On the Relationship between Correlation Power Analysis and the Stochastic Approach: An ASIC Designer Perspective

Fabrizio De Santis[1,2], Michael Kasper[3,4], Stefan Mangard[2],
Georg Sigl[1], Oliver Stein[4,5], and Marc Stöttinger[6]

[1] Technische Universität München, Germany
desantis@tum.de
[2] Infineon Technologies AG, Germany
stefan.mangard@infineon.com
[3] Fraunhofer Institute for Secure Information Technology, Germany
michael.kasper@sit.fraunhofer.de
[4] Center for Advanced Security Research Darmstadt, Germany
{michael.kasper,oliver.stein}@cased.de
[5] Technische Hochschule Regensburg, Germany
oliver.stein@hs-regensburg.de
[6] Nanyang Technological University, Singapore
mstottinger@ntu.edu.sg

Abstract. The design and the security verification of side-channel resistant cryptographic hardware often represent an iterative process. This process essentially consists of a *detection phase* ($\mathcal{DP}$), where the information leakage is identified and a *correction phase* ($\mathcal{CP}$), where design flaws are corrected. Correlation Power Analysis (CPA) and the Stochastic Approach (SA) are two candidate tools to perform the $\mathcal{DP}$ and to *support* designers in the $\mathcal{CP}$. However, until now, the relationship between these two tools has not been discussed yet and it is uncertain from a designer point of view, what informative feedback can be gained from these methods, especially when it comes to evaluate high-dimensional leakage models. In this work, we investigate the relationship between CPA and the SA from both a mathematical and empirical point of view. In particular, we demonstrate that the informative feedback provided by the SA is transferable to a linear combination of CPA attacks and discuss the implications of this entanglement, when it comes to pinpoint the high-dimensional leakage of simulated leakage data and simulated power traces of an ASIC implementation of Present.

1 Introduction

The analysis of side-channel leakage is an integral part of the design of secure cryptographic hardware, which is performed at early stages of development already. Typically, the earlier a design flaw is discovered, the less re-engineering effort has to be spent in correcting the issue [1,19]. In practice, two different challenges have to be accomplished during the development of side-channel resistant

G. Paul and S. Vaudenay (Eds.): INDOCRYPT 2013, LNCS 8250, pp. 215–226, 2013.

cryptographic hardware: first, the *leakage* must be identified in a so-called *detection phase* ($\mathcal{DP}$). Secondly, the *design flaws*, which have caused the leakage, must be repaired in a so-called *correction phase* ($\mathcal{CP}$).

Recently, different works have been published addressing the $\mathcal{DP}$, cf. [7,17,23], whereas the $\mathcal{CP}$ was not addressed. However, previous contributions have shown how the Correlation Power Analysis (CPA) and the Stochastic Approach (SA) are indeed useful tools to constructively *support* designers in the correction of design flaws: in [15, 16] CPA was used to pinpoint a design flaw of a masked ASIC implementation of AES, while a constructive usage of the SA is reported in [9], where the SA was used to pinpoint a design flaw in the routing of an FPGA implementation of AES. Yet, it remains uncertain from a designer perspective, which different informative feedback can be gained from CPA and the SA to support designers in the $\mathcal{CP}$. Also, it remains unclear, to which extent these tools can be used to pinpoint the *leakage* of bit interactions using high-dimensional leakage models. In fact, aforementioned works only considered the switching activity of *independent* leakage contributions, whereas [3, 8, 20] have shown that dependent contributions may also produce exploitable information leakage *e.g.* due to the occurrence of glitches in the combinational logic path or due to technology factors like the scaling of the CMOS technology.

In this work, we address these open questions by investigating the relationship between CPA and the SA from both a mathematical and empirical point of view. In particular, we provide the following contributions:

- We proof that the SA can be expressed as a linear combination of CPA attacks and provide two Corollaries, which demonstrate that the informative feedback provided by CPA and the SA is indeed equivalent, in some specific, yet practically relevant cases.
- We extend previous works [9, 15, 16] by considering high-dimensional leakage models. In particular, we show that the SA can precisely identify and quantify each contribution to the leakages, once an adequate approximation subspace is selected and properly estimated. On the other hand, we show that CPA can only loosely point to individual leakage contributions, being inherently unable to capture bit interactions when high-dimensional leakage models are used.

The rest of the paper is structured as follows. In Sect. 2, we provide the necessary background information. In Sect 3, we discuss the *mathematical* relationship between CPA and the SA. In Sect. 4, we *empirically* evaluate and discuss the application of CPA and the SA to simulated high-dimensional leakage data as well as to simulated power traces of an ASIC implementation of Present. Finally, we draw conclusions in Sect. 5.

2 Background

The Leakage Model. During the execution of a cryptographic implementation θ, *sensitive* intermediate values $v_{\theta,t}(x,k) \in \{0,1\}^w$ are computed at certain time t. Sensitive intermediate values typically depends on a public input

$x \in \{0,1\}^p$ and a secret key $k \in \{0,1\}^s$ and their size w depends on the particular design specifications. The computation of sensitive intermediate values $v_{\theta,t}(x,k)$ for randomly chosen inputs $x \in \{0,1\}^p$ and a random, but fixed, key $k \in \{0,1\}^s$ can be interpreted as a random experiment, which defines the random variable $V_{\theta,t}(\cdot,k)$. Similarly, the measurable leakage $\ell_{\theta,t}$, which is observed during the computation of $v_{\theta,t}(\cdot,k)$, can be treated as the realization of yet another random variable $L_{\theta,t}(\cdot,k)$ taking values in $\mathbb{R}$. A central assumption in side-channel analysis is to consider the leakage as the sum of a deterministic contribution $\delta_{\theta,t}$ and *independent* noise R_t:

$$L_{\theta,t}(\cdot,k) = \delta_{\theta,t}(V_{\theta,t}(\cdot,k)) + R_t. \tag{1}$$

Tools for Side-Channel Analysis. CPA and the SA are side-channel analysis tools which are typically employed by designers to analyse the leakages of cryptographic hardware implementations and identify possible design flaws [9,15,16]. Both techniques consider the leakage $\ell_{\theta,t}(\cdot,k)$ of N cryptographic operations under public inputs $x_0,\ldots,x_{N-1}$, being observed at M points in time $T = \{t_0,t_1,\ldots,t_{M-1}\}$. Let the matrix $\mathbf{L}_\theta = \big(\ell_{\theta,t_j}(x_i,k)\big)_{\substack{0\le i\le N-1\\ 0\le j\le M-1}}$ denote the $N \times M$ *leakage matrix* and $\boldsymbol{\ell}_{\theta,t}$ a column vector thereof. For any $u \ge 0$, let $\mathbf{A}_\theta$ be the $N \times (u+1)$ *model matrix* defined as follows:

$$\mathbf{A}_\theta := \begin{pmatrix} g_{\theta,0}(x_0,k^*) & \ldots & g_{\theta,u}(x_0,k^*) \\ \vdots & \ddots & \vdots \\ g_{\theta,0}(x_{N-1},k^*) & \ldots & g_{\theta,u}(x_{N-1},k^*) \end{pmatrix}. \tag{2}$$

where $g_{\theta,j}\colon \{0,1\}^p \times \{0,1\}^q \to \mathbb{R}$ are the analytical *model functions*, which model the leakages of arbitrary selected *target* intermediate values. The target intermediate values are typically a function of the public inputs $x_0,\ldots,x_{N-1}$ and a *small* key hypothesis $k^* \in \{0,1\}^q$ ($q \ll s$) and do not necessarily correspond to $v_{\theta,t}(\cdot,k)$, being arbitrarily selected by designers during the security verifications (*e.g.* only a subset of bits can be targeted).

Correlation Power Analysis (CPA). The basic idea of CPA is to evaluate the linear strength between the observed leakages and the leakage models specified via the $\mathbf{A}_\theta$ matrix, cf. [4]. Hence, given the leakages $\boldsymbol{\ell}_{\theta,t}$ and the corresponding model matrix $\mathbf{A}_\theta$, CPA computes the sample Pearson's correlation coefficient $\widetilde{\rho}$ between the measured leakages $\boldsymbol{\ell}_{\theta,t}$ and each column $\mathbf{A}_{\theta,j}$ of the model matrix $\mathbf{A}_\theta$, as follows:

$$\widetilde{\rho}_{\theta,j,t}(\mathbf{A}_{\theta,j},\boldsymbol{\ell}_{\theta,t}) := \frac{\sum_{n=0}^{N-1}(g_{\theta,j}(x_n,k^*) - \widetilde{\mathbb{E}}(\mathbf{A}_{\theta,j}))(\ell_{\theta,t}(x_n,k) - \widetilde{\mathbb{E}}(\boldsymbol{\ell}_{\theta,t}))}{\sqrt{\sum_{n=0}^{N-1}(g_{\theta,j}(x_n,k^*) - \widetilde{\mathbb{E}}(\mathbf{A}_{\theta,j}))^2 \sum_{n=0}^{N-1}(\ell_{\theta,t}(x_n,k) - \widetilde{\mathbb{E}}(\boldsymbol{\ell}_{\theta,t}))^2}},$$

where $\widetilde{\mathbb{E}}(\cdot)$ denotes the sample mean operator.

The Stochastic Approach (SA). The core idea of the SA is to estimate the leakage function, which underlies the origin of the leakage, by approximating the deterministic part $\delta_{\theta,t}$ and the noise R_t of Eq. (1) in a chosen subspace. In here, we only recall the approximation of the deterministic part, and refer to [9,11,22] for a comprehensive study of the SA. The SA approximates the deterministic part in a $(u+1)$-dimensional subspace $\mathcal{F}_{u+1,t}$ spanned by the analytical model functions $g_{\theta,j}(\cdot,\cdot)$ as follows:

$$\mathcal{F}_{u+1,t} := \left\{ \widetilde{\delta} : \{0,1\}^p \times \{0,1\}^q \to \mathbb{R}; \quad \widetilde{\delta} = \sum_{\mathrm{j}=0}^{\mathrm{u}} \widetilde{\beta}_{\theta,\mathrm{j},\mathrm{t}} \mathrm{g}_{\theta,\mathrm{j}} \right\}. \tag{3}$$

The ordinary least square method, cf. [18], is used to derive the optimal approximation $\widetilde{\delta}_{\theta,t}(\cdot)$ in terms of square errors. The least square estimators $\widetilde{\boldsymbol{\beta}}_\theta$ are uniquely determined by the solution which minimizes $\|\mathbf{L}_\theta - \mathbf{A}_\theta \boldsymbol{\beta}_\theta\|_2^2$:

$$\widetilde{\boldsymbol{\beta}}_\theta = (\mathbf{A}_\theta^T \mathbf{A}_\theta)^{-1} \mathbf{A}_\theta^T \mathbf{L}_\theta, \tag{4}$$

where $\mathbf{A}_\theta^T \mathbf{A}_\theta$ must be an invertible $(u+1) \times (u+1)$-matrix and, typically, $\mathbf{A}_{\theta,0} = \mathbf{1}_N$. In side-channel analysis, the regression coefficients $\boldsymbol{\beta}_\theta$ are usually referred to as the *leakage characteristic* of the design θ under test or, simply, $\boldsymbol{\beta}_\theta$-characteristic [9].

Other Tools. For the sake of completeness, we briefly recall the Kocher's original DPA attack [10] and the Mutual Information Analysis (MIA) [2]. Kocher's original DPA attack assumes a single-bit model (*e.g.* $\mathbf{A}_\theta$ defined over $\{0,1\}$) and computes $\widetilde{\mathbb{E}}[\boldsymbol{\ell}_{\theta,t}|\mathbf{A}_{\theta,j}=1] - \widetilde{\mathbb{E}}[\boldsymbol{\ell}_{\theta,t}|\mathbf{A}_{\theta,j}=0]$. Similarly, MIA estimates the mutual information between leakages and models as $\widetilde{I}(\boldsymbol{\ell}_{\theta,t}, \mathbf{A}_{\theta,j})$ to quantify the leakage amount in bits of information. Interestingly, these attacks are *asymptotically* equivalent to CPA attacks when single-bit models are employed [14].

Security Verification. The goal of a side-channel security analysis is to make conclusive statements about the side-channel security of a design θ under test. However, since both leakages and models are realizations of random experiments, security statements are only possible in a stochastic sense, that is, only relatively to selected $(\mathbf{A}_\theta, \mathsf{T}, \gamma)$-adversaries, where $\mathbf{A}_\theta$ denotes the selected leakage models, T denotes the selected side-channel analysis tool (*e.g.* CPA or the SA) and γ denotes the desired statistical confidence[1], unless equivalences are demonstrated, cf. [6, 14]. Hence, the experience of designers play a fundamental role to select appropriate adversaries in order to anticipate the attackers working conditions and pre-emptively remove possible design flaws. Clearly, an improper selection of the adversaries could possibly leave exploitable flaws behind and, consequently, lead to an overestimated security confidence and produce higher security risks. In this respect, the task of designers is to properly define the model matrix $\mathbf{A}_\theta$ by accurately choosing the target intermediate values, the model selection functions and the side-channel analysis tools. A typical choice for designers is to select those intermediate values which can be computed from a small key hypothesis

[1] In this view, we can speak of $(\mathbf{A}_\theta, \mathsf{T}, \gamma)$-security. The number of traces N is implicitly accounted in by γ.

and then use Boolean model functions in the form $g_{\theta,j} : \mathbb{F}_2^p \times \mathbb{F}_2^q \rightarrow \mathbb{F}_2$ to map intermediate values to single-bits [15, 16], possibly considering any polynomial of the target bits, as follows:

$$g_{\theta,j}(x,k^*) = \prod_{p \in P_j} f_{\theta,p}(x,k^*), \quad j = 1, \ldots, \sum_{i=1}^{b} \binom{b}{i} - 1, \tag{5}$$

where $f_{\theta,p}$ denotes the p^{th} bit of the target intermediate values obtained by combining the public and secret material by an algorithm specific function $f_\theta(\cdot,\cdot)$, b is the size of target intermediate values, $\mathcal{P}$ denotes the power set operator and P_j is the j^{th} element of the set $\mathcal{P}(b)\backslash\{\emptyset\}$. The choice of using single-bit models is well-known to be suboptimal for attackers in terms of efficiency [13]. However, using the single-bit models of Eq. (5) still represents the most effective way for designers to understand the leakage characteristic of digital circuits. In fact, single-bit models make the least assumptions on the leakages and enable designers to collect informative feedback relatively to every single signal in the design. Additionally, using quadratic or higher-degree polynomials of the target bits, denoted as *high-dimensional models* in [8], allow to capture also those effects which arise from the interactions of logical transitions *e.g.* like those occurring during the asynchronous activity of the combinational logic. It is worth noting that, contrary to attackers, hardware designers typically work under more favourable conditions during security verifications for at least three reasons: first, designers have *full* control over the design under test and possess the knowledge of the secret key processed by the device. Therefore, they can skip over testing multiple key hypothesis and *significant* computational effort can be saved. Second, designers are not only interested in exploiting the leakage in the most efficient way, rather they are mainly interested in identifying and correcting design flaws. Third, hardware designers can simulate the switching activity of their implementation, or even single parts of it, with a sampling frequency in the range of THz, which is typically not available to real attackers. For instance, designers can simulate the switching activity of individual hardware modules at gate or transistor level, with a resolution of a few picoseconds [1, 19].

3 On the Relation between CPA and the SA

In this section we discuss the *mathematical* relation between CPA and the SA by proving that the leakage characteristic of the SA can be expressed as a linear combination of CPA attacks. Therefore, we derive two Corollaries of specific, yet practical relevance.

Proposition 1. *Let* $\mathbf{A}_\theta$ *and* $\widetilde{\boldsymbol{\beta}}_{\theta,t}$ *be defined as in Eq.* (2) *and* (4)*. Let* $\widetilde{\boldsymbol{\beta}}^*_{\theta,t}$ *be the vector* $\widetilde{\boldsymbol{\beta}}_{\theta,t}\backslash\{\widetilde{\beta}_{\theta,0,t}\}$ *and* $\mathbf{A}_{\theta,0} = \mathbf{1}_N$*. Then, the* $\widetilde{\boldsymbol{\beta}}^*_{\theta,t}$*-characteristic can be expressed as a linear combination of sample correlation coefficients:*

$$\widetilde{\beta}_{\theta,i,t} = \sum_{j=1}^{u} w_{i,j}\widetilde{\rho}_{\theta,j,t}(\mathbf{A}_{\theta,j}, \boldsymbol{\ell}_{\theta,t}),\ i \in [1,u],\ w_{i,j} \in \mathbf{W}\ \textit{as defined in Eq.}\ (8)$$

Proof. Starting from Eq. (2) and (4), we derive:

$$\begin{array}{ccc} (\mathbf{A}_\theta^T\mathbf{A}_\theta)\widetilde{\boldsymbol{\beta}}_{\theta,t} & = & \mathbf{A}_\theta^T\boldsymbol{\ell}_{\theta,t} \iff \\ \left(\sum_{n=0}^{N-1} g_{\theta,i}(x_n,k^*)g_{\theta,j}(x_n,k^*)\right)_{0\le i,j\le u}\widetilde{\boldsymbol{\beta}}_{\theta,t} & = & \left(\sum_{n=0}^{N-1} g_{\theta,i}(x_n,k^*)\ell_{\theta,t}(x_n,k^*)\right)_{0\le i\le u} \iff \\ \left(\sum_{j=0}^{u}\widetilde{\mathbb{E}}\left[\mathbf{A}_{\theta,i}\cdot\mathbf{A}_{\theta,j}\right]\widetilde{\beta}_{\theta,j,t}\right)_{0\le i\le u} & = & \left(\widetilde{\mathbb{E}}\left[\mathbf{A}_{\theta,i}\cdot\boldsymbol{\ell}_{\theta,t}\right]\right)_{0\le i\le u}, \end{array} \tag{6}$$

where the $\cdot$ operator denotes the element-by-element multiplication between vectors. From Eq. (6): $\widetilde{\beta}_{\theta,0,t} = \widetilde{\mathbb{E}}\left[\boldsymbol{\ell}_{\theta,t}\right] - \sum_{j=1}^{u}\widetilde{\beta}_{\theta,j,t}\widetilde{\mathbb{E}}\left[\mathbf{A}_{\theta,j}\right]$ for $i = 0$, since $\mathbf{A}_{\theta,0} = \mathbf{1}_N$. By replacing $\widetilde{\beta}_{\theta,0,t}$ in Eq. (6) and grouping terms together, we obtain:

$$\left(\sum_{j=1}^{u}\widetilde{\sigma}(\mathbf{A}_{\theta,i},\mathbf{A}_{\theta,j})\widetilde{\beta}_{\theta,j,t}\right)_{0\le i\le u} = \left(\widetilde{\sigma}(\mathbf{A}_{\theta,i},\boldsymbol{\ell}_{\theta,t})\right)_{0\le i\le u}, \tag{7}$$

where $\widetilde{\sigma}(\cdot,\cdot)$ is the sample covariance estimator. By rewriting Eq. (7) in a matrix form we obtain $\widetilde{\boldsymbol{\Sigma}}_{\mathbf{A}_{\theta,i},\mathbf{A}_{\theta,j}}\widetilde{\boldsymbol{\beta}}^*_{\theta,t} = \widetilde{\boldsymbol{\Sigma}}_{\mathbf{A}_{\theta,i},\boldsymbol{\ell}_{\theta,t}}$, where the matrices $\widetilde{\boldsymbol{\Sigma}}_{\mathbf{A}_{\theta,i},\mathbf{A}_{\theta,j}} = \left(\widetilde{\sigma}(\mathbf{A}_{\theta,i},\mathbf{A}_{\theta,j})\right)_{1\le i,j\le u}$ and $\widetilde{\boldsymbol{\Sigma}}_{\mathbf{A}_{\theta,i},\boldsymbol{\ell}_{\theta,t}} = \left(\widetilde{\sigma}(\mathbf{A}_{\theta,i},\boldsymbol{\ell}_{\theta,t})\right)_{1\le i\le u}$ are sample covariance matrices. Therefore, solving for $\widetilde{\boldsymbol{\beta}}^*_{\theta,t}$:

$$\widetilde{\boldsymbol{\beta}}^*_{\theta,t} = \mathbf{W}\mathbf{R}_{\mathbf{A}_{\theta,i},\boldsymbol{\ell}_{\theta,t}}, \tag{8}$$

where $\mathbf{R}_{\mathbf{A}_{\theta,i},\boldsymbol{\ell}_{\theta,t}} = \left(\widetilde{\rho}_{\theta,i,t}(\mathbf{A}_{\theta,i},\boldsymbol{\ell}_{\theta,t})\right)_{1\le i\le u}$ is a sample correlation matrix and the weighting matrix is $\mathbf{W} = \operatorname{diag}\left(\left(\widetilde{\sigma}(\mathbf{A}_{\theta,i})\widetilde{\sigma}(\boldsymbol{\ell}_{\theta,t})\right)_{1\le i\le u}\right)\widetilde{\boldsymbol{\Sigma}}^{-1}_{\mathbf{A}_{\theta,i},\mathbf{A}_{\theta,j}}$. ■

Proposition 1 establishes a mathematical relation between CPA and the SA, which entails the following theoretical, yet practical, consequences. First, the $\widetilde{\boldsymbol{\beta}}^*_{\theta,t}$-characteristic can be computed from CPA attacks. For instance, designers can evaluate CPA attacks in first place and then estimate the $\widetilde{\boldsymbol{\beta}}^*_{\theta,t}$-characteristic only afterwards, with a consequent saving of computational resources. It is worth noting that $\widetilde{\boldsymbol{\beta}}_{\theta,t}$ and $\widetilde{\boldsymbol{\beta}}^*_{\theta,t}$ are equivalent from a designer perspective, since $\widetilde{\beta}_{\theta,0,t}$ only accounts to an offset in the leakages, which is typically not relevant for the sake of side-channel analysis. Secondly, Eq. (8) clearly shows that the information feedback provided by the $\widetilde{\boldsymbol{\beta}}_{\theta,t}$-characteristic results necessarily biased, in case the leakages contain dependent contributions which are not accounted by the selected subspace. In this respect, only an appropriate selection of the subspace can precisely identify the leakage contributions. In contrast, an improper selection would provide a biased leakage characteristic and therefore lead designers to misinterpret the leakage contributions and the design flaws there of.

Finally, it can be observed from Eq. (8) that, if all the correlations $\rho_{\theta,i,t}$ equal zero, then also the $\boldsymbol{\beta}^*_{\theta,t}$-characteristic necessarily equal zero. This is of course most unlikely to happen for any particular leakage realization, but hypothesis testing can be used to verify if there exists enough evidence in the realizations to reject the idea that populations values equal zero [13, 18].

Corollary 1. *If all the columns of the model matrix $\mathbf{A}_\theta$ are pairwise uncorrelated, then the $\widetilde{\boldsymbol{\beta}}^*_{\theta,t}$-characteristic can be expressed as scaled correlation coefficients:*

$$\widetilde{\beta}_{\theta,i,t} = \widetilde{\rho}_{\theta,i,t}(\mathbf{A}_{\theta,i}, \boldsymbol{\ell}_{\theta,t})\frac{\widetilde{\sigma}_{\boldsymbol{\ell}_{\theta,t}}}{\widetilde{\sigma}_{\mathbf{A}_{\theta,i}}}, \ \forall i \in [1, u]$$

Proof. If $\forall i \neq j\colon\ \widetilde{\sigma}(\mathbf{A}_{\theta,i}, \mathbf{A}_{\theta,j}) = 0$, then $\left(\widetilde{\boldsymbol{\Sigma}}_{\mathbf{A}_{\theta,i},\mathbf{A}_{\theta,j}}\right)^{-1} = \operatorname{diag}\left(\widetilde{\boldsymbol{\Sigma}}_{\mathbf{A}_{\theta,i},\mathbf{A}_{\theta,j}}\right)^{-1}$. Therefore, the Corollary follows immediately from Equation (8).

■

Corollary 2. *If all the columns of the model matrix $\mathbf{A}_\theta$ are pairwise uncorrelated* and *all $\mathbf{A}_{\theta,i}$ are defined over two constants $\{a_{0,i}, a_{1,i}\}$, then the $\widetilde{\boldsymbol{\beta}}^*_{\theta,t}$-characteristic can be expressed as a scaled difference of means:*

$$\widetilde{\beta}_{\theta,i,t} = \frac{1}{a_{1,i} - a_{0,i}}\left(\widetilde{\mathbb{E}}\left[\boldsymbol{\ell}_{\theta,t}|\mathbf{A}_{\theta,i} = a_{1,i}\right] - \widetilde{\mathbb{E}}\left[\boldsymbol{\ell}_{\theta,t}|\mathbf{A}_{\theta,i} = a_{0,i}\right]\right), \ \forall i \in [1, u]$$

Proof. If $\forall i \neq j\colon\ \widetilde{\sigma}(\mathbf{A}_{\theta,i}, \mathbf{A}_{\theta,j}) = 0$, then $\widetilde{\beta}_{\theta,i,t} = \widetilde{\sigma}(\mathbf{L}_{\theta,t}, \mathbf{A}_{\theta,i})/\widetilde{\sigma}^2(\mathbf{A}_{\theta,i})$ from Corollary 1. Since each $\mathbf{A}_{\theta,i}$ can be viewed as the realization of a binary random variable with probability p_i, then $\sigma^2(\mathbf{A}_{\theta,i}) = p_i(1-p_i)(a_{1,i} - a_{0,i})^2$ and $\sigma(\mathbf{L}_{\theta,t}, \mathbf{A}_{\theta,i}) = p_i(1-p_i)(a_{1,i} - a_{0,i})(\mathbb{E}\left[\mathbf{L}_{\theta,t}|\mathbf{A}_{\theta,i}{=}a_{1,i}\right] - \mathbb{E}\left[\mathbf{L}_{\theta,t}|\mathbf{A}_{\theta,i} = a_{0,i}\right])$. Therefore, the Corollary follows immediately.

■

The Corollaries above are valid if and only if $\mathbf{A}_{\theta,i}, \mathbf{A}_{\theta,j}$ are uncorrelated, that is $\widetilde{\sigma}(\mathbf{A}_{\theta,i}, \mathbf{A}_{\theta,j}) = 0$. As previously discussed, it is very unlikely to happen that $\widetilde{\sigma}(\mathbf{A}_{\theta,i}, \mathbf{A}_{\theta,j}) = 0$. However, the model matrix $\mathbf{A}_\theta$ is under the control of designers and can be constructed in such a way that the condition $\widetilde{\sigma}(\mathbf{A}_{\theta,i}, \mathbf{A}_{\theta,j}) = 0$ holds true. This is indeed the practical case when *balanced* inputs go through bijective functions (or, composition thereof, *e.g.* $S \circ \oplus$) and only first degree polynomials of the target intermediates are considered [5]. In particular, Corollary 1 implies that, under the conditions stated, the informative feedback provided by the $\widetilde{\boldsymbol{\beta}}^*_{\theta,t}$-characteristic corresponds to CPA attacks. They are indeed equivalent, in case both $\boldsymbol{\ell}_{\theta,t}$ and $\mathbf{A}_{\theta,i}$ are standardized. Therefore, in this case, testing whether $\rho_{\theta,i,t} = 0$ is equivalent to test whether $\beta_{\theta,i,t} = 0$ and require the same sample size. It should be noted that standardization is a common pre-processing technique for the SA [18] to reduce problems which arise from round-off errors and does not affect CPA, being the Pearson's correlation coefficient invariant to linear transformations of the inputs. Finally, Corollary 2 implies that, if single-bit models are used under the conditions stated, then the informative feedback of the SA is equivalent to the informative feedback provided by the Kocher's original DPA attacks and similar considerations regarding hypothesis testing hold, as in the previous case.

4 Empirical Validation

In this section, we evaluate the relationship between CPA and the SA from an *empirical* point of view, by considering the information feedback of CPA and the SA, when applied to simulated high-dimensional leakage data as well as to simulated power traces of an ASIC implementation of PRESENT [21].

Validation Using Simulated Leakage Data. Simulated high dimensional leakage data were generated by drawing inputs X uniformly at random and using the PRESENT S-box input $v_{\mathsf{Sim},t_0}(X,k) = X \oplus k$ and S-box output $v_{\mathsf{Sim},t_1}(X,k) = v_{\mathsf{Sim},t_2}(X,k) = \mathsf{S}(X \oplus k)$ as sensitive intermediate values. Table 1 summarizes the results of conducted experiments by the way of three exemplary simulated leakage functions $L^{\mathsf{Sbox_{IN}}}_{\mathsf{Sim},t_0}$, $L^{\mathsf{Sbox_{OUT}}}_{\mathsf{Sim},t_1}$ and $L^{\mathsf{Sbox_{OUT}}}_{\mathsf{Sim},t_2}$. For the sake of analysis, the very same intermediate values were targeted, using the $g_{\mathsf{Sim},j}$ model functions as defined by Eq. (5). When applying the SA, two different subspaces were considered, namely the so-called linear subspace $\mathcal{F}_5$ (c.f. $\widetilde{\beta}^{\mathsf{Lin}}_{\mathsf{Sim},j,t}$) and the so-called full subspace $\mathcal{F}_{16}$ (c.f. $\widetilde{\beta}^{\mathsf{Full}}_{\mathsf{Sim},j,t}$). Finally, bold faced fonts were used to mark values significantly different from zero with a confidence level of $\gamma = 0.999$ [13,18] and the coefficient of determination $\widetilde{R}^2$ [6,18] was used to quantify the model fit. From Table 1, it can be observed that only the SA is able to precisely identify each contribution to the leakages, given that the selected subspace properly accounts all the existing dependent contributions contained in the leakage function, c.f. $\widetilde{\beta}^{\mathsf{Lin}}_{\mathsf{Sim},j,t_1}$.

Table 1. Simulated Leakage Data

$g_{\mathsf{Sim},j}$	v_1	v_2	v_3	v_4	v_1v_2	v_1v_3	v_1v_4	v_2v_3	v_2v_4	v_3v_4	$v_1v_2v_3$	$v_1v_2v_4$	$v_1v_3v_4$	$v_2v_3v_4$	$v_1v_2v_3v_4$	$\widetilde{R}^2$
$L^{\mathsf{Sbox_{IN}}}_{\mathsf{Sim},t_0} = 5v_1 + 2.5v_1v_2 + 2.5v_1v_3 + 2.5v_1v_4 + \mathcal{N}(0, 0.001)$																
$\widetilde{\rho}_{\mathsf{Sim},j,t_0}$	**0.944**	**0.126**	**0.124**	**0.137**	**0.696**	**0.696**	**0.697**	**0.134**	**0.149**	**0.155**	**0.553**	**0.554**	**0.557**	**0.139**	**0.445**	0.459
$\widetilde{\beta}^{\mathsf{Full}}_{\mathsf{Sim},j,t_0}$	**5.000**	0.000	0.000	0.000	**2.500**	**2.500**	**2.500**	−0.000	−0.000	−0.000	0.000	−0.000	0.000	0.000	−0.000	1.000
$\widetilde{\beta}^{\mathsf{Lin}}_{\mathsf{Sim},j,t_0}$S	**8.751**	**1.234**	**1.252**	**1.267**												0.946
$L^{\mathsf{Sbox_{OUT}}}_{\mathsf{Sim},t_1} = v_1 + v_2 + v_3 + v_4 + \mathcal{N}(0, 0.001)$																
$\widetilde{\rho}_{\mathsf{Sim},j,t_1}$	**0.497**	**0.503**	**0.497**	**0.495**	**0.577**	**0.570**	**0.573**	**0.571**	**0.571**	**0.569**	**0.559**	**0.562**	**0.557**	**0.557**	**0.506**	0.848
$\widetilde{\beta}^{\mathsf{Full}}_{\mathsf{Sim},j,t_1}$	**1.000**	**1.000**	**1.000**	**1.000**	0.000	−0.000	0.000	−0.000	0.000	−0.000	0.000	−0.000	0.000	0.000	−0.000	1.000
$\widetilde{\beta}^{\mathsf{Lin}}_{\mathsf{Sim},j,t_1}$	**1.000**	**1.000**	**1.000**	**1.000**												1.000
$L^{\mathsf{Sbox_{OUT}}}_{\mathsf{Sim},t_2} = 5v_1 - 5v_2 + 10v_1v_2 + \mathcal{N}(0, 0.001)$																
$\widetilde{\rho}_{\mathsf{Sim},j,t_2}$	**0.895**	0.000	−0.000	−0.010	**0.775**	**0.510**	**0.515**	−0.002	−0.010	−0.009	**0.502**	**0.507**	**0.328**	−0.013	**0.337**	0.228
$\widetilde{\beta}^{\mathsf{Full}}_{\mathsf{Sim},j,t_2}$	**5.000**	**−5.000**	0.000	−0.000	**10.000**	−0.000	0.000	−0.000	0.000	−0.000	0.000	−0.000	0.000	0.000	−0.000	1.000
$\widetilde{\beta}^{\mathsf{Lin}}_{\mathsf{Sim},j,t_2}$	**10.039**	0.040	−0.029	−0.020												0.801

In contrast, if the employed subspace does not properly account the leakage contributions, c.f. $\widetilde{\beta}^{\mathsf{Lin}}_{\mathsf{Sim},j,t_0}$ or $\widetilde{\beta}^{\mathsf{Lin}}_{\mathsf{Sim},j,t_2}$, the information feedback delivered by the SA is biased and it is unable to identify the origins of the leakage correctly, although a notably high measure of model fit. In this sense, a biased leakage characteristic provides designers with wrong indications about the origins of the leakage and the flaws there of. On the other hand, it can be noted that CPA can only loosely explain the leakages in the three cases, since it can only consider contributions individually as if they were independent, although they are necessarily dependent when high-dimensional models are considered. Yet, it might be possible to grasp individual contributions using CPA by studying how the obtained individual contributions relate to each other. However, while this strategy would certainly be successful for relatively simple leakages, *e.g.* like those typically happening during the synchronous update of registers [12], it would not definitely be an option to explain arbitrary complicated leakage functions in general.

Validation Using Simulated Power Traces. In order to map previous experiments to a concrete application scenario, the leakage of an ASIC implementation of PRESENT [21] was investigated. Please note that only the analysis of an *unprotected* implementation is reported, since the goal here is to evaluate how CPA and the SA can *support* designers in the identification of design flaws by detecting each contribution to the leakages, and *not* to pin-point the flaws of a *specific* protected implementation. The design was synthesized using *Synopsys DesignCompiler* version *G-2012.06-SP3* in a TSMC 150nm process. The circuit was simulated at 25 MHz using *Synopsys VCS* version *F-2011.12-SP1* and power estimations were performed using *Synopsys PrimeTime* version *F-2011.06-SP3* with a resolution of 100 ps, resulting in 400 points per clock cycle. The data path processes 4 bits per clock cycle and performs one round in 17 clock cycles. The first 16 clock cycles perform $\mathsf{S}_i(X_i \oplus k_i)$ of input nibble $i \in [0, 15]$, while the 17^{th} perform the permutation layer. The investigation focused on the processing of the first nibble in the first round and only the power consumption *after* the register updates was considered in the analysis. Hence, only the asynchronous switching activity of the combinational path was considered to validate the presence of bit interactions, due to glitches and different propagation delay characteristics.

Table 2 summarizes the results of conducted experiments by the way of three exemplary cases, obtained by targeting the intermediate values of the S-box output over a clock cycle. It can be observed that, the information feedback provided by CPA and the SA at time t_0 is fairly equivalent, since they both point to the same leakage contributions, although CPA only loosely. A similar situation can be observed a couple of hundreds picoseconds later, at time t_1, where the issue persists only for the second contribution, but it occurs that the $\widetilde{R}^2$ is halved. In this case, the selected subspace generated from the S-box output is not able to completely capture all the contributions in the leakages. This fact can be explained by observing that the considered implementation, though serialized, actually moves all the nibbles every clock cycles through a shift register

Table 2. Simulated Power Traces

$g_{\mathsf{Ser},j}$	v_1	v_2	v_3	v_4	v_1v_2	v_1v_3	v_1v_4	v_2v_3	v_2v_4	v_3v_4	$v_1v_2v_3$	$v_1v_2v_4$	$v_1v_3v_4$	$v_2v_3v_4$	$v_1v_2v_3v_4$	$\widetilde{R}^2$
$\widetilde{\rho}_{\mathsf{Ser},j,t_0}$	**0.732**	**0.632**	**0.104**	**−0.102**	**0.779**	**0.528**	**0.338**	**0.469**	**0.272**	**0.067**	**0.607**	**0.424**	**0.355**	**0.315**	**0.424**	0.458
$\widetilde{\beta}^{\mathsf{Full}}_{\mathsf{Ser},j,t_0}$	**0.358**	**0.316**	−0.001	0.001	0.002	0.003	0.003	0.000	−0.001	0.002	−0.002	0.002	−0.006	−0.003	0.000	0.990
$\widetilde{\beta}^{\mathsf{Lin}}_{\mathsf{Ser},j,t_0}$	**0.360**	**0.315**	−0.001	0.000												0.989
$\widetilde{\rho}_{\mathsf{Ser},j,t_1}$	**−0.053**	**0.731**	**0.049**	**−0.056**	**0.380**	0.022	**−0.088**	**0.469**	**0.383**	0.022	**0.299**	**0.202**	0.014	**0.298**	**0.202**	0.110
$\widetilde{\beta}^{\mathsf{Full}}_{\mathsf{Ser},j,t_1}$	−0.004	**0.116**	−0.000	−0.003	−0.001	0.003	0.008	0.002	−0.000	0.002	−0.001	0.004	−0.009	−0.006	0.000	0.535
$\widetilde{\beta}^{\mathsf{Lin}}_{\mathsf{Ser},j,t_1}$	−0.002	**0.115**	0.000	−0.002												0.535
$\widetilde{\rho}_{\mathsf{Ser},j,t_2}$	**0.413**	**0.426**	**0.345**	**0.139**	**0.642**	**0.281**	**0.307**	**0.472**	**0.399**	**0.324**	**0.605**	**0.428**	**0.280**	**0.494**	**0.428**	0.430
$\widetilde{\beta}^{\mathsf{Full}}_{\mathsf{Ser},j,t_2}$	**0.414**	**0.266**	**0.601**	**0.092**	**−0.099**	**−0.862**	−0.003	**−0.676**	0.007	**−0.337**	**1.119**	**−0.857**	**0.446**	**0.660**	0.000	0.935
$\widetilde{\beta}^{\mathsf{Lin}}_{\mathsf{Ser},j,t_2}$	**0.205**	**0.210**	**0.127**	**0.084**												0.483

producing surrounding activity which is not modelled by the targeted S-box. Finally, at time t_3, which happens closely before the end of the combinational logic activity, the $\mathcal{F}_{16}$ subspace clearly shows several high-dimensional contributions, which would be quite difficult to appreciate from either CPA or the SA in the $\mathcal{F}_5$ subspace. The propagation of the leakage characteristic over a clock cycle towards higher-degree polynomials results fairly clear and understandable if considering that the analysed implementation is unprotected and the sampling resolution used for power estimations is fine enough to be able to measure the effects of different path delays and the glitches there of, occurring during the switching activity of the logic gates in the combinational path.

5 Conclusion

In this work, we investigated the relationship between CPA and the SA as candidate tools to identify the leakage contributions and *support* hardware designers in the correction of design flaws. First, we investigated the mathematical relationship of CPA and the SA and showed how in some specific, yet practically relevant cases, they convey the same information feedback to designers. Secondly, we analysed the results of CPA and the SA when applied to simulated data and power traces, and showed the importance of high-dimensional leakage models when it comes to pinpoint the leakage of bit interactions during the switching activity of the combinational logic path. In particular, we have shown that the SA is able to precisely quantify high-dimensional leakages given that an adequate approximation subspace is selected, whereas CPA can only loosely point to the leakage contributions of high-dimensional leakages, being inherently unable to consider dependent variables jointly. On the other hand, we have shown that the SA has some notable limitations over CPA in practice, since establishing whether a selected subspace is adequate to properly explain the leakage contributions, using the coefficient of determination as a measure of model fit, is not an easy task. In this respect, we have shown that it is not always straightforward to interpret the measure of model fit in practice, since a notably high fit generated a false positive, cf. Table 1, while a relatively low fit provided a false negative, cf. Table 2.

Additionally, contrary to CPA which has quite good convergence properties, the estimation effort of the SA grows with the size of the selected subspace, which is exponential in the size of the considered target bits. Hence, the analysis of large datapath designs with the SA might be much more laborious, than the provided exemplary analysis of a 4-bit datapath design. In this case, the use of stepwise regression as proposed in [24] can be of help to systematically evaluate different subspaces, but still liable to similar interpretation issues in the context of side-channel security verifications. To conclude, the constructive usage of CPA and the SA offers a crucial *support* to designers for the identification and correction of design flaws, but ultimately the engineering experience and expertise are still decisive to determine the success of the correction phase.

Acknowledgements. This work has been funded in part by the German Federal Ministry of Education and Research 163Y1200D (HIVE) and 01DP12037A (SMERCS). The authors would like to thank Oscar Guillen and Stefan Rass for their useful comments.

References

1. Barenghi, A., Bertoni, G., De Santis, F., Melzani, F.: On the Efficiency of Design Time Evaluation of the Resistance to Power Attacks. In: DSD, pp. 777–785. IEEE (2011)
2. Batina, L., Gierlichs, B., Prouff, E., Rivain, M., Standaert, F.X., Veyrat-Charvillon, N.: Mutual Information Analysis: A Comprehensive Study. J. Cryptology 24(2), 269–291 (2011)
3. Bhasin, S., Guilley, S., Heuser, A., Danger, J.L.: From Cryptography to Hardware: Analyzing and Protecting Embedded Xilinx BRAM for Cryptographic Applications. Journal of Cryptographic Engineering, 1–13 (2013)
4. Brier, E., Clavier, C., Olivier, F.: Correlation Power Analysis with a Leakage Model. In: Joye, M., Quisquater, J.-J. (eds.) CHES 2004. LNCS, vol. 3156, pp. 16–29. Springer, Heidelberg (2004)
5. Doget, J., Dabosville, G., Prouff, E.: A New Second Order Side Channel Attack Based on Linear Regression. Cryptology ePrint Archive, Report 2011/505 (2011)
6. Doget, J., Prouff, E., Rivain, M., Standaert, F.X.: Univariate Side Channel Attacks and Leakage Modeling. J. Cryptographic Engineering 1(2), 123–144 (2011)
7. Goodwill, G., Jun, B., Jaffe, J., Rohatgi, P.: A testing methodology for side-channel resistance validation. In: NIST Non-invasive Attack Testing Workshop (2011)
8. Heuser, A., Schindler, W., Stöttinger, M.: Revealing Side-channel Issues of Complex Circuits by Enhanced Leakage Models. In: Rosenstiel, W., Thiele, L. (eds.) DATE, pp. 1179–1184. IEEE (2012)
9. Kasper, M., Schindler, W., Stöttinger, M.: A Stochastic Method for Security Evaluation of Cryptographic FPGA Implementations. In: Bian, J., Zhou, Q., Athanas, P., Ha, Y., Zhao, K. (eds.) FPT, pp. 146–153. IEEE (2010)
10. Kocher, P.C., Jaffe, J., Jun, B.: Differential power analysis. In: Wiener, M. (ed.) CRYPTO 1999. LNCS, vol. 1666, pp. 388–397. Springer, Heidelberg (1999)
11. Lemke-Rust, K., Paar, C.: Analyzing Side Channel Leakage of Masked Implementations with Stochastic Methods. In: Biskup, J., López, J. (eds.) ESORICS 2007. LNCS, vol. 4734, pp. 454–468. Springer, Heidelberg (2007)

12. Mangard, S.: Hardware countermeasures against DPA – A statistical analysis of their effectiveness. In: Okamoto, T. (ed.) CT-RSA 2004. LNCS, vol. 2964, pp. 222–235. Springer, Heidelberg (2004)
13. Mangard, S., Oswald, E., Popp, T.: Power Analysis Attacks - Revealing the Secrets of Smart Cards. Springer (2007)
14. Mangard, S., Oswald, E., Standaert, F.X.: One for All - All for One: Unifying Standard Differential Power Analysis Attacks. IET 5(2), 100–110 (2011)
15. Mangard, S., Pramstaller, N., Oswald, E.: Successfully Attacking Masked AES Hardware Implementations. In: Rao, J.R., Sunar, B. (eds.) CHES 2005. LNCS, vol. 3659, pp. 157–171. Springer, Heidelberg (2005)
16. Mangard, S., Schramm, K.: Pinpointing the Side-Channel Leakage of Masked AES Hardware Implementations. In: Goubin, L., Matsui, M. (eds.) CHES 2006. LNCS, vol. 4249, pp. 76–90. Springer, Heidelberg (2006)
17. Mather, L., Oswald, E., Bandenburg, J., Wójcik, M.: A Comparison of Statistical Techniques for Detecting Side-Channel Information Leakage in Cryptographic Devices. Cryptology ePrint Archive, Report 2013/298 (2013)
18. Montgomery, D.C., Peck, E.A., Vining, G.G.: Introduction to Linear Regression Analysis. Wiley & Sons (2012)
19. Regazzoni, F.: A Design Flow and Evaluation Framework for DPA-resistant Embedded Systems. Ph.D. thesis, University of Lugano, Lugano, Switzerland (2010)
20. Renauld, M., Standaert, F.-X., Veyrat-Charvillon, N., Kamel, D., Flandre, D.: A Formal Study of Power Variability Issues and Side-Channel Attacks for Nanoscale Devices. In: Paterson, K.G. (ed.) EUROCRYPT 2011. LNCS, vol. 6632, pp. 109–128. Springer, Heidelberg (2011)
21. Rolfes, C., Poschmann, A., Leander, G., Paar, C.: Ultra-Lightweight Implementations for Smart Devices – Security for 1000 Gate Equivalents. In: Grimaud, G., Standaert, F.-X. (eds.) CARDIS 2008. LNCS, vol. 5189, pp. 89–103. Springer, Heidelberg (2008)
22. Schindler, W.: Advanced Stochastic Methods in Side Channel Analysis on Block Ciphers in the Presence of Masking. J. Math. Cryptology 2(3), 291–310 (2008)
23. Whitnall, C., Oswald, E.: Profiling DPA: Efficacy and Efficiency Trade-offs. Cryptology ePrint Archive, Report 2013/353 (2013)
24. Whitnall, C., Oswald, E., Standaert, F.X.: The Myth of Generic DPA . . . And the Magic of Learning. IACR Cryptology ePrint Archive 2012, 256 (2012)