Generalized Correlation Power Analysis

Sébastien Aumônier

Oberthur Card Systems SA,
71-73, rue des Hautes Pâtures
92726 Nanterre, France
s.aumonier@oberthurcs.com

Abstract. The correlation power attack [4] is a widespread attack used to recover secret data based on leakages of a device. An enhancement of this method was proposed in [5] which improves the CPA by restricting normalization factor. But these attacks only focus on the linear relationship between the leakages and the hamming distance model. In this article, we propose a generalization of the correlation attack which does not restrict on the linearity but also takes into account the non linear relationships. Our attack uses the theory of nonparametric statistics [10] which enables us to solve estimation problems.
Keywords: CPA, DPA, Side Channel Attacks, Power Analyis, Mutual Information, Non parametric Statistics

1 Introduction

Most smart card components are based on the CMOS logic. The power consumption characteristics of CMOS circuits can be summarized shortly as follows. Whenever a circuit is clocked, the circuits gates change their states simultaneously. This leads to a charging and discharging of the internal capacitors and this in turn results in a current flow which is measurable at the outside of the device. Such measurements can be conducted easily. One needs either a data acquisition card or a digital oscilloscope to acquire the measurements. The current flow can be measured directly with a current probe, or by putting a small resistor in series with the ground input or power input of the device. By using statistical properties and leaked data from devices, Kocher, Jaffe and Jun [6] were able to recover secret keys which were believed to be sealed in a secure environment. They named their attack DPA which is an acronym for Differential Power Analysis. Since that moment, numerous countermeasures have been invented and the attacks also have been improved. The Correlation power analysis attack [4] which can be viewed as a multi-bit DPA method focuses on the linear relationship between the consumption curves and the hamming weight model. Experiments [4] show that this attack can not

work on all the components. That's why we are interested in non linear relationship between the consumption power leakages and the hamming weight that should give better results when you process them by using another statistical tool based on the notion of Mutual Information (noted MI in the following). We do not investigate in this article the copula functions [11] which can also capture non linear relationships among variables. But this should be done in the future. The model which is commonly used to describe the power consumption in a smart card is the following:
Let's R be a buffer (for example of 8 bits) with an assigned value a and let's b be the result in that buffer after a computation. The power consumption used to pass from the value a to the value b in the buffer R can be represented by the following model:

$$C(t)^{(a,b)} = \lambda HW(a \oplus b) + \beta_t$$

where HW is a hamming weight function, λ is the power consumption used to switch a bit from 0 to 1 as from 1 to 0 and β_t is a white noise (its distribution follows a normal law $N(0,1)$).

This model is used to perform a CPA attack. But in order to obtain better results two ways can be investigate :

- either you modify the model and you apply the CPA with this new model (because the relationship between the leakages and the model would be linear). In this case the consumption must be represented as follows $C(t)^{(a,b)} = F(a,b)$ where F is a function to make explicit.
- either you do not search to explicit F and in that case you must use statistical tools which are able to quantify both linear and non linear relationships in order to recover secret data

In this article we will focus on the second strategy; in the next section we will briefly make a review on the notion of Mutual Information and we will develop the tools needed to set up the Generalized Correlation Power Analysis.
Notation:
Let C_i, be a power curve. We can represent C_i by the vector $[C_i(1)....C_i(T)]$ where T is the sampling rate.
pdf: probability density function.

2 Mutual Information

Mutual Information (MI) is a measure of general dependance (both linear and non linear) between random variables X and Y. This concept was

developed in communication theory and cross domains. Considering two random variables X and Y, the MI, denoted by $I(X,Y)$, is defined as

$$I(X,Y) = H(Y) - H(Y|X) = H(X) + H(Y) - H(X,Y),$$

where $H(X)$ or $H(Y)$ is the marginal information entropy which measures the information content in a signal and $H(X,Y)$ is the joint information entropy which measures the information content in a joint system X and Y. The MI between two random variables X and Y can also be defined as :

$$I(X,Y) = \int_Y \int_X \rho_{XY}(x,y) log \frac{\rho_{XY}(x,y)}{\rho_X(x)\rho_Y(y)} dxdy$$

where $\rho_{XY}(x,y)$ is the joint probability density function (pdf) between X and Y, and $\rho_X(x)$ and $\rho_Y(y)$ are the marginal *pdfs*.
A comparison of MI based dependence with traditional measures of dependence, such as Pearson linear correlation coefficient, Spearman rank order correlation and Kendall's tau is done in [7]. MI seems to be a good tool for recovering secret information as it provides information about both linear and non linear dependencies between two sets of data.

3 The Linear Correlation Coefficient and the Mutual Information based non linear Correlation Coefficient

In this section we remind the basis of the CPA; this attack uses an estimator of the Pearson linear correlation coefficient, r, defined as follows for two random variables X and Y:

$$r(X,Y) = \frac{cov(X,Y)}{\sigma_X \sigma_Y} = \frac{E(XY) - E(X)E(Y)}{E((X-m_1)^2)E((Y-m_2)^2)}$$

The Pearson correlation coefficient can take values from -1 to $+1$. A value of $+1$ shows that the variables are perfectly linear related by an increasing relationship, a value of -1 shows that the variables are perfectly linear related by a decreasing relationship, and a value of 0 shows that the variables are not linear related by each other. There is considered a strong correlation if the correlation coefficient is greater than 0.8 and a

weak correlation if the correlation coefficient is less than 0.5. When you look at the result of the CPA made in [4] on a 32 bit implementation of a DES algorithm, the correlation rate is not high on boxes 5 to 8 (it is around 0.5). The justification of the authors for this bad result is because of partial and imperfect modelling. To overcome this problem, we must look at non linear relationship between the leakages and the hamming distance model. To do that we will use the Granger coefficient which is a stronger tool than the Pearson linear correlation coefficient. Indeed while the correlation coefficient quantifies the dependance purely in terms of the linear information content, the Granger coefficient, λ, quantifies the complete (linear and non linear) information content. It is defined as follows:

$$\lambda(X,Y) = \sqrt{1 - exp(-2 * I(X,Y))}$$

Remark that λ ranges from 0 to 1; for the Generalized Correlation attack we must estimate that coefficient in the following way

$$\widehat{\lambda}(X,Y) = \sqrt{1 - exp(-2 * \widehat{I}(X,Y))}$$

where $\widehat{I}(X,Y)$ is the estimated MI between X and Y.
Henceforth in order to compute this coefficient, we must use a good estimator of the mutual information. The next section focuses on this subject.

4 The MI estimation methods

The estimation of the MI is not an easy problem : it requires the estimation of the joint and marginal *pdf*; the most widespread technique is to approximate the MI integral by binning the coordinate axes and counting the number of data points per bin, which is computationally intensive and prone to systematic errors. Indeed histograms are not the good tool for the estimation in our problematic; histograms are not smooth, depend on the width of the bins and the end points of the bins.
To overcome these problems several methodologies have been explored such as

- Kernel density estimators (KDE) [1]
- k-nearest neighbors (KNN) [2]
- Edgeworth approximation of differential entropy [3]
- Wavelet based density estimation

– Finite mixtures

We will describe some of these methods in our problematic; further work must be done in order to classify these methods.
The estimation of the MI is the following: Let X be the random variable $\widehat{C}(t)$ (the consumption at time t). Let Y be $HW(\widehat{M} \oplus K_j)$ where $\widehat{M}$ is a random variable controlled by the attacker (the messages). It mustn't follow a uniform law otherwise we can't extract information on K_j. Hence in the general case we have:

$$\widehat{I}(X,Y) = \int_{-\infty}^{+\infty} \sum_{i=0}^{8} \widehat{\rho}_{X,Y}(x,i) log \frac{\widehat{\rho}_{X,Y}(x,i)}{\widehat{\rho}_X(x)\widehat{\rho}_Y(i)} dx \tag{1}$$

where $\widehat{\rho}_{XY}(x,i)$ is the estimated *joint pdf*, and $\widehat{\rho}_X(x)$ and $\widehat{\rho}_Y(i)$ are the estimated marginal *pdfs* at (x,i). Hence the MI estimates are obtained by first estimating the different *pdfs*.
In order to force the structure and the shape of the *pdf*, we use specific messages; for a supposition K_j, you must make $\frac{N}{2}$ acquisitions by sending the message M_1 such that $HW(M_1 \oplus K_j) = 0$ and make $\frac{N}{2}$ acquisitions by sending the message M_2 such that $HW(M_2 \oplus K_j) = 8$. Obviously the drawback of this method is that you need to perform $N * 2^s$ acquisitions where s is the size of K_j in bits. Now Equation (1) becomes (in the case of the good guessing of the key):

$$\widehat{I}(X,Y) = \int_{-\infty}^{+\infty} (\widehat{\rho}_{X,Y}(x,0) log \frac{\widehat{\rho}_{X,Y}(x,0)}{\widehat{\rho}_X(x)\frac{1}{2}} + \widehat{\rho}_{X,Y}(x,8) log \frac{\widehat{\rho}_{X,Y}(x,8)}{\widehat{\rho}_X(x)\frac{1}{2}}) dx$$

At each time t, you must compute this value of $\widehat{I}(X,Y)$; when the good guessing on the key will be done, the estimated MI will be maximal at the time t_0 when the key is manipulated (Meaning that a relation either linear or not exists between the consumption and the hamming weight model)

4.1 Kernel Density Estimator

Kernel density estimation (or Parzen window method, named after Emanuel Parzen) is a way of estimating the probability density function of a random variable. In our case we don't need to estimate the *pdf* of the variable Y which is the hamming weight of the xor operation between the guessed key and part of the message. But the *pdf* of the variable X and the joint distribution must be estimated : as we get the power curves C_i (a sample

of a random variable) then the kernel density approximation of its density function at time t is:

$$\widehat{\rho}_X(x) = \frac{1}{Nh}\sum_{i=1}^{N} G_1\left(\frac{x - C_i(t)}{h}\right)$$

where G_1 is some kernel function (see annex A), h is a bandwith (smoothing parameter).
The kernel density estimation of the joint *pdf* is given by:

$$\widehat{\rho}_{X,Y}(x,y) = \frac{1}{Nh_xh_y}\sum_{i=1}^{N} G_2\left(\frac{x - C_i(t)}{h_x}, \frac{y - y_i}{h_y}\right)$$

where h_x and h_y are the smoothing parameters and y_i hamming weight equals to 0 or 8 (depending of the sample) and G_2 is a kernel function with 2 parameters (see annex A).

Notice that when h, h_x, h_y are small, we get a lot of noise or spurious structure in the estimate. On the other side, when these elements are larger, we get a smoother estimate, but there is the possibility that we might obscure bumps or other interesting structure in the estimate. In practice, it is recommended that the analyst examines kernel density estimates for different window widths to explore the data and to search for structures such as nodes or bumps.
The method used to choose the optimal bandwidth is to minimise the optimality criterion AMISE (Asymptotic Mean Integrated Squared Error). In general, the AMISE still depends of the true density (which of course we don't have) and so we need to estimate the AMISE from our data as well. This means that the chosen bandwidth is an estimate of an asymptotic approximation.

4.2 Nearest Neighbor Method

The basic idea of this method is to control the degree of smoothing in the density estimate based on the size of a box required to contain a given number of observations. The size of the box is controlled by using an integer k that is smaller than the sample size (a typical choice would be $k \approx N^{\frac{1}{2}}$). Suppose we have the *ordered* data sample $(x_{(1)}, ..., x_{(N)})$). For any point x on the line we define the distance between x and the points on the sample by

$$d_i(x) = |x_i - x|$$

so that

$$d_1(x) \leq d_2(x) \leq ... \leq d_N(x)$$

Then the kth nearest neighbor density estimation is defined as follows :

$$\widehat{\rho}_X(x) = \frac{k-1}{2 * N * d_k(x)}$$

The problem with this estimator is that the integral of $\widehat{p}_X(x)$ is infinite; but this can be fixed by using the generalized kth nearest neighbor density estimate:

$$\widehat{\rho}_X(x) = \frac{1}{N * d_k(x)} \sum_{k=1}^{N} G_1(\frac{x - C_k(t)}{d_k(x)})$$

In fact this is just a kernel estimate evaluated at x with window width $d_k(t)$. Overall smoothing is controlled by choice of k, with the window width at any specific point depending on the density of points surrounding it. However, the main drawback of this method is the high computational cost associated with the search for the nearest neighbors.

4.3 Edgeworth Approximation

In [3] an estimation of the mutual information based on Edgeworth approximation is introduced; it seems that this method can be more accurate than the nearest neighbor method; more studies on this topic should be conducted.

4.4 Wavelet based density estimation

In statistics, amongst other applications, wavelets [9] have been used to build suitable non parametric density estimators.What do wavelet estimators have to offer in comparison with more classical estimators of the same type? A major drawback of classical series estimators is that they appear to be poor in estimating local properties of the density. This is due to the fact that orthogonal systems, like the Fourier one, have poor time/frequency localization properties. On the contrary, as previously pointed out, wavelets are localized both in time and in frequency. This makes wavelet estimators well able to capture local features. Indeed recently it has been shown that kernel density estimations tend to be inferior to wavelet-based density estimates [8]. Ideas of [13] can be adapted to our problematic.

Those previous methods require a choice of a smoothing parameter (h,..); it influences the estimated *pdf*. It would be better to avoid choosing

such parameter. The finite mixture method could be a good solution to avoid this problem. Instead of determining the smoothing parameter, we must determine the number of terms in the mixture which could be done easily as we will see.

4.5 Finite mixtures

The finite mixture methods assumes the density $\rho(x)$ (which is either $\rho_X(x)$ or $\rho_{X,Y}(x,y)$) can be modeled as the sum of c weighted densities, with $c << n$ (where n is the size of the sample). The most general case for the univariate finite mixture is

$$\rho(x) = \sum_{i=1}^{c} p_i g(x;\theta_i)$$

where p_i represents the weight or mixing coefficients for the i-th term, and $g(x;\theta_i)$ denotes the gaussian probability density function, with parameters represented by the vector $\theta_i = (\mu_i, \sigma_i)$.To make sure that is a bona fide density, we must impose the condition $p_1 + .. + p_c = 1$ and $p_i > 0$. To evaluate $\rho(x)$, we take our point x, find the value of the component densities $g(x, \theta_i)$ at that point and take the weighted sum of these values. The number of components c acts something like a smoothing parameter. Smaller numbers of components will behave more like parametric models and can lead to specification bias. Greater flexibility can be obtained by letting the number of components grow, although too many components can lead to overfitting and excessive variation.
The major interest of the Gaussian mixture is its capacity to produce a quick and useful approximation to a multi-modal histogram. Indeed an efficient algorithm called the *Expectation Maximization* (EM) algorithm allows us to estimate the unknown parameters $\theta_1, .., \theta_c, p_1, .., p_c$. Remark that the use of the EM algorithm requires that the number of components c in the mixture is available. Nevertheless you can overcome this problem by making a guess on c, then execute the EM algorithm and at last perform a Kolmogorov Smirnov test (KS test) in order to validate the guessing on c. Remark that in [12], an extended KS test is proposed to determine the number of components in a mixture model: an approach based on the EM procedure is constructed by the extended KS test in parallel to do the computation. This method seems to be more efficient.

4.6 Application on the attacks

Now that we are able to compute efficiently *pdfs*, we will focus on how to perform an attack. Remind that in the case of the CPA, the following estimator is used :

$$\widehat{r_{K_j}}(t) = \frac{N\sum_{k=1}^{N} C_k(t)HW(M_k \oplus K_j) - \sum_{k=1}^{N} C_k(t)\sum_{k=1}^{N} HW(M_k \oplus K_j)}{\sqrt{N\sum_{k=1}^{N}(C_k(t) - \overline{C_k(t)})^2}\sqrt{N\sum_{k=1}^{N}(HW(M_k \oplus K_j) - \overline{HW(M_k \oplus K_j)})^2}}$$

At each time t, an attacker computes for all the values K_i this estimator and the more the correlation is, the better the hypothesis made on the secret key is at the time when the real key is manipulated.

In the case of the Generalized Correlation Power Analysis (GCPA), the following estimator is used :

$$\widehat{\lambda_{K_j}}(t) = \sqrt{1 - exp(-2\widehat{I}(\widehat{C}(t), HW(\widehat{M} \oplus K_j)))}$$

where $\widehat{I}(X, Y)$ is the estimated MI between X and Y computed by one of the method described in the section 4 and $\widehat{C}(t)$ is a random variable for which we have a sample of size $N * 2^s$ at each time (i.e N consumption curves for each guessed key).
At each time t, an attacker computes for all the values K_j this estimator and the closer to one the coefficient is, the better the hypothesis made on the secret key is at the time when the real key is manipulated.
Remind that for this attack you mustn't use random messages, you must use chosen messages in order to have characteristic *pdf* as described in the beginning of section 4.

5 Conclusion

The GCPA is a new attack which takes into account both the linear and non linear relationship between the leakages and the hamming weight model. But in order to validate this new attack and compare it with the CPA (for the number of curve needed and the computation timings; obviously the GCPA will be much more time consuming compared to the CPA), an attack must be conducted with real data. Moreover the choice of the method used to estimate the MI will influence the results of this experiment. Another research topics would be (if the result of the

GCPA are good) to develop a new power consumption model in order to apply a classical CPA (as the relation between the new model and the consumption would be linear). At last it would be interesting to use the copula methods in order to define a new attack.

References

1. Moon, Y., B. Rajagopalan, and U.Lall (1995), Estimation of mutual information using kernel density estimators, *Phys. Rev. E, 52(3), 2318-2321*
2. Kraskov, A., H. Stogbauer, and P.Grassberger (2004), Estimating mutual information *Phys. Rev. E, 69, 066138*
3. Hulle, M. M. V. (2005), Edgeworth approximation of multivariate differential entropy *Neural Comput., 17,1903-1910*
4. Brier E., Clavier C. and Olivier F., Correlation Power Analysis with a Leakage Model *CHES 04*
5. Le T., Clédière J., Canovas C., Robisson B., Servière C. and Lacoume J., A Proposition for Correlation Power Analysis Enhancement *CHES 06*
6. P.Kocher, J.Jaffe, B.Jun, Differential Power Analysis. *In proceedings of CRYPTO 1999, LNCS 1666, pp. 388-397, Springer-Verlag,1999.*
7. Celluci, C. J.,A.M.Albano, and P.E.Rapp (2005), Statistical validation of mutual information calculations: Comparisons of alternative numerical algorithms *Phys. Rev. E, 71,066208*
8. M. Vannucci, Nonparametric density estimation using wavelets. Technical report, Duke University, 1998.
9. S.Mallat, A Wavelet Tour of Signal Processing, Second Edition *ACADEMIC PRESS*
10. L.Wasserman, All of Nonprametric Statistics *Springer Texts in Statistics*
11. U.Cherubini, E.Luciano, and W.Vecchiato, Copula methods in finance *John Wiley and Sons, Ltd, 2004*
12. Ming-Heng Zhang, Qian-Sheng Cheng, Determine the number of components in a mixture model by the extended KS test *Pattern Recognition Letters archive Volume 25 , Issue 2 (January 2004), Pages: 211 - 216, Elsevier Science Inc.*
13. D.R.M. Herrick, G.P.Nason, B.W.Silverman, Some New Methods for Wavelet Density Estimation 2001

6 Annex A

The kernel function G is usually chosen to be a smooth unimodal function with a peak at 0. Even though Gaussian kernels are the most often used, there are various choices among kernels as shown in the table below.

Kernel	$K(u)$
Uniform	$\frac{1}{2}I(u)$
Triangle	$(1-\|u\|)I(u)$
Epanechnikov	$\frac{3}{4}(1-u^2)I(u)$
Quartic	$\frac{15}{16}(1-u^2)^2I(u)$
Triweight	$\frac{35}{32}(1-u^2)^3I(u)$
Tricube	$\frac{70}{81}(1-\|u\|^3)^3I(u)$
Gaussian	$\frac{1}{\sqrt{2\pi}}exp(-\frac{1}{2}u^2)$
Cosinus	$\frac{\pi}{4}cos(\frac{\pi}{2}u)I(u)$

where the function I is defined as : $I(x) = 1$ if $|x| \leq 1$ else $I(x) = 0$

The quality of a kernel estimate depends less on the shape of the G than on the value of its bandwidth h. It's important to choose the most appropriate bandwidth as a value that is too small or too large is not useful. Small values of h lead to very spiky estimates (not much smoothing) while larger h values lead to oversmoothing.

Remarks

The choice of the function G weights more or less some points.

A kernel function with 2 parameters means that the vector u is of dimension 2 hence the previous function can be adapted to multivariate inputs.