SIDE CHANNEL ANALYSIS

WITH MIMO ESTIMATION TECHNIQUES

by

Ge Xiao

Submitted in partial fulfilment of the requirements

for the degree of Master of Applied Science

at

Dalhousie University

Halifax, Nova Scotia

August 2019

Dedication

To my beloved father Dongliang Xiao and mother Zhanping Fu.

TABLE OF CONTENTS

LIST OF FIGURES

ABSTRACT

Encryption and decryption technology have been the focus of cryptography research. Side channel attacks have attracted the attention of cryptanalysts as it is a method to attack encrypted electronic devices against their leaked physical information, which is far more effective than the mathematical methods of cryptographic analysis.

In this thesis, the model of side channel analysis (SCA) attack is considered as a communication system and MIMO channel estimation is used as a new method of obtaining hypothetical leakage information from the power measurements. It reduces the computational complexity of SCA that uses correlation power analysis (CPA) and compensate for the interruption of the leakage information. The least-square (LS) algorithm is used in estimation and Python examples are provided. The final results demonstrate that our method could effectively help to find the correct secret key.

LIST OF ABBREVIATIONS AND SYMBOLS USED

SCA	Side Channel Analysis
CPA	Correlation Power Analysis
DPA	Differential Power Analysis
LS	Least Squares
RSA	Rivest–Shamir–Adleman
MMSE	Minimum Mean-Square
DES	Data Encryption Standard
AES	Advanced Encryption Standard
FIR	Finite Impulse Response
MIMO	Multiple-Input Multiple-Output
SPA	Simple Power Analysis
IOT	Internet of Things
PITA	Portable Instrument for Trace Acquisition
OFDM	Orthogonal Frequency Division Multiplexing
CSI	Channel State Information
SISO	Single-Input Single-Output
SNR	Signal to Noise Ratio
EM	Electromagnetic
HW	Hamming Weight
ML	Maximum Likelihood
LMS	Least Mean Square
SHA	Secure Hash Algorithm
σ_X	Standard deviations of X
σ_X^2	Variance of X

μ_X	Mathematical expected value of X
$r(t)$	Continuous-time function of time t
$r[n]$	Discrete-time function of sample n
$\oplus$	X-OR Operation
k	Key value
b	Plaintext byte
t^+	Pseudo inverse of t

ACKNOWLEDGEMENTS

Rome was not built in a day. I have learned a lot during the two years of my postgraduate study. I always believe that everything is arranged for the best, and I am grateful that I was lucky enough to meet the people who accompany me in these two years.

First of all, I would like to sincerely thank my supervisor, Dr. Zhizhang (David) Chen for his guidance in study and help in life. His serious attitude towards research and great enthusiasm for life have infected me, and I have always regarded him as my role model. He taught me to face difficulties independently and positively, which made me keep an optimistic mind during the whole research process.

Under the guidance of my other supervisor, Dr. Colin O'Flynn, I learnt about the area of side channel analysis and got interested in it. The Chipwhisperer-Lite designed by Dr. Colin showed me relative knowledge about side channel and inspired me that I could apply what I have learned to practice, just like him.

Furthermore, I would like to thank my committee members, Dr.Jean-Francois Bousquet, and Dr.Guy Kember, for their encouragements and feedbacks.

The emails sent by our administrative staff, Nicole Smith and Rebecca Baccardax remind me to finish my paper step by step and I want to thank them for their kind help.

During the process of writing the thesis, the support and encouragement from my closest friends, Krystal Zhang, Ziwa Yu and my boyfriend, Zhengming Wang, gave me great strength. I thank them for always being with me.

Of course, I owe a great debt to my parents, Dongliang Xiao and Zhanping Fu, for their unconditional love and support.

CHAPTER 1 INTRODUCTION

1.1 Background

In the era of rapid technological development, each of us is using different kinds of electronic products. Encryption is widely used in these electronic products, such as the SIM card in the mobile phone, bank cards of major Banks, mobile phone, laptop, etc.; all need to be kept secret, so as to ensure our private information is secure and not exposed. However, there are ways to attack devices to get the information. People used to attack the target to carry out violent cracking of the encryption[1]. With the development of modern encryption technology, the time in the cracking becomes longer, resulting in a greatly increased difficulty in the success. Still the side channel is one of the methods that was used to attack encrypted electronic devices.

Side channel attack (SCA) is a cryptographic analysis technique proposed by Kocher et al in 1991 [1]. With more and more powerful hardware capability and lower implementation cost of SCA, SCA is used to conduct more and more cryptographic devices, which have attracted the attention of cryptanalysts worldwide as it is far more effective than the mathematical methods of cryptographic analysis. Fig. 1 shows the flowchart example of the CPA that use the correlation power analysis (CPA) to decrypt the secrete key of an encrypted device.

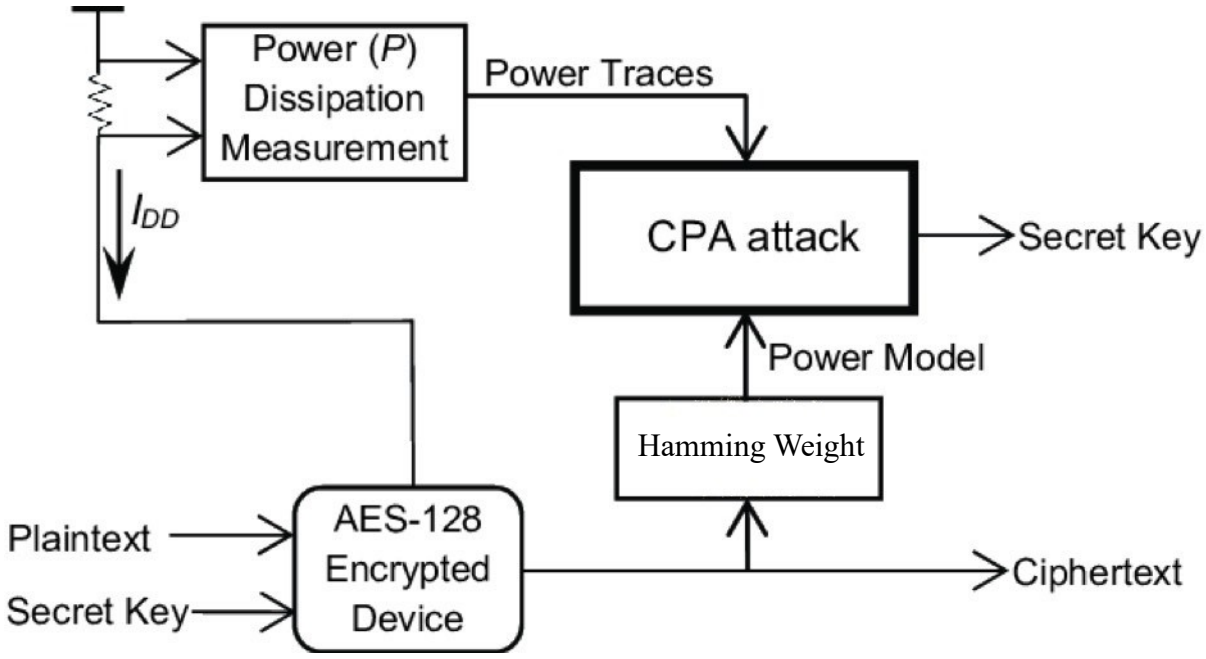


Fig 1.1 The flow chart of CPA[1]

Among all widely used side channel analysis techniques, differential power analysis (DPA) can be regarded as the most powerful attack to block ciphers, such as Advanced Encryption Standard (AES) and Data Encryption Standard (DES) and so on [2]. Correlation power analysis (CPA) is another but more general technique that focuses on the linear relationship between the power consumption and the hypothetical power leakage; it can be viewed as a multi-bit differential power analysis (DPA) method[3].

1.2 Motivation

Nowadays, side channel attack has attracted broad attention. With the development of Internet of Things (IOT), smart home and other technologies, security will become a primary issue of concern, and the need to secure devices and IC chips will increase. The unique feature of side channel attack is that it is not achieved by brute force cracking or algorithm analysis [2] but takes advantage of the design defects of a device (either software or hardware) and cracking is attained through monitoring power consumption, electromagnetic leakage and other

information of the device.

This work considers the side channel with the communication model and simplifies the calculation process of side channel attack by using communication channel estimation techniques. The novelty lines in the fact that it converts an entire set of measured power traces to a single point[5] and uses a straightforward communication method in correlation power analysis (CPA) to acquire the leakage information; as a result, it simplifies the process of obtaining the secret key[6]. In addition, an excellent hardware & software combined toolchain, called Chipwhisperer-Lite, is used, which makes the research threshold low and eliminates the need for complex devices.

The channel model we used is the multiple-input and multiple-output (MIMO) channel and least-square (LS) method used for MIMO channel estimation. Since a large body of open literature has been available for the MIMO channel analysis, we could use them for our analysis and estimation whenever possible.

In the later chapters of this thesis, we will begin with descriptions of multiple-input and multiple-output (MIMO) channel and the least-square (LS) channel estimation, and then the MIMO channel estimation will be introduced and be applied to side channel analysis.

1.3 Contributions

This paper discusses a new method to simplify side channel attacks. The main contributions are (1) the application of the communication channel estimation methods to the side channel attack analysis and (2) development of a new method of obtaining leakage information used in correlation power analysis. They are elaborated as follows:

1) A multiple input multiple output channel estimation is presented with two transmit

antennas and two receive antennas. It can make full use of space resources and increase the capacity and analysis capability of system channel analysis. Compared to the single input and single output (MIMO) channel, it contains more information and it therefore more powerful.

2) The model of side channel analysis (SCA) attack is modeled as a communication MIMO system and analyzed with the MIMO channel estimation technique; with the MIMO theory and techniques developed so far, the computational complexity of SCA with the correlation power analysis (CPA) is reduced by collapsing an entire power trace curve to a single point.

1.4 Outline of the Thesis

This thesis is divided into five chapters and is organized as follows:

Chapter 1 presents the relative information and brief introduction of this thesis, including background, motivations, contributions and organization. Chapter 2 introduces the communication systems first and then discusses the multiple input multiple output (MIMO) channel model and channel estimation using the least square (LS) technique. Data estimation are introduced in this chapter and the simulation results are presented. Chapter 3 gives the introduction to the side channel attack (SCA). The detailed process of performing an attack is presented and the correlation power analysis (CPA) is used as the specific analysis technique for the side channel attack. Chapter 4 presents the applicability of the channel estimation to the side channel attack and the associated results of the attack that involves the hamming weight estimation and correlation power analysis are given. Finally, Chapter 5 draws the conclusion and discusses the future work.

CHAPTER 2 COMMUNICATION SYSTEM AND ESTIMATION

In this chapter, we will first introduce communication systems with basic theory and the matched filter. Then multiple-input multiple-output (MIMO) system and channel estimation are described and the least-square (LS) algorithm used for channel estimation is introduced. Finally, the simulation results of data estimation are presented to check the effectiveness of the channel estimator.

2.1 Communication System

2.1.1 Basic Theory

In the field of communications, additive white Gaussian noise (AWGN) refers to a random wireless noise signal, whose power spectral function is a constant (i.e., white noise), and whose probability density function conforms to Gaussian distribution. The noise signal is an ideal noise signal for analysis, which is not exactly the same as the real-world noise signal but a very good approximation[2].

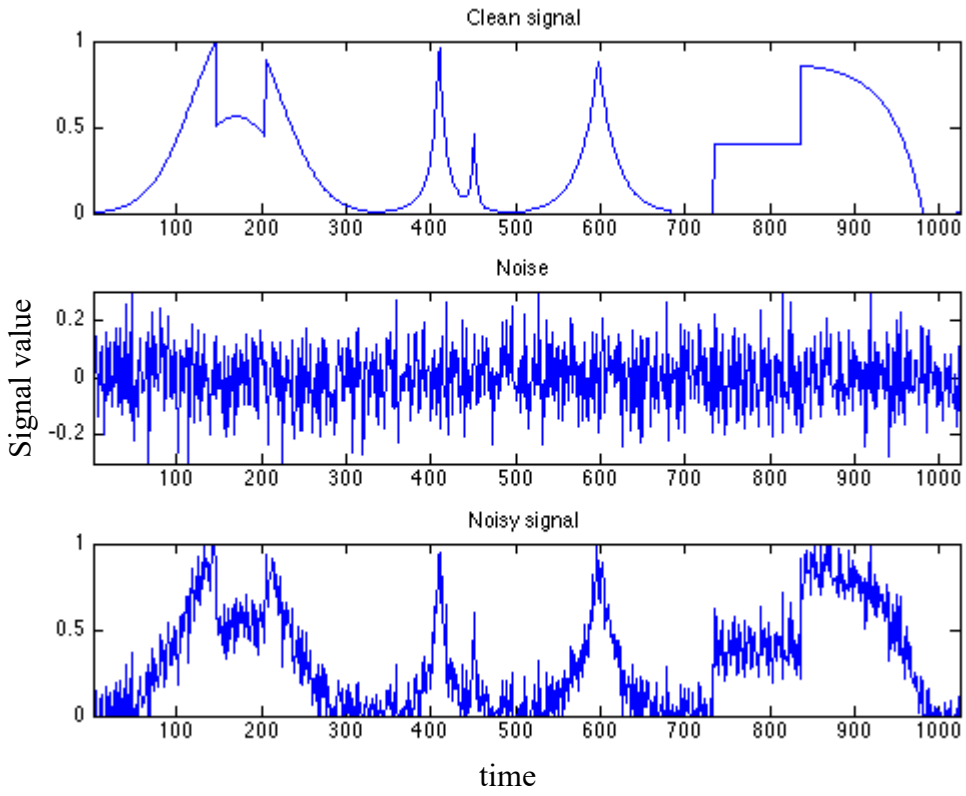


Fig 2. 1 A signal corrupted by noise[1]

In communications, the original signals are often considered to be transmitted through a channel with AWGN. Fig. 2.1 shows the change that a signal corrupted by the noise. The basic problem is how to correctly recover the signals that have been contaminated or corrupted by AWGN. The continuous-time and discrete-time mathematical model can be represented as follows:

$$r(t) = As(t) + n(t) \quad , \tag{2.1}$$

$$r[n] = As[n] + w(n) \quad . \tag{2.2}$$

Here $s(t)$ or $s[n]$ are the transmitted signals or sequences is in the continuous form or in the discrete form, $r(t)$ or $r(n)$ are the received signals in the continuous form or in the discrete form. A is the coefficient related to the channel impulse response, s is one column vector that contains N signals, $s_1(t)$, $s_2(t) \cdots$, $s_N(t)$. r is also the one-column vector that contains the N signals,

$r_1(t), r_2(t) \cdots, r_N(t)$. *n(t)* or *w(t)* refer to the noise that corrupt signals in the continuous form or in the discrete form.

2.1.2 The Matched Filter

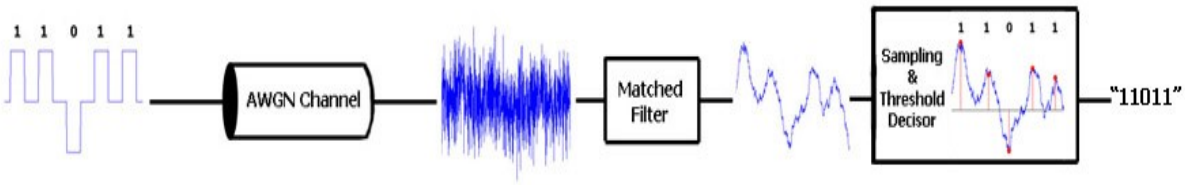


Fig 2. 2 The matched filter system in the binary digital transmission[22]

In an AWGN channel, we can use the matched filter to obtain the best performance in receiving a known signal. The matched filter is an optimal linear filter, which is designed under the maximum criterion of output signal-to-noise ratio (SNR). The matched filter is a very important concept. For example, in binary digital transmission, the matched filter will give the best signal-to-noise (SNR) under which the best two possible signals can be obtained from the noise as shown in Fig 2.2.

Suppose the transfer function of the matched filter is *H(w)* and the corresponding impulse response is *h(t)*. Denotes the input and output of the matched filter as *r(t)* and *y(t)*, shown at Fig 2.3. The input of the matched filter in the figure is

$$r(t) = S(t) + n(t) \quad . \quad (2.3)$$

where *S(t)* is the input signal of the matched filter and *n(t)* is zero-mean gaussian white noise.

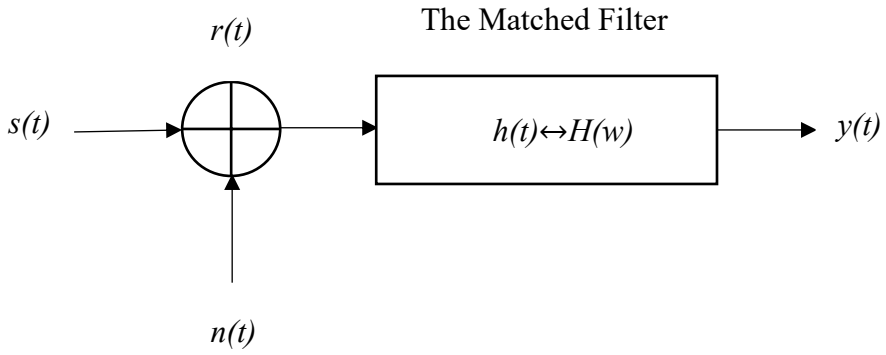


Fig 2. 3 Block diagram of the Matched Filter

The received signal *r(t)* goes through the matched filter:

$$y(t) = r(t) * h(t) = \int r(\tau)h(t-\tau)d\tau \quad . \quad (2.4)$$

Impulse response *h*(*t*) of the matched filter is a time-reversed and shifted copy of the transmitted signal:

$$h(t) = s(T-t), 0 \le t \le T \quad . \quad (2.5)$$

And then we could get:

$$y(t) = \int r(\tau)h(t-\tau)d\tau = \int_0^t r(\tau)s(T-t+\tau)d\tau \quad . \quad (2.6)$$

If it is sampled at *t*=*T*, we could find that:

$$y(T) = \int_0^T r(\tau)s(\tau)d\tau \quad . \quad (2.7)$$

At time *t* = *T*, the output SNR of the matched filter is maximized in reference to the transmitted signal *s(t)*. In other words, we use this filter to convolve the impulse response with our received signal, and then sample the output of this convolution at time *t* = *T*. We just need

to perform *N* convolutions for every known candidate $S_n(t)$. We would be able to select the most likely candidate that becomes the arg max of the convolution at $t = T$. This can be considered as the correlation of received signal r(t) with all candidates $S_n(t)$ at $t = 0$, which would be:

$$arg_n \max(r(t) * S_n(t)|_{t=T}) = arg_n max \int_0^T y(\tau) s_n(\tau) d\tau \quad . \quad (2.8)$$

In communication systems, a known training sets of signals are sent, and the matched filter is used to determine which signal was transmitted specifically. The definition of $s(t)$ is different for side channel analysis, which actually refers to the number of observed cryptographic operations. Function $s_n(d)$ reflects the hypothetical value of the secret key and the matched filter comparison is finished at the same time point in each power trace.

2.2 MIMO System and Channel Estimation

2.2.1 MIMO Channel model

With the rapid developments of mobile communication services, how to effectively use the relatively limited spectrum resources to provide higher and higher quality and data rate transmissions has become the focus of the industry[21]. The conventional single antenna transceiver communication system faces severe challenges and could not solve the problem of large capacity and high reliability requirements of new generation wireless communication system[22]. The multi-input multi-output (MIMO) technology presents a new way to solve this problem, which multiplies the capacity and spectrum utilization of communication systems with no need to increase bandwidth. MIMO technology is one of the most competitive technologies in the next generation mobile communication system[23].

Analyses of MIMO systems are usually based on independent Rayleigh fading channels,

and the spatial correlation of received signals is rarely considered. In practice, when the distance between antenna elements and the angle of arrival wave are small, effects of spatial correlation cannot be ignored. The correlation is more dependent on the spectrum distribution of the angles of signal arrivals. Therefore, in order to construct and analyze MIMO system accurately, it is necessary to systematically analyze and evaluate the spatial correlation of received signals[24].

The communication model we used for channel estimation is Multiple-Input Multiple-Output (MIMO) channel. MIMO technology refers to the use of multiple transmitting antennas and receiving antennas for transmitting and receiving the signals, thus improving the communication quality[23]. Compared with the Single-Input Single-Output (SISO) technology, it has the advantage of making full use of space diversity and multiplicity of signal transmissions and reception. In addition, MIMO technology has the advantages in increasing system channel capacity without increasing antenna transmitting power and spectrum. It is also the core technology of the next generation mobile communications[21].

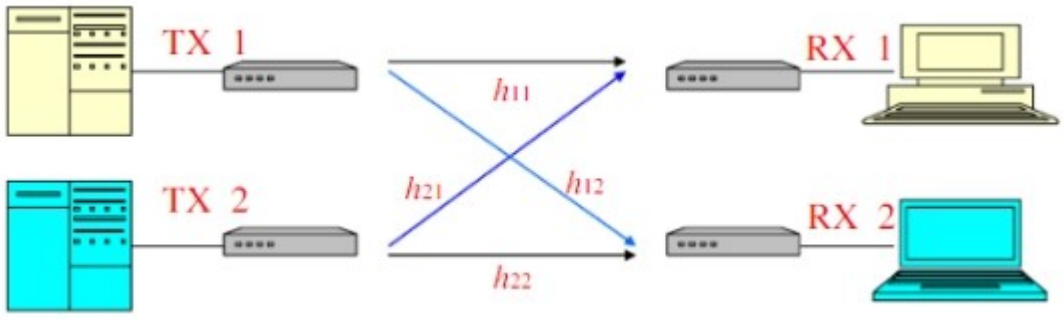


Fig 2. 4 A 2x2 MIMO transmission model[21]

A transmission model for a 2x2 MIMO system is shown in Fig 2.4. It consists of two transmit ($M_t = 2$) and two receive ($M_r = 2$) antennas. h_{11} and h_{22} are the transmission coefficients between the same transmitters and receivers, while h_{12} and h_{21} are the transmission coefficients between the different transmitters and receivers.

The received signal in the system can be written as

$$Y = H\,S + N \quad , \tag{2.9}$$

where *Y, S* and *N* are the received signal vector on the receive antennas, transmitted signal vector on transmit antennas, and additive noise vector, respectively. *H* is the matrix that contains all information about the transmission coefficients of two channels. As matrix *H* could be used to estimate the transmitted signals from received signals, it is considered as channel estimator.

The elements of the noise vector are independent to each other and are of Gaussian random variables with zero-mean and variance of σ_n^2. The correlation of *N* is then given by

$$R_{nn} = E\{N^H.N\} = \sigma_n^2.M_t.M_r\ I_{N_P} \quad , \tag{2.10}$$

where $(.)^H$ denotes the complex conjugate (Hermitian) transpose, *E*(.) is the mathematical expectation, and I_{N_P} denotes the $M_t \times M_r$ identity matrix.

As mentioned before, channel estimation is to recover the channel matrix *H* based on the knowledge of *Y* and *S*. For the 2x2 MIMO,

$$y_1 = h_{11} \cdot s_1 + h_{21} \cdot s_2 + n_1 \quad , \tag{2.11}$$

$$y_2 = h_{12} \cdot s_1 + h_{22} \cdot s_2 + n_2 \quad , \tag{2.12}$$

where $Y = [y_1, y_2]^T$, $S = [s_1, s_2]^T$, $H = \begin{bmatrix} h_{11} & h_{12} \\ h_{21} & h_{22} \end{bmatrix}$, and $N = [n_1, n_2]^T$. The multiplication above should be convolution in time; however, we can consider the special case where no channel delay exists and the convolution becomes a simple arithmetic multiplication; the main reason is that when the MIMO estimation is applied to the side channel attack in the

later chapters, it is used for the data processing where data received at different time instances are considered independent to each other.

2.2.2 Signal Model

Consider a MIMO system with M_t transmit and M_r receive antennas. The *m-th* signals transmitted at time instant k is denoted by $S_m(k)$[27]. The transmitted signals are denoted as:

$$S_m(k) = [S_1(k) \cdots S_{M_t}(k)]^T \tag{2.13}$$

of length M_t , where $(.)^T$denotes the transpose operation. Then, between transmit signal m_1 and received signal m_2, there is a complex single-input single-output (SISO) channel impulse $h_{m_2,m_1}(k)$ of length *K+1 (k=0,1,2,…K)*, described by the vector

$$h_{m_2,m_1} = [h_{m_2,m_1}(0) \cdots h_{m_2,m_1}(K)]^T \ , \ m_1 = 1,2,\dots,M_t \text{ and } m_2 = 1,2,\dots,M_r \ . \tag{2.14}$$

Assume the same length *K* for all the signals, the MIMO channel can be described by *K*+1 complex channel matrix

$$H(k) = \begin{bmatrix} h_{11}(k) & \cdots & h_{1\,M_t}(k) \\ \vdots & \ddots & \vdots \\ h_{M_r\,1}(k) & \cdots & h_{M_r\,M_t}(k) \end{bmatrix} \ , \quad k = 0,\cdots,K \ . \tag{2.15}$$

Of the dimension $M_r \times M_t$. The M_r signals received at the M_r antennas can be represented in a vector:

$$Y(k) = [y_1(k),\cdots,y_{M_r}(k)]^T \quad . \tag{2.16}$$

and

$$y_{m_2}(k) = S_1(k)\, h_{m_2\ 1}(k) + \cdots + S_{M_t}(k)\, h_{m_2\ 1}(k) + n$$

$$= [S_1(k), \cdots, S_{M_t}(k)] \begin{bmatrix} h_{m_2\ 1}(k) \\ \vdots \\ h_{m_2\ M_t}(k) \end{bmatrix} + n \quad . \qquad (2.17)$$

As indicated before, assume the additive white Gaussian noise (AWGN) with zero mean and variance of σ_n^2, the spatial correlation matrix of the noise is then given by

$$R_{NN} = E\{n(k)n^H(k).\} = \sigma_n^2\, M_t\ M_r\ I_{N_r} \quad , \qquad (2.18)$$

where I_{N_r} is the identity matrix and $(.)^H$ denotes the complex conjugate (Hermitian) transpose.

2.2.3 Description of Channel Estimation

Channel estimation is the process of estimating the parameters of an assumed channel model from the analysis of the received data. If the channel is linear, it is then to estimate the impulse response of the system. It is a mathematical representation of the influence of channel on input signal, and an estimation algorithm that minimizes the estimation errors can be referred as a "good" channel estimation[25]. It essentially consists of a set of linear equations, where transmitting and receiving signals are represented respectively by transmitting signal vectors and receiving signal vectors. The actual transmission characteristics, or the current channel state, are summed up in a matrix to represent the channel effects on the signals.

Channel estimation can be carried out in various ways: blind or training-based methods, adaptive or non-adaptive methods, with or without the help of parametric models and so on. Based on the types of input data, it can be conducted in either time domain or frequency domain. The frequency domain method is widely used for multi-carrier systems. However, the time-domain method is applicable to both single-carrier and multi-carrier systems; they are based

on the statistical properties of reference signals or transmitted data[33].

The blind estimation method uses decision feedback based on characteristics of modulated signals that are irrelevant to the specific transmitted information[26]. This method is carried out without any pilot sequence in time domain and frequency domain and channel characteristics are estimated at the receiving end based on statistical information of the received data. Therefore, the spectral efficiency could be improved with no need to send training sequences[37]. However, in mobile wireless communication systems, the environment of the whole channel is time-varying, especially when fast movement occurs and it is hard for channel to be stable and maintain constant statistical characteristics in a certain period of time.

The training-based method is the most widely used method. It sends a certain, known signal for estimation of the channel. Compared with the blind estimation, the estimation is based on training of the system and it is widely used in mobile communication systems due to its higher accuracy, lower complexity and shorter statistical time[28].

The pilot tones can be inserted in symbols and subcarriers. The former is the block pilot for slow fading channel[36]; it works well when the transfer function of the channel does not change rapidly; pilot tones are inserted in each symbol so better performance can be achieved in variable environments. The latter is with comb pilots and could be applied to monitor some fast channels;

Different computational algorithms of the channel estimation have been developed by researchers in recent years. The Least Square (LS) algorithm and Minimum Mean-Square （MMSE）are two common ones. The advantage of the LS channel estimation algorithm lies in its simple structure and low computational complexity. However, the estimated value of LS channel estimation could be easily affected by noise and intercarrier-interference (ICI)[36].

The accuracy of the estimator will be greatly reduced if the interference level is too high. MMSE channel estimation is a Bayesian estimation method. Different from LS estimation, the objective function optimized by MMSE is statistical average of errors rather than determined error energy. When the statistical performance of channels is known, this estimation method shows better performance than LS[34]. This algorithm is not applicable when the channel statistical performance is unknown. In addition, the MMSE estimation is often applied to the frequency domain response of the channel, since it is easier than to the time domain[34].

2.2.4 The Least Square (LS) Technique for Channel Estimation

The computational algorithm of channel estimation we used is the least square (LS) algorithm. It is originated from the field of astronomy and geodesy. Scientists and mathematicians were seeking for the solution to the problem of ocean navigation in the age of exploration. Accurately describing the behavior of celestial bodies is the key to enabling ships to navigate through the high seas, where sailors can no longer rely on land-based observations[34].

In 1805, Legendre published his first concise exposition of the least square method. The technique is described as an algebraic process of fitting linear equations to data[34]. The value of Legendre's least-squares method was immediately recognized by leading astronomers and geodesists of the time.

2.2.4.1 General Operational Principle

Least square is a method in regression analysis to approximate the solution of over determined systems, which is shown in Fig 2.5. “Least squares” means that the overall solution minimizes the sum of the squares of the residuals or errors made in the results of each single equation.

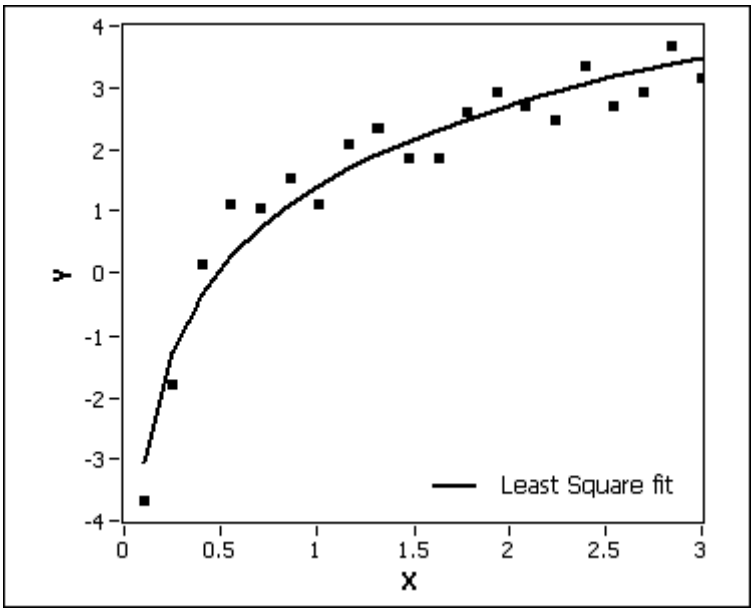


Fig 2. 5 Least Square Fitting

The objective of the least square method is to adjust the parameters of the model function to best fit the data set. For example, there is a simple data set which consists of *j* points (data pairs) (x_i, y_i), *i=1,...,j*, where the value of *y* is dependent on *x*. The form of the model function is $f(x, \beta)$, where adjustable parameters are held in vector β . The goal is to find the parameter values for the model that "best fits" the data. The fitting between the model and data points is measured by its residual, which is defined as the difference between the actual value of the dependent variable and the predicted value of the model:

$$r_i = y_i - f(x_i, \beta) \quad . \tag{2.19}$$

The least-square method finds the optimal parameter values by minimizing the sum, *S*, of squared residuals:

$$S = \sum_{i=1}^{j} {r_i}^2 \quad . \tag{2.20}$$

The least square method can deal with two kinds of problems: linear or ordinary least

squares and nonlinear least squares, which depend on whether the residuals are linear in all unknowns or not. The linear least square problem exists in statistical regression analysis. It has a closed form solution. Nonlinear problems are usually solved by the iterative refinement method. In each iteration, the system is approximately linear, so the core computations are similar to the linear case[37].

Note that we assume that the channel matrix H shown in equation (2.9) is a linear estimator, however, the channel itself may be non-linear. The linear least square method is used to obtain the channel estimator.

2.2.4.2 Linear Least Squares

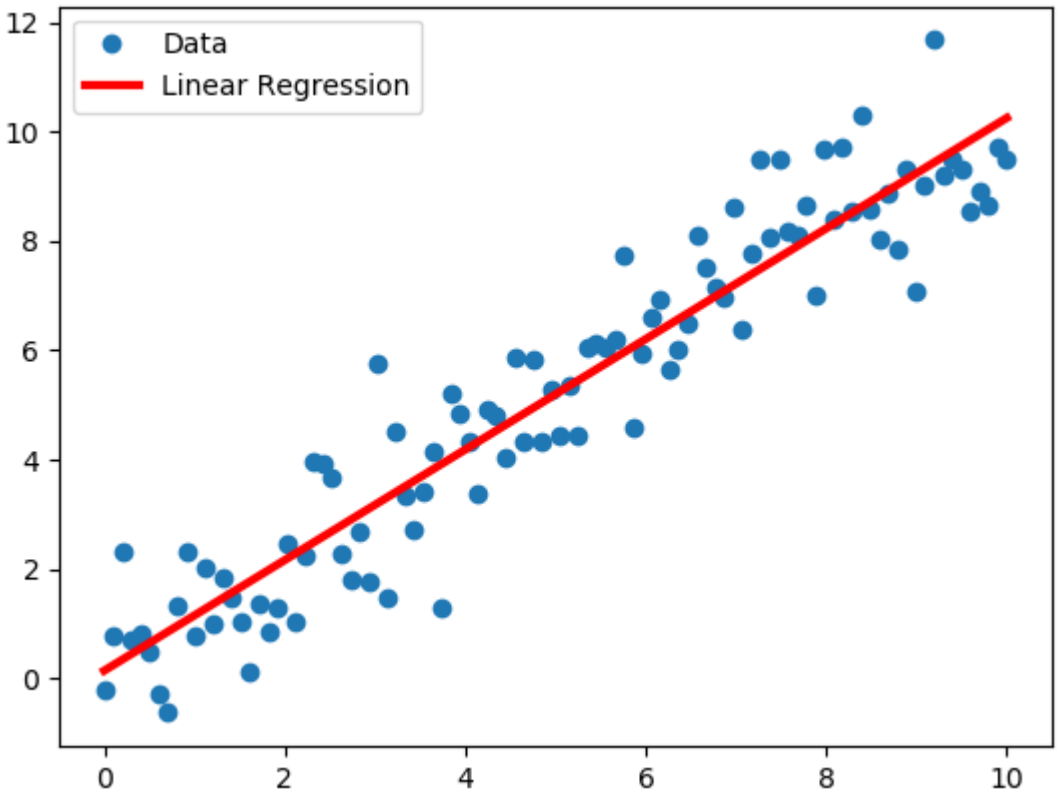


Fig 2. 6 Linear least square fitting example[34]

In the linear least square method, the regression model is considered for the solution shown in Fig 3.3. It is a linear model comprised of linear combinations of the basis functions:

$$f(x,\beta) = \sum_{j=1}^{m} \beta_j \emptyset_j(x) \quad , \tag{2.21}$$

where function $\emptyset_j$ is a known or preselected basis function of x and β_j is the expansion coefficients to be found.

Let

$$X_{ij} = \emptyset_j(x_i) \quad . \tag{2.22}$$

We can then find the least square estimate (or estimator, in the context of a random sample) with β given by

$$\hat{\beta} = (X^T X)^{-1} X^T y \quad , \tag{2.23}$$

where X is a matrix whose element is X_{ij}, y is a vector whose ith element is the ith observation of the dependent variable and $\hat{\beta}$ is a closed-form expression for the estimated value of the unknown parameter vector β. Fig. 3.3 shows an example of the linear LS where the red line is the result.

2.5 Data Estimation

In order to confirm the accuracy of our channel estimator, we need to compare the transmitted signals, received signals as well as the signals go through our channel estimator. Thus "data estimation" are introduced to help us get the estimated transmitted data.

The signals we received are normally influenced by various kinds of energy components such as energy due to inter symbol interference, co-channel interference, adjacent channel interference and noise. These energies are not from the transmitted signals and would increase or decrease the total energy of the received signals. The channel estimator can be used

to detect the transmitted signals from the received signals and the progress of estimating transmitted signals is defined as data estimation, which is given by

$$D = Y\left(\widetilde{H}\right)^{-1} \quad , \qquad (2.24)$$

where *Y, H, D* are received signals, channel estimator matrix and estimated transmitted signals respectively.

Note that training-based method is used to carry out our channel estimation, it means the training sequences with transmitted data values and received data values are known at first. In order to get the channel estimator, the known training sequences are needed for calculation.

2.5.1 Simulation Setup

Based on the above analysis and description, we use Python programming to simulate multi-input and multi-output (MIMO) channels. The MIMO transmission system we used consists of two transmit and two receive antennas[29]. Two transmitters send ten signals at a time. The signal will be contaminated by noise when it passes through the channel; the signal received at the receiving end will be different from the signal sent. Then we do data estimation and restore the received signal to the transmitted signal as much as possible[31].

2.5.2 Simulation Results

Let's try sending two sets of data to see the results. First, send a set of data with all 1's, and then send a set of data with all 2's. The results are shown in Fig 2.7- Fig 2.12.

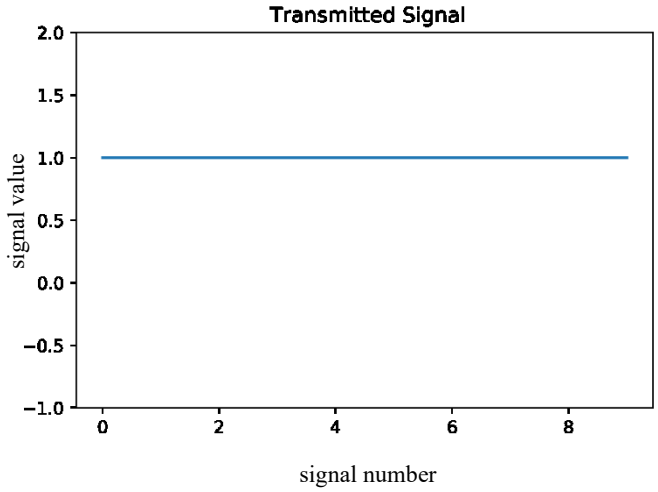


Fig 2. 7 The transmitted signals over two channels: all 1's

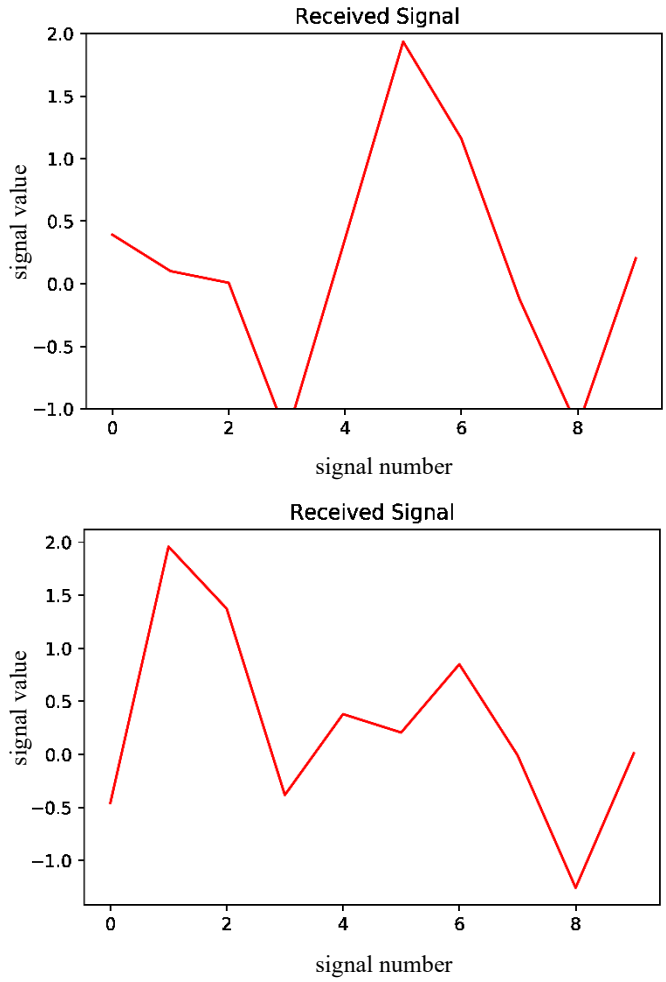


Fig 2. 8 Signal data received by two receivers respectively

From Fig 3.10 and Fig 3.11, we can see that compared with the transmitted signals of all ones, the signal data received are significantly different from the transmitted signals. Let's

look at the estimated signal data results:

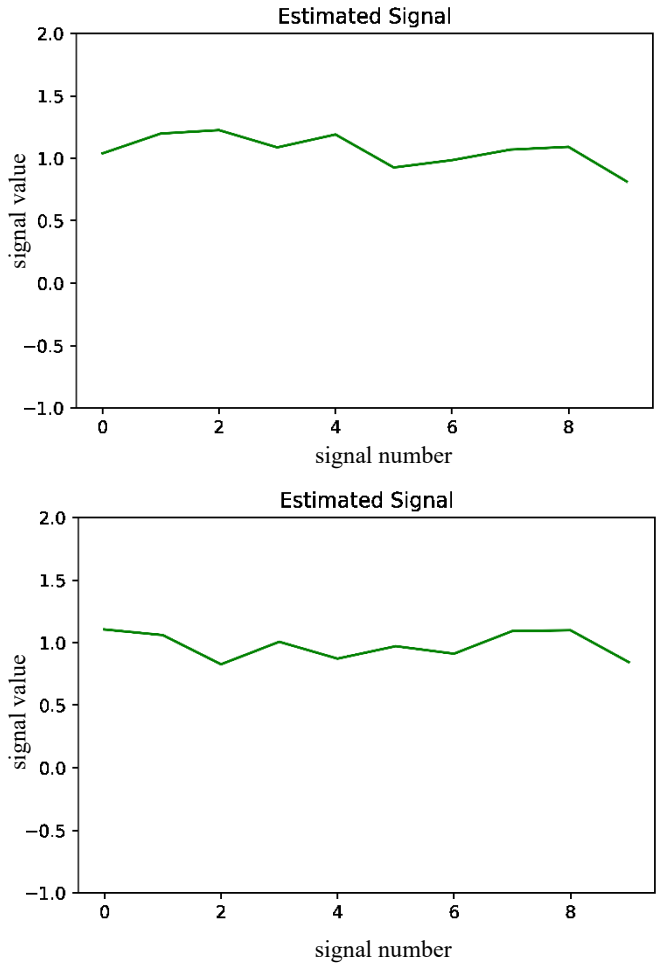


Fig 2. 9 The signals estimated with the channel estimation.

By comparison, it can be seen that the results after MIMO channel estimation are closer to the transmitted signal compared with the previous results.

Let's move on to the next signal transmission, this time we send a set of data with all 2's. The simulation results are as follows.

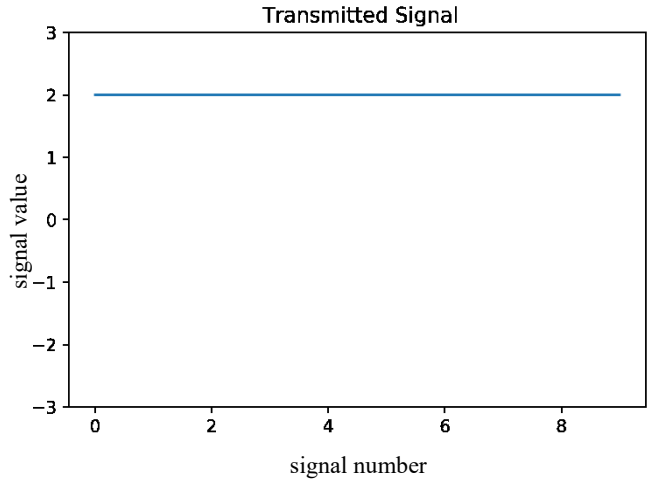
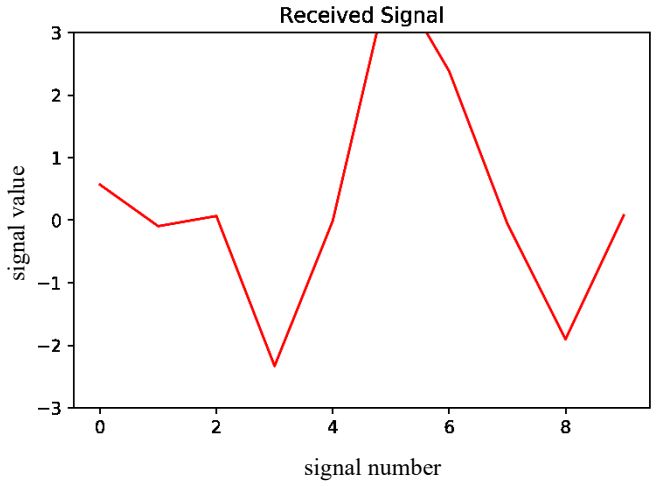


Fig 2. 10 Transmitted signal over two channels: all 2's



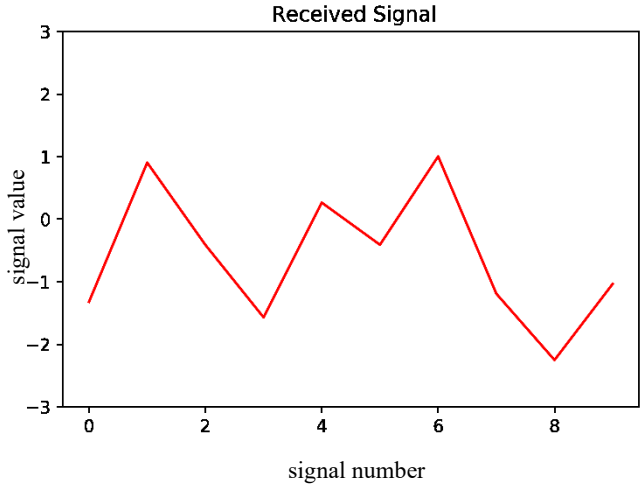


Fig 2. 11 Signal received by the two receivers

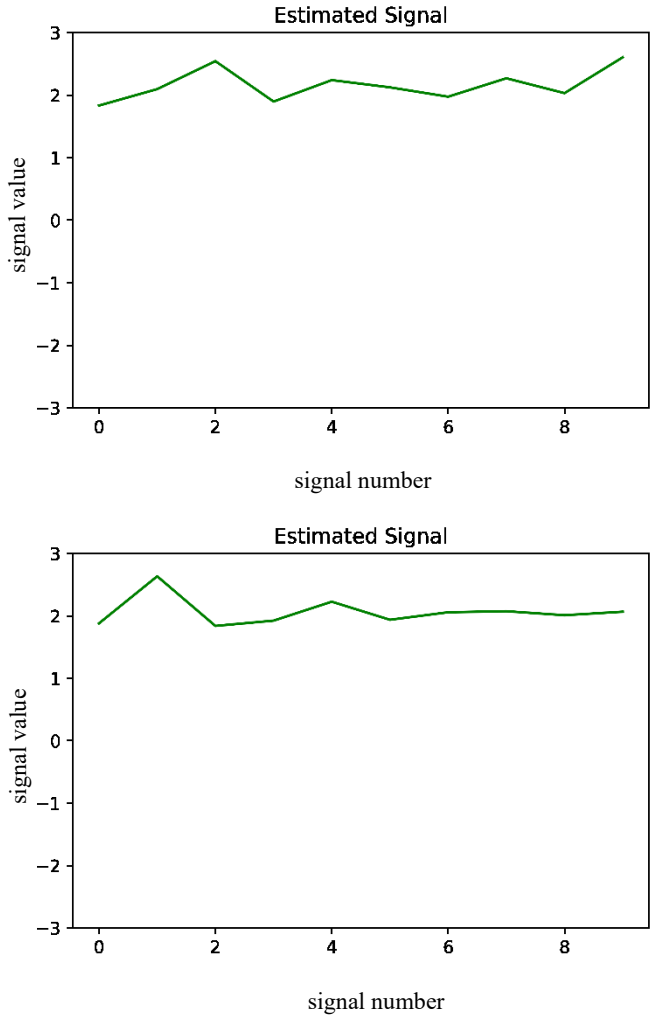


Fig 2. 12 Signal data processed by the estimation matrix of two channels respectively

As seen in Fig 2.10, Fig 2.11 and Fig 2.12, the estimated signals are close to our transmitted ones. But there still uncertainty about the exact transmitted signals between “2” or “3”.

Further simulations are performed with the transmitted data being all “5”s, “6”s and “8”s, respectively. The simulation results are shown below.

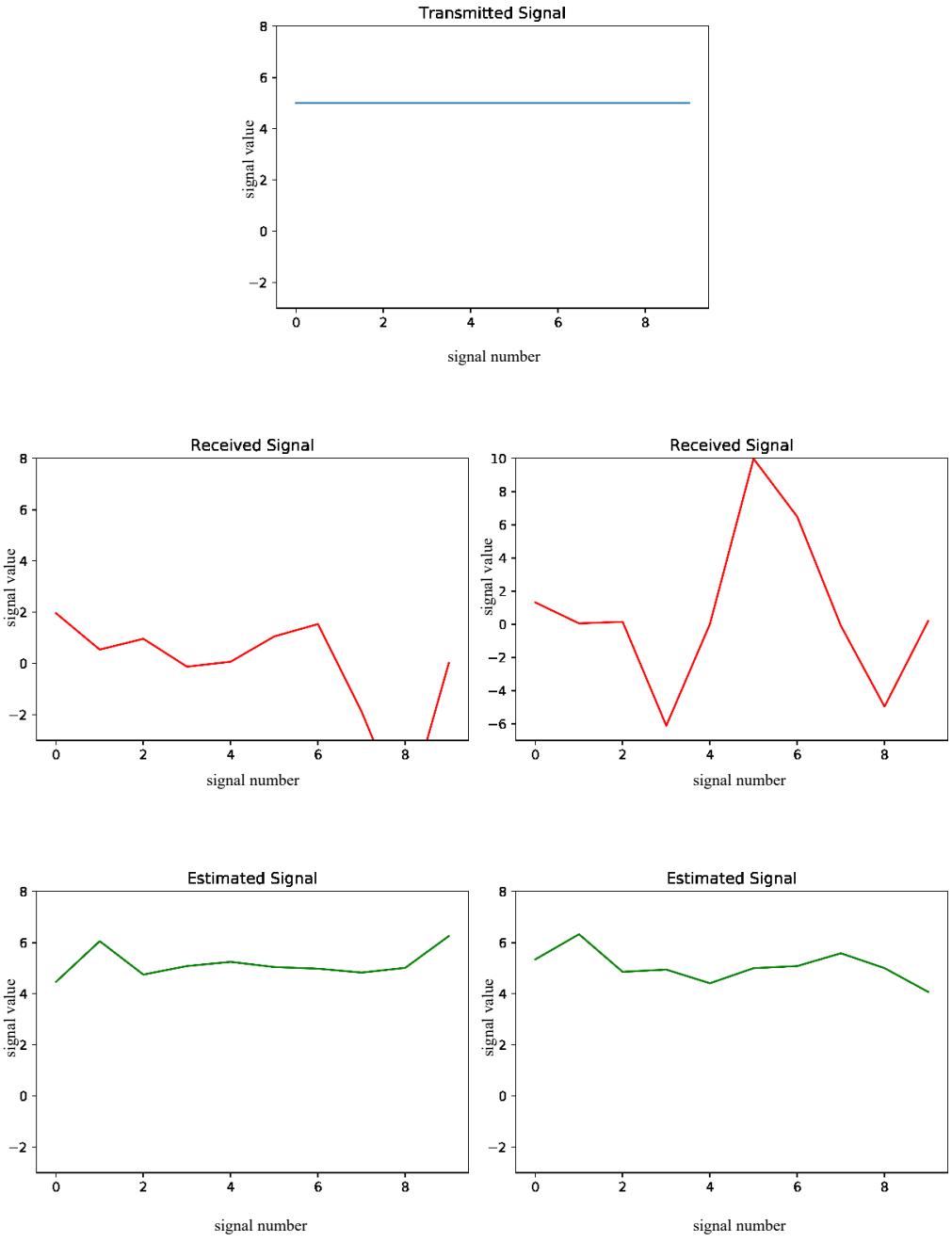


Fig 2. 13 The comparison of transmitted signals (all 5's), received signals and estimated signals

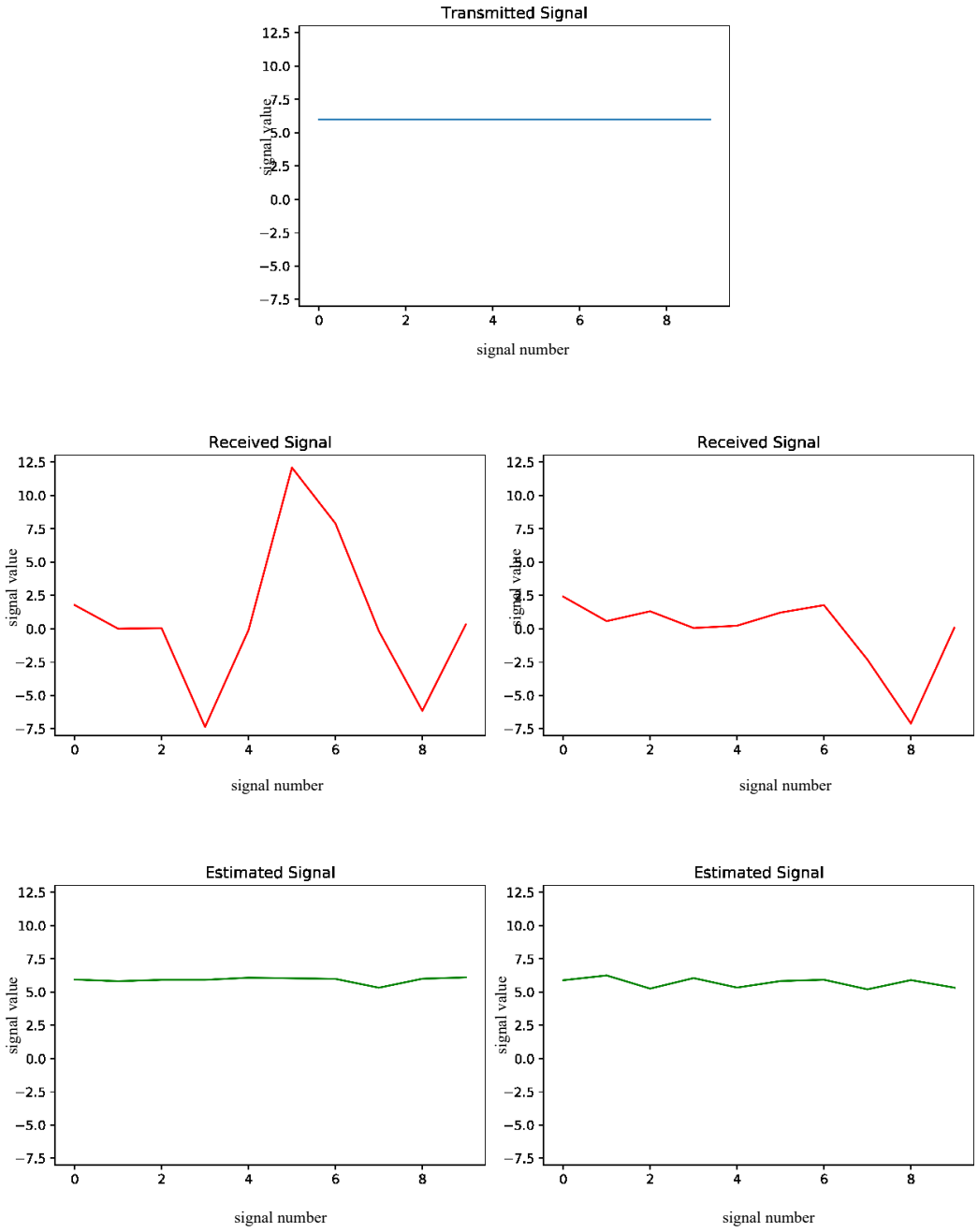


Fig 2. 14 The comparison of transmitted signals (all 6's), received signals and estimated signals

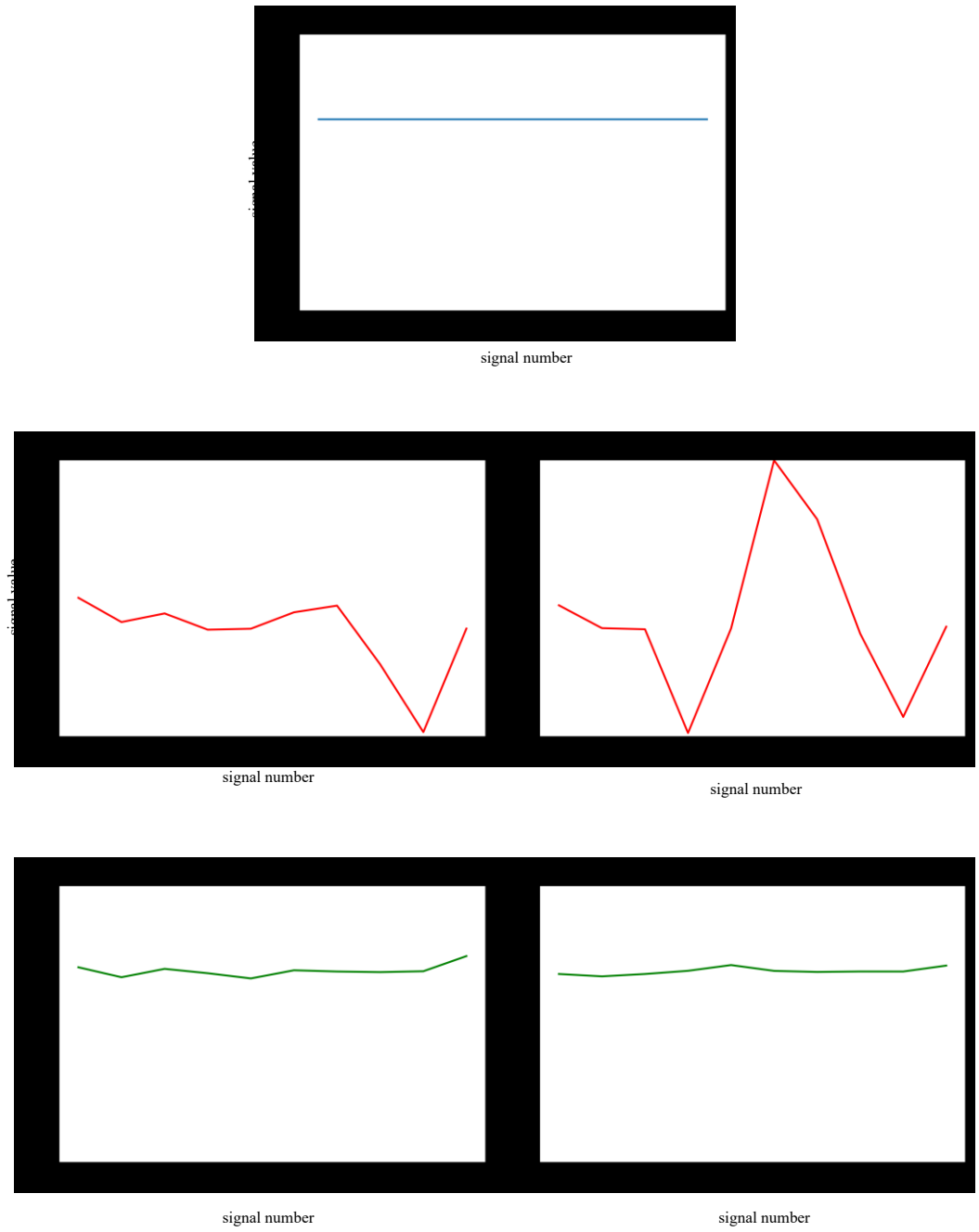


Fig 2. 15 The comparison of transmitted signals (all 8's), received signals and estimated signals

The results show good match between the estimated data and the transmitted data. It shows that the data estimation using the MIMO channel estimation is effective at recovering transmitted signals under noises and interferences.

2.5.3 Overall Comparison of the Results

The above results are for a single transmission of signal data only. In order to observe the overall effect of the data estimation, training sets of 50 were given and variance between transmitted, received and estimated signals was calculated to compare the differences among the groups of data.

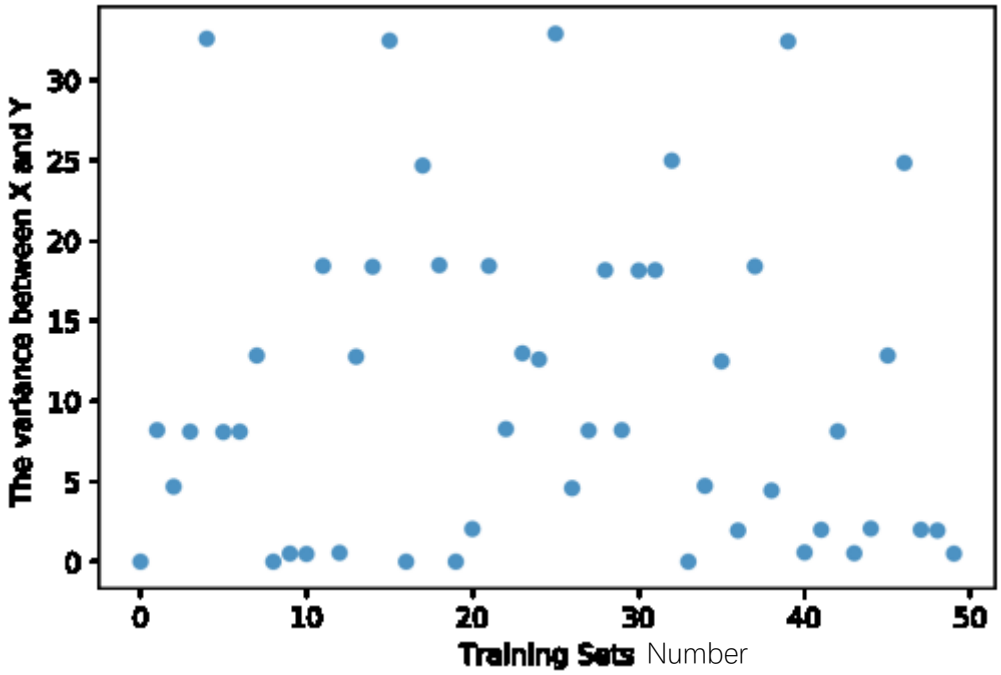


Fig 2. 16 The variance between transmitted and received signals（training sets=50）. *X* refers to the received signal values at received antennas and *Y* refers to the transmitted signal values at transmitted antennas.

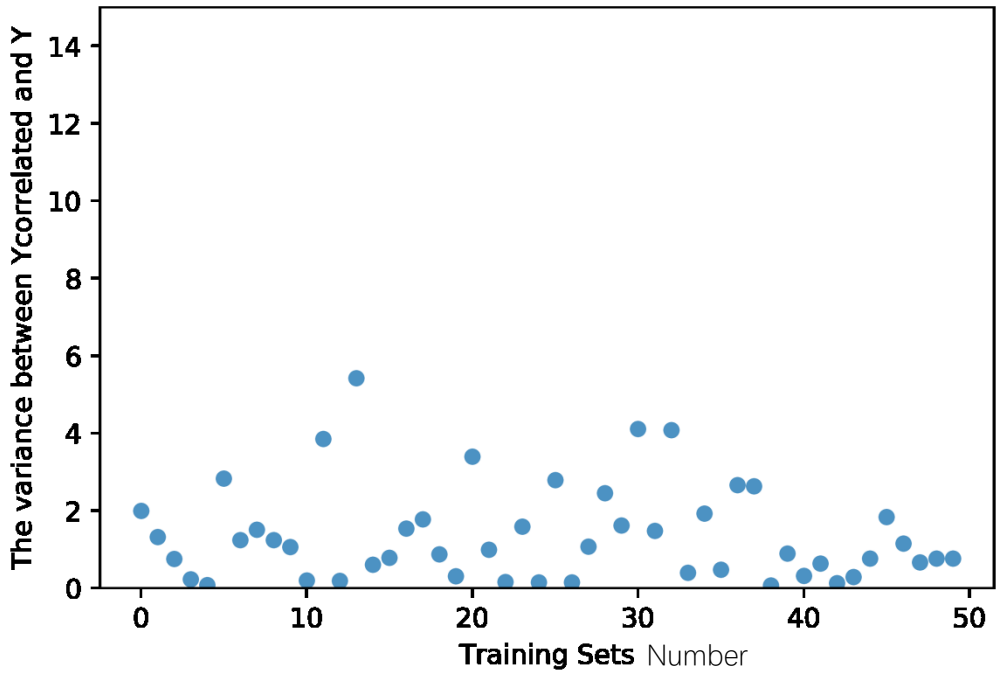


Fig 2. 17 The variance between transmitted and the estimated signals（training sets=50）. *Y* correlated refers to the signal values that have been estimated by the channel estimators and *Y* refers to the transmitted signal values at transmitted antennas.

As seen from Fig 2.16 and Fig 2.17, variances between estimated signals and transmitted signals (training sets =50) is considerably lower than the variances between transmitted signals and received signals.

Next the results from 1000 training sets are shown.

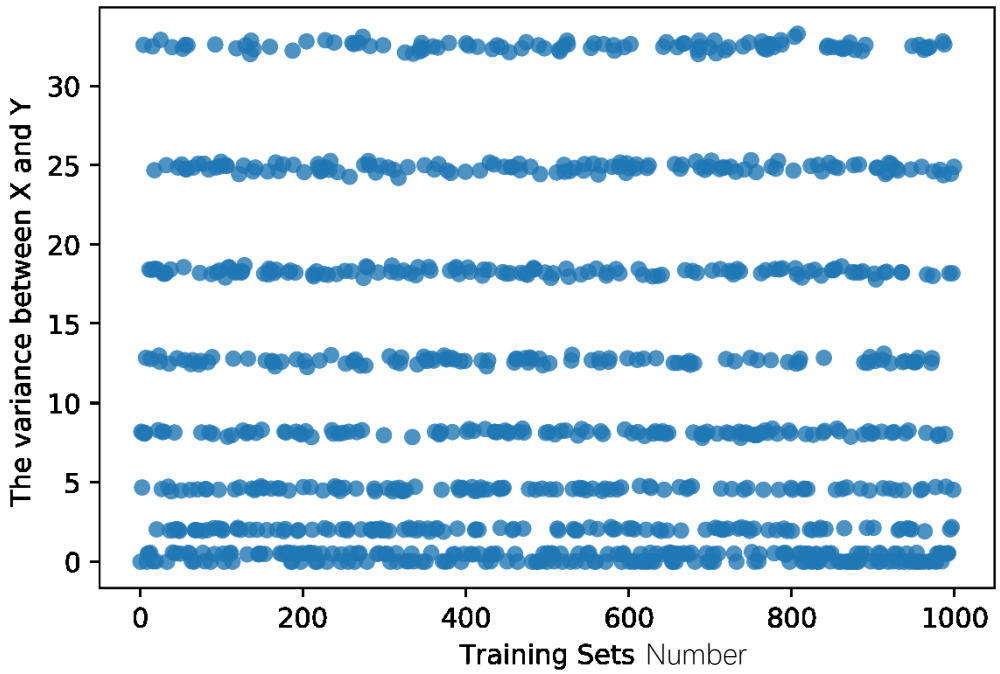


Fig 2. 18 The variance between transmitted and received signals（training sets=1000）. *X* refers to the received signal values at received antennas and *Y* refers to the transmitted signal values at transmitted antennas.

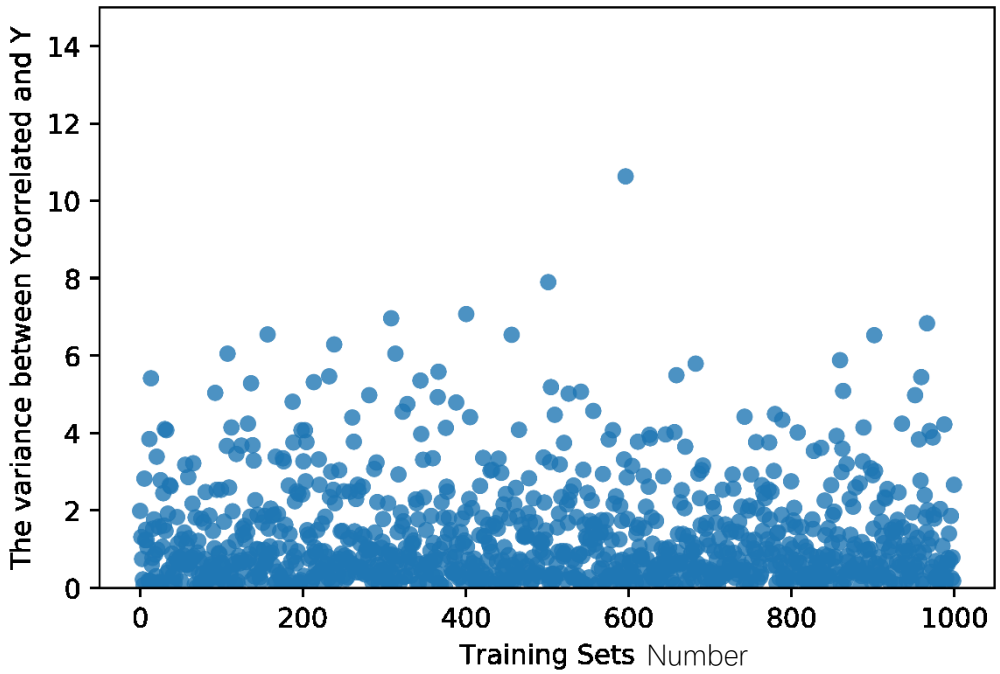


Fig 2. 19 The variance between transmitted and estimated signals（training sets=1000）.

***Y*correlated refers to the signal values that have been estimated by channel estimators and *Y* refers to the transmitted signal values at transmitted antennas.**

As seen from Fig 2.18 with Fig 2.19, the difference between our estimated signal and transmitted signal (maximum variance is around 11) is much smaller than the difference between received signal and transmitted signal (maximum variance is 34). Our data estimation results are positive and valid in most cases.

2.6 Summary

This chapter introduces communication systems and MIMO channel estimation. The Least Square (LS) algorithm and its basic principle presented. The performance of the data estimation using the least-square (LS) channel estimator is described and its performances are assessed by simulations in comparison of transmitted, received and estimated data. A good estimation is obtained when a reasonable number of training sets are selected, and its effectiveness lays the foundation for later use in side channel attacks.

CHAPTER 3 SIDE CHANNEL ATTACKS

The objective of this chapter is to give a detailed description of the side channel attack (SCA), which includes its classifications and applications, the methods to measure power consumption, tools used in SCA and the attack process. Then the correlation power analysis (CPA) are discussed as the specific technique we used to do side channel attack.

3.1 Introduction to the Side Channel Attacks (SCA)

During world war II, secure typewriter communications were developed by Bell Telephone for military use. “When one of these mixers was being tested in a Bell laboratory, a researcher noticed, quite by accident, that each time the machine stepped, a spike appeared on an oscilloscope in a distant part of the lab. After he examined these spikes more carefully, he found that he could read the plain text of the message being enciphered by the machine!” [39] This is the origin of side channel attacks (SCA)[4]. In 1951, it was discovered that a plain text could be recovered one mile away. In 1985, Dutch scientist Wim Van Eck published his research paper demonstrating that computer monitor emissions could be detected and displayed on televisions in adjacent buildings[40].

Side channel attack is a method to attack encrypted electronic devices against their leaked physical information (such as time consumption, power consumption or electromagnetic radiation) during operation[4]. This new type of attack is far more effective than the mathematical methods of cryptographic analysis; it poses a serious threat to cryptographic devices[8].

3.1.1 Classifications

There are several broad categories of side channel attacks depending on the medium

used, and here we introduce three main types[3]:

1) Timing attack. Timing attacks generally infer the operation used by a device based on time of the operation, or by comparing the time of the operation to infer the storage location of the data, or by using the communication time difference to steal relevant data. Timing attack is not a theoretical attack method; OpenSSL, OpenSSH and other applications have had timing attack vulnerabilities. For example, suppose that a function is set up to make comparisons between the password in the system and from the user; if the function begins with the first comparison, find and return the difference immediately, we can calculate rate of the returns and find which bit of the key begins to be different[9]. This allows the successful breaking the code bit by bit, just like in a movie. The complexity of the password cracking is then reduced by thousands or even millions of times.
2) Electromagnetic attack. Some equipment will leak out of electromagnetic radiation in the process of operation; relevant information (such as text, voice, image, etc.) could be figured out from electromagnetic leakages after proper analysis. This attack method is widely used in cryptographic attack and could be used in some non-cryptography attacks such as eavesdropping behavior and the tempest attack (for example, Van Eck phreaking and radiation monitoring).
3) Power-monitoring attack. Power-monitoring attack is seen as one of the most common side-channel attacks in modern computer systems. The power consumption of different hardware circuit units in the same device is different. Therefore, the power consumption of the device varies depending on the hardware circuit unit that is in use. This information could be used to determine the data stored in a device. Moreover, techniques, such as the simple power analysis (SPA), differential power analysis (DPA) or correlation power analysis (CPA), can be applied to the measured power traces to

decipher the secret keys[5].

3.1.2 Applications

Side channel attack has attracted the attention of researchers in many countries in recent years. At Crypto 2008 conference, Eisenbarth et al. published the use of energy attack technology to crack KeeLoq Remote Keyless Entry Systems, one of the typical remote-control car locking systems[9]. At FC 2013 conference, researchers published a successful energy attack on the mobile phone card which uses the comp128-1 algorithm. At Black Hat 2015 conference, researchers from Shanghai Jiao Tong University demonstrated how to use energy attack technology to crack 3G/4G sim cards[13].

The mostly used side channel analysis method so far is power analysis. From power analysis, relationship between the cracking information and the intermediate value generated by the sensitive security parameters or the processed secret key could be figured out[8]. Most of the side- channel attacks, timing attack, electromagnetic radiation attack, sound attack, etc., follow the ideas of power or energy analysis[6].

A research team from Tel Aviv university have made significant advances in the power analysis. Starting in 2014, they published several actual attacks on PC software using the power analysis. At Crypto 2014 conference, they successfully demonstrated the cracking of 4096-bit RSA (Rivest–Shamir–Adleman), one of the most secure encryption algorithms, by listening to high-pitched sounds generated by users' computers to decrypt data[9]. At CHES 2015 conference, they took another step forward, showing a way to steal computer keys by using a specially designed machine that can collect electromagnetic signals from computers within 50 meters; the collected signals are stored in a microSD card, and then the secret key could be deduced after the side channel analysis[10]. The machine designed by them is called Portable

Instrument for Trace Acquisition (PITA)[9], which is an untethered measurement device for low-bandwidth key extractions from electromagnetic attacks. The structure of the machine and the measurement process are shown in the following figure:

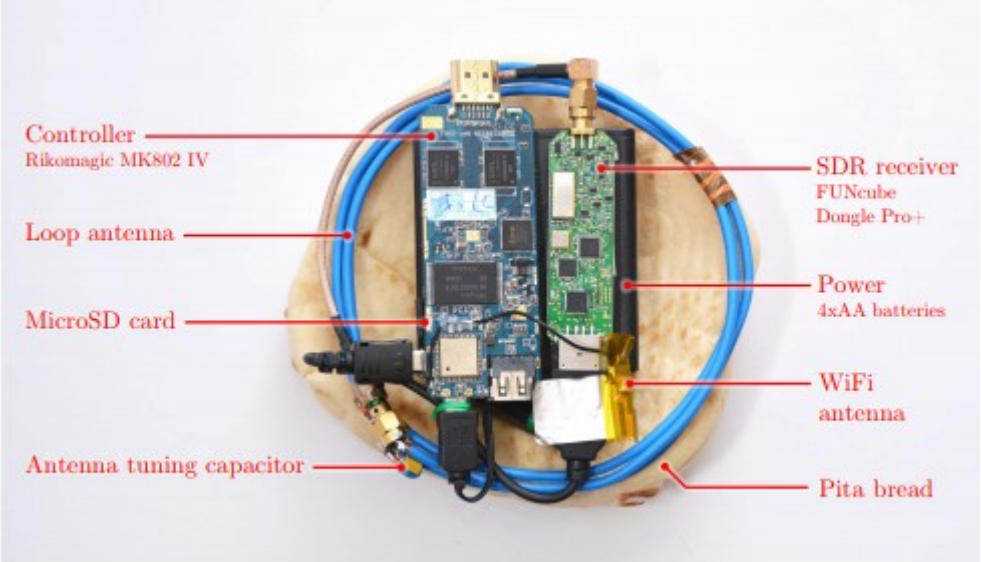


Fig 3. 1 The structure of the portable instrument for trace acquisition (PITA)[19]

Fig 3. 2 Schematic diagram of the measurement process. The PITA first measures the target computer (left) at a specific frequency band and then transmit digital signals over Wi-Fi to the attacker's computer (right) in real time.[19]

3.2 Side Channel Attack with Power Measurements

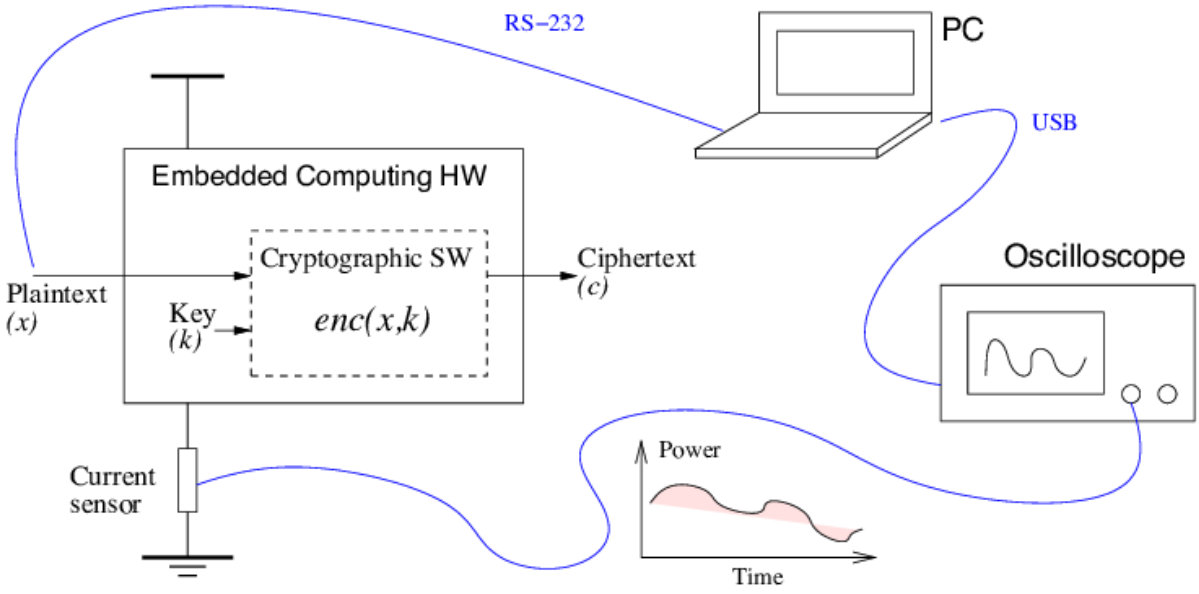


Fig 3. 3 The side-channel attack setup[1]

The process of side channel attack is described in[13]. It is aimed at getting the side channel information leakage such as the time consumption, power consumption or electromagnetic radiation of encrypted electronic devices. In our case, the attack is based on acquisition of the power consumption[20]. Fig 3.3 shows the power measurement set up of the side channel attack.

There are generally two ways to measure the power consumptions: one is to use a shunt resistor and the other is to use an electromagnetic probe.

A resistor which has a low value of resistance connected in parallel with another resistor is called shunt resistance as shown in Fig 3.4. A shunt resistor is mainly made of materials with low-temperature coefficient of resistance, such as insulating ceramics with evenly distributed holes[7]. The shunt resistor has a high-precision resistance that can be used to measure the current flowing through a circuit. According to ohm's law, we know that the voltage over the resistance divided by the resistance of the resistor is equal to the current. Since the biasing

voltage of a circuit is normally fixed, the current fluctuations measured then reflect the power fluctuations[15].

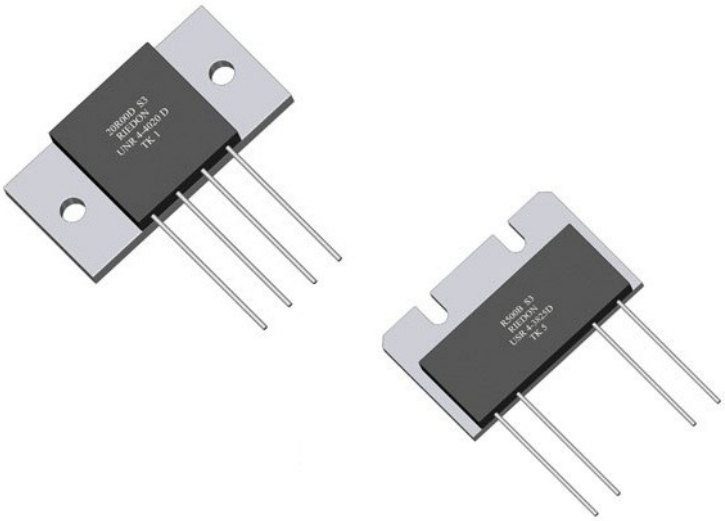


Fig 3. 4 The high-precision shunt resistor

An electromagnetic (EM) probe shown in Fig 3.5 is a simple metal probe that captures the electromagnetic leakage of the chip, which is widely used and has achieved good results[7]. The probe can directly obtain the information leaking out of the chip. One advantage of the electromagnetic probe is that it does not need to change the internal structure of the device being attacked. There have been the reports of successful attacks using only $2 electromagnetic probes in extracting secret keys from the bitcoin software running on IOS devices. The reports also indicate that such attack scenarios are not easy to implement. However, given enough time and the right tools, most devices can be hacked[19]. If the device is placed in the hands of an attacker, owner of a phone will be completely unaware that the private key data has been stolen. Smaller probes can be used to scan the chip face in order to pick out specific features such as data lines[16].

Fig 3. 5 Electromagentic probe set

3.3 Main Tools Used for our Attack

3.3.1 Chipwhisperer-Lite Attack Platform

We use a toolchain called Chipwhisperer-Lite shown in Fig 3.6 for our side-channel power analysis. Chipwhisperer-Lite is an ultimate toolchain concentrating on embedded hardware security, which is maintained by NewAE Technology Inc. It combines software and hardware and provides training to enable users to master hardware security issues.

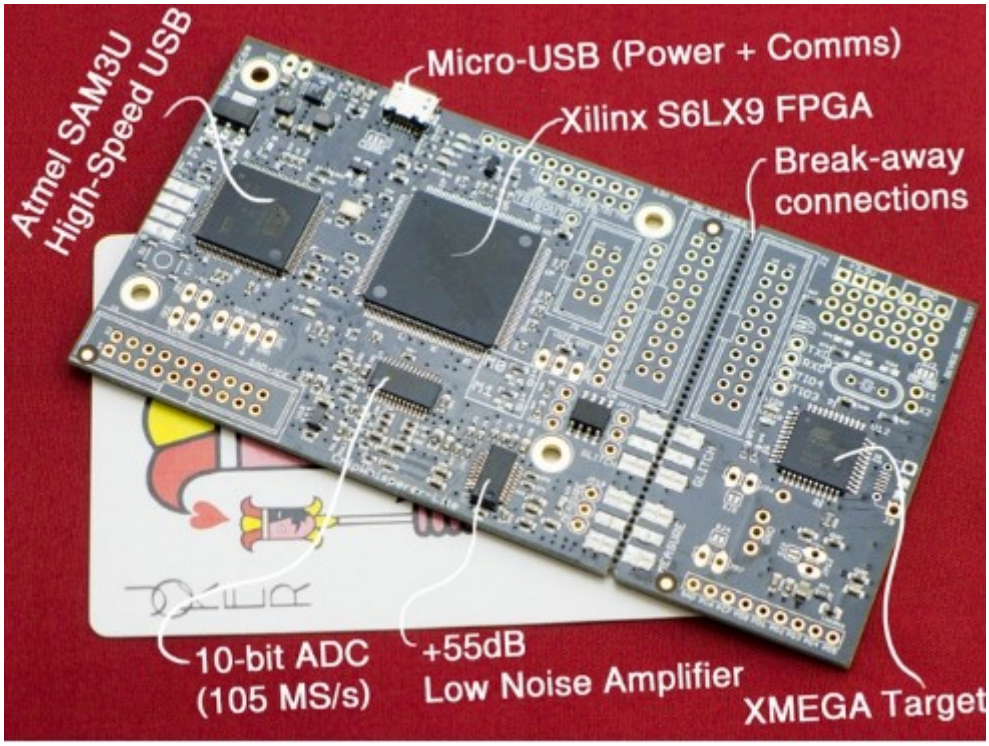


Fig 3. 6 Chipwhisperer-Lite

In our measurement, we have two items in order to perform a side channel attack. The first one is the target circuit board, it is a standard microcontroller which you could implement algorithms onto. The second one is a multi-purpose power analysis capture instrument. Chipwhisperer-Lite have the capture hardware and the target both. Its' software is open-source and available online freely, which is friendly for beginners.

3.3.2 Programming Language

The programming language we use is Python. Python is an object-oriented, dynamically typed language; it was originally intended for writing automated scripts(shell) but is now increasingly used for development on large, independent projects as versions are updated and new features are added.

Python is an interpretive scripting language which can be used at web and Internet developments, scientific computing, software development, desktop interface development,

and so on. The Python language was born in the early 1990s by a Dutchman named Guido van Rossum. Inspired by the British television comedy Monty Python's Flying Circus, which first aired in the 1970s, he chose Python as the name of the programming language. The development environment of the Python language and its numerous extension libraries are ideal for engineering and scientific researchers to process experimental data, make charts, and even develop scientific computing applications. Python maintains a clear and uniform style in its design, which makes it a widely used language that is easy to read, easy to maintain, and popular with a large number of users.

Compared with MATLAB, Python is an easier and more rigorous programming language to learn. It allows users to write code that is easier to read and maintain. MATLAB focuses on engineering and scientific computing. However, even in the field of computing, file management, interface design, network communication and other requirements are often encountered. Python, on the other hand, has a rich library of extensions that allow developers to easily accomplish a variety of advanced tasks, allowing developers to use Python to implement all the functions needed for a complete application.

When Python is executed, the source codes in the file are first compiled into byte codes, which are then executed by the Python Virtual Machine. One of Python's design goals is to make code highly readable.

It is designed to use punctuation marks and English words often used in other languages to keep the code clean and beautiful. A python example is shown in Fig 3.7.

PythonExample.py

```
# Program published on https://beginnersbook.com

# Python program to perform Addition Subtraction Multiplication
# and Division of two numbers

num1 = int(input("Enter First Number: "))
num2 = int(input("Enter Second Number: "))

print("Enter which operation would you like to perform?")
ch = input("Enter any of these char for specific operation +,-,*,/: ")

result = 0
if ch == '+':
    result = num1 + num2
elif ch == '-':
    result = num1 - num2
elif ch == '*':
    result = num1 * num2
elif ch == '/':
    result = num1 / num2
else:
    print("Input character is not recognized!")

print(num1, ch , num2, ":", result)

```

Fig 3. 7 Python Example

3.2.3 The Virtual System

We use a virtual machine called VirtualBox to run the whole side channel attack process. The virtual system has exactly the same function as the real windows system. After entering the virtual system, all operations are carried out in this new independent virtual system. We can install and run software independently, save data, and have our own desktop, which will not have any impact on the real system. We can switch between the existing system and the virtual image flexibly. The virtual system will not degrade the performance of the computer. Starting the virtual system does not need to be as time-consuming as starting the windows system. It is more convenient and faster to run the program.

3.3 Execution of an Attack

After the code is written, we could start to implement our side channel attack. First, open the virtual system software, connect the computer to the Chipwhisperer-Lite and the software interface is shown in the figure:

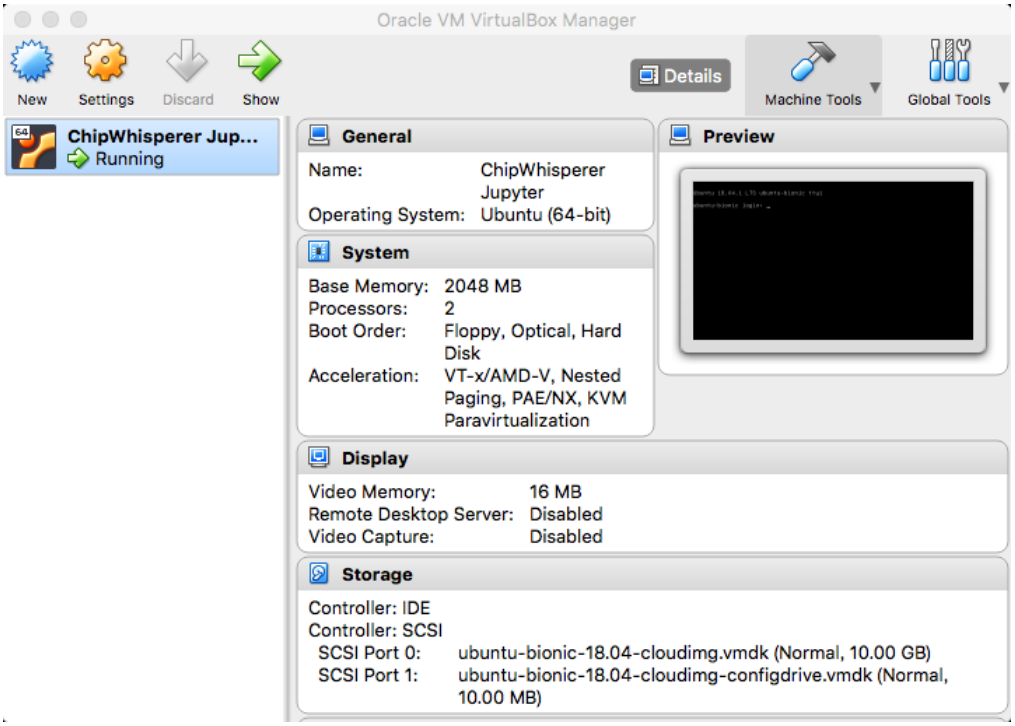


Fig 3. 8 Main page of VirtualBox

Then click the run button. A page will pop up, as shown below:

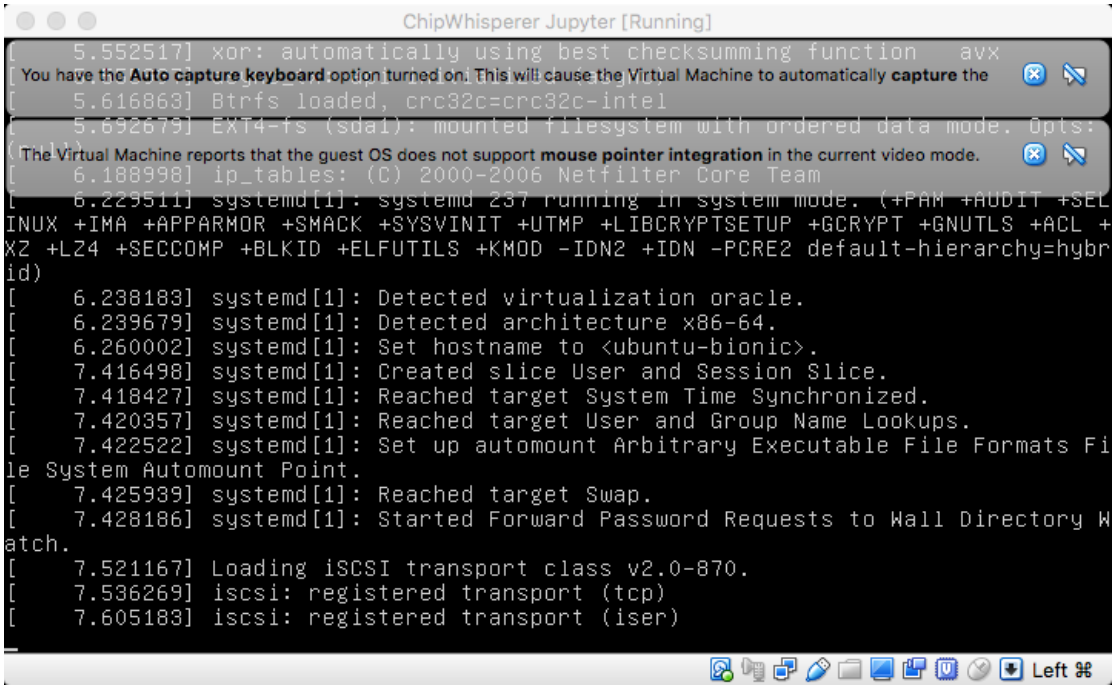


Fig 3. 9 Loading interface display

After the loading, the following screen appears

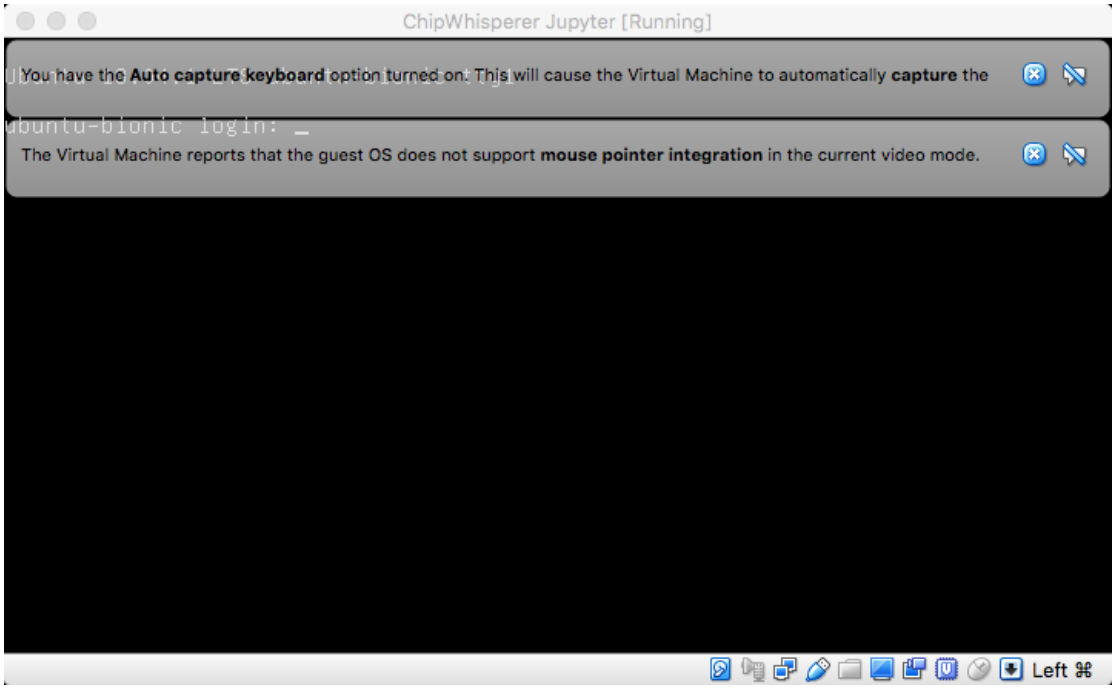


Fig 3. 10 Operation interface display

This shows that the connection was successful. After finishing the setup, the real data

could be obtained from the simulation by Python example.

3.4 Correlation Power Analysis (CPA)

The specific method for us to do side channel attack analysis is called the correlation power analysis (CPA). CPA is a technique that allows us to find the encryption key of a device. It has been widely used in total block ciphers and public key ciphers. It uses the linear correlation coefficient in statistics to analyze the key[2].

The power consumption of an electronic computer (microcontroller, FPGAS, etc.) consists of two main components. The first is the static power which is the amount of power needed to keep a device running normally[7]. This static power is usually affected by the number of transistors inside a device. The second but more important component is the dynamic power consumption which is largely dependent on data moving within a running device.

One of the most widely used and simplest analysis models to analyze power consumption is the Hamming weight model. The Hamming weight of a string is the number of symbols that are different from the zero-symbol of the alphabet used. In this case the classic hamming weight (HW) correlation power analysis (CPA) assumption is used.

When we successfully model power consumption, we need a way to compare our power estimate to our measured traces. A helpful tool to figure out this is through Pearson's correlation coefficient, which is defined as:

$$\rho_{X,Y} = \frac{cov(X,Y)}{\sigma_X \sigma_Y} = \frac{E\,[(X-\mu_X)(Y-\mu_Y)]}{\sqrt{E[(X-\mu_X)^2]E\,[(Y-\mu_Y)^2]}} \quad , \qquad (3.1)$$

where μ_X and μ_Y are the mathematical expected values of X and Y, σ_X and σ_Y are the standard deviations of X and Y, $E(.)$ is the operator of the mathematical expected value, and cov(.) is the covariance.

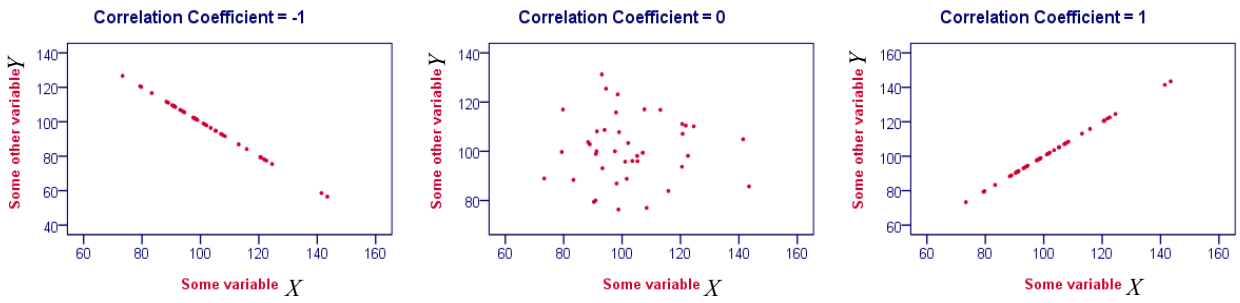


Fig 3. 11 Examples of the Pearson correlation coefficients

Examples of the correlation coefficient is shown in Fig 3.11 and it will always be within the range of [-1, 1]. It is a quantity that represents the degree of the relationship between two random variables. It reflects how closely the random variables X and Y are related or the normalized cross-correlation.

In a side channel attack, the two variables we need to compare is measured power consumption (power traces) and the hypothetical power leakage (hamming weight). Among different key hypothesis, the one that makes the correlation coefficient $\rho_{X,Y}$ largest is accepted as the correct secret key.

We assume that D power traces of t are obtained and each of these traces has T data points in time. Denote $t_{d,j}$ as point j in power trace d $(1 \leq d \leq D, 0 \leq j \leq T)$, $p_{d,i}$ is the hypothetical power consumption (hamming weight) of hypothesis i for trace number d, $r_{i,j}$ as the correlation coefficient at point j for hypothesis i, $p_{d,i}$. Then, with the definition of equation (3.1), X=p, and Y=t, the correlation coefficient is computed as:

$$r_{i,j} = \frac{\sum_{d=1}^{D} [(p_{d,i} - \overline{p_i})(t_{d,j} - \overline{t_j})]}{\sqrt{\sum_{d=1}^{D}(p_{d,i} - \overline{p_i})^2 \sum_{d=1}^{D}(t_{d,j} - \overline{t_j})^2}} \quad . \qquad (3.2)$$

The above equation is widely used, most typically in image processing applications. It is used to match images to known templates[18]. The correlation coefficient reflects the similarity between an image diagram and the template (or reference) diagram.

3.5 Summary

In this chapter, detailed description of side channel attack (SCA) are given. We use Chipwhisperer-Lite as our attack platform and python language is used for programming. Virtual Box is introduced to run the attack process and then the correlation power analysis is presented as the specific method to do side channel attack analysis with its major equation.

CHAPTER 4 SIDE CHANNEL ATTACK WITH MIMO CHANNEL ESTIMATION TECHNIQUE

The objective of this chapter is to connect channel model with side channel attack and apply channel estimation to side channel attack. Then the real data obtained from the attack such as power trace curves and hamming weights are presented as the known training sequences of hamming weight estimation. The real attack results containing hamming weight estimation result and correlation power analysis results are given at last.

4.1 Adaption of the Channel Estimation to Side Channel Attack

Using a channel model for side channel attack means we assume that the hamming weight value is generated from the power trace, via the channel model[2]. Note that in the communication systems, the estimator coefficients at the receiver requires that the transmitter sends a known 'training sequence'. By comparison, for the side channel analysis, the leaked information (hamming weight) is used to instead the 'known sequence', and the "channel" is the medium of everything between the power measurement and the leaked information.

In our multiple input multiple output communication system, the model that used is described by equation (5.11).

$$Y = H\,S + N \quad , \tag{4.1}$$

where S are the training sequences sent by the transmitter and N is additive noise. Again Y is received signal. H is channel estimator matrix that assist data estimation.

For side channel attack, the channel describes the relationship between the power measurement and the hypothetical leaked information. The model we assumed is given below[2]:

$$p_{d,s} = (t_d - \mu_{td}) \cdot h_s + \mu_{p,s} \tag{4.2}$$

where $p_{d,s}$ is the information being leaked by the device. The leaked information is Hamming weight about subkey s related to trace d. $p_{d,s}$ refers to the hamming weight, $p_{d,s} = HW(SubByte(b \oplus k)$, where k is a key value and b is the plaintext byte. The estimation vector for subkey s is h_s and t_d is the vector of power measurements[2].

We assume t_d and p_d to be zero-mean, and then we could remove $\mu_{p,s}\ and\ \mu_{td}$. As a result, the equation can be simplified as:

$$p_{d,s} = t_d \cdot h_s \quad . \tag{4.3}$$

In comparisons of equations (4.1) and (4.3), we can see the MIMO and Side Channel Attack models are similar where both H and h_s are linear estimator coefficients. In MIMO system, the progress of estimating transmitted signals is defined as data estimation while in side channel attack, the progress of estimating leakage information (hamming weight) is defined as hamming weight estimation. The MIMO channel model we use has two transmitters and two receivers and for side channel attack, we divide the real data including power measurement (power traces) and hypothetical leakage information (hamming weight) into two groups as two channels. We only consider the power leakage in this work and the MIMO channel model could be further used to deal with different kinds of physical leakage information of the encrypted device, such as time consumption and electromagnetic radiation, which can be viewed as different channels.

In communication systems, data acquisition must be performed in real-time to be useful, otherwise the data collected will contain error information. Real channels generally change over time, so tracking channels is necessary. These can be seen as several complex factors in communication systems. For side channel analysis, however, the whole computation process

only needs to be completed within reasonable time. Moreover, due to fixed measuring device, the "channel" varies little over time[2].

4.2 Estimator Coefficients

After data are extracted from a device, estimation coefficients can be calculated from the leaked data, which is considered the known value of $p_{d,s}$. In addition, power measurements t_d will be involved. The known values of hamming weights are calculated from the original way of correlation power analysis. Note that training-based estimation is used to deal with hamming weights and power traces, so if we want to get the estimator coefficient in side channel attack we need to collect the known power traces and hamming weights first. Instead of selecting the entire power curve, a certain number of points are chosen to represent the whole power measurements. The error between the 'known' p_d and the estimated $\hat{p}_d$ is calculated as follow:

$$e(d) = p_{d,s} - \hat{p}_{d,s} = p_{d,s} - (\hat{h}_s \cdot t_d) \quad . \tag{4.4}$$

In order to get the best value of the estimator coefficients, least square (LS) methods are used to minimize this error. As mentioned before, the least-squares method finds the optimal parameter values by minimizing the sum, *S*, of squared residuals:

$$S = \sum_{d=0}^{D-1} e^2(d) = \sum_{d=0}^{D-1} (p_{d,s} - \hat{h}_s \cdot t_d)^2 \quad . \tag{4.5}$$

There is a more direct mathematical method to solve the least-square problem, which use the pseudoinverse:

$$\hat{h}_s = t^+ \cdot p_s \quad , \tag{4.6}$$

where t^+ refers to the pseudoinverse of t.

Side channel analysis is considered as a channel model as we assume that a single piece of data (hamming weight) generated the entire power trace. Like MIMO channel estimation, the estimation coefficients $\hat{h}_s$ which are acquired for each subkey s can be used to convert measured power traces into the hamming weight which contains everything related to the leakage information[2].

4.3 Real Data from Attack

Training sequences dealing with estimation coefficient for side channel attack involve measured power consumptions (power traces) and hypothetical leakage information (hamming weight). We need to run a side channel attack to get our real data first and use them as the known training sequences.

4.3.1 Power Traces Curves

The power consumption that we got from the device are shown in Fig 4.1, which can be viewed as the input data. However, we do not use the entire curve but select a certain number of points in the curve to form a sample to do the specific calculation[2].

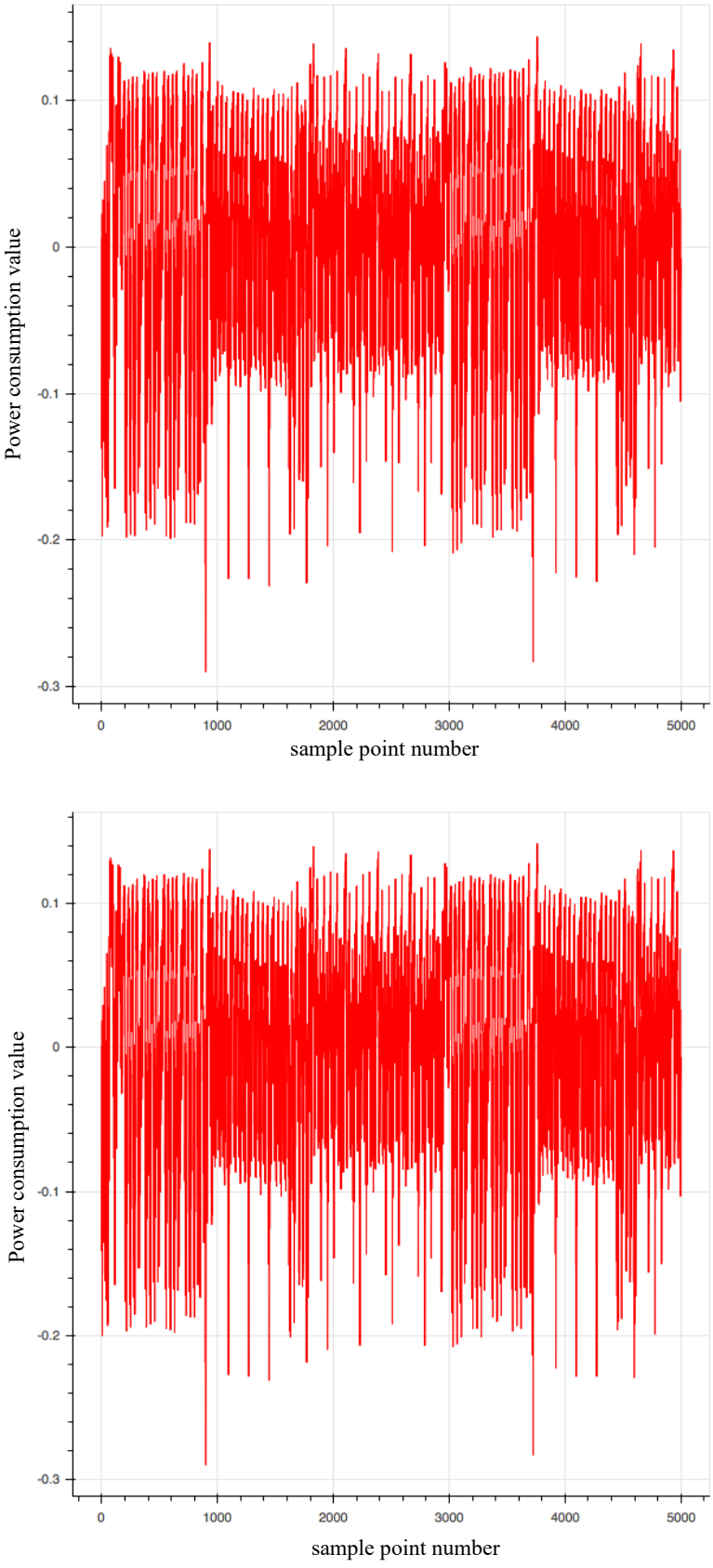


Fig 4. 1 Different power traces

4.3.2 Hamming Weight Results

The hamming weights shown in the following figure are the leakage information needed for the subsequent calculation, which is a value ranging from 0 to 8.

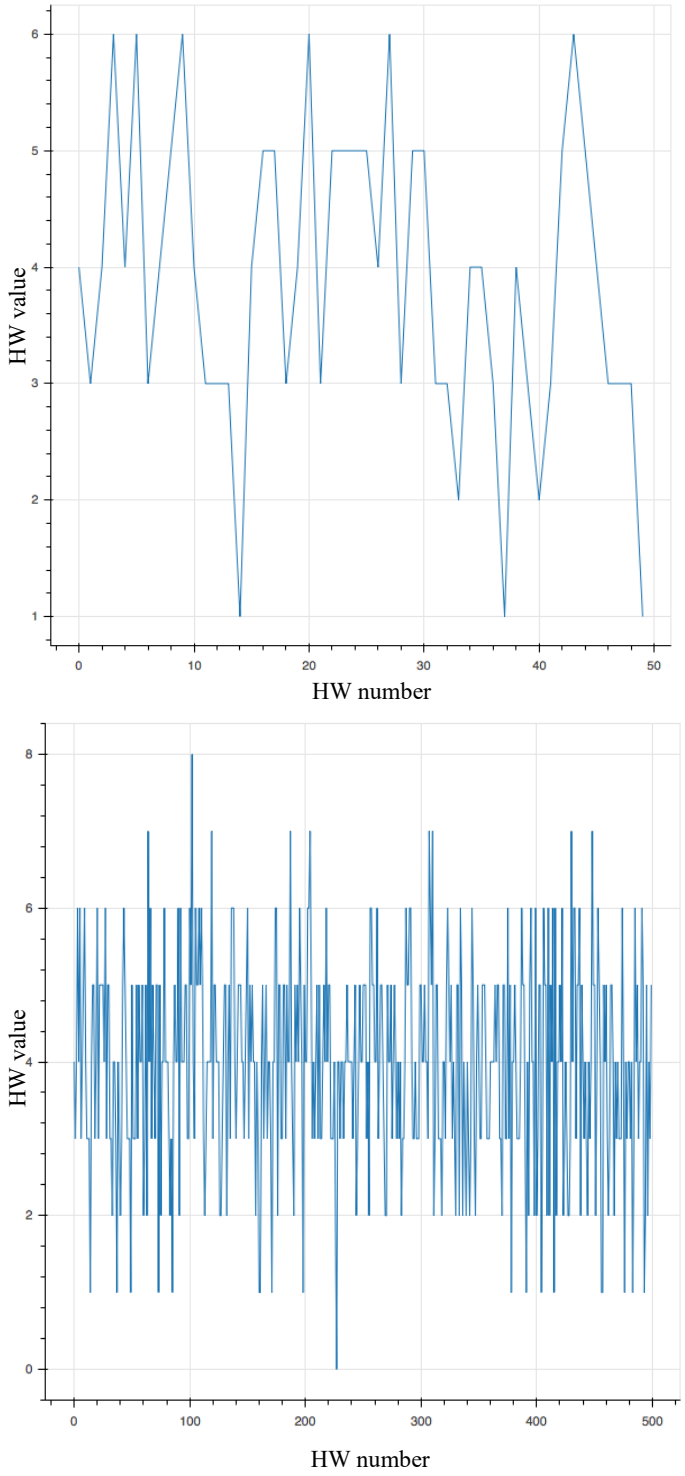


Fig 4. 2 Hamming weight results shown in polyline

To make the results of hamming weights more intuitive, it can also be drawn as dots:

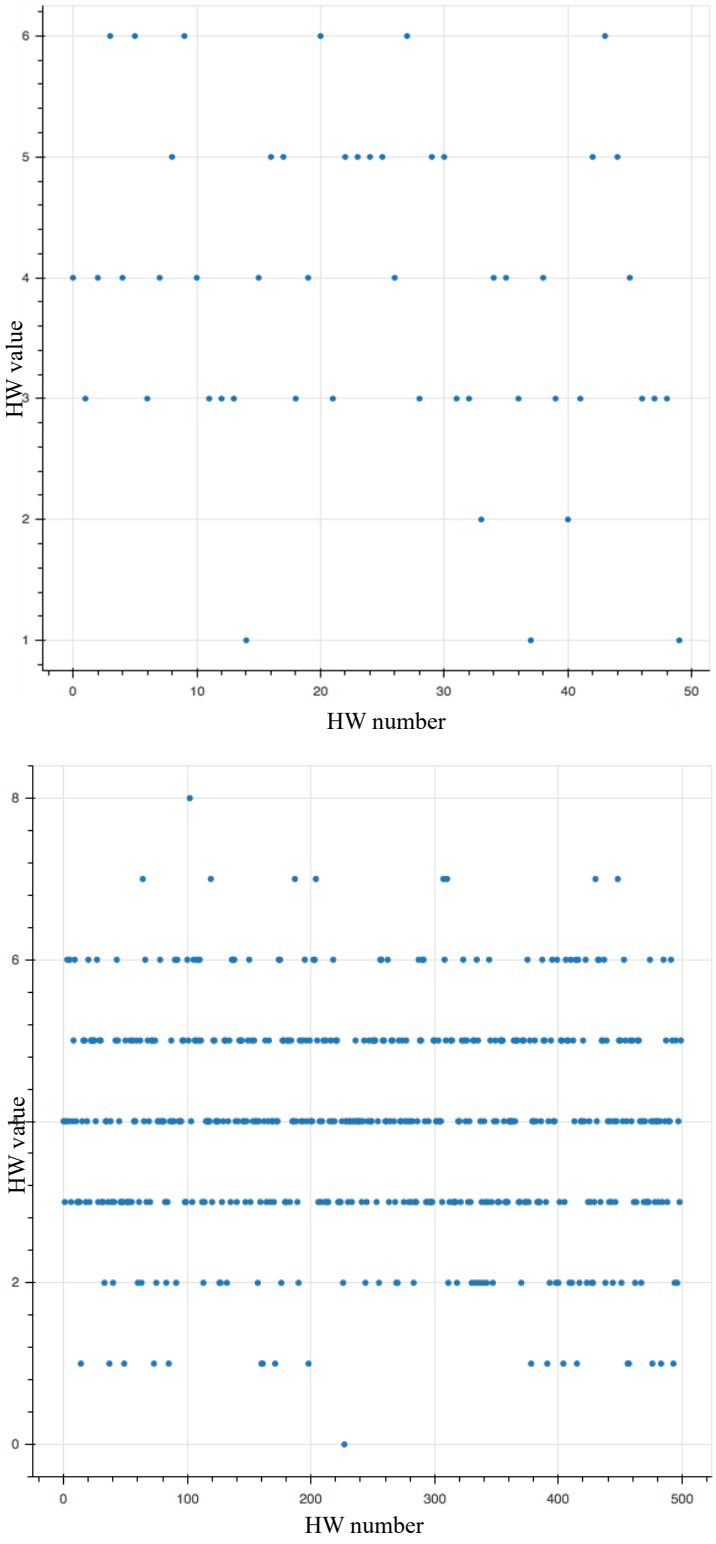


Fig 4. 3 Hamming weight results shown in dot

4.4 Real Attack Results

4.4.1 Hamming Weight Estimation

Our training sequences involving power measurements and calculated hamming weights (HW) are substituted into equation (4.3) and python is used for analysis. The estimated matrix $\hat{h}_s$ will be obtained and then the estimated hamming weight values could be acquired from the estimated matrix. The calculated hamming weights are considered as 'theoretical' values while the estimated hamming weight values are considered as 'Measured' values. Results of 3000 training sets are given as follows:

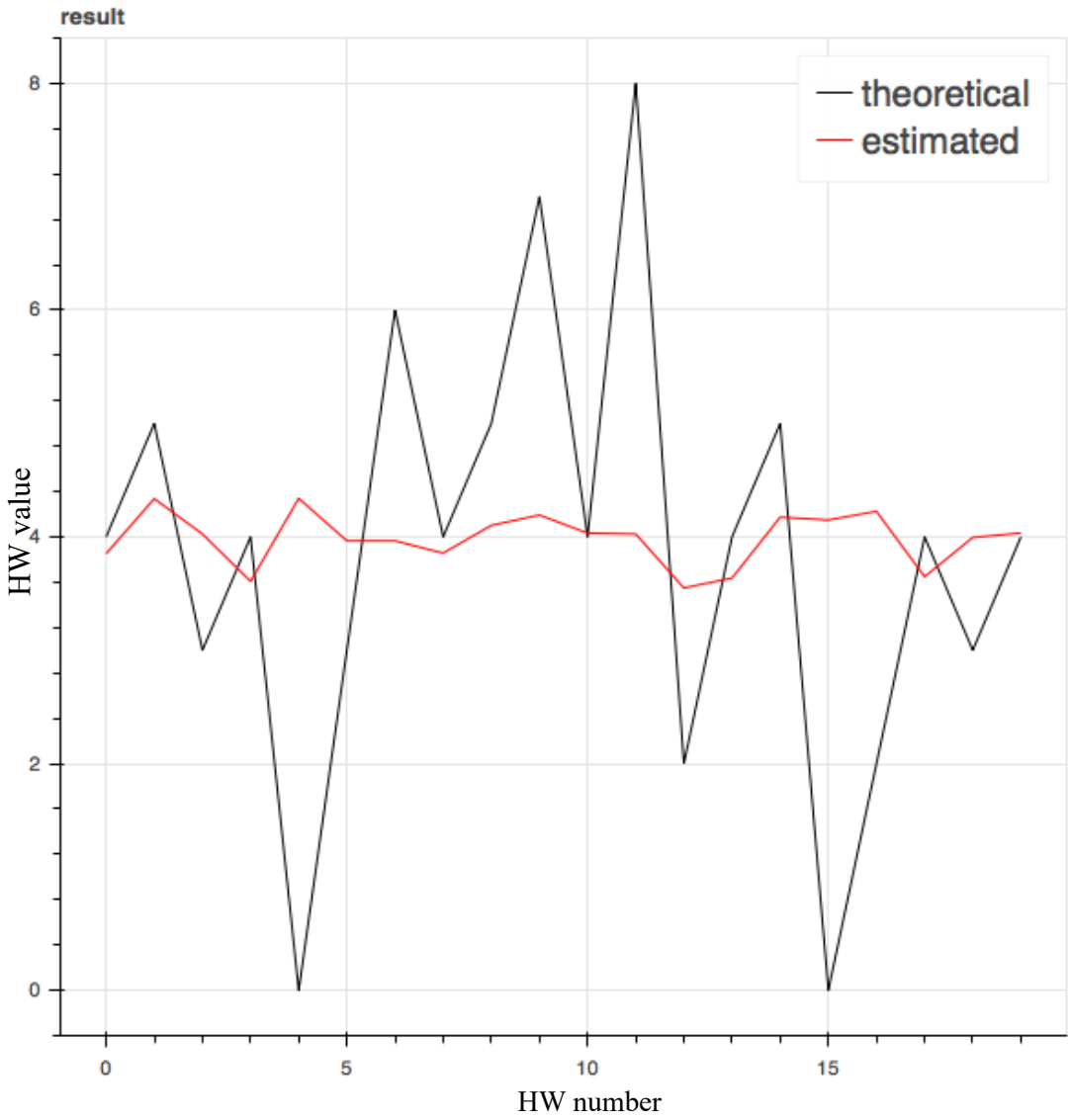


Fig 4. 4 Hamming weight comparison chart (training sets = 300)

To make the results easier to observe, only 0-20 hamming weights were selected for comparison.

It can be seen that there is significant difference between the theoretical values and the measured values, which indicates that the estimation is not good. If the measured value obtained is applied to our correlation power analysis, there will be errors that cannot be ignored, heavily affecting the attack result. In order to make the estimation more accurate, we selected 5000 sets of training data, and the results are as follows:

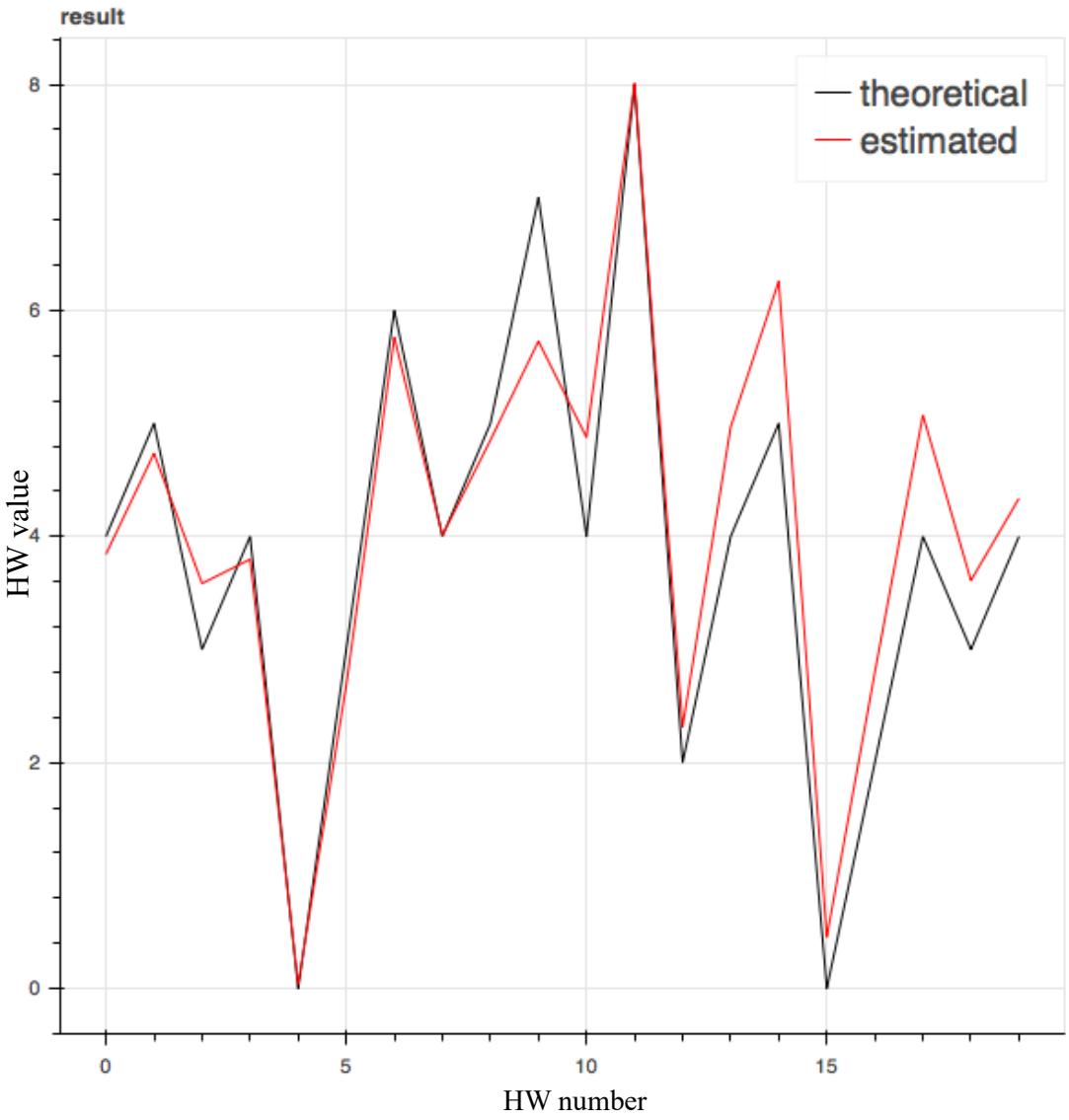


Fig 4. 5 Hamming weight comparison chart (training sets = 5000)

In the above figure, it can be seen that the two curves are almost identical and little differences are shown, indicating that the more training sets are used, the better the fitting results will be. Then the estimated hamming weight values can be applied to our correlation power analysis.

4.4.2 Correlation Power Analysis Results

As shown in equations, correlation coefficient $\rho_{X,Y}$ is widely used to measure the linear dependence between two variables X and Y.

Training sets of 300 are chosen first. The theoretical values refers to the correlation coefficient we calculated with actual measured power and the hamming weights calculated from the algorithm while the 'measured' value refers to the correlation coefficient calculated with actual measured power and estimated hamming weights[13]. Results are shown in Fig. 4.6.

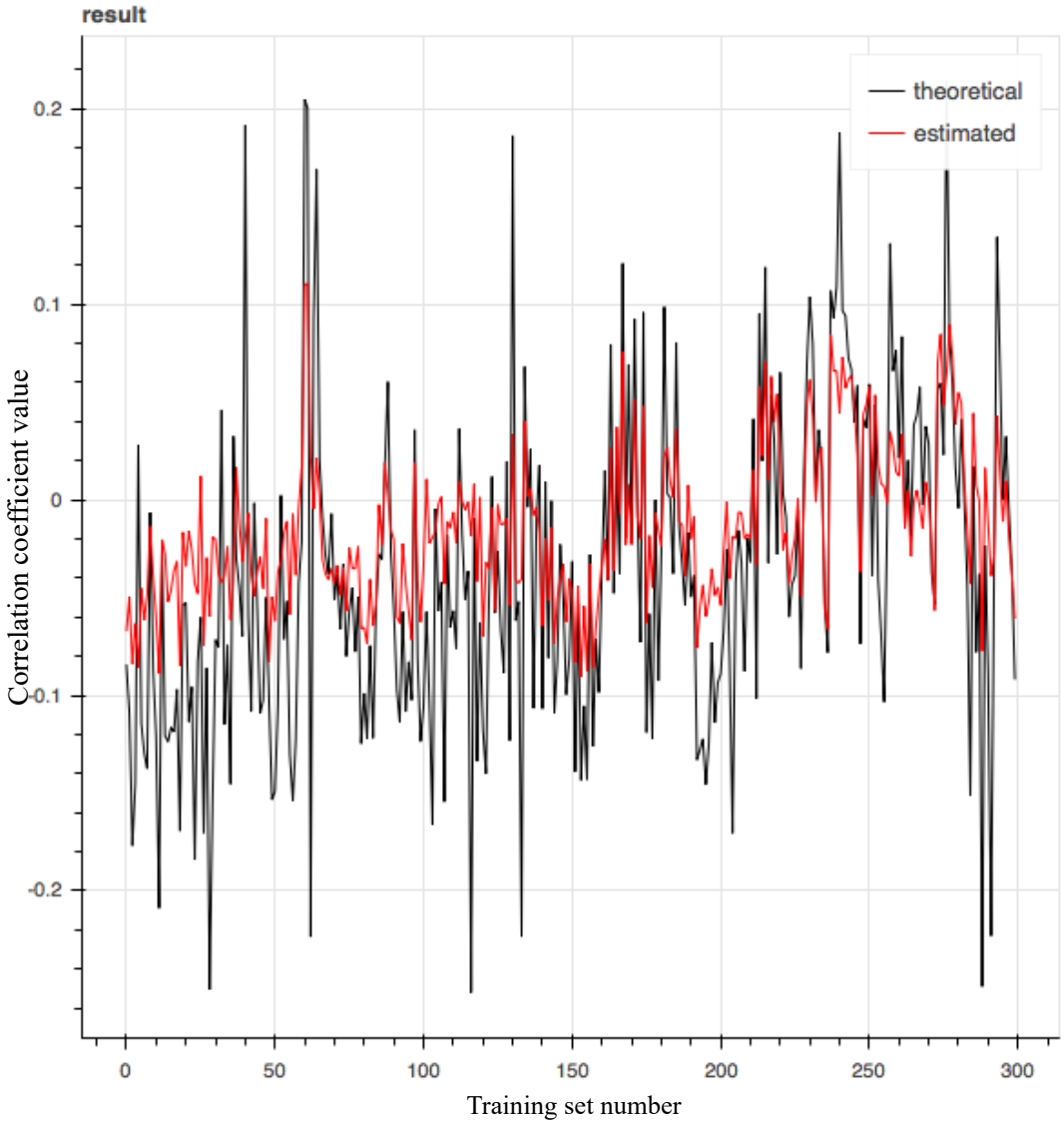


Fig 4. 6 Correlation coefficients corresponding to different hypothesis (Training sets = *300*)

As shown in the figure, the maximum correlation coefficients value does not exceed *0.4*, indicating that the correct secret key does not appear in the figure. This means more training sets need to be involved in order to make our method effective. Next training sets of

3000 are selected, and the results are shown in Fig 4.7.

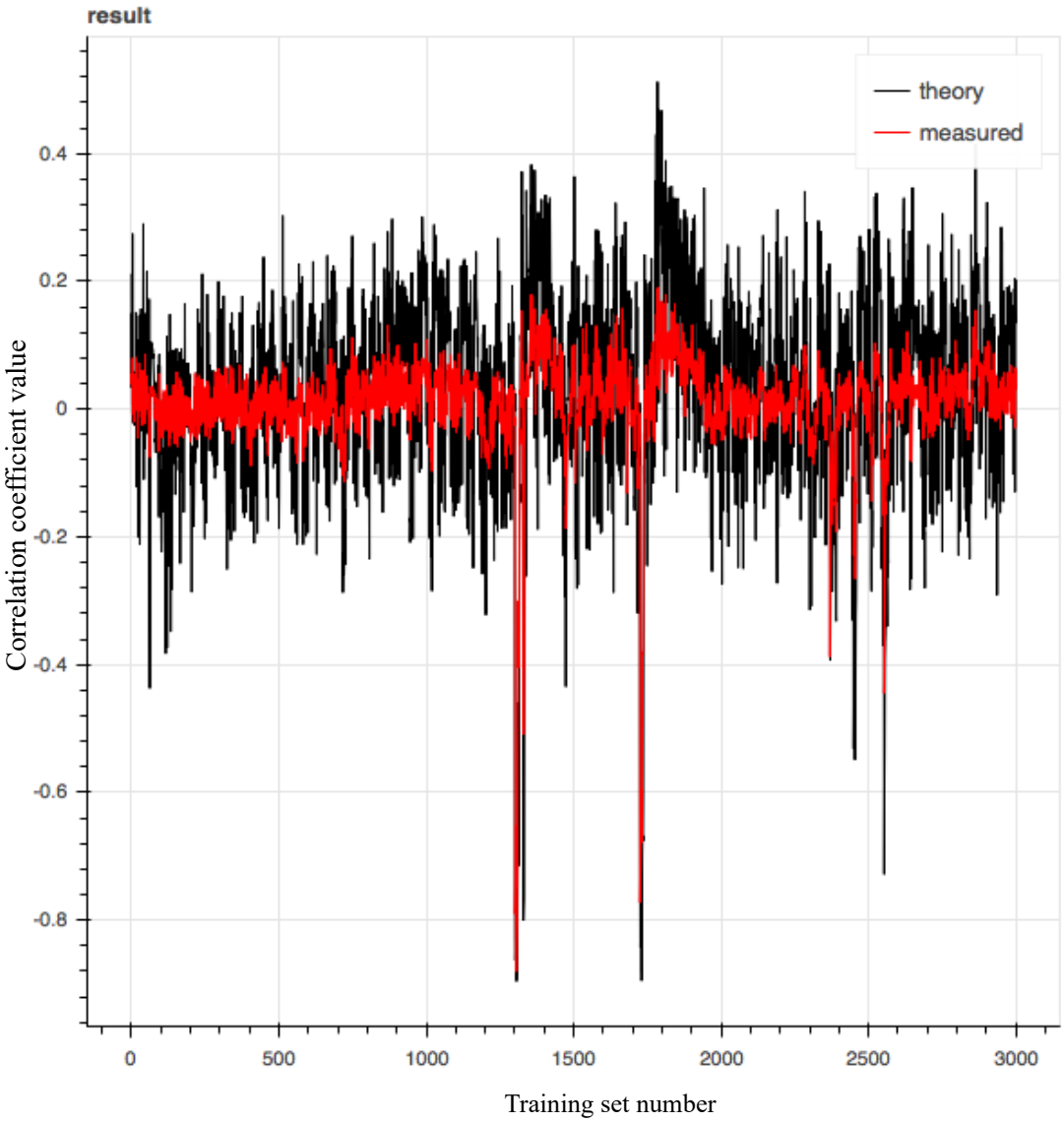


Fig 4. 7 Correlation coefficients corresponding to different hypothesis (Training sets =3000)

From Fig 4.7, it can be seen that when the x-coordinate is around *1305*, the y-coordinate, as well as the correlation coefficient value, is at its peak (the absolute value is above *0.9*). This indicates that the correct key appears in our analysis, and we can tell when the correct key appears from both curves. The fact shows that as long as we select a sufficient number of training sets, our estimated value could be used to find the correct secret key, which greatly simplifies the calculation of hypothetical power leakage (hamming weight).

4.5 Summary

The key difference between side channel attack analysis with and communication system model lies in the specific meanings of the variables. The applicability of channel estimation to side channel attack analysis is discussed in this chapter, including their similarities and differences. The results of hamming weight estimation are shown and the results of correlation power analysis (CPA) are displayed, indicating that the correlation coefficients calculated from the estimated value could effectively help to find the correct secret key.

CHAPTER 5 DISCUSSION

5.1 Conclusion

Side channel attacks have received increasing attention in recent years, and researchers are constantly developing attack methods to increase the accuracy and success of decryption. Correlation power analysis (CPA) attacks own its popularity as it could be applied to various kinds of cryptographic algorithms and could be executed without knowing the details of the attacked implementations.

The main work of this thesis is to apply the MIMO channel estimation to side channel attack analysis effectively in order to simplify the computational complexity of attack analysis; it transfers an entire power measurement trace into the hypothetical leaked information (hamming weight). The side-channel attack system is viewed as a communication system and a linear estimator is used. In communications systems, an estimator is to determine the relationship between the transmit and receive signals, which often ends up to be the impulse response of the entire system; in side-channel attack analysis, an estimator refers to the relationship between the measured power and the leaked information (hamming weight). The estimators are found using the least squares (LS) metrics. In this thesis, Python examples are provided, and the results of hamming weight estimation are represented. The correct secret key could be found from the analysis of correlated coefficients with the estimated hamming weights.

In summary, the work done in this thesis presents an innovative idea for performing a cryptanalytic attack. Specific methods and steps of implementation are given, including comparisons of results. This work considerably reduces attack complexity and is straight-forward to be implemented in real systems.

5.2 Future Work

Side channel attacks involve a wide range of techniques; this thesis only deals with correlation power analysis (CPA) attack. Future work can focus on improving the accuracy and efficiency of the attack analysis based on the work proposed in this thesis.

The leakage information targeted by our side channel attack is the power consumption of a cryptographic device. Other physical measurement indexes of cryptographic devices include electromagnetic radiation and time consumption. They can be modeled and included in further analysis [2].

The Least Square (LS) algorithm is used in our MIMO channel estimation and hamming weight estimation. Other different algorithms such as Maximum Likelihood (ML) algorithm and Least Mean Square (LMS) algorithm can be considered.

More encrypted devices can be attacked to test whether the proposed attack method can be widely applied. Different cryptographic algorithms such as Data Encryption Standard (DES) and Secure Hash Algorithm (SHA) can also be considered.

REFERENCES

[1] S. Mangard, E. Oswald, T. Popp, “Power Analysis Attacks: Revealing the Secrets of Smart Cards”, 2007 Springer Science+Business Media,LLC, pp.119 – 142, 2007.

[2] C. O’Flynn, Z. Chen, “Channel Equalization for Side Channel Attacks”, 10/7 2014. https://eprint.iacr.org/2014/028.pdf

[3] K. Wu, H. Li and F. Yu, “CPA attack for correlation power analysis of synchronous stream cryptography devices”, Institute of Computing Technology, Chinese Academy of Sciences, 08/11 2009.

[4] D. Genkin, L. Pachmanov, I. Pipman and E. Tromer, “Stealing Keys from PCs using a Radio: Cheap Electromagnetic Attacks on Windowed Exponentiation”, 3/3 2015. https://eprint.iacr.org/2015/170.pdf

[5] D. Das, S. Maity, S. B. Nasir, S. Ghosh, A. Raychowdhury and S. Sen, “High Efficiency Power Side-Channel Attack Immunity using Noise Injection in Attenuated Signature Domain”, School of Electrical and Computer Engineering, Purdue University, 2016.

[6] P. Kocher, R. Lee, G. McGraw, “Security as a new dimension in embedded system design”, In DAC 2004: Proceedings of the 41th Design Automation Conference. IEEE Computer Society Press: Los Alamitos, pp.753– 761, 2004.

[7] C. O’Flynn, “A frame work for embedded hardware security analysis”, PhD Thesis, Dalhousie University, June 2017.

[8] E. Oswald, “On side-channel attacks and the application of algorithmic countermeasures”, PhD Thesis, Faculty of Science of the University of Technology Graz (IAIK-TUG), May 2003.

[9] T. Hornby, “Side-Channel Attacks on Everyday Applications: Distinguishing Inputs with FLUSH+RELOAD”, University of Calgary Zcash, 2015.

[10] S. Mangard, E. Oswald, F. Standaert, “A unified framework for the analysis of side-channel key recovery attacks”, In EUROCRYPT 2009: Proceedings of the 28th International Conference on the Theory and Applications of Cryptographic Techniques, vol 5479, pp. 443 – 461, 2009.

[11] M. Agrawal, S. Karmakar, D. Saha,” Scan based side channel attacks on stream ciphers and their counter-measures”, In INDOCRYPT 2008: Proceedings of the 9th International Conference on Cryptology, vol 5365, pp.226 - 238,2008.

[12] S. Burman, D. Mukhopadhyay, K. Veezhmathan, “LFSR based stream ciphers are vulnerable to power attacks”, In INDOCRYPT 2007: Proceedings of the 8th International Conference on Cryptology, vol 4859, pp.379– 392, 2007.

[13] Y. Jia, Y. Hu, F. Wang and H. Wang, “Correlation power analysis of Trivium”, 27/4 2011.https://onlinelibrary-wiley-com.ezproxy.library.dal.ca/doi/full/10.1002/sec.329

[14] Y. Souissi, N. Debande, S. Mekki, S. Guilley, A. Maalaoui, “On the optimality of correlation power attack on embedded cryptographic systems”, WISTP 2012, LNCS 7322, pp.165 – 179, 2012.

[15] M. Renauld, D. Kamel, F. Standaert, D. Flandre, “Information theoretic and security analysis of a 65-nanometer DDSLL AES S-box”, CHES 2011, LNCS 6917, Nara, Japan, pp. 223 – 239, 2011.

[16] A. Elaabid, M. Meynard, O. Guilley, S. Danger, “Combined side-channel attacks”, WISA 2010, LNCS 6513, Jeju Island, Korea, pp. 175-190, 2011.

[17] W. Schindler, K. Lemke, C. Paar,” A Stochastic Model for differential side channel cryptanalysis”, CHES 2005, LNCS 3659, Edinburgh, UK, pp. 30– 46, 2005.

[18] E. Brier, C. Clavier, F. Olivier, “Correlation power analysis with a leakage model. Cryptographic Hardware and Embedded Systems - CHES 2004, pp. 135–152, 2004.

[19] Y. Souissi, N. Debande, S. Mekki, S. Guilley, A. Maalaoui, J. Danger, “On the optimality of correlation power attack on embedded cryptographic systems”, WISTP 2012, LNCS 7322, Egham, UK, pp. 169 – 178, 2012.

[20] H. Zhang, Y. Zhou and D. Feng, “Theoretical and practical aspects of multiple samples correlation power analysis”,27/10 2016. https://onlinelibrary-wiley-com.ezproxy.library.dal.ca/doi/full/10.1002/sec.1686

[21] D. Wang, “Channel Estimation for Wired MIMO Communication Systems”, Multidimensional DSP Project, Spring 2015.

[22] Y. Li, “Simplified channel estimation for OFDM systems with multiple transmit antennas”, IEEE Transactions on Wireless Communications, vol 1, pp.67 - 75, 2002.

[23] M. A. M. MOQBEL, D. Wang, and A. Z. Ali. “MIMO Channel Estimation Using the LS and MMSE Algorithm”, IOSR Journal of Electronics and Communication Engineering (IOSR-JECE), 2015.

[24] T. Yan, S. Yu, P. Su and L. Zhang, “Research on an Iterative Algorithm of LS Channel Estimation in MIMO OFDM Systems”, IEEE Transactions on Broadcasting, vol 51, no. 1, pp. 149-153, March 2005.

[25] C. Murthy, A. Jagannatham, and B. Rao, “Training-Based and Semi-Blind Channel Estimation for MIMO Systems with Maximum Ratio Transmission”, IEEE Transactions on Signal Processing, vol 54, No. 7, 2006.

[26] M. Kanmani, M. Kannan, “Blind channel estimation for Multiple-Input Multiple-Output system using Constant Modulus Algorithm”, Jerusalem College of Engineering, Anna University, 16/4 2015.

[27] Y. Wang, Y. Gou and X. Meng, “Adaptive MIMO Channel Estimation based on Kalman Filtering”, Journal of air force engineering university, China, vol 5, No.6, Dec 2004.

[28] M. Benslama, H. Mokhtari, “Compressed Sensing in Li-Fi and Wi-Fi Networks”, 2017 Elsevier Ltd, pp.232 – 245, 2017.

[29] K. P. Bagadi, S. Das, “MIMO-OFDM Channel Estimation using Pilot Carries”, International Journal of Computer Applications, vol 2, No.3, May 2010.

[30] A. Petropulu, R. Zhang, and R. Lin, “Blind OFDM channel estimation through simple linear pre-coding", IEEE Transactions on Wireless Communications, vol 3, No.2, pp. 647 - 655, March 2004.

[31] S. Schindler and H. Mellein, “Assessing a MIMO Channel”, 2011. https://cdn.rohde-schwarz.com/pws/dl_downloads/dl_application/application_notes/1sp18/1SP18_10e.pdf

[32] R. C. de Lamare and R. Sampaio-Neto, “Detection and Estimation Algorithms in Massive MIMO Systems”, Rodrigo C. de Lamare and Raimundo Sampaio-Neto Centre for Telecommunications Studies (CETUC), 21/8 2014.

[33] S. D. Ma and T. S. Ng, “Time domain signal detection based on second-order statistics for MIMO-OFDM systems,” IEEE Trans.Signal Process. vol 55, No. 3, pp. 1150 – 1158, Mar. 2007.

[34] A. Khlifi and R. Bouallegue, “Performance Analysis of LS and LMMSE Channel Estimation Techniques for LTE Downlink Systems”, International Journal of Wireless & Mobile Networks (IJWMN), vol 3, No. 5, October 2011.

[35] D. Wan, B. Han, J. Zhao, X. Gao, and X. You, “Channel estimation algorithms for broadband MIMO-OFDM sparse channel,” Proc.14th IEEE Int. Symp. on Personal, Indoor and Mobile Radio Communications, pp.1929 – 1933, Beijing, China, Sept. 2003.

[36] Y. Mostof and D. C. Cox, “ICI Mitigation for Pilot-Aided OFDM Mobile Systems”, IEEE Transactions on Wireless Communications, vol 4, NO. 2, March 2005.

[37] H. L. Xiao, Y. S. Ou, Z. P. Nie, “Research status and technical route of MIMO wireless communication system”, University of Electronic Science and Technology of China, 2008.

[38] D. Mavares, P. Rafael, “Space-time code selection for OFDM-MISO system”, ELSEVIER journal on Computer Communications, Vol. 32, Issue 3, pp. 477-481, February 2009.

[39] J.Friedman, “Tempest: A signal problem”, NSA Cryptologic Spectrum, 1972.

[40] J.Matthews, “Side Channel Attacks: Even in a Crypto-Ideal World is Data Truly Secure?”, September 2017.