Adaptive Chosen-Message Side-Channel Attacks

Nicolas Veyrat-Charvillon*, François-Xavier Standaert**,

Université catholique de Louvain, Crypto Group, Belgium.
e-mails: nicolas.veyrat;fstandae@uclouvain.be

Abstract. Most side-channel attacks that have been published in the open literature assume known- or chosen-message adversarial scenarios. In this paper, we analyze the increase of the attacks' efficiencies that can be obtained by adaptively selecting the messages. For this purpose, we first describe a generic strategy that allows an adversary to take advantage of this capability. We show that it can be applied to any differential power or electromagnetic analysis attack, against unprotected or protected devices and exploiting profiled or non-profiled leakage models. Then, we provide various experiments to quantify these improvements. Finally, we discuss the optimality of our strategy and its implications for the security evaluation of leakage-resilient cryptographic hardware.

1 Introduction

In classical cryptanalysis, the adaptive selection of the inputs to a cryptographic primitive is known to be a powerful ability for the adversaries. For example, blockwise-adaptive chosen-message attacks have been used to show the insecurity of different encryption modes in [9]. Similarly, Bleichenbacher has demonstrated in [3] that chosen-ciphertext attacks can be used to attack the RSA Encryption Standard PKCS #1. And in the symmetric setting, boomerang attacks are an example of how the adaptivity can be exploited to reduce the complexity of certain categories of attacks [24]. Quite surprisingly, and although it is frequently suggested as a possible improvement, very few related works have been performed in the context of side-channel attacks. In [20], Schindler presented a timing attack against RSA with the Chinese remainder theorem that requires some form of adaptivity. And more recently, Köpf and Basin provided a careful model and analysis of such an attack scenario. But the investigations in [12] are carried out in a restricted context of noiseless leakage. This typically applies to timing attacks such as [10], but is of limited interested in the case of power or electromagnetic side-channel attacks, in which noise is a typical issue adversaries have to deal with [1, 11, 19]. To the best of the authors's knowledge, the generalization of this previous work, from the context of deterministic leakages to the one of probabilistic (or noisy) leakages was left as an open question.

* Work supported by the Walloon region through the project SCEPTIC.
** Associate researcher of the Belgian Fund for Scientific Research (F.R.S.-FNRS).

In this paper, we consequently tackle this problem and propose a careful investigation of adaptive chosen-message side-channel attacks. We describe a generic strategy that can be applied to improve the efficiency of any distinguisher. As an illustration, we detail its impact for correlation and template attacks [4, 5], both from simulations and actual experiments. Our evaluations show significant increases of the side-channel key-recovery success rates. We additionally evaluate the application of adaptive strategies against implementations protected with masking [8] and observe very similar improvements. Eventually, we discuss the optimality of our approach and compare it with the one in [12].

These results imply direct consequences for the good security evaluations of leaking cryptographic devices. They show that when applicable, adaptive strategies require to take larger security margins than for random-message attacks. In other words, the attacks described in this paper indicate how to best exploit the physical information leakage, in the standard DPA setting formalized in [15]. Hence, they can be used to determine the worst-case number of measurements required to performed a successful key-recovery in this context. We note that considering key-recovery attacks in security evaluations can appear as too weak from a theoretical point of view. But as demonstrated in [2], there is a strong relationship between distinguishing attacks and key-recovery attacks in the context of block ciphers. Hence, this situation is not very different than the one in classical (*e.g.* linear, differential) cryptanalysis, in which one considers the best available attacks in order to approximate the security of a cipher. And the adaptive strategies presented in this paper are part of these evaluation tools.

Note also that our results have interesting connections with recent works in the area of leakage resilient cryptography. Indeed, one of the assumptions in, *e.g.* [6, 16], is that the leakage function can be adaptively selected by the adversary. But as discussed in [23], this is a quite strong requirement that is rarely observed: it would require that the adversary can change his measurement setup in a constructive manner. In practice, most attacks rather rely on a fixed leakage function and measure this function for different plaintexts. It is the combination of several plaintexts that allows increasing the information leakage in such a way that the key is eventually revealed. Hence, our adaptive selection of the messages provides a more realistic counterpart of the adversarial capabilities.

2 Terminology & notations

In a side-channel attack, an adversary tries to recover some secret information from a leaking implementation, *e.g.* a software program or an IC computing a cryptographic algorithm. In this paper, we focus on the divide-and-conquer strategies that are most frequently considered in the literature [14] and are formalized as "standard DPA attacks" in [15]. In the context of a block cipher implementation (that will be our running example), one typically targets small pieces of the master key - called subkeys in the following - one by one. The attacks then follow the different steps illustrated in Figure 1.

Namely, we consider a device performing several cryptographic computations $\mathsf{E}_k(x_i)$ on different plaintexts x_i drawn from the text space $\mathcal{X}$, using some fixed key k drawn from the key space $\mathcal{K}$. While computing $\mathsf{E}_k(x_i)$, the device handles some intermediate values that depend on the known input x_i and the unknown key k (defined as sensitive variables in [18]). In practice, the interesting sensitive variables in a DPA attack are the ones that depend on an enumerable subkey s: we denote them as v_i^s, for a plaintext x_i. Any time such a sensitive intermediate value is computed, the device generates some physical leakage, denoted as l_i^k (where the k superscript indicates that the leakage potentially depend on all the key k, including the subkey s). Hence, in order to perform a key-recovery, an adversary first has to select a sensitive value. Given that this variable only depends on a subkey s, he can then predict its result for the plaintexts x_i that generated l_i^k and enumerate every possible subkey candidate $s^* \in \mathcal{S}$. This leads to different hypothetical intermediate variables $v_i^{s^*}$. Afterwards, the adversary exploits a leakage model to map these values from their original space $\mathcal{V}$ towards a modeled leakage space $\mathcal{M}$. As a result, he obtains $|\mathcal{S}|$ different models, denoted

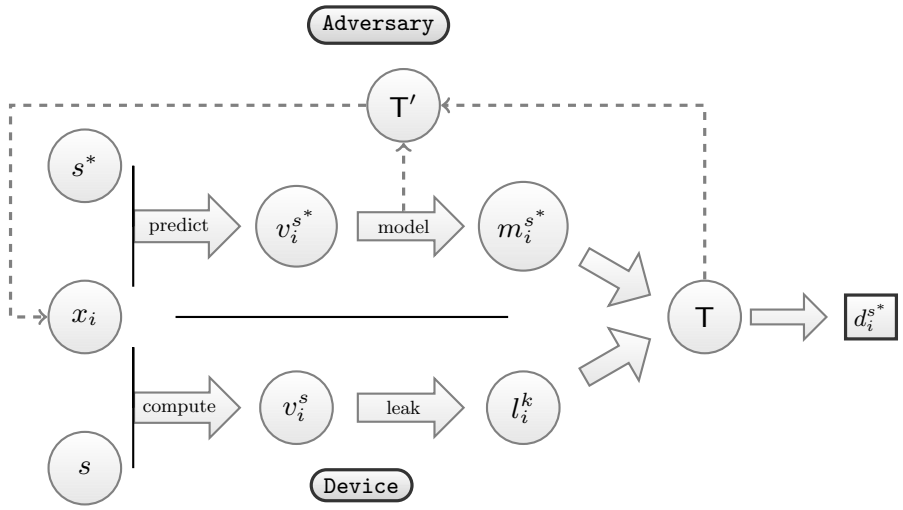


Fig. 1. Schematic description of a side-channel key-recovery attack.

as $m_i^{s^*}$, again corresponding to the different subkey candidates. Eventually, he uses a statistical test T to compare the different models $m_i^{s^*}$ with the actual leakages l_i^k. If the attack is successful, the highest value for this test should occur for the correct subkey candidate $s^* = s$. This procedure can be repeated for different subkeys in order to recover the complete key k.

In view of this description, there are several important parameters that determine the efficiency of a DPA. First, the choice of an intermediate computation and leakage model have a significant impact. For example, it is well known that predicting the first round S-boxes' outputs in a block cipher leads to a better

discrimination of the subkeys than predicting their inputs [17]. As for the leakage models, it mainly relates to the a-priori knowledge of the adversary about the device he targets. One generally distinguishes profiled and non-profiled attacks. In the first ones (*e.g.* template attacks [5]), the adversary can characterize the leakage probability density functions (pdf for short) prior to the online attack. In the second ones, he exploits simpler models (*e.g.* predicting only certain moments of the leakage pdf, as in correlation attacks [4]) or performs the profiling "on-the-fly" [7]. Second, and closely related, the choice of a statistical test is usually determined by the type of models available to the adversary.

Another parameter that is less frequently considered (and evaluated) in the literature is the selection of the plaintexts. That is, in most experimental settings, one generally considers attacks with random input messages. But as illustrated in Figure 1, a more powerful scenario is to adaptively select the plaintexts, in function of the prior knowledge about the secret subkey and an hypothetical leakage model. In this paper, we consequently investigate the statistical tests T' that can be used in order to best exploit the available leakage.

3 Adaptive template attacks

In this section, we present the principles of our adaptive chosen-message strategy. We first describe it in the (profiled) context of template attacks. Then, we discuss how to generalize our solution to non-profiled distinguishers.

3.1 Template attacks

Template attacks, first published in [5], are usually considered as the most powerful type of side-channel attacks, in an information theoretic sense. They work in two main steps. In a first profiling phase, the adversary builds key-dependent templates, *i.e.* he estimates the leakage pdf for different internal configurations of his target device. Then, in a second (online attack) phase, he uses these templates to perform a maximum-likelihood key-recovery. In this paper, we focus on the (most frequently considered) case of Gaussian templates.

Templates construction. Gaussian template attacks assign a Gaussian distribution to a number of different configurations of the target device. In their most generic form, they perform this assignment exhaustively. For example, if an adversary targets the 8 first bits of an AES master key, he will use one Gaussian for any pair (x_i, s^*), out of the 2^{16} possible ones. In practice, different tricks can be used to reduce this number of templates, in order to increase the efficiency of the profiling, *e.g.* by taking advantage of symmetry properties and stochastic models [21]. In this section, we describe the generic approach for simplicity. Suppose that the adversary is provided with N_p traces to estimate the pdf corresponding to a state (x_i, s). He will then assume that the leakage traces $\{l_i^{k,j}\}_{j=1}^{N_p}$ are drawn from the multivariate normal distribution:

$$\mathcal{N}(l_i^{k,j}|\boldsymbol{\mu}_{x_i}^s, \boldsymbol{\Sigma}_{x_i}^s) = \frac{1}{(2\pi)^{\frac{N}{2}}|\boldsymbol{\Sigma}_{x_i}^s|^{\frac{1}{2}}} \exp\left\{-\frac{1}{2}(l_i^{k,j} - \boldsymbol{\mu}_{x_i}^s)^\top (\boldsymbol{\Sigma}_{x_i}^s)^{-1}(l_i^{k,j} - \boldsymbol{\mu}_{x_i}^s)\right\},$$

where the mean $\boldsymbol{\mu}_{x_i}^s$ and the covariance matrix $\boldsymbol{\Sigma}_{x_i}^s$ specify completely the noise distribution associated to the leakage trace of each pair (x_i, s). The templates are built by estimating the sets of parameters $\boldsymbol{\mu}_{x_i}^s$ and $\boldsymbol{\Sigma}_{x_i}^s$ for $x_i \in \mathcal{X}$ and $s \in \mathcal{S}$. Maximum likelihood estimators can be used for this purpose: $\hat{\boldsymbol{\mu}}_{x_i}^s = \frac{1}{N_p}\sum_{j=1}^{N_p} l_i^{k,j}$, and $\hat{\boldsymbol{\Sigma}}_{x_i}^s = \frac{1}{N_p}\sum_{j=1}^{N_p}(l_i^{k,j} - \hat{\boldsymbol{\mu}}_{x_i,s})(l_i^{k,j} - \hat{\boldsymbol{\mu}}_{x_i,s})^\mathsf{T}$.

Online attack. Assume now that there are $|\mathcal{S}|$ possible subkeys. To determine which one is the most likely to have generated a new trace l_i^k, we compute:

$$\tilde{s} = \underset{s^*}{\operatorname{argmax}}\ \hat{\Pr}[s^*|l_i^k] = \underset{s^*}{\operatorname{argmax}}\ \hat{\Pr}[l_i^k|s^*, x_i] \cdot \hat{\Pr}^{(0)}[s^*],$$

where $\hat{\Pr}[l_i^k|s^*, x_i] = \mathcal{N}(l_i^k|\hat{\boldsymbol{\mu}}_{x_i}^{s^*}, \hat{\boldsymbol{\Sigma}}_{x_i}^{s^*})$ and $\hat{\Pr}^{(0)}[s^*]$ is the a priori probability of the subkey candidate s^*, that we assume to be uniform in the following (*i.e.* equal to $1/|\mathcal{S}|$, $\forall s^*$). In other words, the classification rule assigns l_i^k to the candidate s^* with the highest a posteriori probability. Since in practice, a single trace is usually not enough to recover the subkey with high confidence, the adversary finally combines several plaintexts and computes $\tilde{s} = \underset{k^*}{\operatorname{argmax}}\ \hat{\Pr}^{(q)}[s^*]$, with:

$$\hat{\Pr}^{(q)}[s^*] = \frac{\prod_{i=1}^q \Pr[s^*|l_i^k]}{\sum_{s' \in \mathcal{S}} \prod_{i=1}^q \Pr[s'|l_i^k]},$$

and q the number of traces used in the online attack. Note that in the following sections, we will denote as univariate (*resp.* multivariate) the attacks in which the the traces l_i^k contain one (*resp.* several) samples.

3.2 Adaptive selection of the plaintexts

Let us now assume that a template attack has been performed with i traces, corresponding to different plaintexts x_1 to x_i, and giving rise to a certain knowledge about the subkey candidates summarized as $\hat{\Pr}^{(i)}[s^*]$. The objective of this paper, illustrated in Figure 1, is to select the next plaintext x_{i+1} in such a way that it will best discriminate the correct subkey. Ideally, this plaintext could be obtained by computing the success rate of the adversary in step $i+1$ or, similarly, by computing the residual entropy of this correct subkey s (*i.e.* one of the metrics in [22]). But while running an attack, the adversary obviously does not know the value of this correct subkey yet. As a consequence, the only applicable strategy is to exploit a criteria that can be estimated "on-the-fly".

Following the previous section, it appears that a natural criteria is to look at the entropy of the subkey candidates rather than the one of the correct subkey. Indeed, in a successful attack, the entropy of these subkey candidates should eventually be null (*i.e.* we should determine only the correct subkey with probability one). For example, at step i, this entropy can be estimated as:

$$\hat{\mathrm{H}}^{(\mathbf{x}_i)}[S^*] = -\sum_{s^*} \hat{\Pr}^{(i)}[s^*] \cdot \log_2 \hat{\Pr}^{(i)}[s^*],$$

where $\mathbf{x}_i = [x_1, x_2, \ldots, x_i]$ is the vector of plaintexts used in the attack. Using this entropy as a criteria for our adaptive chosen-message attacks implies selecting the plaintext x_{i+1} as the one minimizing $\hat{\mathrm{H}}^{(\mathbf{x}_{i+1})}[S^*]$. This can be done as follows. First, let us observe that for every plaintext candidate x^*_{i+1} and subkey candidate s^*, one can define a random variable $\hat{L}^{s^*}_{x^*_{i+1}}$, corresponding to the simulated leakage trace that perfectly follows the leakage model obtained from the templates construction phase (*i.e.* a normal curve with mean vector $\hat{\boldsymbol{\mu}}^{s^*}_{x^*_{i+1}}$ and covariance matrix $\widehat{\boldsymbol{\Sigma}}^{s^*}_{x^*_{i+1}}$). We can then construct a random variable $\hat{L}^{S^*}_{x^*_{i+1}}$, as a mixture of $\hat{L}^{s^*}_{x^*_{i+1}}$'s, for different plaintext candidates x^*_{i+1}, with probability:

$$\Pr[\hat{L}^{S^*}_{x^*_{i+1}}] = \sum_{s^*} \hat{\Pr}^{(i)}[s^*] \cdot \Pr[\hat{L}^{s^*}_{x^*_{i+1}}]. \tag{1}$$

That is, we have one $\hat{L}^{S^*}_{x^*_{i+1}}$ per plaintext candidate x^*_{i+1}. Exemplary mixtures are represented in Figure 2, for two different plaintexts and in a simple context with only four possible subkeys. The definition of this variable is motivated by the fact that at step i in an attack, the only available knowledge about the subkeys is stored in $\hat{\Pr}^{(i)}[s^*]$. Hence, Equation (1) is the best available estimation of the leakage pdf at this step. For a given mixture and a fixed (simulated) leakage value $\hat{l}^{S^*}_{x^*_{i+1}}$, it is possible to compute the conditional entropy $\hat{\mathrm{H}}^{(\mathbf{x}^*_{i+1})}[S^*|\hat{l}^{S^*}_{x^*_{i+1}}]$, as illustrated in Figure 2 for three exemplary leakage values l_0, l_1 and l_2. Integrating this entropy over the leakages yields the estimations:

$$\hat{\mathrm{H}}^{(\mathbf{x}^*_{i+1})}[S^*] = \int \Pr[\hat{l}^{S^*}_{x^*_{i+1}}] \cdot \hat{\mathrm{H}}^{(\mathbf{x}^*_{i+1})}[S^*|\hat{l}^{S^*}_{x^*_{i+1}}] \; dl^{S^*}_{x^*_{i+1}}.$$

And since we have one such entropy value for every possible choice of x^*_{i+1} in $|\mathcal{X}|$, we finally obtain the following rule to select the plaintexts:

$$\tilde{x}_{i+1} = \underset{x^*_{i+1}}{\operatorname{argmin}} \; \hat{\mathrm{H}}^{\mathbf{x}^*_{i+1}}[S^*]$$

Summarizing, we use the available a-priori subkey information at step i and the leakage model (*i.e.* the templates) to predict how the entropy of the subkey candidates would evolve at step $i+1$, for different plaintext candidates x^*_{i+1}.

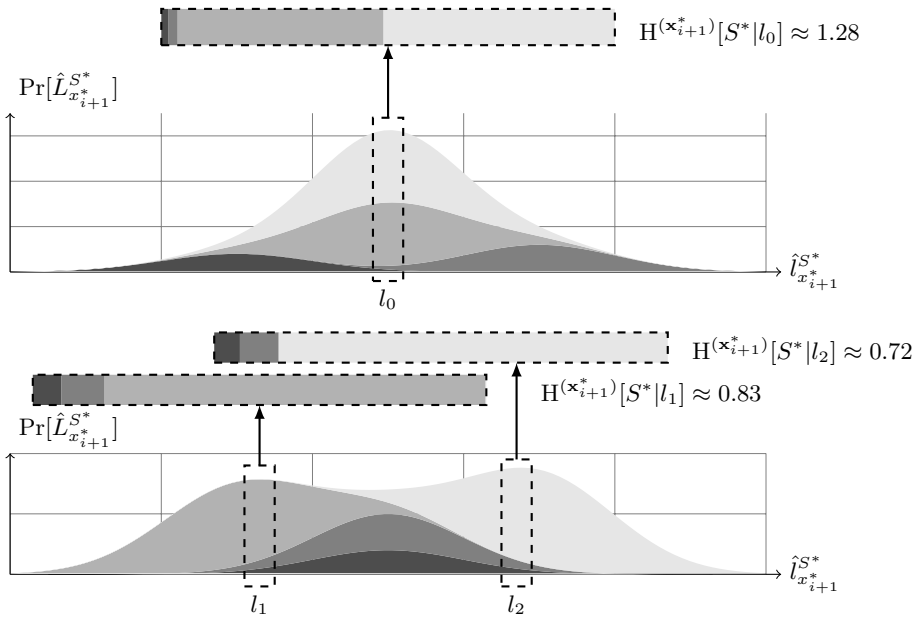


Fig. 2. Adaptive selection of the plaintexts in a simplified context with $|\mathcal{S}| = 4$.

3.3 Generalization to non-profiled attacks

As detailed in the previous section, an important requirement when applying an adaptive strategy (*e.g.* in the case of template attacks) is the availability of a good leakage model. Therefore, an interesting question is to know if such strategies can still help in the context of non-profiled side-channel attacks, where the model is usually less precise. As an illustration, we discuss this problem for the frequently considered correlation power analysis using Pearson's coefficient.

Correlation attacks, as described in [4], use the following distinguisher:

$$\hat{\rho}(\mathbf{M}_q^{s^*}, \mathbf{L}_q) = \frac{\hat{\mathbf{E}}\Big(\big(l_i - \hat{\mathbf{E}}(\mathbf{L}_q)\big) \cdot \big(m_i^{s^*} - \hat{\mathbf{E}}(\mathbf{M}_q^{s^*})\big)\Big)}{\hat{\sigma}(\mathbf{L}_q) \cdot \hat{\sigma}(\mathbf{M}_q^{s^*})},$$

where $\hat{\mathbf{E}}$ and $\hat{\sigma}$ denote the sample means and standard deviations of a random variable, respectively. In this context, the models $m_i^{s^*}$ are not the complete leakage pdf (as in template attacks) but only their mean values (*i.e.* the first-order moments of the pdf). In general, these mean values are not estimated with profiling, but rather taken from engineering intuition. For example, a usual assumption is to use the so-called Hamming Weight or distance leakage models [14].

Adaptive correlation. When trying to apply the strategy of the previous section to correlation attacks, two main problems arise, that we now detail. First, the subkey probability estimation is not straightforward. Whereas template attacks rate these subkey candidates using their probabilities, correlation attacks

return a set of scores, corresponding to the value of Pearson's coefficient. In order to mount an adaptive attack, the adversary consequently needs to use heuristics in order to estimate the subkey distribution $\hat{\Pr}^{(i)}[s^*]$, *e.g.* by:

- using the absolute value of the estimated coefficient $\hat{p}_i^{s^*} = |\hat{\rho}(\mathbf{M}_q^{s^*}, \mathbf{L}_q)|$,
- applying Fisher's transform on this correlation coefficient (in order to get a normal distribution), *i.e.* computing $\hat{p}_i^{s^*} = |\text{arctanh}(\hat{\rho}(\mathbf{M}_q^{s^*}, \mathbf{L}_q))|$,
- computing the p-values associated with each correlation in a hypothesis test. For example, one could estimate the p-value obtained when stating that the subkey candidate in not correlated with the model.

In each case, we then need to normalize the $\hat{p}_i^{s^*}$'s in order to get an estimated probability distribution, as the values we obtain are not actual probabilities, and some wrong subkeys may give a non-zero score (*aka* ghost peaks [4]):

$$\hat{\Pr}^{(i)}[s = s^*] = \frac{\hat{p}_i^{s^*}}{\sum_{s' \in \mathcal{S}} \hat{p}_i^{s'}}$$

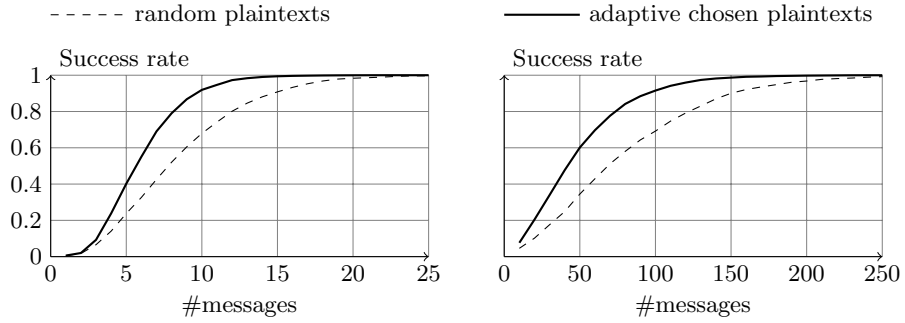
Second, and more critically, the selection procedure of Section 3.2 requires to build a random variable $\hat{L}_{x_{i+1}^*}^{S^*}$ as a mixture of $\hat{L}_{x_{i+1}^*}^{s^*}$, that estimates the leakage distribution given the subkey probabilities at step i in an attack. This requires an estimate of the leakage pdf that is given if the leakage model is probabilistic (as in template attacks), but is not directly available in a correlation attack. Again, a number of heuristics are possible. The simplest one, that we considered in this work, is to combine the (Hamming weight or distance) power models with a Gaussian assumption, *i.e.* to paste a Gaussian curve to the different Hamming weights, of which the variance is estimated "on-the-fly" during the attack.

4 Simulated experiments

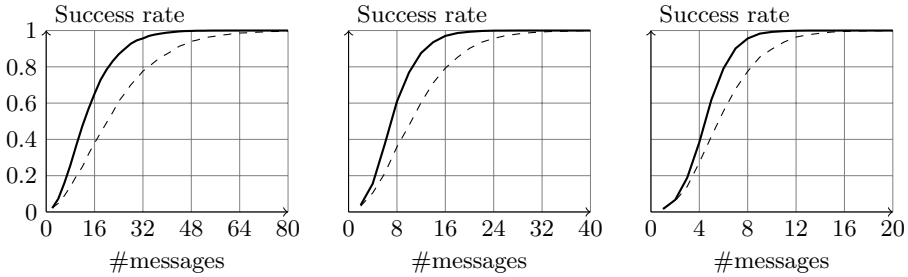
In order to validate our adaptive message selection, we first conducted software simulations. These attacks target the output of a single AES S-box in the first encryption round. Excepted if mentioned otherwise, physical leakages are simulated as the Hamming weight of the S-box outputs, to which is added a normally distributed noise with standard deviation σ_n. The efficiency of an attack is then measured with the success rate, averaged over 1000 independent key recoveries. The results of our experiments are in Figure 3 from which we observe:

1. In all cases, the adaptive strategy leads to increased success rates. It noticeable that the impact of this adaptivity becomes significant as soon as a slight a-priori knowledge is known about the target subkey. Also, and as illustrated in Figure 3**.(a)**, this improvement holds for different noise levels.
2. The same observation also holds for different leakage functions. For example, Figure 3**.(b)** shows the success rates of attacks exploiting three different side-channels of the form: $\mathsf{L}(x) = \sum_i \alpha_i x[i] + n$, where $x[i]$ is the ith bit of the target S-box output and n a Gaussian noise. Interestingly, these examples

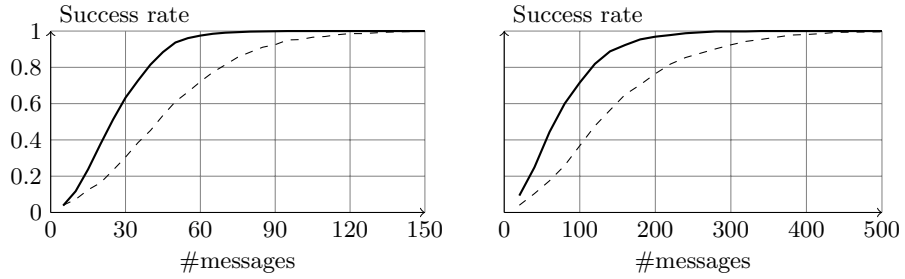
(a) Hamming weight leakage function, $\sigma_n = 1$ (left) and $\sigma_n = 4$ (right).



(b) Different leakage functions with the same noise level $\sigma_n = 1$ and the conditional entropy $\mathrm{H}[S|\mathbf{L}_1] = 7.7$ (left), $\mathrm{H}[S|\mathbf{L}_1] = 7.4$ (middle), $\mathrm{H}[S|\mathbf{L}_1] = 6.9$ (right).



(c) Masked S-box, Hamming weight leakages, $\sigma_n = 0.5$ (left) and $\sigma_n = 1$ (right).



(d) Correlation attacks, Hamming weight leakages, $\sigma_n = 0.5$ (left) and $\sigma_n = 1$ (right).

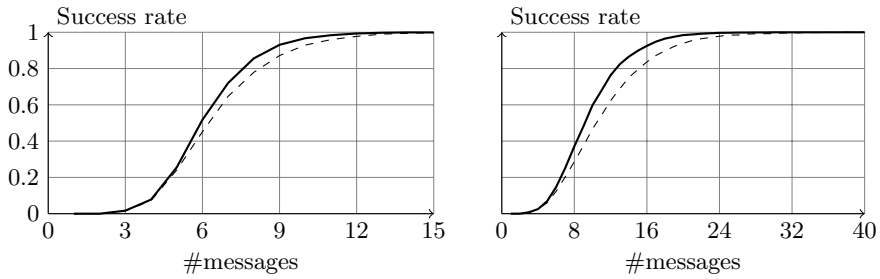


Fig. 3. Success rates of different simulated experiments.

directly connect with the framework in [22]. They show that as in a non-adaptive context, a more informative leakage function (measured with the conditional entropy $\mathrm{H}[S|\mathbf{L}_1]$) leads to more efficient attacks.

3. Although more computationally intensive (because they require to deal with mixtures of probability distributions, *e.g.* as described in [13]), attacks against masked implementations exhibit similar improvements (see Figure 3.**(c)**).
4. Eventually, the results of the heuristics proposed to exploit adaptivity in the context of correlation power analysis are given in Figure 3.**(d)**. As expected, the imperfect approximations of the pdf imply smaller improvements.

This last point implies interesting scopes for further research. For example, it would be interesting to apply adaptive strategies to other non-profiled tools such as the MIA [7], in which an estimation of the leakage pdf is computed as part of the attack. In the same line, it could also be possible to exploit stochastic models in order to obtain a leakage model "on-the-fly". In this respect, it is worth recalling that such distinguishers can also be used for profiling a device, without a-priori knowledge of the key (*i.e.* to obtain templates in a flexible way).

5 Experiments using actual measurements

In order to confirm the previous simulations, we additionally performed actual experiments against an implementation of the AES Rijndael in an Atmel Atmega 644p chip. Such actual measurements are interesting because they allow exploiting the leakage of several time samples, contrary to the simulated case where a single point of interest was considered. In other words, actual experiments allow easily evaluating the impact of multivariate templates. In practice, we compared attacks with up to three samples, for adaptive and random message selection. The points of interest were selected as part of the profiling phase, two of them corresponding to the S-box computation, and one to the first key addition. Again, we estimated the success rates over 1000 independent key recoveries, excepted for the trivariate attack which was only launched against 50 different keys. The smaller number of attacks in this case is due to their computational cost, that grows exponentially with the number of dimensions, and makes the exhaustive analysis of Section 3.2 too intensive to be performed.

The results of these experiments are in Figure 4. They show that the adaptive strategy holds for real world implementations. That is, the leakage models built during profiling can be precise enough[1] so that the estimation of the "next-step entropy" $\hat{\mathrm{H}}^{\mathbf{x}^*_{i+1}}[S^*]$ leads to a meaningful selection of the next plaintext x_{i+1}. It is worth noting the large difference between univariate random-message attacks and trivariate chosen-message ones. It illustrates the variability that can be observed between different attack scenarios in physically observable cryptography.

[1] We used a 1000 traces to characterize each template.

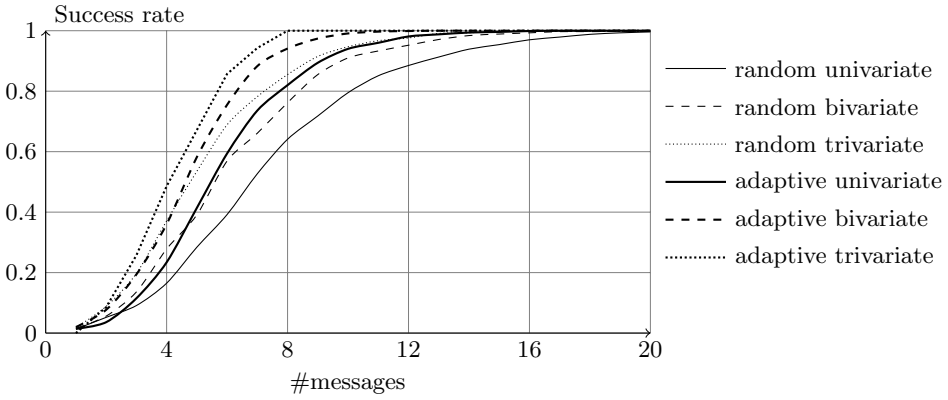


Fig. 4. Success rates of experiments carried out against an AES implementation.

6 Discussion and concluding remarks

Is our strategy optimal? Following the previous sections, a first natural question is to know if the proposed strategy is optimal. For this purpose, it is interesting to relate our work with the one of [12]. The authors estimate the number of queries required for a key-recovery, in the context of deterministic side-channel leakages. For each encryption step, the key candidates are partitioned in r sets, and the side-channel leakage allows the adversary to discriminate one set containing the correct key. The optimal strategy minimizes the number of steps required to reduce the number of key candidates to one. The main limitation is that its computational cost is doubly exponential in the number of attack steps. This is because in this optimal strategy, it is in fact several next plaintexts (x_{i+1}, x_{i+2}, ...) that have to be predicted in order to minimize the entropy of the key candidates. Hence, this strategy is hardly applicable, even for small parameters size. In order to get rid of this limitation, Köpf and Basin propose an alternative greedy heuristic, which predicts only one next plaintext at a time.

The procedure presented in this paper can be seen as the extension of such a greedy strategy, from the deterministic case towards the more general probabilistic case. The main difference is that deterministic leakages allow the adversary to effectively eliminate subkeys, whereas probabilistic leakages only help the adversary to update the subkey candidates' distribution. This extension allows an application of adaptive strategies to a broader class of attacks, including power and electromagnetic leakages, typically. But it comes at a computational cost, since we had to turn deterministic sums into integrals (that are multidimensional in the case of multivariate attacks). Summarizing, our strategy is not optimal. But as indicated in [12], greedy heuristics can provide close to (or even equal to) optimal results in practice. The exact evaluation of the greedy approach with respect to the optimal one and the investigation of alternative solutions to reduce the computational cost of adaptive attacks is a scope for further research.

Implications. Next to optimality, another important question is to determine whether the application of adaptive strategies may have practical impact in certain applications. Looking at the figures in the previous sections indicates that the improvements are not huge, but can be significant. For example, Table 1 shows that the number of measurements required to reach a certain success rate is improved, in particular when combining adaptive attacks with trivariate leakages. But in fact, the consequences of adaptivity are best observed with respect to the global success rates of the attacks. That is, because standard DPA attacks exploit a divide-and-conquer strategy, the overall success rate against the full AES master key can be estimated by simply raising the success rate against an 8-bit byte to the power 16. This assumes that all key bytes are equally difficult to recover, which is reasonable in most applications, in particular software ones as in Section 5. In the case of adaptive attacks, it also means that the selection of all the plaintext bytes are performed concurrently. Table 2 shows these estimated success rates in function of the number of messages in the attack. It clearly illustrates the strong impact that adaptive strategies may have. For example, one can imagine a re-keying scheme where the secret is updated every four encryptions. Our results suggest that the resulting security level would differ by a factor of 2^9 depending on the use or not of adaptive messages. This factor increases to 2^{26} if multivariate leakages are considered. And in the case of attacks against the AES-256, these factors would additionally be squared.

Target success rate	$> 20\%$	$> 40\%$	$> 60\%$	$> 80\%$	$\approx 100\%$
random messages - 1D	5	7	8	11	20
adaptive messages - 1D	4	5	7	8	16
adaptive strategy - 3D	3	4	5	6	8

Table 1. Approximated data complexities for different attacks against an 8-bit subkey.

Number of messages	2	3	4	5	6	7	8
random messages - 1D	2^{-69}	2^{-55}	2^{-42}	2^{-29}	2^{-21}	2^{-15}	2^{-10}
adaptive messages - 1D	2^{-64}	2^{-50}	2^{-33}	2^{-20}	2^{-12}	2^{-7}	2^{-5}
adaptive strategy - 3D	2^{-58}	2^{-32}	2^{-16}	2^{-9}	2^{-4}	2^{-2}	-

Table 2. Approximated success rates for different attacks against a 128-bit key.

It is worth mentioning that targeting hardware implementations, in which all the subkeys are manipulated in parallel, would imply additional questions. For example, in a context where a single key byte has to be recovered with high efficiency, one could also take advantage of chosen plaintexts so that the remaining input bits are constant, in order to reduce the algorithmic noise. But an adaptive strategy would still apply to the target key byte. Extending the experiments of this paper towards more devices and countermeasures against side-channel attacks is anyway another interesting direction for further research.

Eventually, and as discussed in [23], the success rates of adaptive attacks can, when applicable, be used as rough (but only available ones) estimations of the bounded leakage[2] that is necessary to prove the security of certain leakage resilient constructions. Our results can also be directly integrated in the evaluation framework of Eurocrypt 2009 [22]: they exhibit a new type of distinguisher that can take advantage of the information leakage in a close to optimal manner. Summarizing, this paper brings an important contribution to the exploitation of side-channel leakages in both theoretical and practical settings.

References

1. D. Agrawal, B. Archambeault, J. Rao, P. Rohatgi, *The EM Side-Channel(s)*, in the proceedings of CHES 2002, Lecture Notes in Computer Science, vol 2523, pp 29-45, Redwood City, CA, USA, August 2002.
2. T. Baignères, *Quantitative Security of Block Ciphers: Design and Cryptanalysis Tools*, PhD Thesis, EPFL, Lausanne, Switzerland, November 2008.
3. D. Bleichenbacher, *Chosen Ciphertext Attacks Against Protocols Based on the RSA Encryption Standard PKCS #1*, in the proceedings of Crypto 1998, Lecture Notes in Computer Science, vol 1462, pp 1-12, Santa Barbara, CA, USA, August 2002.
4. E. Brier, C. Clavier, F. Olivier, *Correlation Power Analysis with a Leakage Model*, in the proceedings of CHES 2004, Lecture Notes in Computer Science, vol 3156, pp 16-29, Boston, MA, USA, August 2004.
5. S. Chari, J. Rao, P. Rohatgi, *Template Attacks*, in the proceedings of CHES 2002, LNCS, vol 2523, pp 13-28, Redwood Shores, California, USA, August 2002.
6. S. Dziembowski, K. Pietrzak, *Leakage-Resilient Cryptography*, in the proceedings of FOCS 2008, pp 293-302, Washington, DC, USA, October 2008.
7. B. Gierlichs, L. Batina, P. Tuyls, B. Preneel, *Mutual Information Analysis: A Generic Side-Channel Distinguisher*, in the proceedings of CHES 2008, LNCS, vol 5154, pp 396-410, Washington DC, USA, August 2008.
8. L. Goubin, J. Patarin, *DES and Differential Power Analysis*, in the proceedings of CHES 1999, LNCS, vol 1717, pp 158-172, Worcester, MA, USA, August 1999.
9. A. Joux, G. Martinet, F. Valette, *Blockwise-Adaptive Attackers: Revisiting the (In)Security of Some Provably Secure Encryption Models: CBC, GEM, IACBC*, in the proceedings of Crypto 2002, Lecture Notes in Computer Science, vol 2442, pp 17-30, Santa Barbara, California, USA, August 2002.
10. P.C. Kocher, *Timing Attacks on Implementations of Diffie-Hellman, RSA, DSS, and Other Systems*, in the proceedings of Crypto 1996, Lecture Notes in Computer Science, vol 1666, pp 104-113, Santa Barbara, CA, USA, August 2002.
11. P. Kocher, J. Jaffe, B. Jun, *Differential Power Analysis*, in the proceedings of Crypto 1999, LNCS, vol 1666, pp 398-412, Santa-Barbara, CA, USA, August 1999.
12. B. Köpf, D.A. Basin, *An Information-Theoretic Model for Adaptive Side-Channel Attacks*, in the proceedings of the ACM Conference on Computer and Communications Security, pp 286-296, Alexandria, Virginia, USA, October 2007.
13. K. Lemke-Rust, C. Paar, *Gaussian Mixture Models for Higher-Order Side Channel Analysis*, in the proceedings of CHES 2007, Lecture Notes in Computer Science, vol 4727, pp 14-27, Vienna, Austria, September 2007.

[2] Given that the plaintext selection is granted to adversaries. As previously said, it is anyway a more reasonable abstraction than the adaptivity of the leakage function.

14. S. Mangard, E. Oswald, T. Popp, *Power Analysis Attacks*, Springer, 2007.
15. S. Mangard, E. Oswald, F.-X. Standaert, *One for All, All for One: Unifying Standard DPA Attacks*, Cryptology ePrint Archive, Report 2009/449.
16. K. Pietrzak, *A Leakage-Resilient Mode of Operation*, in the proceedings of Eurocrypt 2009, LNCS, vol 5479, pp 462-482, Cologne, Germany, April 2009.
17. E. Prouff, *DPA Attacks and S-Boxes*, in the proceedings of FSE 2005, Lecture Notes in Computer Science, vol 3557, pp 424-441, Paris, France, February 2001.
18. M. Rivain, E. Dottax, E. Prouff, *Block Ciphers Implementations Provably Secure Against Second-Order Side-Channel Analysis*, in the proceedings of FSE 2008, LNCS, vol 5086, pp 127-143, Lausanne, Switzerland, February 2008.
19. J.-J. Quisquater, D. Samyde, *ElectroMagnetic Analysis (EMA): Measures and Counter-Measures for Smart Cards*, in the proceedings of E-smart 2001, Lecture Notes in Computer Science, vol 2140, pp 200-210, Cannes, France, September 2001.
20. W. Schindler, *A Timing Attack against RSA with the Chinese Remainder Theorem*, in the proceedings of CHES 2000, Lecture Notes in Computer Science, vol 2965, pp 109-124, Worcester, MA, USA, Auguest 2000.
21. W. Schindler, K. Lemke, C. Paar, *A Stochastic Model for Differential Side-Channel Cryptanalysis*, in the proceedings of CHES 2005, Lecture Notes in Computer Science, vol 3659, pp 30-46, Edinburgh, Scotland, September 2005.
22. F.-X. Standaert, T.G. Malkin, M. Yung, *A Unified Framework for the Analysis of Side-Channel Key Recovery Attacks*, in the proccedings of Eurocrypt 2009, LNCS, vol 5479, pp 443-461, Cologne, Germany, April 2009, extended version available on the Cryptology ePrint Archive, Report 2006/139.
23. F.-X. Standaert, O. Pereira, Y. Yu, J.-J. Quisquater, M. Yung, E. Oswald, *Leakage Resilient Cryptography in Practice*, Cryptology ePrint Archive, report 2009/341.
24. D. Wagner, *The Boomerang Attack*, in the proceedings of FSE 1999, Lecture Notes in Computer Science, vol 1636, pp 156-170, Rome, Italy, August 2002.