA concepts. We propose in this paper the Partitioning Power Analysis (PPA) method to merge these concepts in a single form.

Lately, the power analysis technique based on the correlation has been widely studied [5, 6, 8, 12]. We propose a reviewing of the correlation approach suggested by Brier et al., named Correlation Power Analysis [5], and the study of its normalization effect. We then propose a way to enhance the performance of CPA. The analytical results are finally confronted with the experimental ones.

The paper is organized as follows. Section 2 starts with an overview of power analysis including the original DPA method, the multi-bit PPA concept and the correlation based CPA method. In Sect. 3, a discussion about the CPA attack and its normalization factor is expressed. We also propose in this section a method to enhance CPA. Experimental results with electromagnetic radiation signals are shown in Sect. 4 and a brief conclusion is given in the last section.

2 Power Analysis Techniques

2.1 Differential Power Analysis

Differential Power Analysis was originally proposed by Kocher et al. [11]. This analysis is based on the fact that the power dissipation to manipulate one bit to 1 is different from the power dissipation to manipulate it to 0. To test different keys K_s, DPA uses N cipher messages (or plain messages) C_i $(i = 1 \dots N)$ and a selection function $D(C_i, b, K_s)$. This boolean function computes the value of an examined bit b, for example a bit of the S-box output. DPA computes a differential trace $\Delta_D(b)$ as the difference between the average of the traces for which $D(C_i, b, K_s)$ is 1 and the average of the traces for which $D(C_i, b, K_s)$ is 0. If we note $W(C_i)$ the power consumption or electromagnetic radiation signal corresponding to the message C_i, the differential trace $\Delta_D(b)$ is computed as follows:

$$\Delta_D(b) = \frac{\sum_{i=1}^{N} D(C_i, b, K_s) W(C_i)}{\sum_{i=1}^{N} D(C_i, b, K_s)} - \frac{\sum_{i=1}^{N} (1 - D(C_i, b, K_s)) W(C_i)}{\sum_{i=1}^{N} (1 - D(C_i, b, K_s))} \tag{1}$$

If the bits calculated during the cryptographic algorithm are statistically uniformly distributed and if the number of ciphering traces is sufficient, $\Delta_D(b)$ tends to 0 for wrong key hypothesis and $\Delta_D(b) \neq 0$ for the correct key K_s hypothesis at the instant τ where the bit b is handled, this is the DPA peak. However, in practice, the bit distribution is correlated to S-box output and so some peaks can be observed on wrong key differential traces. This is the ghost peak problem explained for example in [5, 15]. For the correct key, peaks can also appear at instants other than τ due to the correlation between transient results during the cryptographic computation.

Note that there exist three main aspects to be considered for applying a power analysis method. The first one is how to choose ***target bits*** and ***cipher messages***. For example, the bit b in DPA method is well chosen if the highest peak belongs to the differential trace of the correct hypothesis, which is not always true for any choice of b. The cipher messages can be random or chosen.

By using chosen messages, attackers can reduce the algorithmic noise and also simplify the Hamming distance to the Hamming weight for hardware implementation [14, 15]. However a chosen message attack implies that the bits inside the algorithm are not independently distributed. So unexpected peaks related to the bits other than b can be observed.

The second aspect is how to determine different ***classes***. In mono-bit DPA method, Kocher has proposed two classes:

$$G_0 = \{W(C_i), i = 1 \ldots N | D(C_i, b, K_s) = 0\}$$
$$G_1 = \{W(C_i), i = 1 \ldots N | D(C_i, b, K_s) = 1\}$$

These classes are computed with the Hamming weight, but can be extended with the Hamming distance considering a previous state for b.

The third aspect is related to the function that calculates differential traces in order to evaluate and detect efficiently the correct hypothesis. These traces can be called as the ***decision signals***. In the mono-bit case, this decision signal is $\Delta_D(b)$. Different kinds of classes and decision signals will be discussed in further sections.

2.2 Partitioning Power Analysis

Multi-bit DPA: To enhance the original DPA, some authors have introduced d-bit DPA attacks which means that d bits are used instead of only one bit. The method proposed by Messerges et al [14] is still based on the idea of dividing power consumption signals into two classes. For a d-bit set $\mathcal{B} = b_1 b_2 \ldots b_d$, two classes of their multi-bit DPA are defined as follows:

$$G_0 = \left\{ W(C_i), i = 1 \ldots N | H(C_i, \mathcal{B}, K_s) < \frac{d}{2} \right\}$$

$$G_1 = \left\{ W(C_i), i = 1 \ldots N | H(C_i, \mathcal{B}, K_s) \geq \frac{d}{2} \right\}$$

where $H(C_i, \mathcal{B}, K_s)$ denotes the Hamming weight of $\mathcal{B}$ corresponding to K_s and C_i. Note that if we consider a previous state R of $\mathcal{B}$ as the reference state, $H(C_i, \mathcal{B}, K_s)$ can be used as the Hamming distance between R and the actual state of $\mathcal{B}$. [1]

The decision signal becomes:

$$\Delta_H(\mathcal{B}) = \frac{\sum_{G_1} W(C_i)}{N_1} - \frac{\sum_{G_0} W(C_i)}{N_0} \tag{2}$$

with $N_0 = card(G_0)$ and $N_1 = card(G_1)$.

[1] In the research of Brier et al.[5], the Hamming distance is used and defined as the number of flipping bits to switch from a reference state R to another state D, and is given by $H(R \oplus D)$.When the reference state R is 0, the Hamming distance $H(R \oplus D)$ becomes the Hamming weight of D.

In a 4-bit DPA case, Bevan et al. [4] suggested combining the $\varDelta_D(b_i)$ computed independently for each bit b_i $(i = 1 \ldots 4)$ of $\mathcal{B}$:

$$\varSigma_D(\mathcal{B}) = \varDelta_D(b_1) + \varDelta_D(b_2) + \varDelta_D(b_3) + \varDelta_D(b_4) \tag{3}$$

The notion of class in this case is the same as the one of mono-bit DPA but it is defined for each bit b_i $(i = 1 \ldots 4)$ of $\mathcal{B}$. The decision signal $\varSigma_D(\mathcal{B})$ is the sum of four other decision signals $\varDelta_D(b_i)$ $(i = 1 \ldots 4)$. This method is efficient only if the values of the four bits influence the power consumption at the same time and in the same way. [2]

Partitioning Power Analysis: In order to generalize the multi-bit DPA methods, we propose here the Partitioning Power Analysis (PPA) method based on the Hamming distance. The multi-partition method has been suggested by Akkar et al.[3] with DiPA, but these authors did not formalize the concept.

We consider d-bit set $\mathcal{B} = b_1 b_2 \ldots b_d$ and divide N power consumption signals $W(C_i)$ $(i = 1 \ldots N)$ into $(d+1)$ partitions (classes) $G_0, G_1, \ldots, G_d$.

$$G_j = \{W(C_i), i = 1 \ldots N | H(C_i, \mathcal{B}, K_s) = j\}$$

where $H(C_i, \mathcal{B}, K_s)$ denotes the Hamming distance between a previous state and the actual state of $\mathcal{B}$, corresponding to the message C_i and the key guess K_s. We note $N_j = card(G_j)$, so $\sum_{j=0}^{d} N_j = N$. The decision signal of PPA is given as follows, where a_j $(j = 1 \ldots N)$ are chosen weights.

$$\varSigma_H(\mathcal{B}) = \sum_{j=0}^{d} a_j \frac{\sum_{G_j} W(C_i)}{N_j} \tag{4}$$

The choice of these weights can be determined with a known key algorithm or with a selection function based on known bits, for example input message bits.

Note: By the previous definition of PPA, the multi-bit DPA concepts proposed by Messerges and Bevan are two cases of PPA with special coefficients a_j. For the Messerges' method, $\varDelta_H(\mathcal{B})$ derived from (2) can be formulated as (4), H being the Hamming distance and $a_j = -1$ for $0 \le j < d/2$ and $a_j = 1$ for $d/2 \le j \le d$. Referring to Bevan's concept, in order to use the Hamming distance notion, we can choose the reference state of $\mathcal{B}$ as '0000'. After some algebraical manipulation, the $\varSigma_D(\mathcal{B})$ of (3) can be rewritten under a form of (4) as follows:

$$\varSigma_D(\mathcal{B}) = -\frac{1}{8}\sum_{G_0} W(C_i) - \frac{1}{4}\sum_{G_1} W(C_i) + \frac{1}{4}\sum_{G_3} W(C_i) + \frac{1}{8}\sum_{G_4} W(C_i) \tag{5}$$

By the same way, if we consider the reference state of the target bit b is '0', the original DPA proposed by Kocher becomes the simplest PPA with a coefficient -1 for the group G_0 and a coefficient 1 for the group G_1.

[2] This point may be true for a hardware algorithm, but false for a software one.

2.3 Correlation Power Analysis

Correlation approaches are based on the dependence between the power consumption of the circuit and the Hamming weight [8, 12] or the Hamming distance [5] of manipulated data. According to Brier's model, the relationship between the power consumption W and $H(R \oplus D)$ is linear ($W = aH + b$, a and b are constant). The correct key is the one which maximizes the correlation factor ρ_{WH}.

If we denote $H_{i,R} = H(R \oplus C_i)$ the Hamming distance between the actual state of the message C_i and the reference state R, the decision signal of the CPA method is the correlation factor $\hat{\rho}_{WH}$ [5]:

$$\hat{\rho}_{WH}(R) = \frac{N \sum W(C_i) H_{i,R} - \sum W(C_i) \sum H_{i,R}}{\sqrt{N \sum W(C_i)^2 - (\sum W(C_i))^2} \sqrt{N \sum {H_{i,R}}^2 - (\sum H_{i,R})^2}} \tag{6}$$

According to this concept, the notion of class is not explicitly used, i.e., N power consumption signals $W(C_i)$ corresponding to N cipher messages C_i ($i = 1 \ldots N$) are not classified in to different classes. However, this notion can be introduced here by grouping the power consumption signals $W(C_i)$ where C_i has the same Hamming distance with a reference state R. Considering a d-bit set $\mathcal{B}$ of messages C_i and using the same notation described in the previous section, we divide N power consumption signals $W(C_i)$ ($i = 1 \ldots N$) into $(d+1)$ classes $G_0, G_1, \ldots, G_d$ with

$$G_j = \{W(C_i), i \in 1 \ldots N | H(C_i, \mathcal{B}, K_s) = j\}$$

We develop now the term $A = N \sum W(C_i) H_{i,R} - \sum W(C_i) \sum H_{i,R}$ by splitting N power consumption signals $W(C_i)$ into $(d+1)$ partitions. The term A becomes:

$$\begin{aligned}
A &= N \sum_{j=0}^{d} \sum_{G_j} W(C_i).j - (\sum_{j=0}^{d} \sum_{G_j} W(C_i))(\sum_{k=0}^{d} \sum_{G_k} k) \\
&= \sum_{j=0}^{d} N.j \sum_{G_j} W(C_i) - (\sum_{j=0}^{d} \sum_{G_j} W(C_i))(\sum_{k=0}^{d} N_k.k) \\
&= \sum_{j=0}^{d} \left(N.j - \sum_{k=0}^{d} N_k.k \right) \sum_{G_j} W(C_i)
\end{aligned}$$

By denoting $\alpha_j = \frac{N_j}{N} \left(j - \sum_{k=0}^{d} \frac{N_k}{N}.k \right)$, the term A becomes:

$$A = N^2 \sum_{j=0}^{d} \left(\alpha_j \frac{\sum_{G_j} W(C_i)}{N_j} \right) \tag{7}$$

Accordingly, from (6) and (7) the correlation between power consumption W and Hamming distance H is then rewritten as:

$$\hat{\rho}_{WH}(R) = \frac{\sum_{j=0}^{d} \left(\alpha_j \frac{\sum_{G_j} W(C_i)}{N_j} \right)}{\sigma_W \sigma_H} \tag{8}$$

Equation (8) shows that the differences between CPA and PPA are the coefficients α_j and the normalization factor $\sigma_W \sigma_H$. Note that while the coefficients α_j of CPA (see (8)) depend on the distribution of N_j, the a_j of PPA (see (4)) are flexibly chosen. If N is large and the bits of $\mathcal{B}$ are uniformly distributed, the coefficients α_j of CPA tend to constant values and can be calculated in function of d and j by the following formula:

$$\alpha_j = \frac{C_d^j}{2^d}(j - \sum_{k=0}^{d} \frac{C_d^k}{2^d}.k)$$

where $C_d^j = \frac{d!}{j!(d-j)!}$ is the number of combinations of d elements taken j at a time.

Some values of α_j when $d = 1 \ldots 4$ are given in the Table 1.

d	α_0	α_1	α_2	α_3	α_4
1	-1/4	1/4	-	-	-
2	-1/4	0	1/4	-	-
3	-3/16	-3/16	3/16	-3/16	-
4	-1/8	-1/4	0	1/4	1/8

Table 1. Coefficients α_j for an uniform distribution of $\mathcal{B}$

We can notice that the coefficients α_j $(i = 1 \ldots 4)$ for $d = 4$ are identical to those of Bevan's method. This interesting remark shows the relation between the multi-bit DPA method of Bevan, a special case of PPA, and the correlation concept of Brier. The difference between these methods is the normalization by $\sigma_W \sigma_H$. This point is studied in the next section.

3 CPA and normalization effect

In this section, we discuss the normalization effects by $\sigma_W \sigma_H$ of CPA signals. For $d = 4$, we examine only 4 bits instead of all bits of messages C_i. If the bits of $\mathcal{B}$ are uniformly distributed and N is large enough, according to the Table 1, the correlation factor $\hat{\rho}_{WH}(R)$ given by the formula (8) becomes:

$$\hat{\rho}_{WH}(R) = \frac{-\frac{1}{8}\frac{\sum_{G_0} W(C_i)}{N_0} - \frac{1}{4}\frac{\sum_{G_1} W(C_i)}{N_1} + \frac{1}{4}\frac{\sum_{G_3} W(C_i)}{N_3} + \frac{1}{8}\frac{\sum_{G_4} W(C_i)}{N_4}}{\sigma_W \sigma_H} \quad (9)$$

Note that the numerator is equal to $\Sigma_H(\mathcal{B})$ given in (4) with $d = 4$, $a_0 = -\frac{1}{8}, a_1 = -\frac{1}{4}, a_2 = 0, a_3 = \frac{1}{4}$ and $a_4 = \frac{1}{8}$. With such choice of PPA weights $a_j = \alpha_j$, we can observe the effect of the normalization factor $\sigma_W \sigma_H$, which is the only difference between PPA and CPA in this case. Furthermore, if the messages are random, the number of messages N is large and if d bits are uniformly distributed, σ_H is independent to key hypothesis and equal to $\frac{d}{4}$. The normalization effect finally depends only on σ_W.

In order to have a better knowledge of σ_W, we use N power consumption signals to compute the standard deviation $\sigma_W(t)$ at every instant t. Because data are handled at clock edges, $\sigma_W(t)$ is larger at theses points of time than at other instants. Hence, $\Sigma_H(\mathcal{B})$ is divided by significant values at clock edges and by smaller values at other moments. Consequently, the noise level of the correlation factor $\hat{\rho}_{WH}(R)$ rises. It can be very high if $\sigma_W(t)$ tends toward zero.

A common numerical method [23] to reduce this normalization effect consists in adding to $\sigma_W(t)$ a positive constant ε. If the ε is correctly chosen, the noise should be reduced without modifying any principal result. We now obtain for the correlation factor:

$$\hat{\rho}_{WH}(R) = \frac{\Sigma_H(\mathcal{B})}{(\sigma_W + \varepsilon)\sigma_H} \quad (10)$$

In our case, the choice of ε is delicate. If ε tends to zero, the CPA signals are always normalized by small values at the non-clock-edge moments. Thus, the noise level of CPA signal is still high. On the other hand, if ε is great in comparison with $\sigma_W(t)$ at clock edges, the correlation between H and W is not respected any more. The object of the following section is to explain the choice of ε that allows an improvement of the CPA detection capacity.

4 Experimental Confrontation

Experimental results from real measured signals shown in this section allow to compare the three techniques DPA, PPA, CPA and to valid our CPA enhancement proposition. Here we compare the PPA and CPA by observing 4 examined bits. The coefficients a_j of PPA and α_j of CPA are identically chosen for $j = 1 \ldots 4$. This choice of coefficients helps us to see the normalization effect. The results of original mono-bit DPA (i.e., $d = 1$) are also shown as a reference for comparison.

Signal acquisition: In our experiment, the electromagnetic radiation of a synthesized ASIC during a DES operation was measured. Up to 10000 messages randomly generated were used. The upper curve of Fig. 1 represents an experimental electromagnetic signal where the 16 peaks corresponding to 16 rounds of the DES can be observed. As the electromagnetic signal is used instead of the power consumption ones, the notation $W(C_i)$ represents here the voltage at the output of our electromagnetic sensor for the processing of the message C_i.

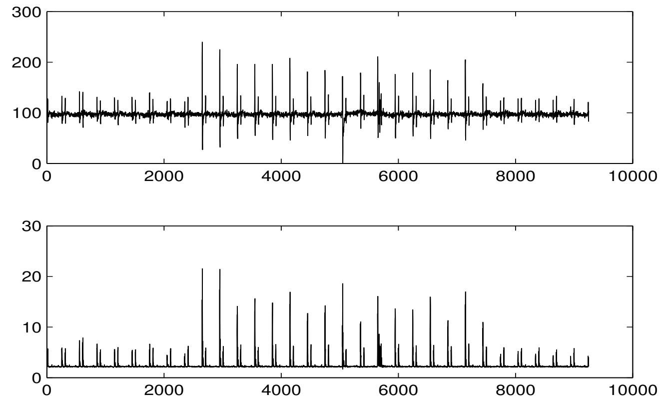

Fig. 1. The horizontal axes represent the time sampling proportional to clock cycles. The upper vertical axis represents the potential difference on the output of an electromagnetic sensor (mV) and the lower one represents its standard deviation.

Variation of $\sigma_{\mathbf{W}}(\mathbf{t})$: As mentioned in the previous section, we compute the standard deviation $\sigma_W(t)$ at each instant t to observe its variation. This one is depicted in the lower curve of Fig. 1. The figure validates our analytical results that the σ_W increases rapidly at each clock edge.

Signal observation: In the first experiment, we used 2000 cipher messages to test 64 key hypothesis with DPA, PPA, CPA and enhanced CPA methods. In Fig. 2, we present the DPA, PPA, CPA and enhanced CPA signals for the correct key (left column) and for a wrong key (right column) resulting in the highest ghost peak. From these figures, we realize that the unexpected peaks for the correct key and for the wrong key appear clearly in the DPA signals. We also see that the PPA method performs better than DPA in terms of the appearance of these unexpected peaks. This result shows the advantage of multi-bit concept compared to the mono-bit one. For the CPA method, the expected peak is clear and the signals coincide with our analysis in Sect. 3: the level of noise in the CPA signal is higher. We can also note that ghost peaks in CPA (see Fig. 2 for the wrong key) are overwhelmed in this described noise.

Evaluation and validation of the proposed method: In order to evaluate the success of an attack, we define two attack-efficient indexes which reflect the possibility of key detection. The ***first index***, i_1, is defined as the ratio between the DPA/PPA/CPA peak (expected peak) corresponding to the correct hypothesis at the moment τ where the bits are manipulated and the highest

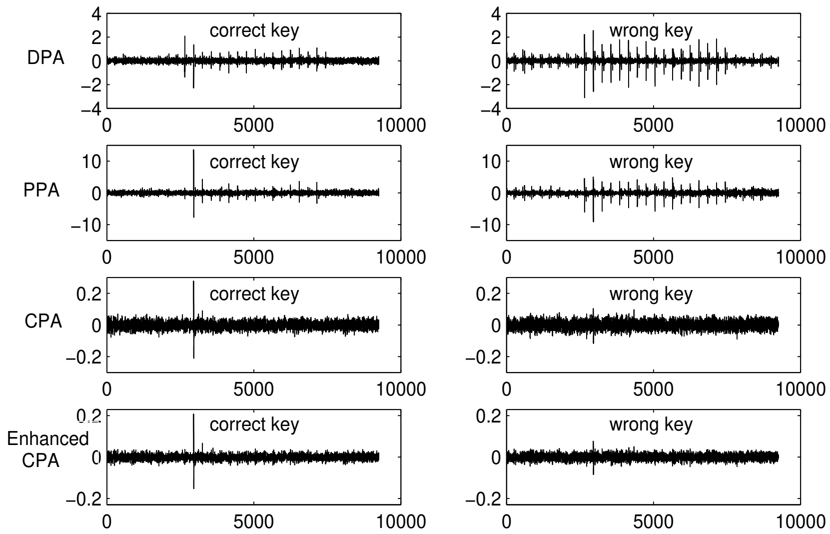

Fig. 2. Power analysis signals with 2000 messages. 1st line: DPA method, 2nd line: PPA method, 3rd line: CPA method and 4th line: Enhanced CPA method. Left column: correct key guess, Right column: wrong key guess resulting in the highest ghost peak. Horizontal axes: time sampling proportional to clock cycle, 1st line vertical axis: $\Delta_D(b)$, 2nd line vertical axis: $\Sigma_H(\mathcal{B})$, 3rd and 4th line vertical axes: $\hat{\rho}_{WH}(R)$.

DPA/PPA/CPA peak among incorrect hypothesis at this instant. If this index is greater than 1, the expected peak is the highest one and the key detection is reliable. On the contrary, if this index is smaller than 1, there exists another peak higher than the expected peak, i.e the key detection is impossible.

The values of i_1 when the number of cipher messages varying from 100 to 10000 messages is illustrated in Fig. 3 and enlarged in Fig. 4. The attack-efficient index i_1 of DPA is represented by the dotted curve, that of PPA is the dashed curve and that of CPA is the dashdot curve. The solid curve corresponds to our proposed method to enhance the CPA. Figure 3 shows that the values i_1 of CPA are always greater than those of DPA/PPA. The better performance of CPA against DPA can easily be explained by the fact that the DPA method is based on the weighting with a single bit b and the CPA method is based on a weighting with 4 examined bits of the cipher messages. The result of CPA against PPA confirms the efficiency of the normalization factor of CPA. When comparing DPA and PPA, we observe that the index i_1 of PPA is always higher than DPA's index. Hence, the multi-bit attack PPA (4 bits in our case) performs better than the mono-bit attack DPA. This conclusion is also confirmed by Fig. 2 in which we observe that the PPA peak is much higher and clearer than the DPA one.

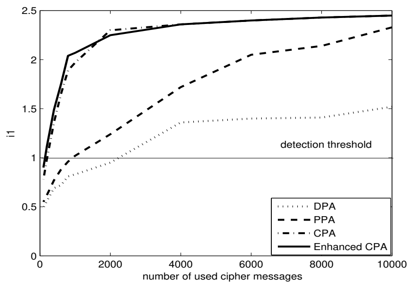

Fig. 3. First attack-efficient index

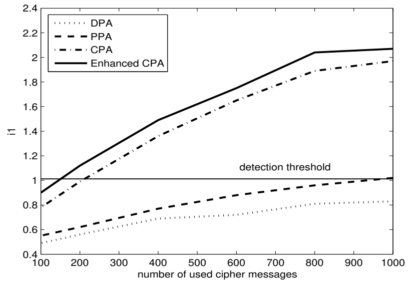

Fig. 4. A zoom of Fig. 3

The ***second index***, i_2, is the signal to noise ratio of the DPA/PPA/ CPA signal corresponding to the correct key. The DPA/PPA/CPA peak is considered as *signal* and the rest as *noise.* If i_2 is not large enough, the expected peak corresponding to the correct key does not appear and we can not confirm which key is correct. The limit is chosen equal to 3 through our experiment results. Figure 5 illustrates the variation of the second attack-efficient index i_2 as function of the number of curves C_i. We observe that the values i_2 of CPA are much lower than those of DPA/PPA. Accordingly, the noise in the CPA signal for correct key is more significant. By using our enhanced CPA method, we reduce this noise.

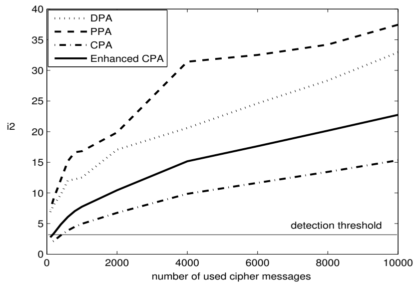

Fig. 5. Second attack-efficient index

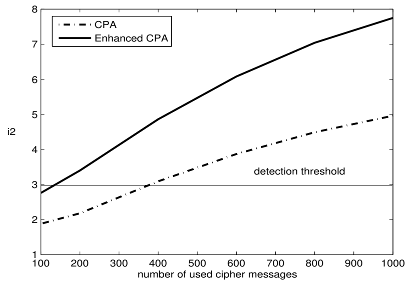

Fig. 6. A zoom of Fig. 5

We choose $\varepsilon = 2$ which is about 10 % of $\sigma_W(\tau)$. This value is quite small compared to $\sigma_W(\tau)$ so that its influence on i_1 index, which is computed at instant τ, is negligible. This explains why the CPA and enhanced CPA curves in Fig. 3 are very close. On the other hand, the value $\varepsilon = 2$ is large enough to reduce the noise level observed in Fig. 2 (the rate of noise in the enhanced CPA signals is weaker than in the CPA signals) and Fig. 5 (the enhanced CPA curve is above the CPA curve) .

Number of cipher messages required for key detection: The key detection depends on both i_1 and i_2 indexes. The key detection is only feasible and reliable if the two following conditions are satisfied: $i_1 > 1$ and $i_2 > 3$. The first condition is trivial. The second condition is chosen through our experiment results.

Hence, if we take into account both indexes i_1 and i_2, according to Fig. 3 and Fig. 5, the DPA method needs about 2500 messages, the PPA needs about 1000 messages and the CPA needs about 400 messages to detect the correct key, i.e. both indexes are above the detection threshold. By using our proposed enhanced CPA method, only 200 messages are required to retrieve the coding key. Figure 7 confirms again our conclusion: with only 200 messages, **our enhanced CPA can detect the key but the original CPA can not.** The use of ε, that restricts the standard deviation used in CPA, allows us to considerably reduce the noise level (see Fig. 2), [3] and to retrieve the key with a lower number of curves (see Fig. 7). This restricted normalization can also be applied to PPA and DPA.

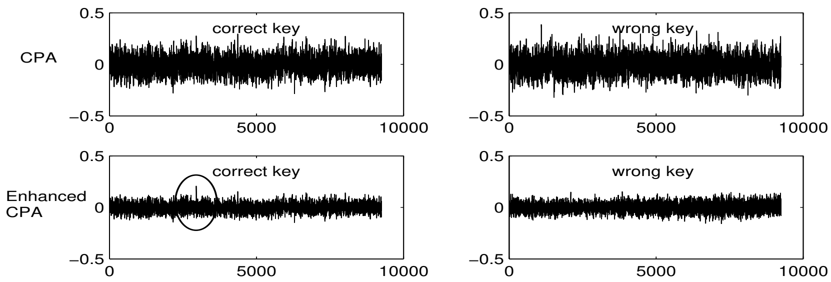

Fig. 7. Power analysis signals with 200 used messages, 1st line: CPA method, 2nd line: enhanced CPA method. Left column: correct key guess, Right column: wrong key guess resulting in the highest ghost peak. Horizontal axes: time sampling proportional to clock cycle, Vertical axes: $\hat{\rho}_{WH}$.

Let's also note that interesting clock cycles can be firstly investigated without normalization and then the restricted factor can be used to fully perform the differential analysis around the selected areas to find the correct keys.

5 Conclusions

First, we merged all existing multi-bit DPA methods into the PPA concept which consists of dividing power consumption signals into partitions. PPA could also

[3] Note that the ghost peaks in CPA are hidden in the noise and are better revealed with the enhanced method proposed here.

be merged into existing cryptanalysis techniques such as partitioning attacks (see for example [18, 19]).

We demonstrate that CPA is, in fact, a special form of PPA normalized by the standard deviation of power consumption signals. This normalization is efficient because it allows us to reduce significantly the number of messages required to break the cryptographic secrets. However, the normalization also increases the noise level of the CPA signal. This noise level can be reduced by using the proposed method with the restriction ε. Through the experiments, our enhanced CPA performs better than original CPA, DPA and four-bit PPA in terms of number of messages required for key detection. In future work, we would like to find the coefficients a_j that optimize the PPA efficiency. From this optimized PPA, we would expect to be able to propose a new power consumption model, taking into account for example the different bit contributions as suggested in [3, 24, 25].

References

1. M.L. Akkar, C. Giraud: An Implementation of DES and AES Secure Against Some Attacks. *In proceedings of CHES 2001*, LNCS 2162, pp. 309-318, Springer-Verlag, 2001.
2. M.L. Akkar, L. Goubin: A Generic Protection Against High-Order Differential Power Analysis. *In proceedings of FSE 2003*, LNCS 2887, pp. 192 - 205, Springer-Verlag, 2003.
3. M.L. Akkar, R. Bevan, P. Dischamp, D. Moyart: Power Analysis, What Is Now Possible... *In proceedings of ASIACRYPT 2000*, LNCS 1976, pp. 489 - 502, Springer-Verlag, 2000.
4. R. Bevan, E. Knudsen: Ways to Enhance DPA. *In proceedings of ICISC 2002*, LNCS 2587, pp.327-342, Springer-Verlag, 2003.
5. E. Brier, C. Clavier, F. Olivier: Correlation Power Analysis with a Leakage Model, *In proceedings of CHES 2004*, LNCS 3156, pp. 16-29, Springer-Verlag, 2004.
6. S. Chari, C.S. Jutla, J.R. Rao, P. Rohatgi: Towards Sound Approaches to Counteract Power Analysis Attacks. *In proceedings of CRYPTO 1999*, LNCS 1666, pp. 348-412, Springer-Verlag, 1999.
7. J.S. Coron, L. Goubin: On Boolean and Arithmetic Masking Against Differential Power Analysis. *In proceedings of CHES 2000*, LNCS 1965, pp. 231-237, Springer-Verlag, 2000.
8. J.S. Coron, P. Kocher, D. Naccache: Statistics and Secret Leakage. *In proceedings of Financial Cryptography*, LNCS 1972, pp. 157-173, Springer-Verlag, 2000.
9. L. Goubin, J. Patarin: DES and Differential Power Analysis: The Duplication Method. *In proceedings of CHES 1999*, LNCS 1717, pp. 158-172, Springer-Verlag, 1999.
10. P. Kocher, J. Jaffe, B. Jun: Introduction to Differential Power Analysis and related attacks. `http://www.cryptography.com`.
11. P. Kocher, J. Jaffe, B. Jun: Differential Power Analysis. *In proceedings of CRYPTO 1999*, LNCS 1666, pp. 388-397, Springer-Verlag, 1999.
12. R. Mayer-Sommer: Smartly Analysing the Simplicity and the Power of Simple Power Analysis on Smartcards. *In proceedings of CHES 2000*, LNCS 1965, pp. 78-92, Springer-Verlag, 2000.

13. T. S. Messerges, E. A. Dabbish, R. H. Sloan: Investigations of Power Analysis Attacks on Smartcards. *In proceedings of the USENIX Workshop on Smart Card Technology 1999*, `http://www.usenix.org/`, 1999.
14. T. S. Messerges, E. A. Dabbish, R. H. Sloan: Examining Smart-Card Security under the Threat of Power Analysis Attacks. *IEEE Transactions on Computers*, Vol. 51, N5, pp. 541-552, May 2002.
15. C. Canovas, J. Clédière: What do S-boxes Say in Differential Side Channel Attacks? *Cryptology ePrint Archive*, http://eprint.iacr.org/, Report 20085/311, 2005.
16. S. Guilley, P. Hoogvorst, R. Pacalet: Differential Power Analysis Model and some Results *In proceedings of CARDIS 2004,* Kluwer Academic Publishers, pp. 127-142, 2004.
17. K. Gandolfi, C.Mourtel, F.Olivier: Electromagnetic Attacks: Concrete Results. *In proceedings of CHES 2001,* LNCS 2162, pp. 252-261, Springer, 2001.
18. Carlo Harpes: Partitioning Cryptanalysis. Post-Diploma Thesis, Signal and Information Processing Lab., Swiss Federal Institute of Technology, Zurich, March 1995. http://www.isi.ee.ethz.ch/ harpes/pc.ps.
19. Thomas Jakobsen: Correlation Attacks on Block Ciphers, Master's Thesis, Dept. of Mathematics, Technical University of Denmark, January 1996.
20. J.J. Quisquater, D. Samyde: Electromagnetic Analysis (EMA): Measures and Countermeasures for Smart Cards. *In proceedings of e-Smart 2001,* LNCS 2140, pp. 200-201, Springer, 2001.
21. J.R. Rao, P. Rohatgi: EMpowering Side-Channel Attacks. *Cryptology ePrint Archive,* http://eprint.iacr.org/, Report 2001/037, 2001.
22. R. Bevan: Estimation statistique et sécurité des cartes à puces, évaluation d'attaques DPA évolués. OCS, rapport de thèse, 2004.
23. W.H. Press, S.A. Teukolsky, W.T. Vetterling, and B.P. Flannery: Numerical Recipes in C++. *Cambridge University Press,* Second Edition, 1002pp, New York, 2002.
24. J. R. Rao, P. Rohatgi, H. Scherzer, S. Tinguely : Partitioning Attacks : Or How to Rapidly Clone Some GSM Cards. *In proceedings of the 2002 IEEE Symposium on Security and Privacy,* pp. 31-41, IEEE Computer Society, 2002.
25. F.-X. Standaert, F. Mace, E. Peeters, J.-J. Quisquater: Updates on the Security of FPGAs Against Power Analysis Attacks. *In proceedings of ARC 2006*, LNCS 3985, pp. 335-346, Springer-Verlag, 2006.