CMC, vol.56, no.2, pp.325-338, 2018

A Highly Effective DPA Attack Method Based on Genetic Algorithm

Shuaiwei Zhang[1], Xiaoyuan Yang[1, *], Weidong Zhong[1] and Yujuan Sun[2]

Abstract: As one of the typical method for side channel attack, DPA has become a serious trouble for the security of encryption algorithm implementation. The potential capability of DPA attack induces researchers making a lot of efforts in this area, which significantly improved the attack efficiency of DPA. However, most of these efforts were made based on the hypothesis that the gathered power consumption data from the target device were stable and low noise. If large deviation happens in part of the power consumption data sample, the efficiency of DPA attack will be reduced rapidly. In this work, a highly efficient method for DPA attack is proposed with the inspiration of genetic algorithm. Based on the designed fitness function, power consumption data that is stable and less noisy will be selected and the noisy ones will be eliminated. In this way, not only improves the robustness and efficiency of DPA attack, but also reduces the number of samples needed. With experiments on block cipher algorithms of DES and SM4, 10% and 12.5% of the number of power consumption curves have been reduced in average with the proposed DPAG algorithm compared to original DPA attack respectively. The high efficiency and correctness of the proposed algorithm and novel model are proved by experiments.

Keywords: DPA, efficiency, noise, genetic algorithm, fitness function, novel model.

1 Introduction

Side channel attack has become a powerful attack method after being studied by different researchers all over the world [Kocher (1996)], leading great threat to the security of cryptographic devices. Currently, profiled attack and non-profiled attack are the two main strategies of side channel attack. Profiled attack [Fahn and Pearson (1999)] introduced as the strongest leakage analysis in an information theoretic sense and is divided into two phases: profiled phase and attacking phase such as template attack [Chari, Rao and Rohatgi (2002)] or stochastic attacks [Schindler, Lemke and Paar (2005)]. On the other hand, non-profiled attack is based on Differential Power Analysis (DPA) [Kocher, Jaffe and Jun (1999)] and Correlation Power Analysis (CPA) [Brier, Clavier and Olivier (2004)]. Instead of acquiring and analyzing cryptographic devices beforehand, like in

[1] Key Laboratory of Network & Information Security of People's Armed Police, Engineering University of People's Armed Police, Wu Jing Road, No.1, Xi'an, 710086, China.

[2] Department of Electrical and Computer Engineering University of Toronto, 111 St. George Street, Toronto, M5S 2E8, Canada.

[*] Corresponding Author: Xiaoyuan Yang. Email: yxyangyxyang@163.com.

CMC. doi:10.3970/cmc.2018.03611

profiled attack, non-profiled attack obtains the secret keys based on processing the information of power consumption gathered from cryptographic devices on-site. Among which, DPA has become the most popular strategy of power attack, benefiting from its low cost and high efficiency. Kocher et al. [Kocher, Jaffe and Jun (1999)] successfully obtained the secret key by exploiting Simple Power Analysis (SPA) and DPA targeting at DES algorithm in 1999. Since DPA utilizes the statistically differential technology to guess secret keys, without having the detailed knowledge of encryption algorithm, a large amount of power consumption data is needed to improve the SNR, and then to accurately recover secret keys. However, acquiring larger amount of power consumption data takes longer time, if under critical conditions (e.g. limited time for the attack) will result in small amount of power consumption sample or high noise power in the acquired samples, which invalidates the accuracy of the obtained secret keys.

2 Related work

Many attempts and contributions have been made, under the same outside environment and controlled parameters, to tackle the problem of how to improve the efficiency of DPA attack. Durvaux et al. [Durvaux and Standaert (2016)] pointed out that finding appropriate attack points at the beginning is the first step to increase the efficiency; after analyzing different ways of finding appropriate attack points, an approach of detecting those points based on *t*-test has been proposed. Hajra et al. [Hajra and Mukhopadhyay (2013)] come up with a multivariate model for an FPGA platform, which significantly improved the efficiency of DPA attack under the condition of high noise power in 2013. And in 2015, they proposed the multivariate leakage model for the optimal combining of non-profiled DPA attack [Hajra and Mukhopadhyay (2015)], and validated their theory by experiments. Ren et al. [Ren, Wu, Li et al. (2016)] applied advanced correlation power analysis attack to smart card with triple-DES, the attack efficiency was enhanced by combining multivariate leakage points in the process of DES encryption algorithm. Zhang et al. [Zhang, Wu, Wang et al. (2014)] firstly exploited genetic algorithm and put forward a new accurate leakage model based on the power consumption of multi S-box instead of the conventional one single S-box, which tremendously increases the attack efficiency. Together with the development of Artificial Intelligence and Big Data, optimization algorithms are trending to be applied to the area of side channel attack. Many attentions have been paid to profiled attack. As artificial intelligence and machine learning become strong tools to tackle a lot of problems in different research fields, the cryptographic community has been exploring the potential of profiled attacks based on machine learning models [Bartkewitz and Lemke-Rust (2012); He, Jaffe and Zou (2012); Heuser and Zohner (2012); Hospodar, Gierlichs, De Mulder et al. (2011); Jap and Breier (2014); Lerman, Bontempi and Markowitch (2014); Lerman, Bontempi and Markowitch (2015)], and Lerman et al. [Lerman, Martinasek and Markowitch (2016)] has concluded that under the circumstances of having “Dirty Data” in the acquired power consumption samples, the robustness and efficiency of profiling attack based on machine learning are better than template attack.

However, as for non-profiled attack, “Dirty Data” need to be taken into considerations as well. “Dirty Data” is the measured power consumption during the processing of algorithm within the encryption chip, which vastly differs from regular power consumption value because of the influence of outer environment and high noise. Carrying samples of "Dirty Data" into the leakage model and further applied into the DPA attack, will reduce the SNR of entire power consumption sample, significantly decreasing the attack efficiency. Thus, verifying "Dirty Data" from regular power consumption is one of the key factors to improve attack efficiency.

This work is inspired by conventional procedures of DPA attack and evaluation model of efficiency, while realizing that there is a possibility of acquiring "Dirty Data" during the real attacks. Hence, we put forward a high efficiency method for DPA attack based on genetic algorithm. The gathered power consumption data will be selected, during which the "Dirty Data" will be eliminated, integrated and assorted by the specifically designed fitness function, and then combined with conventional DPA attack procedure to recover secret keys. Besides, we also propose a highly practical evaluation model of DPA attack efficiency. And by experimenting with power consumption data from DES and SM4 algorithm, the amount of power consumption samples is proven to be reduced with our algorithm, and the proposed evaluation model of efficiency has better accuracy than the conventional model.

Our contribution. The novel contributions of this paper are as follows:

(1) In this paper, we put forward a highly efficient DPA attack based on genetic algorithm, which is able to eliminate most of the "Dirty Data" generated by influence of noises in the raw power consumption data, and in the meanwhile, integrates and assorts the effective data, down-sizing the amount of samples for attack curve, elevating the attack efficiency.

(2) We come up with a new evaluation model of DPA attack efficiency. Comparing to the conventional model without taking "Dirty Data" into consideration, which severely interferes with the information provided by effective data, our model processes "Dirty Data" to develop the utilization of effective data, resulting in a much more accurate model.

(3) The genetic algorithm and evaluation model of efficiency proposed in our works can be applied to any encryption algorithm based on DPA. Furthermore, regarding other power consumption attack method, similar results can be achieved by slightly adjusting the fitness function.

This paper is organized as follows. Section 3 includes preliminaries of conventional DPA procedures. Section 4 introduces our highly effective DPA attack method. In Section 5, the results of the experiments are presented for validation of our algorithm and novel model. Section 6 presents the conclusions. Section 7 is dedicated to future work.

3 Preliminaries

3.1 Conventional procedures of DPA attack

(1) n sets of known plaintext $M_1, M_2, M_3, ..., M_n$ was selected and encrypted with same cryptographic key K, and the power consumption curves are measured and recorded as $T_i[j]$ respectively, i is the set number $(1 \le i \le n)$ and j is the sample point.

(2) $D(M_i, b, K_s)$ represents the bit b of median L at the end of first iteration, among which M is the known plaintext, $0 \le K_s \le 2^q$ is the q bits cryptographic key of the S-box of bit b.

(3) Based on guessed value of K_s and known plaintext, distinguisher $D(M_i, b, K_s)$ can be calculated, and then average all the power consumption curves for $D(M_i, b, K_s)$ equals "0" and "1" respectively, compute differential power consumption using the following formula:

$$\Delta D[j] = \frac{\sum_{i=1}^{n} D(M_i, b, K_s) T_i[j]}{\sum_{i=1}^{n} D(M_i, b, K_s)} - \frac{\sum_{i=1}^{n} (1 - D(M_i, b, K_s)) T_i[j]}{\sum_{i=1}^{n} (1 - D(M_i, b, K_s))} \quad (1)$$

(4) Observing the differential power consumption curves, if there is one peak point, q bits cryptographic key is correctly guessed, otherwise, it is a false guessing, a new round of anticipation should be started.

(5) Applying the same procedures to anticipate q bits cryptographic key to other S-boxes.

3.2 Conventional evaluation model of DPA attack efficiency

DPA attack efficiency has two important factors:

(1) Minimum number of power consumption curve needed to recover key, model is as follows:

$$A = \{M_1, M_2 ... M_n\} \quad (2)$$

$$A_0, A_1 \xleftarrow{divide} C_{algorithm}(K_{guess}, m_1, m_2 ... m_n) \quad K_s \in \left(0, 2^q\right) \quad (3)$$

$$V = \max\left(\frac{aver(A_0)}{n_0} - \frac{aver(A_1)}{n_1}\right) = \Delta D[j]_{\max} \quad (4)$$

$$n_0 + n_1 = n \quad (5)$$

Among which, A is the obtained power consumption curves; A_0 and A_1 stands for power consumption curves with $D(M_i, b, K_s)$ equals to "0" and "1" respectively; n_0 and n_1 are the number of power consumption curves in A_0 and A_1 respectively; $C_{algorithm}$ represents for encryption algorithm. The minimum number of power consumption curves is obtained using the evaluation algorithm in Tab. 1.

Fig. 1 is an illustration of array V, G_r and G_w are the maximum differential power consumption as a function of n, when guessing K_s is true or false respectively. Point d, the intersection point of G_r and G_w, is the minimum number of power consumption curves needed for DPA attack.

Table 1: Evaluation algorithm for minimum number of power consumption curves

```
Input: $A$ , $C_{algorithm}$
Output: $V_{K_r,d}$

input $A$ , $C_{algorithm}$
  for $K_{guess} = 0$ to $2^q$
    for $i = 1$ to $n$
      do $A_0, A_1 \xleftarrow{divide} C_{algorithm}(K_{guess}, A)$
        $V_{K_{guess},i} = \max(\frac{aver(A_0)}{n_0} - \frac{aver(A_1)}{n_1})$
    end for
      $V_{K_{guess}} = [V_{K_{guess},1}, V_{K_{guess},2}, \cdots, V_{K_{guess},d}, \cdots, V_{K_{guess},n}]$
  end for
      $V = \begin{bmatrix} V_{0,1} & V_{0,2} & \cdots & V_{0,d} & \cdots & V_{0,n} \\ V_{1,1} & V_{1,2} & \cdots & V_{1,d} & \cdots & V_{1,n} \\ \vdots & \vdots & \ddots & \vdots & \vdots & \vdots \\ V_{K_r,1} & V_{K_r,2} & \cdots & V_{K_r,d} & \cdots & V_{K_r,n} \\ \vdots & \vdots & \vdots & \vdots & \ddots & \vdots \\ V_{2^q,1} & V_{2^q,2} & \cdots & V_{2^q,d} & \cdots & V_{2^q,n} \end{bmatrix}$
    return $V_{K_r,d}$
```

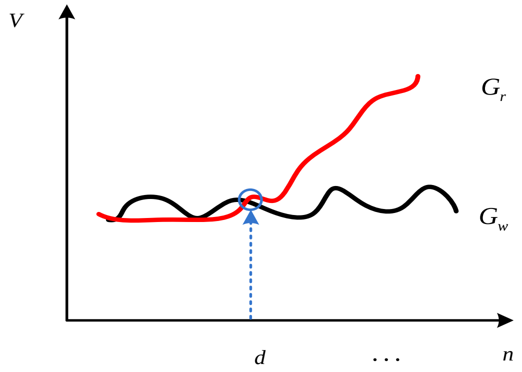

Figure 1: Efficiency function for conventional DPA attack

(2) Possibility of cryptographic key recovering:

Repeat procedures in (1) for p times, vector $D=\{d_1,d_2,d_3,\cdots,d_p\}$ with value d of p dimension can be achieved（$d_1\le d_2\le d_3\le\cdots\le d_p$）, then the possibility of cryptographic key recovering i/p can be calculated from d_i. As it is shown in Fig. 2, the largest element $d_{\max}=d_p$ is the minimum number of power consumption curves needed for recovering of cryptographic key with possibility of 100%.

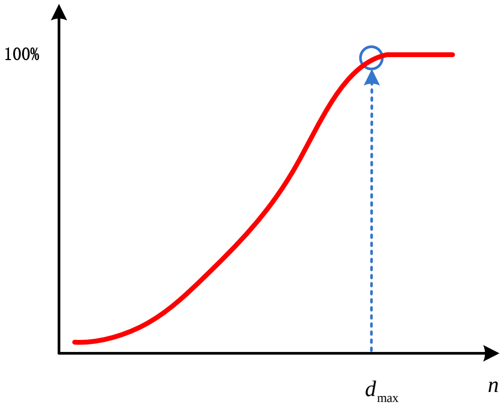

Figure 2: Success rate with conventional DPA attack

4 DPA attack based on genetic algorithm (DPAG)

4.1 Main idea of our algorithm

According to the analysis above, the gathered power consumption data contains many “Dirty Data”, due to the fact that attacker would not be able to anticipate the environment while attacking beforehand, and the working environment for the chip, which processes the cryptographic key, is not ideal and interfered by noises. Thus, in order to improve the attack efficiency and reduce the number of power consumption curve sample, a fitness function is designed as in Eq. (6). Through evaluating the fitness value of each power consumption curve with Eq. (6), curves with the high value of fitness will be preserved and curves with the low value of fitness or sample with “Dirty Data” will be eliminated.

$$H=acorr(L,m_i)-b\frac{\sum_{j=1}^{k}(L^j-m_i^j)^2}{\sum_{j=1}^{k}(L^j)^2}+C_0 \tag{6}$$

Among which, H stands for the fitness value of the i^{th} curve; L is the average value of all the acquired power consumption curves, and L^j is the j^{th}（$1\le j\le k$）sampling point of

L; m represents the target power consumption curve, while m_i^j means the j^{th} sampling point of i^{th} target power consumption curve; C_0 is the parameter of power consumption leakage point; and a , b are coefficients.

4.2 DPAG algorithm process

Step 1. Based on different values of $D(M_i, b, K_s)$, samples of power consumption curve A_0 can be separated from A;

Step 2. Calculating the average curve L_0 in A_0 with the number of n_0, and the fitness value H_0 between each curve of A_0 and L_0. Then sort all the curves in A_0 with its fitness value from largest to smallest, select a number of $\frac{n_0}{2}$ curves with largest fitness value, and copy them into *B* and *C* (*B*=*C*);

Step 3. Defining *C* as the set of selected curves, each curve belonged to C has a probability of selection proportional to the sequence of sorting in Step 2, which is $\frac{4n_0 - 8i + 8}{{n_0}^2 + 2n_0}$ for the i^{th} curve ($i \in (1, \frac{n_0}{2})$).

Step 4. Set *B* is regarded as initial population, and the probability of selection for each curves in *B* is $\frac{4n_0 - 8i + 8}{{n_0}^2 + 2n_0}$; *P* represents the probability of crossover, and the crossover population is in B'.

Step 5. Sorting the individuals in *B'*, with its fitness value in H_0, like in Step 2, selecting a number of $\frac{n_0}{2}$ curves with largest fitness value, as the next generation *B1*.

Step 6. Repeating Step 4 and Step 5 for a number of times or values in H_0 no longer increasing, recording the curves with largest H_0 in each generation and their parents; processing the data in A_1, similarly as in A_0, with the only difference that sorting with fitness value from smallest to largest, and then $H = |H_0 - H_1|$.

4.3 Novel evaluation model of DPA attack efficiency

(1) Minimum number of power consumption curve needed to recover key.

Different from conventional evaluation model of DPA attack efficiency, our novel model adopts the optimal number of power consumption curve as an independent variable for fitness function *H*, while evaluating the minimum number of power consumption curve needed to recover the key. The model is as follows:

$$A = \{M_1, M_2 ... M_n\} \tag{7}$$

$$A_0, A_1 \xleftarrow{divide} C_{al\,\mathrm{g}orithm}(K_s, m_1, m_2 ... m_n) \qquad K_s \in \left(0, 2^{\mathrm{q}}\right) \tag{8}$$

$$H_1 = acorr(L, m_i) - b\frac{\sum_{j=1}^{k}(L^j - m_i^j)^2}{\sum_{j=1}^{k}(L^j)^2} + C_0 \qquad m_i \in A_1 \tag{9}$$

$$H_0 = acorr(L, m_i) - b\frac{\sum_{j=1}^{k}(L^j - m_i^j)^2}{\sum_{j=1}^{k}(L^j)^2} + C_0 \qquad m_i \in A_0 \tag{10}$$

$$H = \left|H_0 - H_1\right| \tag{11}$$

Evaluation figure of the minimum number of power consumption curves can be obtained with the evaluation algorithm in Tab. 2.

Table 2: Novel evaluation algorithm for minimum number of power consumption curves

```
Input: $A$ , $C_{algorithm}$
Output: $H$
input $A$ , $C_{algorithm}$
$A_0, A_1 \xleftarrow{divide} C_{algorithm}(K_{guess}, A)$
$H^1 = acorr(L, m_i) - b\frac{\sum_{j=1}^{k}(L^j - m_i^j)^2}{\sum_{j=1}^{k}(L^j)^2} + C_0 \qquad m_i \in A_1$
$H^0 = acorr(L, m_i) - b\frac{\sum_{j=1}^{k}(L^j - m_i^j)^2}{\sum_{j=1}^{k}(L^j)^2} + C_0 \qquad m_i \in A_0$
for $K_{guess} = 0$ to $2^q$
    do $DPAG^0$
```

```
7:    $H^0 = \begin{bmatrix} H^0_{0,B^1_n} & H^0_{0,B^2_n} & \cdots & H^0_{0,B^d_n} & \cdots & H^0_{0,B^n_n} \\ H^0_{1,B^1_n} & H^0_{1,B^2_n} & \cdots & H^0_{1,B^d_n} & \cdots & H^0_{1,B^n_n} \\ \vdots & \vdots & \ddots & \vdots & \ddots & \vdots \\ H^0_{K_r,B^1_n} & H^0_{K_r,B^2_n} & \cdots & H^0_{K_r,B^d_n} & \cdots & H^0_{K_r,B^n_n} \\ \vdots & \vdots & \ddots & \vdots & \ddots & \vdots \\ H^0_{2^q,B^1_n} & H^0_{2^q,B^2_n} & \cdots & H^0_{2^q,B^d_n} & \cdots & H^0_{2^q,B^n_n} \end{bmatrix}$
8:    for $K_{guess} = 0$ to $2^q$
9:    do $DPAG^1$
10:   $H^1 = \begin{bmatrix} H^1_{0,B^1_n} & H^1_{0,B^2_n} & \cdots & H^1_{0,B^d_n} & \cdots & H^1_{0,B^n_n} \\ H^1_{1,B^1_n} & H^1_{1,B^2_n} & \cdots & H^1_{1,B^d_n} & \cdots & H^1_{1,B^n_n} \\ \vdots & \vdots & \ddots & \vdots & \ddots & \vdots \\ H^1_{K_r,B^1_n} & H^1_{K_r,B^2_n} & \cdots & H^1_{K_r,B^d_n} & \cdots & H^1_{K_r,B^n_n} \\ \vdots & \vdots & \ddots & \vdots & \ddots & \vdots \\ H^1_{2^q,B^1_n} & H^1_{2^q,B^2_n} & \cdots & H^1_{2^q,B^d_n} & \cdots & H^1_{2^q,B^n_n} \end{bmatrix}$
11:   return $H = \left| H^0_{K_r,B^d_n} - H^1_{K_r,B^d_n} \right|$
```

Fig. 3 shows the minimum number of power consumption curve needed to recover the secret key. Curve G_r is the changing tendency of H, with the increasing of B_n^n, while the anticipation of K_s is true; G_w is the changing tendency of the maximum of H, with the increasing of B_n^n, while the anticipation of K_s is false. Curve G_w and G_r separates at intersection point of B_n^d; meaning that B_n^d is the minimum number of power consumption curve needed for DPA attack.

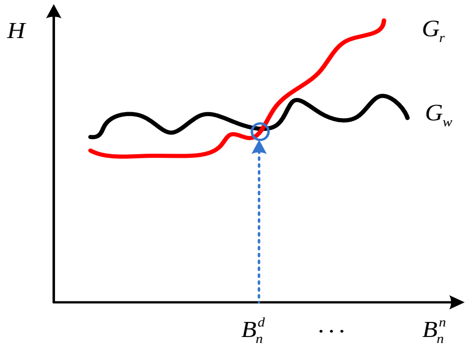

Figure 3: Efficiency function for novel DPA attack

(2) Possibility of cryptographic key recovering:

Repeating procedures in Eq. (1) for *p* times, $B_n^D = \left\{ B_n^{d_1}, B_n^{d_2}, B_n^{d_3}, \cdots, B_n^{d_p} \right\}$ can be achieved, and then calculate the B:

$$B = B_n^D = \left\{ (B_n^{d_1}), (B_n^{d_2}), \cdots, (B_n^{d_p}) \right\} = \left\{ B_1, B_2, \cdots, B_p \right\} \tag{12}$$

Among which, $(B_1 \le B_2 \le \cdots \le B_p)$, then i/p, the recovering possibility of corresponding cryptographic key can be calculated from B_i. As it is shown in Fig. 4, the largest element $B_{\max} = B_p$ is the minimum number of power consumption curves needed for recovering of key with possibility of 100%.

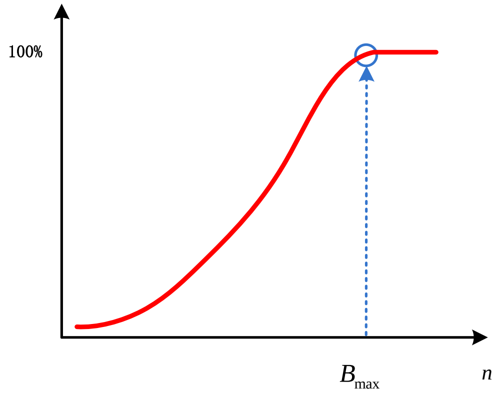

Figure 4: Efficiency function for novel DPA attack

5 Experiments

In order to validate the efficiency and correctness of the proposed **DPAG** and novel evaluation model for DPA attack efficiency, experiments with the power consumption data gathered form the data acquisition system, as Fig. 5 shown, running block cipher algorithm of DES and SM4 separately on FPGA platform have been carried out. Results are as shown below:

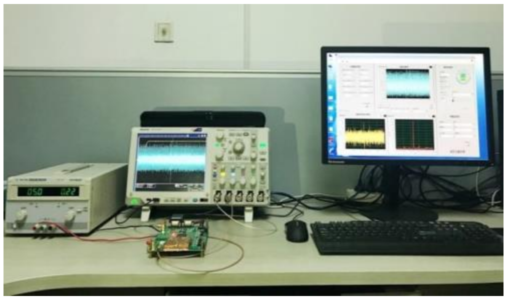

Figure 5: System for acquisition of power consumption from target device

(1) Experiments on DES algorithm:

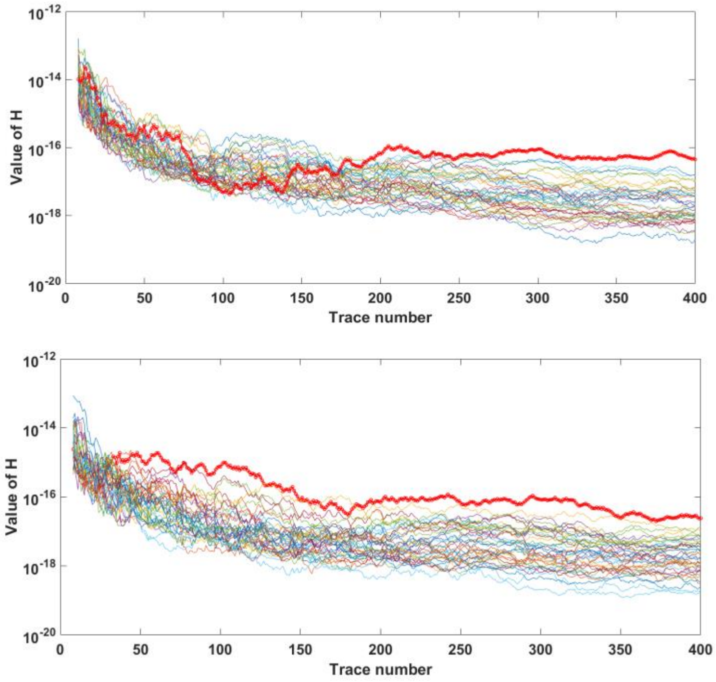

Figure 6: Comparison between original DPA and DPAG in DES

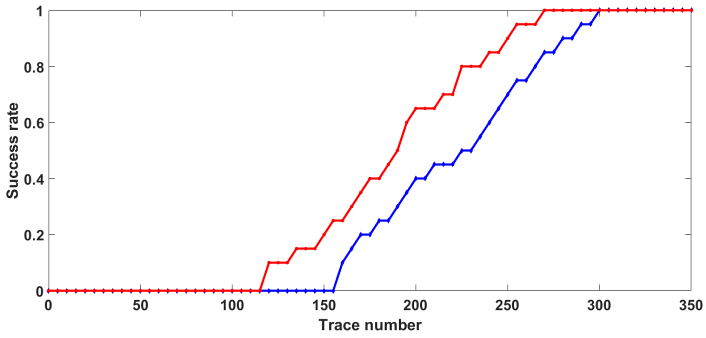

Figure 7: Success rate between original DPA and DPAG in DES

Illustrated in Fig. 6**,** as the results of attack between original DPA and DPAG in the DES algorithm, the trace number reduced from 267 to 184 when the right key curve can be separated from the wrong key curves. And in Fig. 7, clearly shows that under 100% success rate, 10% of the number of power consumption curves have been reduced in average with the proposed DPAG algorithm compared to original DPA attack.

(2) Experiments on SM4 algorithm:

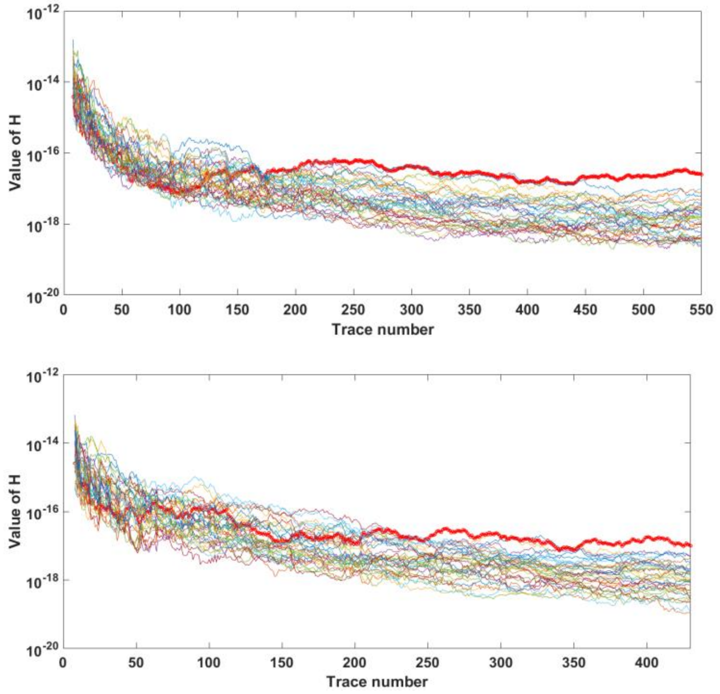

Figure 8: Comparison between original DPA and DPAG in SM4

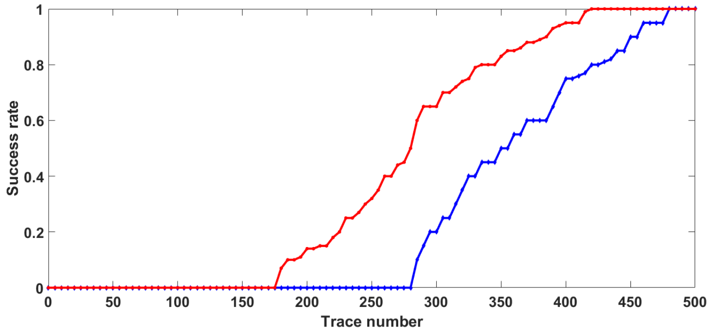

Figure 9: Success rate between original DPA and DPAG in SM4

Demonstrated in Fig. 8, as the results of attack between original DPA and DPAG in the SM4 algorithm, the trace number reduced from 443 to 353 when the right key curve can be separated from the wrong key curves. And in Fig. 9, clearly shows that under 100% success rate, 12.5% of the number of power consumption curves have been reduced in average with the proposed DPAG algorithm compared to original DPA attack.

To sum up, the high efficiency and correctness of the proposed method in this paper have been proved by experiments on block cipher algorithm of DES and SM4.

6 Conclusion

In this work, a highly efficient DPA attack based on genetic algorithm has been designed. With the established fitness function, power consumption data curve with larger or smaller value of fitness can be selected, sorted and integrated into effective data, eliminating samples with “Dirty Data” introduced by noise interference. Furthermore, a novel evaluation model of DPA attack efficiency has been proposed based on the designed algorithm. Comparing to conventional evaluation model of DPA attack efficiency, our model adopts the optimal number of power consumption curve as independent variable for fitness function *H*, instead of using superposition of single curve samples as an independent variable for differential power consumption, while evaluating the minimum number of power consumption curve needed to recover the key. After the experiments, both the DPA attack based on genetic algorithm and novel evaluation model of DPA attack efficiency are supported correctly and accurately by experimental evidence: power consumption data of DES and SM4 algorithm processed with FPGA platform.

7 Future works

On the one hand, development of conventional methods of side channel attack has reached bottleneck; on the other hand, with the rapid development of Artificial Intelligence and Big Data, more and more optimization algorithms with excellent performances have been invented and improved. There is a great potential for the improvement of attack efficiency by applying algorithm of artificial intelligence into side channel attack. Our next step is to investigate more appropriate, better performed optimization algorithms for side channel attack to further improve the attack efficiency.

Acknowledgement: This work was supported by National Key R&D Program of China (Grant No. 2017YFB0802000), National Natural Science Foundation of China (Grant No. U1636114, 61772550, 61572521), National Cryptography Development Fund of China (Grant No. MMJJ20170112).

References

Bartkewitz, T.; Lemke-Rust, K. (2012): Efficient template attacks based on probabilistic multi-class support vector machines. *International Conference on Smart Card Research and Advanced Applications*, pp. 263-276.

Brier, E.; Clavier, C.; Olivier, F. (2004): Correlation power analysis with a leakage model. *International Workshop on Cryptographic Hardware and Embedded Systems*, pp. 16-29.

Chari, S.; Rao, J. R.; Rohatgi, P. (2002): Template attacks. *International Workshop on Cryptographic Hardware and Embedded Systems*, pp. 13-28.

Durvaux, F.; Standaert, F. X. (2016): From improved leakage detection to the detection of points of interests in leakage traces. *Annual International Conference on the Theory and Applications of Cryptographic Techniques*, pp. 240-262.

Fahn, P. N.; Pearson, P. K. (1999): IPA: A new class of power attacks. *International Workshop on Cryptographic Hardware and Embedded Systems*, pp. 173-186.

Hajra, S.; Mukhopadhyay, D. (2013): Multivariate leakage model for improving non-profiling DPA on noisy power traces. *International Conference on Information Security and Cryptology*, pp. 325-342.

Hajra, S.; Mukhopadhyay, D. (2015): Reaching the limit of non-profiling DPA. *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems*, vol. 34, no. 6, pp. 915-927.

He, H.; Jaffe, J.; Zou, L. (2012): Side channel cryptanalysis using machine learning. *CS229 Project*, pp. 1-392.

Heuser, A.; Zohner, M. (2012): Intelligent machine homicide. *International Workshop on Constructive Side-Channel Analysis and Secure Design*, pp. 249-264.

Hospodar, G.; Gierlichs, B.; De Mulder, E.; Verbauwhede, I.; Vandewalle, J. (2011): Machine learning in side-channel analysis: A first study. *Journal of Cryptographic Engineering*, vol. 1, no. 4, pp. 293-302.

Jap, D.; Breier, J. (2014): Overview of machine learning based side-channel analysis methods. *International Symposium on Integrated Circuits*, pp. 38-41.

Kocher, P. C. (1996): Timing attacks on implementations of Diffie-Hellman, RSA, DSS, and other systems. *Annual International Cryptology Conference*, pp. 104-113.

Kocher, P.; Jaffe, J.; Jun, B. (1999): Differential power analysis. *Annual International Cryptology Conference*, pp. 388-397.

Lerman, L.; Bontempi, G.; Markowitch, O. (2014): Power analysis attack: An approach based on machine learning. *International Journal of Applied Cryptography*, vol. 3, no. 2, pp. 97-115.

Lerman, L.; Bontempi, G.; Markowitch, O. (2015): A machine learning approach against a masked AES. *Journal of Cryptographic Engineering*, vol. 5, no. 2, pp. 123-139.

Lerman, L.; Martinasek, Z.; Markowitch, O. (2016): Robust profiled attacks: Should the adversary trust the dataset? *IET Information Security*, vol. 11, no. 4, pp. 188-194.

Ren, Y.; Wu, L.; Li, H.; Li, X.; Zhang, X. et al. (2016): Key recovery against 3DES in CPU smart card based on improved correlation power analysis. *Tsinghua Science and Technology*, vol. 21, no. 2, pp. 210-220.

Schindler, W.; Lemke, K.; Paar, C. (2005): A stochastic model for differential side channel cryptanalysis. *International Workshop on Cryptographic Hardware and Embedded Systems*, pp. 30-46.

Zhang, Z.; Wu, L.; Wang, A.; Mu, Z. (2014): Improved leakage model based on genetic algorithm. *IACR Cryptology EPrint Archive*, pp. 314.