Efficient Entropy Estimation for Mutual Information Analysis using B-splines

Alexandre VENELLI[1,2]

[1] IML - ERISCS Université de la Méditerranée,
Case 907, 163 Avenue de Luminy
13288 Marseille Cedex 09, FRANCE
[2] ATMEL Secure Microcontroller Solutions
Zone Industrielle, 13106 Rousset, FRANCE
alexandre.venelli@atmel.com

Abstract. The Correlation Power Analysis (CPA) is probably the most used side-channel attack because it seems to fit the power model of most standard CMOS devices and is very efficiently computed. However, the Pearson correlation coefficient used in the CPA measures only linear statistical dependences where the Mutual Information (MI) takes into account both linear and nonlinear dependences. Even if there can be simultaneously large correlation coefficients quantified by the correlation coefficient and weak dependences quantified by the MI, we can expect to get a more profound understanding about interactions from an MI Analysis (MIA). We study methods that improve the non-parametric Probability Density Functions (PDF) in the estimation of the entropies and, in particular, the use of B-spline basis functions as pdf estimators. Our results indicate an improvement of two fold in the number of required samples compared to a classic MI estimation. The B-spline smoothing technique can also be applied to the rencently introduced Cramér-von-Mises test.

1 Introduction

Side-channel analysis, and power analysis attacks in particular, are a major concern for the smart card industry. Differential Power Analysis (DPA) is one of the most known and efficient side-channel attacks. Introduced by Kocher et al. [1] in 1999, DPA exploits statistical differences in a large set of observations to deduce the secret key of the attacked algorithm. It uses a partition function to sort the set of observations into two subsets. This partition function simulates an intermediate cryptographic computation of the algorithm where parts of the secret and the plaintext are combined. Then, DPA consists in using the differences between averages of power consumption curves of the two subsets to show a peak when the attack uses a correct key guess. This statistical tool shows how different the subsets are.

In 2004, Brier et al. [2] proposed a Correlation Power Analysis (CPA), an attack using the Pearson correlation coefficient as a statistical distinguisher.

This correlation factor seems to be the most successfull in differential power analysis on standard CMOS devices. It finds the linear dependencies between power consumption curves and a leakage function based on a key guess and a plaintext value. Batina et al. [3] proposed using non-parametric tests when the dependency between the power consumption and the leakage fuction used is not so close to linear. They showed that the non-parametric Spearman rank correlation coefficient outperforms the Pearson coefficient in such case.

Recently in [4], the authors presented the use of mutual information as a distinguisher. This Mutual Information Analysis (MIA) is a more general attack. It makes no assumptions on the relation between observations and the leakage function whereas CPA just recovers the linear correlation. As most standard CMOS devices seem to follow the linear Hamming weight power model, CPA often performs better than MIA. However on special logic, e.g. Wave Dynamic Differential Logic where the assumption on the Hamming weight power model no longer holds, MIA seems to be quite efficient [4].

The MIA, as presented in [4], could perform better in term of efficiency of the results, even on standard CMOS devices. In the present paper, we introduce the use of B-splines as a tool to better estimate entropy. B-spline basis functions can be used as Probability Density Function (PDF). By construction, the B-spline estimation takes into account the measurement noise of the data. We evaluate the efficiency of this improved evaluation and demonstrate significantly better results on practical data. We also apply the B-spline technique to the recently proposed Cramér-von-Mises test [5].

Section 2 summarizes the fundamentals of information theory. Section 3 introduces the classical methods of estimating probabilities and entropies. Section 4 presents our contributions with the use of B-spline functions as estimators and how it particulary fits into the side-channel context. Experimental results are provided in Section 5 and Section 6 concludes the paper.

2 Information Theory Background

In information theory, Mutual Information (MI) is defined as a measure of mutual dependance of two variables. Unlike the linear Pearson correlation coefficient, it is sensitive also to dependencies which do not occur in covariance.

Let X be a random variable, with a finite set of M_X possible states X_i with $i \in \{1, \ldots, M_X\}$ and with a probability distribution $\mathbb{P}_X$, the Shannon entropy of X, noted $H(X)$ or $H(\mathbb{P}_X)$ is defined as:

$$H(X) = -\sum_{i=1}^{M_X} p(X_i) \log(p(X_i)), \tag{1}$$

where $p(X_i)$ is the probability of the state X_i. The Shannon entropy is a measure of how evenly the states of X are distributed.

The joint entropy $H(X,Y)$ of two random variables X and Y is analogously defined as:

$$H(X,Y) = -\sum_{i=1,j=1}^{M_X,M_Y} p(X_i,Y_j)\log(p(X_i,Y_j)), \tag{2}$$

and expresses the uncertainty one variable has about another.

The conditional entropy $H(X|Y)$ expresses the uncertainty of X given Y and is defined as:

$$H(X|Y_j) = -\sum_{i=1}^{M_X} p(X_i|Y_j)\log(p(X_i|Y_j)), \tag{3}$$

$$H(X|Y) = \sum_{j=1}^{M_Y} p(Y_j)H(X|Y_j). \tag{4}$$

The mutual information $I(X;Y)$ can then be defined as:

$$I(X;Y) = H(X) - H(X|Y), \tag{5}$$

$$\text{or} \quad I(X;Y) = H(X) + H(Y) - H(X,Y). \tag{6}$$

3 Classical Techniques for Estimating Mutual Information

There are two basic approaches to estimation, parametric and non-parametric. In this paper we restric ourselves to the non-parametric field. Parametric estimation makes assumptions about the regression function that describes the relationship between dependent variables. Therefore, the density function will assume that the data are from a known family of distributions, such as normal, and the parameters of the function are then optimized by fitting the model to the data set. Non-parametric estimation, by contrast, is a statistical method that has no meaningful associated parameters. There is often no reliable measure used for the choice of the parameters. This paper seeks to introduce efficient non-parametric PDF estimation methods in the context of side-channel analysis.

3.1 The Intuitive Histogram-Based Approach

All definitions in Section 2 involved the explicit knowledge of the probability distributions. However, in practice, these probabilities are not known and have to be estimated from measurements. The most straightforward and widely used approach is the histogram-based algorithm.

Consider a collection of N measurements of two variables X and Y. The data is partitioned into B different bins. The bins are defined through B intervals $a_i = [o + i.h, o + (i+1).h]$ where o is the value of the origin and h is the width of the bins and with $i = 0, \ldots, B-1$. We note k_i the number of mesurements

that lie in the interval a_i. The probabilities $p(a_i)$ are then approximated by the corresponding relative frequencies of occurence:

$$p(a_i) = \frac{k_i}{N}.$$

From these approximated probabilities, we calculate the entropies and finally the mutual information. The choice of the number of bins B is critical. It plays the role of smoothing parameter (see Fig. 1). The number of bins determines two things: how good the statistics will be reflecting the ideal, continuous distribution and how close the partitioning will be to the actual physical data-dependency of the device. Histogram PDF estimation is very computationally efficient, however it can give approximate results.

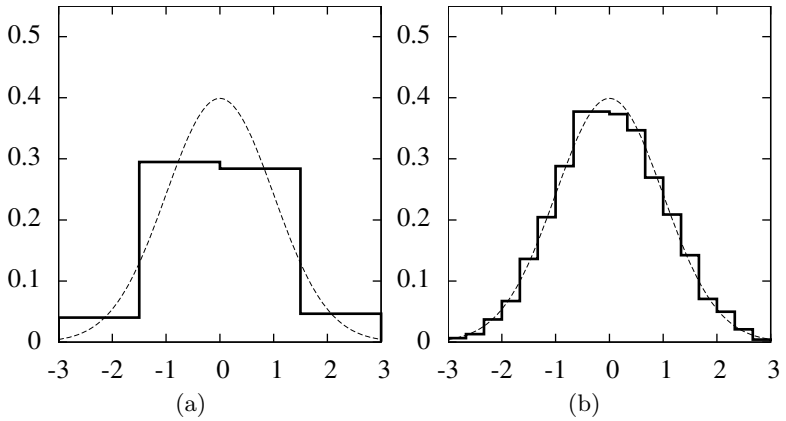

Fig. 1: Effect of the number of bins on how close the estimation is to the actual distribution. The dotted line in these figures is a Gaussian distribution, the solid line is the estimation. Figure 1a shows an estimation with 4 bins. Figure 1b is an estimation with 18 bins.

3.2 Kernel Density Estimation

There exists alternatives to the histogram-based approach. We introduce the Kernel Density Estimation (KDE) [6], also called Parzen window method [7]. Kernel techniques assume that the probability density is smooth enough such that structure below a certain kernel bandwidth can be ignored. The kernels essentially weight the distances of each point in the sample to a reference point depending on the form of the kernel function and according to a given bandwidth h. The simplest possibility is to estimate the density at a point x by the number of points in a box centered at x of size h divided by its volume. Rather than

simply counting the points, kernel functions are used to give them distance-dependant weights. We obtain a naive estimator $f(x)$ that aims at improving the estimate of the probability $p(x)$:

$$f(x) = \frac{1}{2Nh} \sum_{i=1}^{N} \Theta(h - |x - x_i|), \tag{7}$$

where Θ denotes the Heaviside function defined as:

$$\Theta(z) = \begin{cases} 1 & \text{if} \quad z > 0 \\ 0 & \text{if} \quad z \leq 0. \end{cases} \tag{8}$$

For a more general definition, we note $K(x)$ as the kernel function. We then define the kernel density estimor $f(x)$ as

$$f(x) = \frac{1}{Nh} \sum_{i=1}^{N} K\left(\frac{x - x_i}{h}\right). \tag{9}$$

An example of kernel function K often used is the Gaussian kernel, the density estimator is then defined as

$$f(x) = \frac{1}{Nh\sqrt{2\pi}} \sum_{i=1}^{N} \exp\left(-\frac{(x - x_i)^2}{2h^2}\right). \tag{10}$$

The Gaussian estimator can be seen as placing small Gaussian 'bumps' at each observation point x_i. The estimation is then the sum of the 'bumps'.

Instead of the critical choice of bins in the histogram approach, the choice of bandwidth h is now crucial in kernel density estimation. If h is too large, the estimate suffers from too little resolution, whereas if it is too small, the estimate suffers from too much statistical variability (Fig. 2).

Even if KDE estimation is more accurate than the histogram-based approach, it suffers from heavy computational requirements. Recently, Prouff and Rivain [8] presented a parametric estimation of entropy that seems as efficient as the CPA when the noise is increasing. We rather focus on non-parametric methods and in particuliar, a more balanced method between accuracy and efficiency. Using B-splines basis functions, we can greatly improve the histogram-based estimation with an acceptable computational overhead.

4 Improving MIA results using B-spline Functions

4.1 Introduction to Piecewise Polynomials and Splines

Let X be a one-dimensional variable. A piecewise polynomial function $f(X)$ is obtained by dividing the domain of X into contiguous intervals, and representing f by a separate polynomial in each interval. Figures 3a and 3b show simple piecewise polynomials. However, we often prefer smoother functions that can

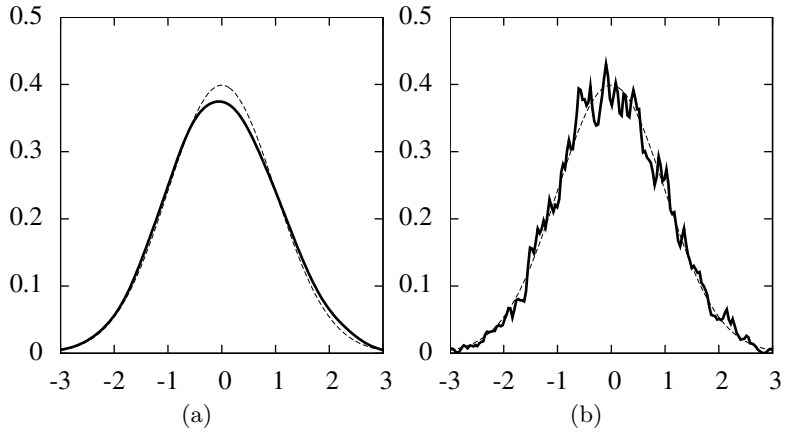

Fig. 2: Kernel density estimation using the Heaviside function with different bandwidth. The dotted line in these figures is a Gaussian distribution, the solid line is the estimation. Figure 2a shows an estimation with a bandwith $h = 0.3$. Figure 2b is an estimation with bandwidth $h = 0.03$.

be obtained by increasing the order of the local polynomial. Figure 3c shows a piecewise-cubic polynomial fit, it is known as a cubic spline. More generally, an order k spline with knots t_i, $i = 0, \ldots, m$ is a piecewise polynomial of order k and has continuous derivatives up to order $k-2$. A cubic spline has order $k = 4$. In fact, Fig. 3a is an order 1 spline and Fig. 3b is an order 2 spline.

4.2 Computation of B-splines

We briefly introduce B-splines which are generalizations of Bézier curves. One can look at [9] for more background on splines.

A B-spline curve defined over an interval $[a, b]$ is specified by:

- the degree d (or order $k = d + 1$), so that each segment of the piecewise polynomial curve has degree d or less,
- a sequence of $m + 1$ numbers, $t_0, \ldots, t_m$, called knot vector, such that $t_i \leq t_{i+1}, \forall i \in \{1, \ldots, m-1\}$,
- control points, $b_0, \ldots, b_n$.

A B-spline curve is defined in terms of B-spline basis functions. The i-th basis function of degree d, noted $B_{i,d}$, defined by the knot vector $t_0, \ldots, t_m$ is defined by the Cox-de Boor recursion formula as follows:

$$B_{i,0}(z) = \begin{cases} 1 & \text{if } t_i \leq z < t_{i+1} \\ 0 & \text{otherwise.} \end{cases} \tag{11}$$

$$B_{i,d}(z) = \frac{z - t_i}{t_{i+d} - t_i} B_{i,d-1}(z) + \frac{t_{i+d+1} - z}{t_{i+d+1} - t_{i+1}} B_{i+1,d-1}(z), \tag{12}$$

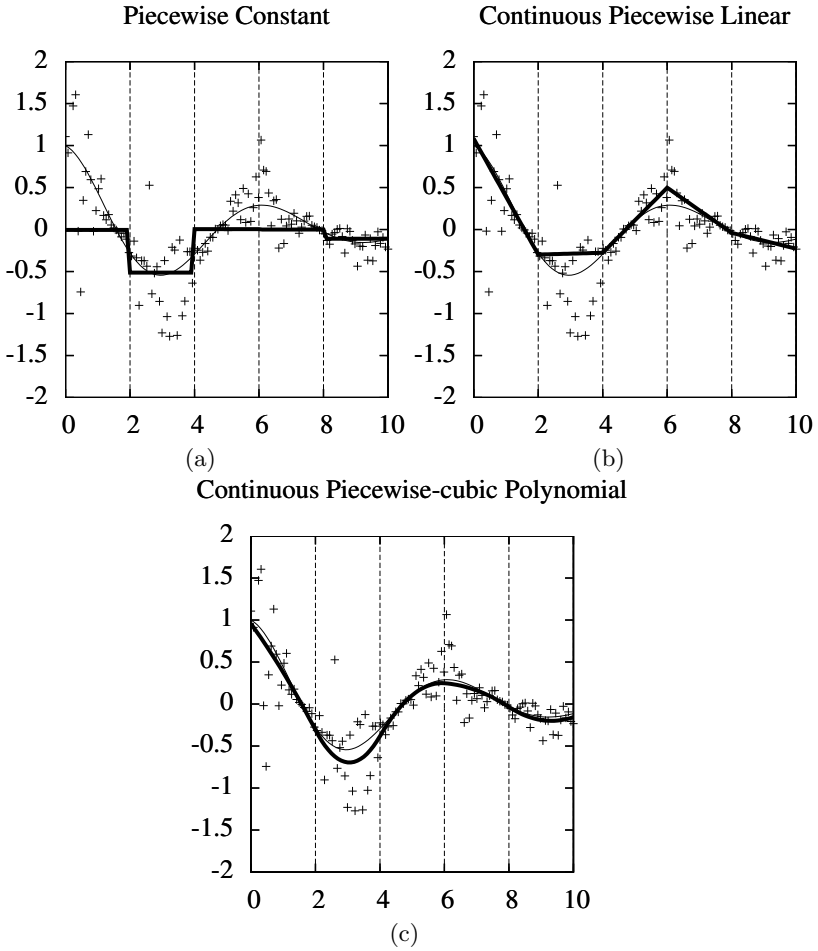

Fig. 3: In each figure, the dotted lines represent the positions of the knots. The thin line is the true function $y(x) = \cos(x)\exp(-x/5)$. The crosses are data generated from the function $y(x)$ with random gaussian noise added. The thick line represents the estimation.

for $i = 0, \ldots, n$ and $d \geq 1$.

The B-spline curve of degree d with control points $b_0, \ldots, b_n$, knots $t_0, \ldots, t_m$ is then defined by:

$$B(z) = \sum_{i=0}^{n} b_i B_{i,d}(z),$$

where $B_{i,d}(z)$ is the B-spline basis function previously defined.

From (12), it is clear that $B_{i,d}(z)$ is non-zero in the interval $[t_i, t_{i+d+1}]$. For example, a cubic B-spline basis function $B_{i,3}(z)$ is non-zero in the interval $[t_i, t_{i+4}]$. This basis function spans the knots $t_i, t_{i+1}, t_{i+2}, t_{i+3}, t_{i+4}$. We also note that, when knots are not repeated, a B-spline is zero at the end-knots t_i and t_{i+d+1}. But B-splines can use repeated knots. If the knot vector contains a sufficient number of repeated knot values, then a division of the form $0/0$ may occur, it is then assumed that $0/0 := 0$. Finally, the property that is essential to this study is:

$$\sum_{i=0}^{n} B_{i,d}(z) = 1,$$

for any value of z. We can adapt easily B-spline basis functions to be a probability distribution function.

4.3 PDF Estimation using B-spline Functions

In [10], authors compare their method with other MI estimators. They show that B-splines offer an increase by roughly two-fold in significance of the MI compared to a simple binning method on an artificially generated dataset. They also noted that using high spline orders doesn't give much better results than a $k = 2$ or $k = 3$ order. As we found the same conclusion with our experiments, we use in the rest of the paper order $k = 3$.

The main problem in a naive histogram approach is that each data point is assigned to only one bin. We loose the information of points near neighbouring bins that could be in either bins depending on the measurement noise for example. The idea of [10] is to allow a data point to be in simultaneously k different neighbouring bins using B-splines basis functions.

We want to imitate the histogram approach replacing the naive binning of the interval of values with a more elaborate partition of the interval using B-spline basis functions. In classic binning as in B-spline estimation, the abscissa axis is broken up into some numbers of intervals, where endpoints of each interval are called breakpoints. To change the shape of a B-spline curve, we already noticed that one can modify: the degree of the curve, the control points or the knot vector. The number of breakpoints is linked to those previously defined values with the formula: $nbreak = n - k + 2$ where n is the number of control points and k is the order of the spline. The B-spline order is generally fixed beforehand, for optimal results we fix $k = 3$ as previously stated. We then have to modify the knot vector and the number of breakpoints so that B-spline functions can act as correct partitions. A practical example of how the parameters are fixed for an

attack on DES is explained in the next Section. In general, B-spline curves are not tangent to the first and last knots. In our case we want to clamp the curve to these extremities. We want the basis functions to fully cover the interval of values. To do so, we have to repeat the first and last knot value $d+1$ times in the knot vector.

B-spline curves that fit the properties previously stated are called open B-spline curves. We construct it with a type of knot vector called uniform non-periodic knots. We use this type of construction for our application to the MIA. First, let's define uniform non-periodic knot vector:

Uniform non-periodic knot vector. Let $B_{i,d}(z)$ be a B-spline of degree d (order $k = d+1$) for $i = 0, \ldots, n$ and $z \in [0, n-k+2]$. We define the knot vector $t_0, \ldots, t_{n+k}$ as follows:

$$t_i = \begin{cases} 0 & \text{if } 0 \leq i < k \\ i-k+1 & \text{if } k \leq i \leq n \\ n-k+2 & \text{if } n < i \leq n+k. \end{cases}$$

For example the uniform non-periodic knot vector for $n = 5$ and $k = 3$ is $[0, 0, 0, 1, 2, 3, 4, 4, 4]$. In general, this type of knot vector has the structure:

$$\underbrace{0, \ldots, 0}_{k \text{ knots}}, 1, 2, \ldots, n-k-1, \underbrace{n-k+2, \ldots, n-k+2}_{k \text{ knots}}.$$

4.4 B-Spline Estimation in the Side-Channel Context

In the PDF estimation context, there is a clear similarity between B-spline estimation and the histogram method as order 1 B-splines are in fact step functions (Fig. 3a). Indeed, instead of affecting a point to only one bin, i.e. one interval, we can spread the same point onto a wider interval with B-spline functions. The higher the degree d of the spline, the wider the considered interval will be. This is particulary interesting in the side-channel context. Each point of the power consumption curve has measurement noise attached to it. This noise could shift points to false neighbouring bins in the classic histogram. B-spline estimation affects a weight to each point so that it covers a larger interval of possible values covering its possible attached noise.

Furthermore, each point is weighted by a curve on an interval when in the histogram it is weighted by a simple step function. With this property, B-spline estimation seems similar to a KDE approach while being more simple, hence less computationally intensive. This claim is adressed in Section 5. B-spline PDF estimation is a good compromise between the time efficient but naive histogram and the complex KDE estimation.

Example of use of B-spline estimation to attack a DES implementation. Using notations from [11], let H be a hypothetical function of the intermediate values targeted by the attacker. The function H is often a surjection from the value

space $\mathcal{V}$ into a hypothetical leakage space $\mathcal{H}$. Let's consider a leakage vector and its partition into B sets. For example, suppose the intermediate value targeted in a DES is the three most significant bits of $SBox(x \oplus k)$. It would seem natural to have $B = 8$ partitions in a classic histogram approach to place the targeted values that range from 0 to 7. If we consider B-spline estimation, we also want to cover the range of the targeted values $[0, 7]$. Recall that B-spline functions are defined on $[0, n - k + 2]$ and using uniform non-periodic knot vector the functions are clamped on the extremities. The parameter k is generally fixed at $k = 2$ or $k = 3$ so that the calculation of B-spline functions is not too complex while the curves are smooth enough [10]. The number of breakpoints $nbreak = n-k+2$ corresponds to B in the classic binning. For our example, with $k = 3$ and $nbreak = 8$, we have $n = nbreak + k - 2 = 9$ basis functions. Hence, we only modify the parameters k and $nbreak$, the number of basis functions n is inferred.

Algorithm computing MI using B-spline PDF estimation. The algorithm to estimate the mutual information using B-spline PDF estimation between two random variables X and Y is as follows [10]:

- **Input:** random variables $X = \{x_1, \ldots, x_N\}$ and $Y = \{y_1, \ldots, y_N\}$, spline order: k, n_X the number of B-spline basis functions for X and n_Y for Y.
- **Output:** $I(X; Y)$.

1. Estimate the entropy of X.
   (a) Determine the n_X Weighting Coefficients (WC) for each $x_u, u = 1, \ldots, N$ as $B_{i,d}(x), i = 1, \ldots, n_X$. Save the matrix $MatrixWC_X$ of $(n_X \times N)$ values containing all the weighting coefficients.
   $$MatrixWC_X[i][u] = B_{i,d}(x_u).$$
   (b) Compute the n_X probabilities $p(a_i)$, $i = 1, \ldots, n_X$:
   $$p(a_i) = \frac{1}{N} \sum_{u=1}^{N} B_{i,d}(x_u).$$
   (c) Compute the entropy (1):
   $$H(X) = -\sum_{i=1}^{n_X} p(a_i) \log(p(a_i)).$$
2. Repeat step 1 for the variable Y to obtain the matrix $MatrixWC_Y$ and the entropy $H(Y)$.
3. Determine the joint probability $p(a_i, b_j)$ for all $(n_X \times n_Y)$ bins:
   $$p(a_i, b_j) = \tfrac{1}{N} \textstyle\sum_{u=1}^{N} (B_{i,k}(x_u).B_{j,k}(y_u))$$
   $$= \tfrac{1}{N} \textstyle\sum_{u=1}^{N} (MatrixWC_X[i][u].MatrixWC_Y[j][u]).$$
4. Calculate the joint entropy $H(X, Y)$ (2).
5. Compute the mutual information (6).

4.5 Combining Cramér-von-Mises Test with B-spline Smoothing

The Cramér-von-Mises (CvM) test is similar to the Kolmogorov-Smirnov (KS) test. The KS statistic is a widely used non-parametric statistical test. The two-sample KS test evaluates the maximal difference between two empirical cumulative distribution functions. The two-sample KS test can also be compared to the non-parametric Mann-Whitney test that is the non-parametric equivalent to the T-test used in DPA. The Mann-Whitney test measures the difference in central tendency between two distributions whereas the KS test seems sensitive to any kind of distributional difference.

We first briefly introduce the principle of the two-sample KS test. Let two samples X_i and Y_j with size n and m. The samples can be characterized by their empirical cumulative density functions:

$$\mathrm{cdf}_X = \frac{\#i : X_i \leq x}{n} \quad \text{and} \quad \mathrm{cdf}_Y = \frac{\#j : Y_j \leq x}{m},$$

that correspond to the proportion of observed values inferior or equal to x. Then, the two-sample KS test is defined as:

$$D_{KS}(X||Y) = \max_x(|\mathrm{cdf}_X - \mathrm{cdf}_Y|).$$

The Cramér-von-Mises test is an alternative to the Kolmogorov-Smirnov test. It is also based on the empirical cumulative density functions but it is defined as:

$$D_{CvM}(X||Y) = \sum_x (\mathrm{cdf}_X - \mathrm{cdf}_Y)^2.$$

In [5], the authors introduce a MIA-inspired distinguisher based on the CvM test and show its efficiency compared to other MIA-like side-channel attacks.

In practice, the empirical cumulative functions are constructed based on histograms. The CvM test does not costs much more than a classical histogram estimation of PDF and is therefore very interesting. The B-spline method previously introduced can also be applied in this context of cumulative function estimations in a similar way. The different values of the samples are affected to more than one bin with an apprioriate weight given by the B-spline functions. Once this smoothed histogram created, the cumulative functions and the CvM test can be computed as originaly. The improvement due to the B-splines is not as advantageous as previously noted with probability density functions. Indeed, the histogram smoothing is less significant in the computation of cumulative density functions than classical PDF. However subtle, the improvement is still noticeable in certain cases.

In the next section, we demonstrate with practical data the two-fold increase of the B-spline estimation compared to an histogram method. Furthermore, this technique adjusts particularly well to side-channel analysis. Indeed, allowing a point of a power consumption curve to be in k different neighbouring bins compensate the measurement noise that might shift the point to a different bin. This observation is demonstrated on practical data sets in the following.

5 Experimental Results

As previously stated, we restrict ourselves to the comparision of the efficiency of non-parametric estimation in the MIA context. We carried out attacks on two different kinds of setups and two different algorithms. For each setup, we measure the efficiency of the attacks using known metrics introduced in the literature:

- the guessed entropy [11] that is the average position of the correct hypothesis in the sorted hypothesis vector of an attack,
- and the first order success rate [11] that is, given a number of traces, the probability that the correct hypothesis is the first best hypothesis of an attack.

We compare the classical MIA attack using histogram estimation [4] (simply noted MIA in the figures), MI estimation using B-spline smoothing of Section 4.4 (noted MIB), the Cramér-von-Mises test [5] (noted CVM), this same test with B-spline smoothing of Section 4.5 (noted CVMB) and finally, as a reference, the CPA [2].

First, we briefly analyse the computational efficiency of these attacks in Fig. 4. The time measurements are recorded on a classical workstation computer with Pentium 4 processor. We remark that, even if the MIB requieres more computational time than a classical MIA, it is clearly more effective than a KDE analysis.

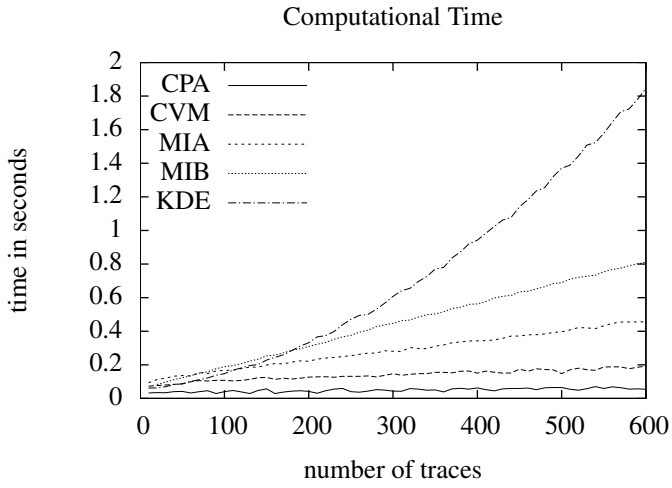

Fig. 4: Comparison of the necessary computational time per byte for each attack on a DES implementation.

Our first set of attacks (Fig. 5) are made using the publicly available traces of the DPA Contest [12]. We remark that the two-fold performance increase of the

MIB is clear compared to the MIA. The closely tied CVM and CVMB perform relatively better. However, their results are still far from the CPA.

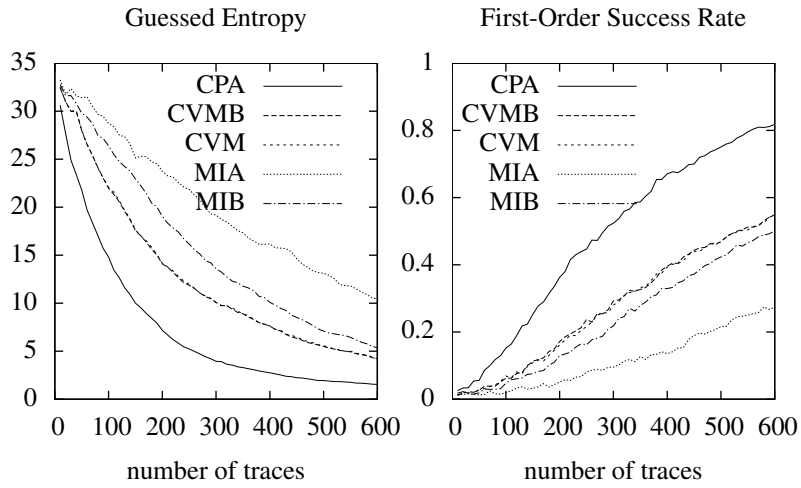

Fig. 5: Comparative results of attacks using the DPA Contest traces implementing a DES.

We then tested the efficiency of the attacks on a different platform and with a different attacked algorithm. We implemented on an Atmel STK600 board with an Atmel AVR ATmega2561 [13] a well-known multi-precision multiplication algorithm using Comba's method [14]. The attacker's goal is to find the bytes of a fixed secret multiplicand, while several random publicly-known multiplicand of the same size are used as input. The major difference between this setup and the one of DPA Contest is that the Atmel STK600 board is not particularly suited for side-channel measurement. Therefore, the power traces contain a lot more noise. The results in this context are particularly interesting (Fig. 6). First, we can observe as previously an increase by roughly two-fold of the MIB efficiency compared to the MIA. The results of the CVM and CVMB and now relatively close the MIB. However, more importantly, we note that the MI-based attacks using the B-spline technique perform globally much better compared to the CPA. In particular, the guessed entropy criterion indicates that the MIB, CVM and CVMB are more efficient than the CPA in this scenario.

6 Conclusion

We present in this paper efficient PDF estimation techniques using B-splines in the side-channel analysis context. The B-spline estimation fits very well as it takes into account the possible measurement noise that can be attached to a data

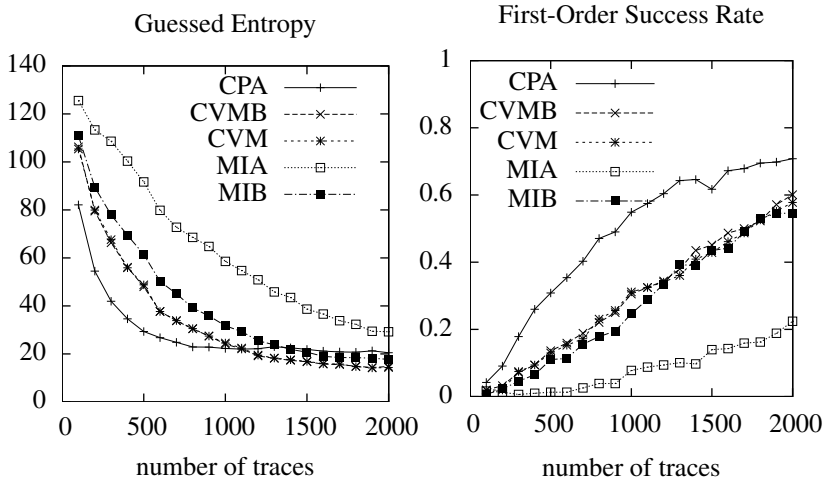

Fig. 6: Comparative results of attacks using traces acquired on an Atmel STK600 board with an Atmel AVR ATmega2561 [13] that implements a multi-precision multiplication algorithm.

point. This is particularly well demonstrated with our comparative analysis in a noisy environment that proved the effect of B-spline smoothing. We even obtain better than the powerful CPA in this scenario. Further research on this topic can include the investigation of other PDF estimators, for example: the parametric edgeworth-based entropy estimation [15], the non-parametric wavelet estimation [16] or the non-parametric method using the k-nearest neighbors algorithm [17].

References

1. Kocher, P., Jaffe, J., Jun, B.: Differential power analysis. CRYPTO 1999, LNCS **1666** (1999) 388–397
2. Brier, E., Clavier, C., Olivier, F.: Correlation power analysis with a leakage model. CHES 2004, LNCS **3156** (2004) 135–152
3. Batina, L., Gierlichs, B., Lemke-Rust, K.: Comparative evaluation of rank correlation based DPA on an AES prototype chip. ISC 2008, LNCS **5222** (2008) 341–354
4. Gierlichs, B., Batina, L., Tuyls, P., Preneel, B.: Mutual information analysis - a generic side-channel distinguisher. CHES 2008, LNCS **5154** (2008) 426–442
5. Veyrat-Charvillon, N., Standaert, F.: Mutual information analysis: How, when and why? CHES 2009, LNCS **5747** (2009) 429–443
6. Moon, Y.I., Rajagopalan, B., Lall, U.: Estimation of mutual information using kernel density estimators. Physical Review E **52**(3) (1995) 2318–2321
7. Parzen, E.: On the estimation of a probability density function and mode. Annals of Mathematical Statistics **33** (1962) 1065–1076
8. Prouff, E., Rivain, M.: Theorical and practical aspects of mutual information based side channel analysis. ACNS 2009, LNCS **5536** (2009) 499–518

9. Deboor, C.: A Practical Guide to Splines. Springer-Verlag Berlin and Heidelberg GmbH & Co. K (December 1978)
10. Daub, C., Steuer, R., Selbig, J., Kloska, S.: Estimating mutual information using B-spline functions - an improved similarity measure for analysing gene expression data. BMC Bioinformatics **5** (2004) 118
11. Standaert, F.X., Gierlichs, B., Verbauwhede, I.: Partition vs . comparison side-channel distinguishers: An empirical evaluation of statistical tests for univariate side-channel attacks against two unprotected CMOS devices. ICISC 2008, LNCS **5461** (2008) 253–267
12. VLSI research group and TELECOM ParisTech: The DPA contest 2008/2009 `http://www.dpacontest.org`.
13. Atmel: ATmega 2561 Data Sheet `http://www.atmel.com/dyn/resources/prod_documents/doc2549.pdf`.
14. Comba, P.G.: Exponentiation cryptosystems on the IBM PC. IBM Syst. J. **29** (1990) 526–538
15. Van Hulle, M.: Multivariate edgeworth-based entropy estimation. In: Machine Learning for Signal Processing, 2005. (2005) 311–316
16. Vannucci, M.: Nonparametric density estimation using wavelets. ISDS, Duke University, Tech. Rep. DP95-26, September 1995, available at http://www.isds.duke.edu (1995)
17. Kraskov, A., Stogbauer, H., Grassberger, P.: Estimating mutual information. Physical Review E **69** (2004) 66138