Combined VMD-GSO Based Points of Interest Selection Method for Profiled Side Channel Attacks

Ngoc Quy Tran[1], Hong Quang Nguyen[1], and Van-Phuc Hoang[2(✉)]

[1] Faculty of Electronics and Telecommunications, Academy of Cryptography Techniques, Hanoi, Vietnam

[2] Institute of System Integration, Le Quy Don Technical University, Hanoi, Vietnam
phuchv@lqdtu.edu.vn

Abstract. Nowadays, one of the most powerful side channel attacks (SCA) is profiled attack. Machine learning algorithms, for example support vector machine, are currently used for improving the effectiveness of the attack. One issue when using SVM-based profiled attack is extracting points of interest, or features from power traces. So far, studies in SCA domain have selected the points of interest (POIs) from the raw power trace for the classifiers. Our work proposes a novel method for finding POIs that based on the combining variational mode decomposition (VMD) and Gram-Schmidt orthogonalization (GSO). That is, VMD is used to decompose the power traces into sub-signals (modes) of different frequencies and POIs selection process based on GSO is conducted on these sub-signals. As a result, the selected POIs are used for SVM classifier to conduct profiled attack. This attack method outperforms other profiled attacks in the same attack scenario. Experiments were performed on a trace data set collected from the Atmega8515 smart card run on the side channel evaluation board Sakura-G/W and the data set of DPA contest v4 to verify the effectiveness of our method in reducing number of power traces for the attacks, especially with noisy power traces.

Keywords: Profiled attack · Side channel attack · Support machine learning · Variational mode decomposition

1 Introduction

Side channel attack (SCA) is one of the most powerful cryptoanalysis technique for revealing secret key or sensitive information stored on cryptographic devices. The conducting of SCA is based on the analyzing of unintended side channel leakages observed from the devices during cryptographic algorithms run on. There are so many forms of the observed leakages, but the time of operation, the power consumption of the devices, or electromagnetic radiation are the most common uses. SCAs based on the power consumption are known as the power analysis attacks first proposed by Kocher et al. in the late 1990s [1]. These attacks rely on the physical nature that the instantaneous power

© ICST Institute for Computer Sciences, Social Informatics and Telecommunications Engineering 2021
Published by Springer Nature Switzerland AG 2021. All Rights Reserved
N.-S. Vo et al. (Eds.): INISCOM 2021, LNICST 379, pp. 483–503, 2021.
https://doi.org/10.1007/978-3-030-77424-0_38

consumption of a cryptographic device depends on the data being processed and the operation being executed. This dependency can be used to expose the data that contains the secret key of a cryptographic device. Depending on the knowledge of attacker about the device under attack as well as the statistical method of analysis and extraction of information from the power consumption traces, SCAs are classified into two main classes: non-profiled attacks and profiled attacks. DPA [1], CPA [2], Mutual Information Analysis (MIA) [3] attacks belong to the first class. These are considered as effective attack methods when the attacker has only an attack device and information of its implementation. The profiled attacks are used when the attacker has the same device as the attack device with full control over. By this device, the attacker is able to accurately characterize the power consumption of the device so that the attack efficiency is much higher than non-profiled attacks in the term of the needed number of power consumption measurements for revealing the secret key successfully.

So far, there is a lot of attention on profiled attack in SCA research community. The first one is called template attack, as proposed in [4] by Chari et al., which relies on an assumption that power consumption characteristic follows multivariate Gaussian distribution. However, in general, this assumption might be not met, so that machine learning (ML) techniques are introduced for profiled attacks. Consequently, several works have applied machine learning techniques to profiled SCA attacks [5, 6]. These works all indicate that ML based profiled attacks are more efficient and SVM is commonly used as ML algorithm. ML based profiled attacks relax the need for probability distributions of side channel leakage traces but still require specific extraction techniques to identify points of interest (POIs) on the traces or feature selection in ML domain. In SCA, POIs are time sample points from the power traces that correspond to the calculation of the sensitive variables being targeted and their values change according to those variables [7]. The POIs selection, as input features to machine learning algorithms is critical for two main reasons as follows: (1) the power traces are usually acquired by a measurement equipment with high sampling rates and so consist of a large amount of time samples. However, often only a relatively small range of these time samples is informative or statistically dependent on a sensitive target variable; (2) power traces are considered as highly multi-dimensional data that results in the curse of dimensionality issues with ML algorithms. That is, computational and runtime complexity for them to solve a task increase. Therefore, POIs selection is critical to the effectiveness of the profiled attacks. The more precisely the POIs are selected, the better the ability to characterize the power consumption of a profiled device, resulting in increasing attack efficiency and vice versa. Our work focuses on a method for finding POIs for SVM-based profiled attacks.

Some studies in the side channel community focus on methods of finding POIs for profiled attacks, which can be classified into four classes: filter methods, dimensionality reduction method, wrapper and hybrid methods, and ML based methods. In filter methods, POIs selection process operates on the base of computation of some sample-wise statistics, whose aim is to quantify a sort of signal strength. The signal-strength estimates are derived from classical SCA distinguishers computed under the right key hypothesis, such as the Difference of Means [4] or Correlation Power Analysis (CPA) [8]. Other deployed estimates are the Sum of Squared Differences [9], the Signal-to-Noise Ratio [10, 11], and the Sum of Squared t-differences, corresponding to the t-test [9]. Once

the chosen signal-strength estimate is computed, all time samples for which the signal strength is higher than a certain threshold are selected as POIs. Of these, the POIs selection method based on CPA estimates is the most common use.

Principal component analysis (PCA), as dimensionality reduction method, is another technique for POIs selection. The time samples on traces have the maximize the variability in the projection space of PCA are remained as POIs. So far, the effectiveness of PCA-based profiled attack is not clear and selecting the number of retained components as well as the threshold of determination in PCA process is also not an easy task [12].

Profiled attacks, as presented in [13], use the wrapper method for finding of POIs. In the wrapper method, subsets of time samples on the power traces are evaluated by the prediction performance of a classifier and the subset has the best performance is selected as POIs. To reduce the number of subsets of the wrapper, hybrid method is used. That is, candidate features are first selected by a filter then furthered refined by an accuracy wrapper. As claimed in [13], wrapper and hybrid methods gave slightly better results. The issue with this approach is that computational complexity and search space increase exponentially as the length of trace increases. Because of the capability of ML algorithms in determining the most informative features from raw data inputs, ML algorithms can be used to finding POIs of power trace. In the first work in this approach [14], SVM has been trained and the sample points of trace which correspond to highly absolute value of weights are selected as POIs. This method is also called normal-based feature selection and strongly recommended by authors in [14].

As our knowledge, there are only few works on finding POIs with noisy traces. Furthermore, there have been no more studies on feature engineering in the machine learning domain as applied to profiled attacks. For noisy traces, the authors in [7] claim that the goodness of POIs selections depends significantly on the noise level: as noise level increases the goodness of POIs selection decreases, while at the same noise level, CPA estimation based POIs selection method is the best. This drawback of the POIs selection method is confirmed by the authors in [13] regarding the wrapper and hybrid method. Inspired by the success of VMD [15] in feature engineering in machine domain, in this work, we propose the method of combining VMD and GSO to find the POIs of power traces. That is, VMD is used to decompose a trace into sub-signals, or modes and POIs are selected from these modes by using GSO as the filer feature selection method. Then, the selected POIs are used for SVM-based profiled attack. We denote SVM_{VMD} for our proposed attack. To demonstrate the efficiency of our proposed attack method, we compare our method with two other SVM-based profiled attacks using the SVM classifier. The first attack uses CPA as the POIs selection method as in [5], so called SVM_{CPA} and is currently considered to be the best method, and the second one uses a normal-based feature selection method as in [14], so called SVM_{NB}. We also investigate the effectiveness of our method with noisy power traces, which often happens in the real attack scenarios.

Our contributions are follows. Firstly, we investigate the ability of combining VMD and GSO for finding POIs of the power traces. Secondly, we propose an SVM-based profiled attack method that uses our POIs selection method. This is a different approach for conducting profiled attack and it is efficient for noisy power traces. The remainder of the paper is structured as follows. In Sect. 2, we describe the background to this research:

the profiled attacks, variational mode decomposition and Gram-Schmidt Orthogonalization and SVM. In Sect. 3, we present our proposed SVM-based profiled attack. The experiments and their results are presented in Sect. 4. Finally, the conclusions of our research are presented in Sect. 5.

2 Background

2.1 Profiled Attack

For profiled attack, the attacker must have a device with full control over that is similar the attack device. This device is called profiling device and used for leakage information characterization by the attacker. In this work, an attack device that runs a block cipher is used for our attack scenario and leakage is in the form of power consumption. The implementation of profiled attack consists of two phases: profiling phase on profiled device and attack phase on attack device.

In the profiling phase, a dataset of N_p profiling traces is acquired from the profiled device. The dataset is seen as the realization of the random variable $S_p \triangleq \{(x_1, z_1), \ldots, (x_{N_p}, z_{N_p})\} \sim Pr[X|Z]^{N_p}$, where x_i are the traces obtained from the device processing the respective intermediate values $z_i = \varphi(P, K)$. Based on S_p, a model is built to characterize the side channel leakage of the cryptographic device for each hypothetical value z_i. This can be modeled as $F(X|Z) : X \rightarrow P(Z)$.

In the attack phase, a dataset of N_a attack traces are acquired from the target device. The dataset is seen as a realization of $S_a \triangleq (k, \{(x_1, p_1), \ldots, (x_{N_a}, p_{N_a})\})$ such that $k \in K$, and for all $i \in [1, N_a]$, $p_i Pr[P] \wedge x_i Pr[X \vee Z = \varphi(p_i, k)]$. Subsequently, a prediction vector is computed for each attack trace, based on a previously built model: $y_i = F(x_i), \forall i \in [1, N_a]$. A score, for example the probability, is assigned to each trace for each intermediate value hypothesis z_j, $with j \in$. The j-value of y_i describes the probability of z_j according to the model when the attack trace is x_i. These scores are combined over all the attack traces to output a *likelihood* for each key hypothesis and the candidate with the highest likelihood is predicted to be the correct key. The maximum likelihood score can be used for prediction. For every key hypothesis $k \in K$, this likelihood score is defined by Eq. (1) with the key assigned the highest score predicted as being the most likely.

$$d_{S_a}[k] \triangleq \prod_{i=1}^{N_a} y_i[z_i] \; where \; z_i = \varphi(p_i, k) \tag{1}$$

2.2 Variational Mode Decomposition (VMD)

VMD is a method used to decompose a real valued signal into narrowband sub-signals, also known as intrinsic mode functions (IMFs) or simply VMD modes [15] by Eq. (2). In that $x(t)$ is the orginal signal and $u_k(t) = A_k(t)cos(\phi_k(t))$, called the k^{th} mode, is the amplitude-modulation and frequency-modulation signal where $A_k(t)$ is the slowly

varying, positive envelope and $\phi_k(t)$ is the phase. Each mode has a central frequency f_k that its instantaneous frequency $\phi'(k)$ varies around.

$$x(t) = \sum_{k=1}^{K} u_k(t) \tag{2}$$

The finding simultaneously a set of modes and their central frequencies by VMD is done by solving the optimization problem given by expression (3). This is the constrained minimization process of sum of all mode's bandwidth. The bandwidth of each mode is estimated by 3 steps: compute the analytic signal of each mode by using Hilbert transform so its spectrum is positive; multiply the analytic signal with a complex exponential for shifting its frequency spectrum to baseband; compute the squared 2-norm of the gradient of the baseband signal.

$$\begin{gathered} \min_{u_k f_k} \left\{ \sum_k \left\| \frac{d}{dt} \left[\left(\delta(t) + \frac{j}{\pi t} \right) * u_k(t) \right] e^{-j2\pi f_k t} \right\|_2^2 \right\} \\ \text{s.t. } \sum_k u_k(t) = x(t) \end{gathered} \tag{3}$$

The solution for (3) provides the optimal point of an unconstrainted augmented Lagrangian given by (4) where α is the penalty factor and $\lambda(t)$ is Lagrangian multiplier. This optimization could be solved by using the alternate direction method of multipliers algorithm [16]. All modes of $x(t)$ are computed in frequency domain by (5) and (6) at each iteration of algorithm until the condition (7) is met.

$$\begin{aligned} L\{u_k(t), f_k, \lambda(t)\} = \alpha \sum_{k=1}^{K} \left\| \frac{d}{dt} \left[\left(\delta(t) + \frac{j}{\pi t} \right) * u_k(t) \right] e^{-j2\pi f_k t} \right\|_2^2 \\ + \left\| x(t) - \sum_{i=1}^{K} u_k(t) \right\|_2^2 + \left\langle \lambda(t), x(t) - \sum_{k=1}^{K} u_k(t) \right\rangle \end{aligned} \tag{4}$$

$$U_k^{n+1}(f) = \frac{X(f) - \sum_{i<k} U_k^{n+1}(f) - \sum_{i>k} U_k^n(f) + \frac{\wedge^n}{2}(f)}{1 + 2\alpha \left\{ 2\pi \left(f - f_k^n \right) \right\}^2} \tag{5}$$

$$f_k^{n+1} = \frac{\int_0^\infty \left| U_k^{n+1}(f) \right|^2 f df}{\int_0^\infty \left| U_k^{n+1}(f) \right|^2 df} \approx \frac{\sum f \left| U_k^{n+1}(f) \right|^2}{\sum \left| U_k^{n+1}(f) \right|^2} \tag{6}$$

$$\sum_k \frac{\left\| u_k^{n+1}(t) - u_k^n(t) \right\|_2^2}{\left\| u_k^n(t) \right\|_2^2} < \epsilon \tag{7}$$

After the convergence of this optimization, the inverse Fourier transform is applied to (5) to obtain the waveform of each mode. Because of the combination of Wiener filtering, Hilbert transform and ADMM in VMD, VMD modes are highly accurate in describing the different components of the original signal and robust to noise.

2.3 SVM Method

SVM algorithms [17] is used to construct classifiers. The basic form of SVM is the binary classifier which can classify two class by the largest-margin separating hyperplane between them. Let $D_M = \{(x_i, y_i) \vee x_i \in R^N, y_i \in \{-1, +1\}, i = 1, 2, .., M\}$ represent a training set, where x_i is a training vector, and y_i is the label of x_i. The training vector x is mapped into feature space by the nonlinear function $\phi(.)$. Consequently, the maximum margin of a binary-class SVM classifier is a constrained optimization problem as follows:

$$\min_{\omega,b,\xi}\left(\frac{1}{2}\|\omega\|^2 + C\sum_{i=1}^{M}\xi_i\right), s.t. y_i\left(\omega^T\phi(x_i) + b\right) \geq 1 - \xi_i, \xi_i \geq 0, \quad i = 1, 2, \ldots, M \tag{8}$$

where $\omega \in R^N, b \in R$, and $C > 0$ is the penalty parameter which evaluates the trade-off between training error and margin size, and ξ_i is the training error of x_i. After the Lagrange multiplier is introduced, the optimization problem in (8) is simplified as follows:

$$\min_{\alpha} \frac{1}{2}\sum_{i=1}^{M}\sum_{j=1}^{M}\alpha_i\alpha_j y_i y_j K\left(x_i, x_j\right) - \sum_{i=1}^{M}\alpha_i, s.t. \sum_{i=1}^{M}\alpha_i y_i = 0, 0 \leq \alpha_i \leq C, \quad i = 1, 2, \ldots, M \tag{9}$$

where α_i are Lagrange multipliers and the kernel function is $K\left(x_i, x_j\right) = \phi(x_i)\phi\left(x_j\right)$.

The kernel function maintains the reasonable computational complexity of SVM in feature space. The common kernel functions are linear kernel and RBF kernel.

$$\begin{aligned} K_{Linear}\left(x_i, x_j\right) &= x_i^T x_j \\ K_{RBF} &= \left(x_i, x_j\right) = exp \end{aligned} \tag{10}$$

where γ is the hyperparameter in (10) and the notation $\|.\|$ represents the L^2 norm between two vectors.

For consideration of training time and accuracy, the one-against-one strategy can be used to train an SVM classifier for each pair of possible classes. In order to use the maximum likelihood estimation to recover the secret key, an attacker is more interested in the probability of an instance x_i belonging to the class c. Accordingly, we give the posterior conditional probability $P_{SVM}(x_i \vee c)$ of each instance [18].

2.4 GSO-Based Feature Selection

As mentioned above, the finding of POIs on the power trace is also known as the feature selection in ML domain. In this paper, Gram-Schmidt orthogonalization based feature selection method is used to select the POIs of the traces. This method is in the form of filter method, independent of the advance classifiers and is effective in ranking he features contained in the traces based on criteria computed directly from the traces. Indeed, this method allows the features to be determined without weighting all of features in the traces. It ranks features based on the correlation between features and the output target

of a prediction model or the pre-assigned label of features. Let $x_k = [x_{k1}, x_{k2}, \ldots, x_{kN}]^T$ be the k^{th} feature vector of N instances, $y = \left[y_1, y_2, .., y_N\right]^T$ be the output target and Q is the number of features. This results in (N, Q) matrix feature data set. To define the relation between each feature and output target, the correlation is calculated by (11) [19].

$$cos(\alpha_k) = \frac{\langle x_k.y \rangle}{\|x_k.y\|} \tag{11}$$

In formula (1), x_k is a column vector containing N values of the k^{th} feature in all Q features, α_k is the angle of vectors x_k and y. If they are perpendicular to each other, the cosine of α_k equals 0 meaning there is no correlation between them, whereas when the angle between them becomes smaller, this correlation increases and the maximum value is 1 when they are completely correlated.

The GSO-based feature selection process uses the formula (11) to quantify the degree to which features are related to the output target. The first selected feature is the most correlated input features with the output target by the cosine calculation. The next features are selected according to the iteration process as follows until all input features are ranked, or until a stopping condition is met [19]: (1) the rest input features and output target are projected on the subspace orthogonal to the selected feature; (2) the cosine calculations are done on this subspace for all projected features and target output to find out the most correlated feature. This feature is added to selected feature list.

3 Proposed Method

In this part, we present our proposed SVM-based profiled attack that uses the combination of VMD and GSO for POIs selection of power traces.

3.1 SVM-Based Profiled Attack

The proposed SVM-based profiled attack, as shown in Fig. 1, is carried out in two phases: a profiling phase and an attack phase. In the profiling phase, power traces are collected from the profiled device while it is executing a cryptographic algorithm to form a trace data set. This trace data set is labeled according to the Hamming weight of targeted value of the algorithm that needs to be profiled $Z_1, \ldots, Z_m$. Usually these targeted values are taken at the output of the S-box. Because, they are 8-bit values that result in 9 Hamming weight classes from 0 to 8 denoted as $c_0, c_1, \ldots, c_8$. This labeled set of traces is fed to the feature extraction and selection block for mapping traces into feature space and the best features are selected. These selected features are considered as POIs of the power trace in feature space that should describe the statistical dependency of the Hamming weight of the targeted value Z_i with the power consumption. In the final step of the profiled phase, POIs of all traces are used to train SVM to model the power consumption characteristic of the profiled device. For training SVM classifier, its parameters are selected as follows: the kernel function is RBF, the penalty factor and width of kernel of RBF are optimized by Grey wolf optimization algorithm as presented in [20].

During the attack phase, unlabeled traces collected from the attack device are fed into the feature extraction and selection block to select POIs and they are next classified by the trained SVM model to determine the probabilities of the traces for classes $c_i \in \{0, 1, \ldots 8\}$. Finally, we compute the log likelihood for each hypothesis value of the key byte that is used by attack device as follows:

$$logL_k \equiv log \prod_{i=1}^{N_a} P_{SVM}(x_i|c_i) = \sum_{i=1}^{N_a} log\, P_{SVM}(x_i|c_i) \tag{12}$$

where k is a hypothesis key byte value, $c_i = Hammingweight(Sbox(p_i, k))$, p_i is the plaintext associated with trace x_i, and the number of attack traces is N_a. The key k_c that maximizes the log likelihood in (13) is predicted to be the correct key.

$$k^c = \underset{k \in K}{argmax}\, logL_k \tag{13}$$

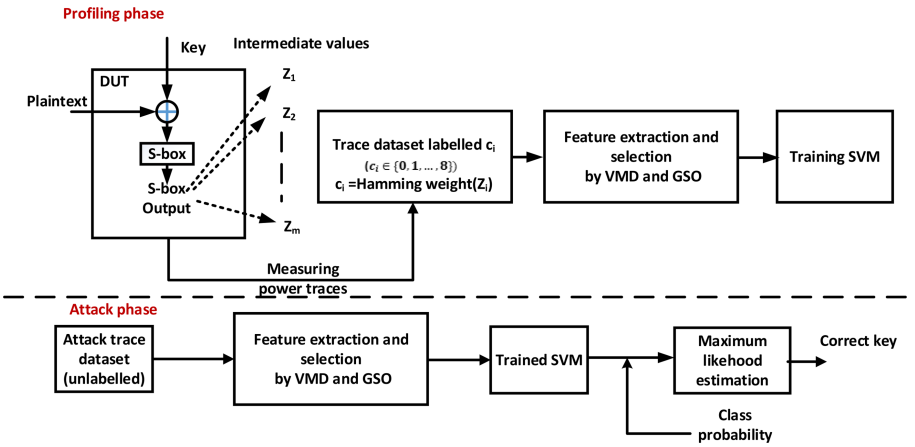

Fig. 1. SVM-based profiled attacks framework.

3.2 Feature Extraction and Selection

Features or POIs selection is critical to the effectiveness of the profiled attacks. The more precisely the features are selected, the better the ability to characterize the power consumption of a profiled device, resulting in increasing attack efficiency and vice versa. This section presents a new method for finding features of power traces for SVM-based profiled attack. First, power trace characteristics are discovered as follows:

The power trace collected during the operation of a cryptographic device describes its power consumption. It consists of many components in which dynamic power dissipation is the most important [10]. This component depends on the processed data of the circuit and is useful information leakage for power analysis attack. The dynamic power

dissipation mainly caused by the switching activity of logic gates in a circuit which is controlled by the operating clock frequency so the dynamic power consumption is driven by the clock frequency of circuit. Therefore, in spectrum of power trace, it is expected that the clock frequency component has significant magnitude compared to the other components. The information leakage is nearly in the form of a both amplitude and frequency modulation signal and the central frequency of its spectrum is the clock frequency. Generally, in a device, the different parts of its circuit are controlled by different operating frequencies through the clock division system, so the dynamic power dissipation is the combination of some amplitude - frequency modulation signals with different center frequencies. So, if it is possible to separate the dynamic power dissipation to the amplitude - frequency modulation signals with different center frequencies, one of these signals contains significant information leakage related to target circuit part while the other does not.

As a result, the feature extraction process from power traces should ensure: (1) the remaining features contain the most important information of the trace which is the dynamic power dissipation caused by the targeted circuit; (2) it could remove the other components of power traces; (3) it could reduce noise in the power traces. Fortunately, these requirements can be fulfilled by using VMD method because VMD decomposes a trace into different components and it is robust to the efects of sampling and noise.

In our proposed method, VMD is used for extracting features from power traces. VMD decomposes the signal into sub-signals, called VMD mode in this paper, which are amplitude-modulated frequency modulated signals so each mode contains a specific frequency spectrum with different center frequency. So, the VMD mode which center frequency relates to clock frequency could be use as feature of the power trace. Indeed, VMD can discover signal changes more accurately so that features of power traces can be recognized more accurately. Moreover, VMD is robust to noise thanks to the use of Wiener filter technique. Thus, VMD should be useful for using noisy traces.

Unfortunately, VMD mode still contains redundant features which not related to target variable that has been profiled. Therefore, they must be eliminated for increasing the generalization capability of the classifier and reducing the volume of training data. The elimination of redundant features is known as the feature selection. In previous related works, all features that is higher than a certain threshold are selected. In this paper, we recommend using GSO to selection feature of selected VMD mode.

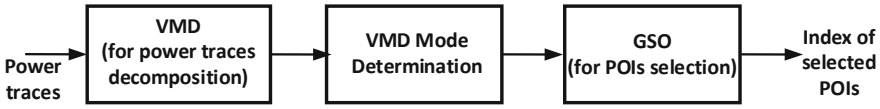

Fig. 2. Feature selection procedure of power trace.

To sum up, there are three phases in the proposed features extraction and selection method as illustrated in Fig. 2. Firstly, VMD is used to decompose original traces to VMD modes. In the VMD process, it is necessary to set parameters. VMD needs to preset the value of number of decomposed modes (*K).* If K is too small, all the decomposition modes cannot be captured. In contrast, if K is too large, the interfering signal will be

over decomposed such that the center frequencies of modes will be mixed. The penalty factor (α) affects the bandwidth of the decomposed signal. To decompose the traces by the VMD, the number of IMFs (K) and the quadratic penalty factor (a) should be determined beforehand. In this work, the parameters, K and a, were determined according to the following steps: (1) Decompose a power trace into modes for different $K = [1, 20]$ and $a = [5, 2000]$; (2) Add up the modes for each of the K and a values to obtain the reconstructed power trace and estimate the values of Pearson correlation coefficient for the reconstructed and original power trace; (3) Select the sets of K and a values for maximum of Pearson correlation coefficient. Others input parameters of VMD including update rate (τ) and convergence condition (ϵ) are selected by standardization values in range of $(0 : 1e - 6)$ [15].

In VMD mode determination phase, the frequency range that containing the clock operating frequency of our attack device is selected. This VMD mode can be used as features of the power trace and contains the most useful information for the SCA. Therefore, CPA attack on the selected VMD mode should give the best results among all VMD modes. Hence, the method for determination of VMD mode is as follows: perform CPA attacks on all the VMD modes and based on the results of these CPA attacks, the VMD mode that has the largest correlation coefficient is selected.

In GSO feature selection phase, it is necessary to set the number of selected features (N). Our principle of finding the value of N is to find a trade-off between the accuracy and the computation cost or execution time. So, the value of N that SVM has the highest accuracy together with the lowest execution time is selected.

4 Experimental Results

In this section, we show the experimental results of implementing profiled attacks with the proposed new SVM_{VMD} approach, which is based on SVM and the combining of VMD and GSO for feature extraction and selection. We compared the effectiveness of the proposed method with the two profiled attacks based on SVM with points of interest selection by CPA in [5] called SVM_{CPA}, and the normal-based feature selection method in [14] called SVM_{NB}. The following parameters are used to evaluate the effectiveness of an attack:

- The ability to reveal the correct key: To confirm that our profiled attacks can reveal the correct key used by AES-128, we figure out the probability of each key being the actual key used. The key with highest probability is the most likely one.
- Guessing Entropy [21]: This score is also known as average rank of correct key is widely used to rate the effectiveness of side channel attack according to number of attack traces. By conducting the attack several times independently, guessing entropy is calculated as follows: (1) the rank of correct key in all guessing keys are computed. This is the index of the correct one in the list of all ranked keys; (2) calculate the average indexes of the correct key. In this paper, this guessing entropy is estimated over 10 independent attacks.

4.1 Dataset

Dataset 1: The set consists of 60000 traces collected while AES-128 processed intermediate values at S-box output. AES-128 was implemented on a Smartcard Atmega8515 runing on Sakura G/W. A sample of one of the collected power traces has 2500 time-samples which is titled '**Original trace**' in Fig. 3.

Dataset 2: This data set consists of 100000 traces downloaded from public DPA contest v4 website at: http://www.dpacontest.org/v4. There is 4000 time-samples in a trace of a first-order masked AES implementation which the output of S-box is $Sbox(P_i + k) \oplus M$, where M is a mask [22]. When the mask values are known, this data set are considered as an unmasked case.

4.2 Results

4.2.1 Feature Selection Phase

In this section, we investigated the effect of the feature selection on the classification accuracy of the proposed method. First, VMD is used to decompose original traces to VMD modes. For VMD, two main parameters: the number of VMD modes (K) and penalty factor (α) are initialized with $K = 5$, $\alpha = 1000$ according to procedure as described in Sect. 3.2. The VMD modes of both Dataset 1 and Dataset 2 are depicted in Fig. 3 and Fig. 4, respectively. As expected, VMD modes contain different components of the original signal at different central frequencies. For selection of VMD mode as feature of the power trace, we conduct CPA attacks on all the VMD modes and the results are shown in Table 1. VMD mode 1 and VMD mode 2 are selected as the extraction feature of power trace in Dataset 1 and VMD mode 2 because the CPA attack gives the highest correlation value.

Table 1. Results of correlation power attack on VMD modes.

Mode	Dataset 1		Dataset 2	
	Max correlation	Key found	Max correlation	Key found
VMD mode 1	0.64	63 (correct)	0.52	108 (correct)
VMD mode 2	0.62	63 (correct)	0.87	108 (correct)
VMD mode 3	0.54	63 (correct)	0.80	108 (correct)
VMD mode 4	0.37	255 (wrong)	0.37	188 (wrong)
VMD mode 5	0.35	246 (wrong)	0.34	135 (wrong)

Table 2 represents the classification accuracy of SVM on Dataset 1 when extracted features are VMD mode 1 and the selected features are chosen by GSO. Table 3 represents the classification accuracy of SVM on Dataset 2 when extracted features are VMD mode 2 and the selected features are chosen by GSO. The selected features are put into an SVM classifier for the training phase. As the feature dimension increases, so does the accuracy of the classification, but with too many features the accuracy decreases because the features do not generalize the power consumption characteristic well when used by the classifier. Therefore, the subset of features with the highest accuracy and lowest feature dimensions are selected and shown in bold font.

Table 2. Acquired results considering extraction of features by VMD and selection by GSO on Dataset 1.

Dim	Selected features	Classification accuracy (%)
2	1036 509	18.2
4	1036 509 2261 2262	30.12
6	1036 509 2261 2262 2263 2260	50.31
8	1036 509 2261 2262 2263 2260 2264 2265	81.56
10	1036 509 2261 2262 2263 2260 2264 2265 2259 861	81.78
12	1036 509 2261 2262 2263 2260 2264 2265 2259 861 2267 1038	89.22
14	**1036 509 2261 2262 2263 2260 2264 2265 2259 861 2267 1038 411 577**	**95.03**
16	1036 509 2261 2262 2263 2260 2264 2265 2259 861 2267 1038 411 577 886 1687	95.02
18	1036 509 2261 2262 2263 2260 2264 2265 2259 861 2267 1038 411 577 886 1687 1211 1670	94.27
20	1036 509 2261 2262 2263 2260 2264 2265 2259 861 2267 1038 411 577 886 1687 1211 1670 1576 216	92.84

Table 3. Acquired results considering extraction of features by VMD and selection by GSO on Dataset 2.

Dim	Selected features	Classification accuracy (%)
2	1804 3201	22.6
4	1804 3201 1664 2389	31.89
6	1804 3201 1664 2389 689 3231	60.38
8	1804 3201 1664 2389 689 3231 1524 1556	80.24
10	1804 3201 1664 2389 689 3231 1524 1556 3093 3192	86.66
12	1804 3201 1664 2389 689 3231 1524 1556 3093 3192 2766 2282	90.35
14	1804 3201 1664 2389 689 3231 1524 1556 3093 3192 2766 2282 1244 852	95.68
16	**1804 3201 1664 2389 689 3231 1524 1556 3093 3192 2766 2282 1244 852 2392 1797**	**96.62**
18	1804 3201 1664 2389 689 3231 1524 1556 3093 3192 2766 2282 1244 852 2392 1797 2251 3113	94.58
20	1804 3201 1664 2389 689 3231 1524 1556 3093 3192 2766 2282 1244 852 2392 1797 2251 3113 3108 1095	90.28

4.2.2 Key Recovery Phase

In order to verify our proposed SVM_{VMD} profiled attack has the ability to reveal secret key of attack device, In the attack phase, SVM_{VMD} is used to reveal the secret key when classifying 9 Hamming weight classes of S-box output. Instead of predicting the class HW of each trace, we gave the posterior conditional probability $P_{SVM}(X_i \vee c)$. The estimated probability of the hypothetical keys is determined by the maximum likelihood estimation. The correct key is defined as the key with the highest probability. For Dataset 1, which was collected in this experiment, the first byte of the AES-128 key is 63, and that is assigned the largest probability value, as depicted in Fig. 5. With Dataset 2, the recovery key is 108 which is identical to the key used to install AES in the DPA contest v4 (Fig. 6). These results prove that our attack method could correctly recover the key used by AES-128.

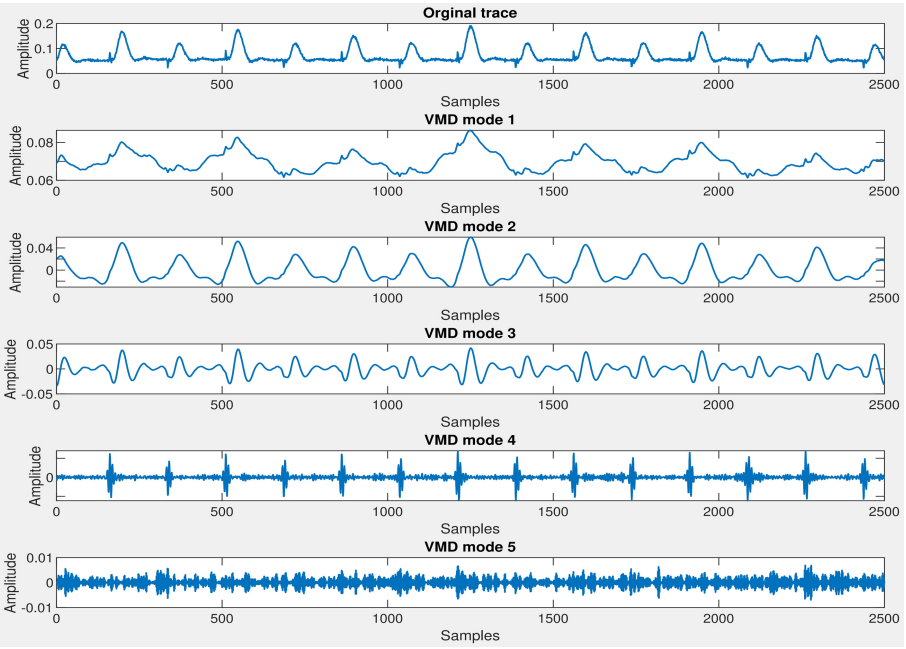

Fig. 3. VMD mode of the power trace on Dataset 1.

Figures 7 and 8 report the GE corresponding to different numbers of traces used for attacks with Dataset 1 when SVM_{VMD}, SVM_{CPA} and SVM_{NB} are used to predict the Hamming weight classes. As expected, the GEs of all attacks decrease as the number of traces increases. Moreover, the larger the size of the training set, the lower the GE. The reason for this is that the performance of SVM is determined by its parameters, and the size of the training set is critical to find the best parameters for the SVM. With Dataset 2, we performed the same experiments as for Dataset 1, and the GE calculated in the attack phases are presented in Fig. 9 and Fig. 10. The overall performance of all attacks are the same as those for Dataset 1. Again, SVM_{VMD} achieves the best GE values.

As shown in Table 4, for each dataset we give the number of traces required by the profiled attacks based on SVM for guessing entropy to reach 0. SVM_{VMD} requires the minimum number of traces to recover the key, 10.2 and 5.3 traces on average, corresponding to 100 and 200 profiling traces, respectively. These empirical results indicate that the SVM-based profiled attack with the VMD feature extraction technique is more effective than the attacks with the CPA and normal-based feature extraction techniques. This can be explained by the VMD extraction technique allowing more effective selection of trace characteristics than the CPA and normal-based POI selection methods.

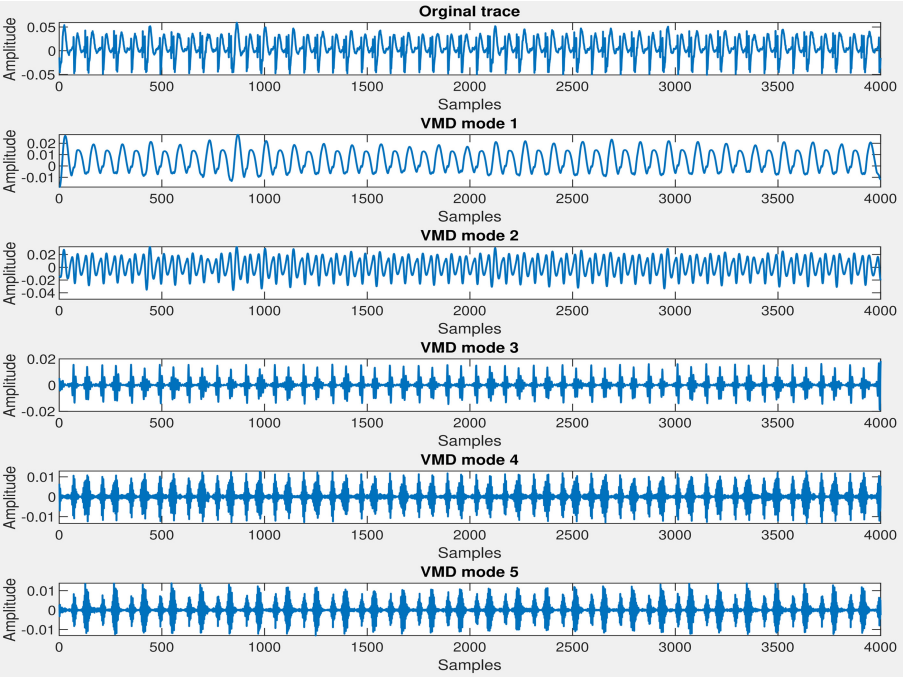

Fig. 4. VMD mode of the power trace on Dataset 2.

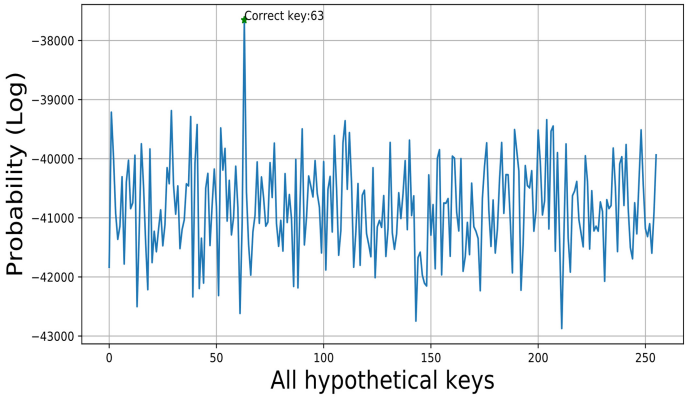

Fig. 5. Probability of all hypothetical keys on Dataset 1.

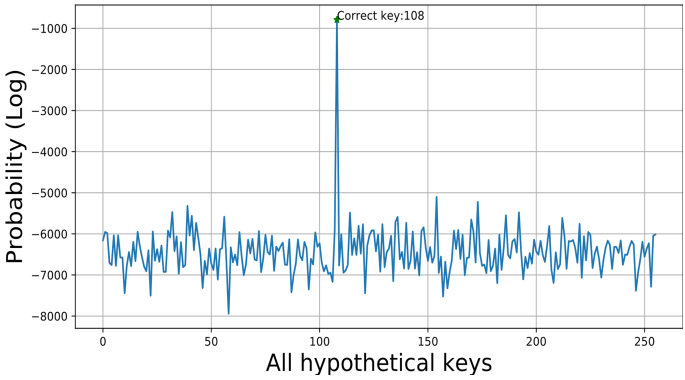

Fig. 6. Probability of all hypothetical keys on Dataset 2.

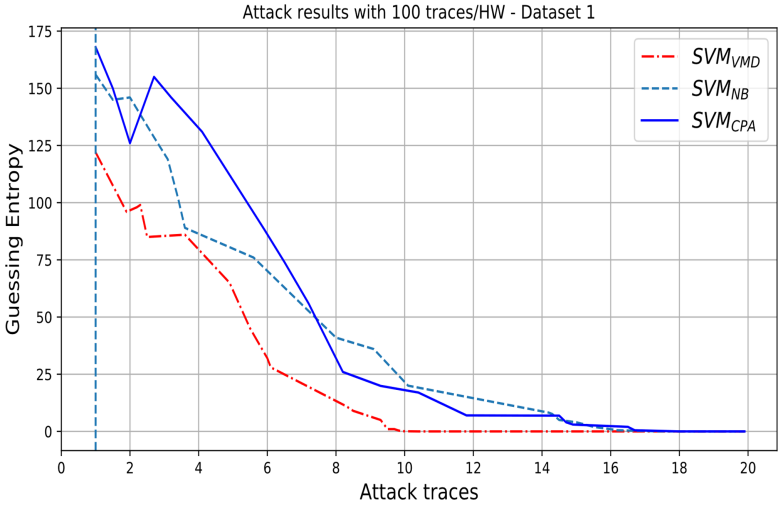

Fig. 7. Attack performance with 100 traces/HW class on Dataset 1.

4.2.3 Results with Noisy Traces

The power traces are usually polluted with noise in practice. To examine the effectiveness of our proposed SVM_{VMD} profiled attack in noisy condition, additive Gaussian noise is added to the power traces. In our experiments, two noise level of standard deviation $\sigma_1 = 5$ and $\sigma_2 = 10$ are added to both Dataset 1 and Dataset 2. In addition, different feature extraction techniques were used for the SVM-based profiled attacks to investigate their effects on the efficiency of the attacks in the presence of noise. Overall, the guessing entropy of all the attacks increase with the level of noise, but the attack based on SVM with combining of VMD and GSO is the least sensitive to noise. The results of our attacks with 200 profiling traces per Hamming weight class, presented in Figs. 11, 12, 13 and 14 and Table 5, show that out of SVM_{CPA}, SVM_{NB} and SVM_{VMD}, the proposed method, SVM_{VMD}, has the best performance at both noise levels while SVM_{CPA} and

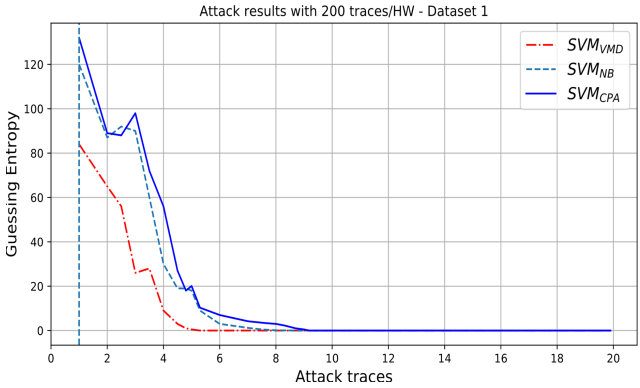

Fig. 8. Attack performance with 200 traces/HW class on Dataset 1.

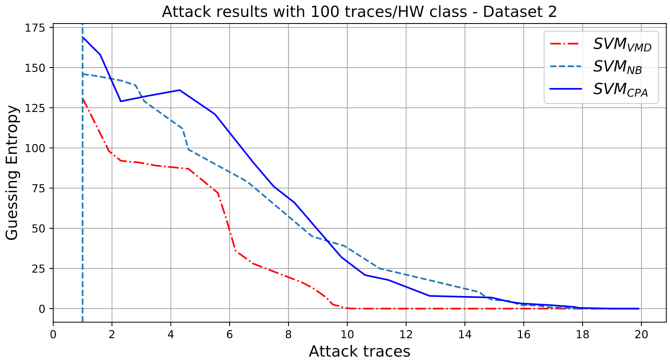

Fig. 9. Attack performance with 100 traces/HW class on Dataset 2.

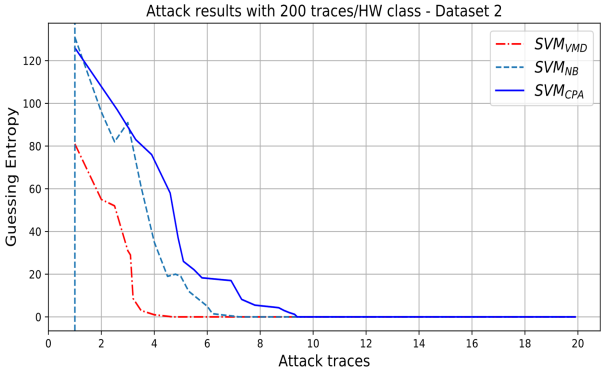

Fig. 10. Attack performance with 200 traces/HW class on Dataset 2.

SVM_{NB} are comparable to each other. After adding noise to the power trace, the number of traces required for GE to reach 0 increased by only 25% approximately with the

Table 4. Number of traces used by the attacks to attain GE = 0.

No. of profiling traces	Dataset 1			Dataset 2		
	SVM_{VMD}	SVM_{CPA}	SVM_{NB}	SVM_{VMD}	SVM_{CPA}	SVM_{NB}
100	10.2	18.1	17.6	10.3	19.2	18.3
200	5.3	9.2	8.7	4.7	9.4	7.3

proposed attack, while it increased by over 100% for the other methods. This proves that the VMD signal is insensitive to noise so the SVM_{VMD} attack should work well under noisy conditions. This property is very useful in real attack scenarios where collected measurement traces invariably contain noise.

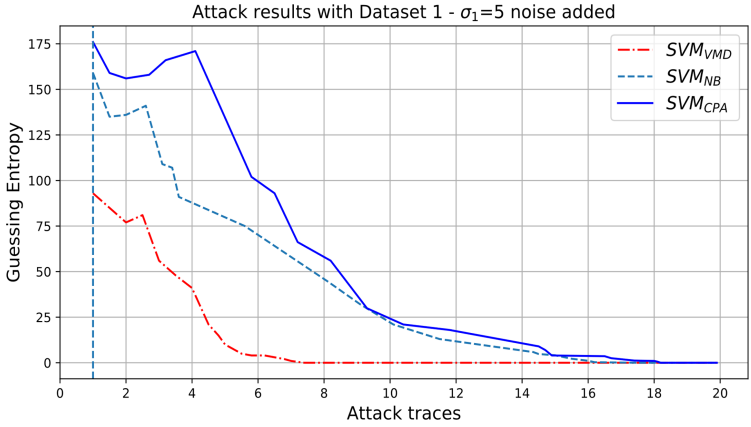

Fig. 11. Attack results on Dataset 1 with $\sigma_1 = 5$ noise added to power traces.

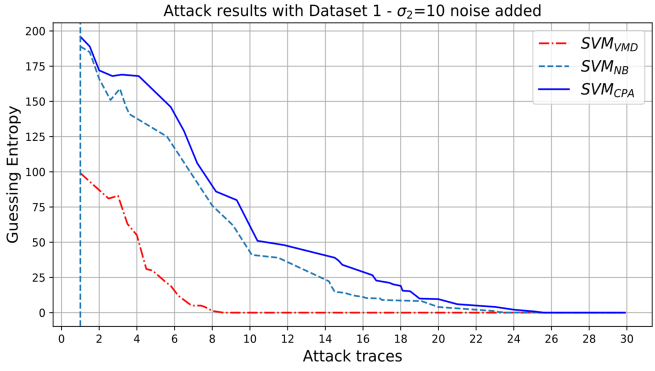

Fig. 12. Attack results on Dataset 1 with $\sigma_1 = 10$ noise added to power traces.

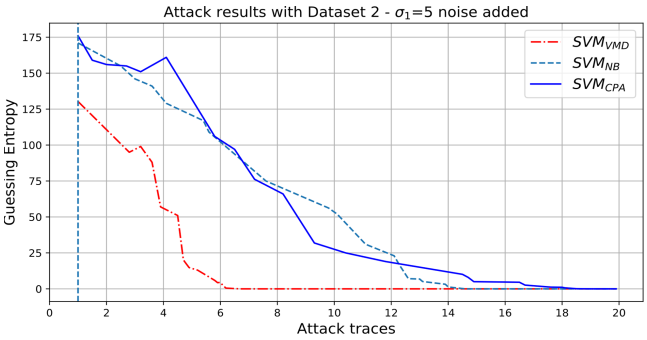

Fig. 13. Attack results on Dataset 2 with $\sigma_1 = 5$ noise added to power traces.

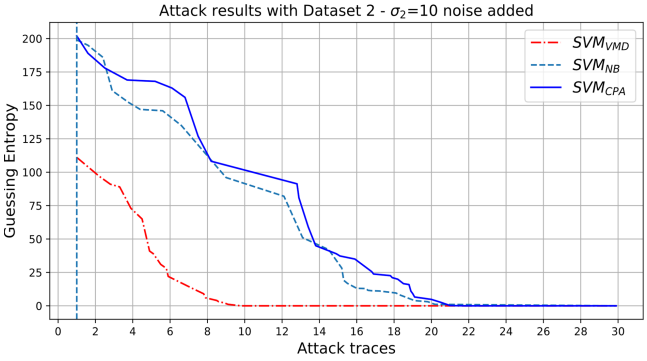

Fig. 14. Attack results on Dataset 2 with $\sigma_1 = 10$ noise added to power traces.

Table 5. Number of noisy traces used by the attacks to attain GE = 0.

Noise level	Dataset 1			Dataset 2		
	SVM_{VMD}	SVM_{CPA}	SVM_{NB}	SVM_{VMD}	SVM_{CPA}	SVM_{NB}
$\sigma_1 = 5$	7.4	19.0	17.0	6.7	18.8	14.6
$\sigma_2 = 10$	8.6	25.7	23.6	9.8	21.6	20.2

5 Conclusion

In this work, the combining of variational mode decomposition and Gram-Schmidt was proposed as a feature extraction and selection method for the power traces. The VMD mode that has central frequency related to clock operation frequency of the attack device can be used as features of power traces and GSO can be used as a feature selection method. Experimental results show that an acceptable classification accuracy can be achieved when SVM classifier uses these selected features as its input. Compared to other SVM-based profiled attacks, the SVM_{VMD} required the minimum number of traces for successful key recovery. Furthermore, SVM_{VMD} is less sensitive to noise so can be

used well with noisy power traces. In our opinion, this work suggests a new approach for feature extraction from power traces using variational mode decomposition, and this method should also be tested in combination with other feature selection method and learning algorithms for the profiled attacks.

Acknowledgment. This research is funded by Vietnam National Foundation for Science and Technology Development (NAFOSTED) under grant number 102.02–2020.14.

References

1. Kocher, P., Jaffe, J., Jun, B.: Differential power analysis. In: Wiener, M. (ed.) CRYPTO 1999. LNCS, vol. 1666, pp. 388–397. Springer, Heidelberg (1999). https://doi.org/10.1007/3-540-48405-1_25
2. 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). https://doi.org/10.1007/978-3-540-28632-5_2
3. Gierlichs, B., Batina, L., Tuyls, P., Preneel, B.: Mutual information analysis. In: Oswald, E., Rohatgi, P. (eds.) CHES 2008. LNCS, vol. 5154, pp. 426–442. Springer, Heidelberg (2008). https://doi.org/10.1007/978-3-540-85053-3_27
4. Chari, S., Rao, J.R., Rohatgi, P.: Template attacks. In: Kaliski, B.S., Koç, K., Paar, C. (eds.) CHES 2002. LNCS, vol. 2523, pp. 13–28. Springer, Heidelberg (2003). https://doi.org/10.1007/3-540-36400-5_3
5. Heuser, A., Zohner, M.: Intelligent machine homicide. In: Schindler, W., Huss, S.A. (eds.) COSADE 2012. LNCS, vol. 7275, pp. 249–264. Springer, Heidelberg (2012). https://doi.org/10.1007/978-3-642-29912-4_18
6. Lerman, L., Bontempi, G., Markowitch, O.: A machine learning approach against a masked AES. J. Cryptogr. Eng. **5**(2), 123–139 (2014)
7. Zheng, Y., Zhou, Y., Yu, Z., Hu, C., Zhang, H.: How to compare selections of points of interest for side-channel distinguishers in practice? In: Hui, L., Qing, S., Shi, E., Yiu, S. (eds.) ICICS 2014. LNCS, vol. 8958, pp. 200–214. Springer, Cham (2015). https://doi.org/10.1007/978-3-319-21966-0_15
8. Rechberger, C., Oswald, E.: Practical template attacks. In: Lim, C.H., Yung, M. (eds.) WISA 2004. LNCS, vol. 3325, pp. 440–456. Springer, Heidelberg (2005). https://doi.org/10.1007/978-3-540-31815-6_35
9. Gierlichs, B., Lemke-Rust, K., Paar, C.: Templates vs. stochastic methods. In: Goubin, L., Matsui, M. (eds.) CHES 2006. LNCS, vol. 4249, pp. 15–29. Springer, Heidelberg (2006). https://doi.org/10.1007/11894063_2
10. Mangard, S., Oswald, E., Popp, T.: Power Analysis Attacks: Revealing the Secrets of Smart Cards. Springer, Heidelberg (2007)
11. Lomné, V., Prouff, E., Roche, T.: Behind the scene of side channel attacks. In: Sako, K., Sarkar, P. (eds.) ASIACRYPT 2013. LNCS, vol. 8269, pp. 506–525. Springer, Heidelberg (2013). https://doi.org/10.1007/978-3-642-42033-7_26
12. Hettwer, B., Gehrer, S., Güneysu, T.: Applications of machine learning techniques in side-channel attacks: a survey. J. Cryptogr. Eng. **10**(2), 135–162 (2019)
13. Picek, S., Heuser, A., Jovic, A., Legay, A.: On the relevance of feature selection for profiled side-channel attacks. Cryptology ePrint Archive (2017)
14. Bartkewitz, T., Lemke-Rust, K.: Efficient template attacks based on probabilistic multi-class support vector machines. In: Mangard, S. (ed.) CARDIS 2012. LNCS, vol. 7771, pp. 263–276. Springer, Heidelberg (2013). https://doi.org/10.1007/978-3-642-37288-9_18

15. Dragomiretskiy, K., Zosso, D.: Variational mode decomposition. IEEE Trans. Signal **62**, 513–544 (2014)
16. Stephen, B., Parikh, N., Chu, E., Peleato, B., Eckstein, J.: Distributed optimization and statistical learning via the alternating direction method of multipliers. Found. Trends® Mach. Learn. **3**(1), 1–122 (2011)
17. Cortes, C., Vapnik, V.: Support-vector networks. J. Mach. Learn. **20**(3), 273–297 (1995)
18. Platt, J.C.: Probabilistic Outputs for Support Vector Machines and Comparisons to Regularized Likelihood Methods, pp. 61–74. Advances in Large Margin Classifiers MIT Press, Cambridge (1999)
19. Stoppiglia, H., Dreyfus, G., Dubois, R., Oussar, Y.: Ranking a random feature for variable and feature selection. J. Mach. Learn. **3**, 1399–1414 (2003)
20. Eswaramoorthy, S., Sivakumaran, N., Sekaran, S.: Grey wolf optimization based parameter selection for support vector machines. Int. J. Comput. Math. Electr. Electron. Eng. **35**(5), 1513–1523 (2016)
21. Standaert, F.-X., Malkin, T.G., Yung, M.: A unified framework for the analysis of side-channel key recovery attacks. In: Joux, A. (ed.) EUROCRYPT 2009. LNCS, vol. 5479, pp. 443–461. Springer, Heidelberg (2009). https://doi.org/10.1007/978-3-642-01001-9_26
22. Nassar, M., Souissi, Y., Guilley, S., Danger, J.: RSM: a small and fast countermeasure for AES, secure against 1st and 2nd-order zero-offset SCAs. In: Design, Automation & Test in Europe Conference & Exhibition (DATE), Dresden (2012)