Yet Another Physical Leakage Assessment with the Wasserstein Distance

Haruka Hirata*, Yusaku Harada*, Yuko Hara†, Kazuo Sakiyama*, Yang Li*
*Department of Informatics, The University of Electro-Communications, Tokyo, Japan
{h.haruka,y.harada,sakiyama,liyang}@uec.ac.jp
†Department of Information and Communications Engineering, Institute of Science Tokyo, Tokyo, Japan
hara@cad.ict.e.titech.ac.jp

***Abstract*—Side-channel attacks pose a significant threat to the security of cryptographic systems by exploiting information leakage such as power consumption. Evaluating the difficulty of such attacks is crucial for developing robust countermeasures. In this study, we propose an analytical methodology using the Wasserstein distance as a metric to assess the vulnerability of cryptographic implementations for power side-channel attacks. We conduct both simulated and FPGA-based experiments to validate the effectiveness of using the Wasserstein distance. Our results demonstrate a clear correlation between the signal-to-noise ratio and the Wasserstein distance, with higher noise leading to smaller distances, thus indicating increased attack difficulty. Notably, the metric stabilizes with a relatively small number of traces, suggesting practical applicability. These findings suggest that the Wasserstein distance can serve as a reliable indicator of attack difficulty, providing a quantifiable measure to aid in the development and evaluation of secure cryptographic systems.**

***Index Terms*—Power analysis, SCA evaluation, Wasserstein distance, Statistical moments**

I. Introduction

In the field of modern cryptography, securing sensitive information against various attacks is a crucial task. Among these threats, side-channel attacks (SCAs) have emerged as particularly insidious, exploiting physical leakages from cryptographic devices to infer secret data. Timing attacks [1], which measure the execution time for encryption or decryption, were the first proposed SCAs. Subsequently, power analysis [2], [3], which measures power consumption during operation, and electromagnetic (EM) analysis [4], which measures EM emanations, have also been proposed. To counteract these attacks, masking [5] has been well-studied as a countermeasure.

Traditionally, Test Vector Leakage Assessment (TVLA) [6] has been used as an evaluation tool for implemented countermeasures. This involves measuring fixed vs. random plaintexts and determining whether the means of these distributions are significantly different using Welch's t-test. Following TVLA, the χ^2-test [7] and Kolmogorov-Smirnov Test [8] were proposed as additional leakage detection tools. However, these hypothesis tests often focus on the mean of distributions and the dependence of probability distributions. Additionally, in higher-order implementations, Welch's t-test may not reveal differences in the first or second moments.

In 2017, a new evaluation metric, the Wasserstein distance, was proposed for generative models in the field of machine learning. The introduction of the Wasserstein metric demonstrated that the training of generative adversarial networks could be stabilized, thereby improving the quality of the generated samples [9]. In their study, the performance of the model is assessed by measuring the distance between the distributions of the real data and the generated data using the Wasserstein distance.

Our motivation, therefore, is to devise a new analytical method that directly examines the distributions themselves utilizing the Wasserstein distance for the field of side-channel analysis. We aim to evaluate the vulnerability of cryptographic implementations to side-channel attacks more effectively, moving beyond traditional hypothesis tests.

Our contributions of this paper are as follows.

- Propose an analytical methodology using the Wasserstein distance.
- Confirm the effectiveness of the method through simulation and practical experiments.

The rest of this paper is organized as follows. In Section II, we provide the background information relevant to our insights. We describe the analytical methodology using the Wasserstein distance in Section III. We perform simulated experiments and practical experiments with an FPGA board in Sections IV and V, respectively. Finally, Section VI concludes the paper. [1]

II. Background

In this section, we provide the background information.

A. Power consumption due to CMOS transitions

In this paper, we focus on an FPGA board consisting of look-up tables and registers. Thus the power consumption of registers accounts for the total power consumption and it can be modeled as Hamming distance leakage.

The power consumption of a register (D Flip-Flop) can be divided into two main components: dynamic power consumption ($P_{dynamic}$) and static power consumption (P_{static}). Dynamic power consumption occurs due to the charging and discharging of CMOS transistors when the register switches

[1]This paper is an extended version of our paper published in the IEEE International Symposium on Asian Hardware Oriented Security and Trust (AsianHOST) in Kobe, Japan, Dec. 2024.

The final published version is available at IEEE Xplore: https://doi.org/10.1109/AsianHOST63913.2024.10838480

TABLE I: Power consumption of the register.

Transition	$0 \to 0,\ 1 \to 1$	$0 \to 1,\ 1 \to 0$
Power consumption	P_{static}	$P_{static} + P_{dynamic}$

states, i.e., $0 \to 1$ and $1 \to 0$, as shown in Table I. Furthermore, the total power consumption is significantly influenced by the dynamic power consumption. Thus, when data stored in the k-bit register transitions from x^t to x^{t+1}, the power consumption leakage $\mathcal{L}$ is formalized as:

$$\mathcal{L} = HD(x^t, x^{t+1}) + Noise \sim N(0, \sigma^2), \tag{1}$$

where HD is the Hamming distance, and the noise follows a Gaussian distribution with variance σ^2.

B. Boolean masking

Masking [5] is a well-studied and widely used to protect cryptographic hardware against side-channel attacks, especially probing attacks. Masking mitigates this threat by obscuring the actual secret values during computation. A secret value is encoded into multiple values called "shares" with a random number.

$$\mathbf{x} \mapsto (x_0, x_1, ..., x_{d-1}, x_d).$$

Thus the original secret value x can be reconstructed only if all the shared values are collected. For Boolean masking, the secret x is divided into $d+1$ shares $x_0, x_1, ..., x_d$ such that:

$$x = x_0 \oplus x_1 \oplus \cdots \oplus x_{d-1} \oplus x_d, \tag{2}$$

where $x_i \in GF(2^k)$, $i > 0$ are random and $x_0 = x \oplus \sum x_i$. Here, $\oplus$ denotes the bit-wise XOR operation

In the masked circuit, cryptographic computations are performed using the shares instead of processing the original secret directory. This ensures that the intermediate values during computation do not reveal the secret.

We consistently indicate masked values in bold in this paper.

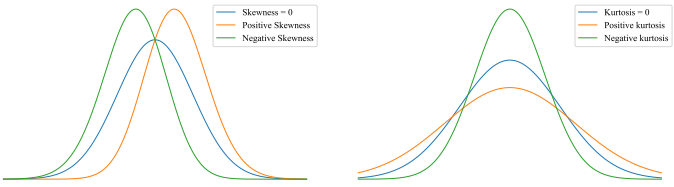
C. Statistical Moments

The 3rd-order moment, skewness, is a statistical measure that describes the asymmetry of the distribution of values in a dataset, while kurtosis (the 4th-order moment) describes the shape of a distribution's tails in relation to its overall shape as shown in Figure 1.

Let X be a random variable and $E(\cdot)$ denotes the expectation operator. Skewness and kurtosis are 3rd- and 4th-order standardized moments, respectively, and can be calculated as follows:

$$SM_3 = E((\frac{X-\mu}{\sigma})^3), \qquad SM_4 = E((\frac{X-\mu}{\sigma})^4), \tag{3}$$

where μ is the mean and σ is the standard deviation. In general, the d-th order standardized moment SM_d is given by $E((\frac{X-\mu}{\sigma})^d)$. These moments are calculated for the higher-order t-test in TVLA [6], [10] and moment correlated DPA attacks [11], [12].



(a) Skewnenss (b) Kurtosis

Fig. 1: Probability density function with different skewness and kurtosis.

D. The Wasserstein distance

The Wasserstein metric, also known as the Earth Mover's Distance [13], [14], is a distance function that measures the distance between two probability distributions. It is often used to quantify the difference between two distributions by calculating the minimum cost needed to transform one distribution into another. Thus the Wasserstein distance can also be obtained by solving the optimal transport problem.

The p-th ($p \geq 1$) Wasserstein distance between probability distributions $\mathcal{A}$ and $\mathcal{B}$ is given by:

$$W_p(\mathcal{A}, \mathcal{B}) = [\inf\ E(d(a,b))^p]^{\frac{1}{p}}, \tag{4}$$

where $E(\cdot)$ denotes the expectation operator, d is the Euclidean distance between two points, and a and b are random variables of $\mathcal{A}$ and $\mathcal{B}$, respectively. Especially, in the case of $p = 1$, the distance can be obtained by:

$$W_1(\mathcal{A}, \mathcal{B}) = \int_{-\infty}^{\infty} |F_{\mathcal{A}}(x) - F_{\mathcal{B}}(x)|\, dx, \tag{5}$$

where $F_{\mathcal{A}}$ and $F_{\mathcal{B}}$ are the cumulative distribution functions.

III. Methodology using Wasserstein distance

In this section, we define a power leakage model and explore an analytical method using the Wasserstein distance as a metric.

A. Power leakage model

Definition 1. *Total power consumption model.*
An attacker obtains the sum of a leakage of all $d+1$ share:

$$\mathcal{L}_{sum} = \mathcal{L}_0 + \mathcal{L}_1 + \cdots + \mathcal{L}_{d-1} + \mathcal{L}_d. \tag{6}$$

The leakage of individual shares cannot be obtained independently.

In the probing model, an attacker is allowed probing up to d out of $d+1$ shares. Despite that, in our model, the attacker who measures the power consumption trace with a probe and an oscilloscope obtains all of the shares as shown in Figure 2.

B. Leakage evaluation with the Wasserstein distance

As described in Section II-D, the metric can be used to evaluate the difference between two probability distributions as a distance. Thus this metric is finite for any given distribution,

$$\infty > W_p(\mathcal{A}, \mathcal{B}) \geq 0. \tag{7}$$

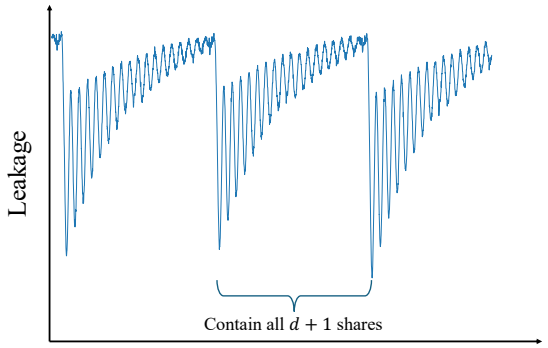


Fig. 2: A power trace of a parallel masked implementation.

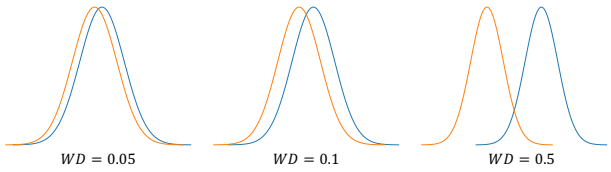


Fig. 3: Comparison of three different Wasserstein distances between two distributions. A smaller Wasserstein distance indicates greater similarity between the two distributions.

The distance is equal to zero if and only if two distributions are identical, i.e., $\mathcal{A} = \mathcal{B}$. As we can see in Figure 3, the smaller the Wasserstein distance, the less difference there is between the two distributions.

Moreover, the distance is determined by the given distribution and is a constant value, i.e., $W_1(\mathcal{A}, \mathcal{B}) = c$. Distributions sampled from $\mathcal{A}$ and $\mathcal{B}$ converge to themselves, following Glivenko–Cantelli theorem (Theorem 1). Therefore, it is reasonable to use the Wasserstein distance as a metric, and this can directly assess the difficulty of side-channel attacks such as DPA.

Theorem 1. *Glivenko–Cantelli theorem.*
Let F_n and F be an empirical distribution and the cumulative distribution, respectively.

$$\|F_n - F\|_\infty = \sup_{x \in \mathbb{R}} \left|F_n(x) - F(x)\right| \longrightarrow 0 \tag{8}$$

This theorem states that the empirical distribution F_n converges to F.

The Wasserstein distance is often used in the field of machine learning. For example, in natural language processing [15], [16] it is employed as the Word Mover's Distance, which measures the similarity between documents by considering them as distributions composed of sets of words. In the area of hardware security, the metric can be used for a system-level tampering detection scheme as a previous work [17]. They detected fluctuations of an impedance characterization caused by tamper events such as probings and electromagnetic radiation to the circuit. Additionally, the metric is used to evaluate the similarity of the feature space of multiple side-channel data [18].

C. Comparison with Kolmogorov-Smirnov, Welch's t- and χ^2-test

The Kolmogorov-Smirnov (KS) test [8] is a non-parametric method used to determine if two distributions differ significantly. It evaluates the maximum difference between the cumulative distribution functions of the two distributions, whereas the Wasserstein distance calculates the integral. The KS test can effectively reject the null hypothesis when there is a significant difference between the distributions, indicating that they do not match. However, one major limitation of the KS test is its inability to confirm that two distributions are identical; it can only fail to reject the null hypothesis, which does not necessarily mean that the distributions are the same. This limitation makes it less conclusive in scenarios where confirming the similarity of distributions is crucial.

For Welch's t-test and the χ^2-test, according to Moradi et al.'s simulated experiments [7], the required plaintexts to detect the leakage considerably increase as the noise and the security order increase. Furthermore, even if leakage can be detected for a certain number of plaintexts, it does not necessarily mean that an attack will be successful with those plaintexts. In other words, these tests cannot directly indicate the difficulty of an attack.

However, because the Wasserstein distance concentrates on the entire distribution, it serves as a more general metric compared to Welch's t-test. We note that these tests provide binary answers, either a leakage found or not found, rather than quantitative evaluations. In contrast, the Wasserstein distance offers a quantifiable measure of the difference between two distributions. It calculates the minimal cost of transforming one distribution into another, capturing subtle differences that the KS test might not detect.

IV. Simulated Experiments

In this section, we perform a software simulation for an 8-bit three-share implementation as a case study. First, we confirm that the Wasserstein distance captures the difference between distributions. We then compare the Wasserstein metric with the t-test and χ^2-test to claim that our evaluation can be easily conducted with a small number of traces.

A. Setup

In the simulation, the signal-to-noise ratio (SNR) is defined as $SNR := k(2\sigma)^{-2}$, where k is bit size and σ is a standard deviation of the Gaussian noise [19], [20]. The simulation settings are the same as [7], namely, we consider three SNRs as follows:

- $SNR = 12.5$, low noise with $\sigma = 0.4$,
- $SNR = 1.0$, middle noise with $\sigma = 1.4$,
- $SNR = 0.1$, high noise with $\sigma = 4.4$.

The power consumption leakage follows Equation 1 and we rely on a Hamming distance leakage model as described in Section II-A, therefore the leakage $\mathcal{L}$ is obtained by

$$\mathcal{L} = \sum_{0}^{2} HD(x_i, y_i) + Noise \sim N(0, \sigma^2). \tag{9}$$

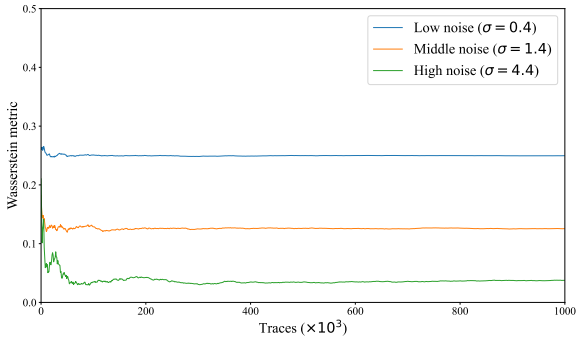


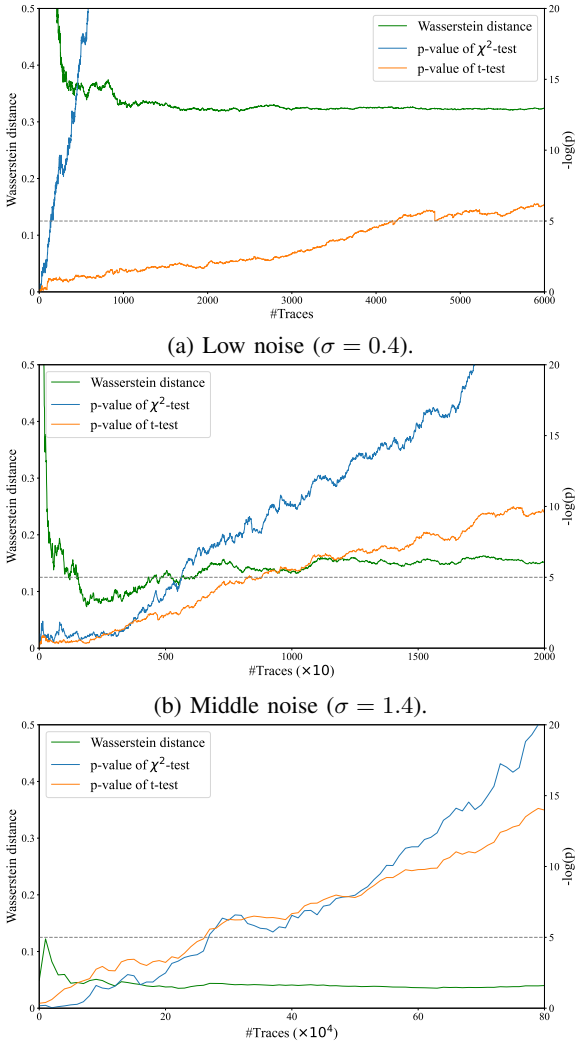
Fig. 4: The Wasserstein distances for three SNRs.

The data are generated for the fixed vs. random plaintexts to make distributions $\mathcal{D}_{fixed}$ and $\mathcal{D}_{random}$. Then we calculate the 1st Wasserstein distance between the obtained distributions $W_1(\mathcal{D}_{fixed}, \mathcal{D}_{random})$ by using a scipy Python library.

B. Results

Figure 4 shows the Wasserstein distance between the distributions for fixed and random data $W_1(\mathcal{D}_{fixed}, \mathcal{D}_{random})$ for three SNR patterns. As shown in the figure, the distance depends on the noise and this indicates that when the noise is high, the difference between the distributions becomes smaller, increasing the difficulty of attacks. Additionally, it can be seen that the Wasserstein distance stabilizes at approximately 400,000 data points for all noise levels.

A Comparison of the performance of the tests and the Wasserstein distances is depicted in Figure 5 for three noise levels. For this evaluation, we round the leakage (after the addition of the noise) in Equation 9 because the χ^2-test uses histograms. As a metric for the t-test and χ^2-test, we compute p-values and the threshold is $p = 10^{-5}$, which corresponds to $t = 4.5$, often determined in the t-test. For low and middle noise cases as shown in Figures 5(a) and 5(b), the χ^2-test outperforms the t-test, but the differences in their performance are negligible. On the other hand, the Wasserstein distance is higher than zero (i.e., $W_1(\mathcal{D}_{fixed}, \mathcal{D}_{random}) \gg 0$) for all noise level. Moreover, the difference in the distributions is observed before the p-values exceed the threshold $p > 10^{-5}$. Although a definitive threshold has not been established so far, the metric is superior to these hypothesis tests because the security evaluation can be conducted with fewer traces.



(a) Low noise ($\sigma = 0.4$).

(b) Middle noise ($\sigma = 1.4$).

(c) High noise ($\sigma = 4.4$).

Fig. 5: Performance of the t-test (orange), χ^2-test (blue) and the Wasserstein distance (green) for simulated 3rd-order leakage with $\sigma = 0.4$, $\sigma = 1.4$ and $\sigma = 4.4$.

TABLE II: Equipment used in experiments

Equipment	Product name and model number
Oscilloscope	Tektronix MSO64
Waveform generator	Keysight 33600A
DC power supply	Tektronix KEITHLEY 2260B
FPGA	SAKURA-G (Spartan-6)

V. Practical Experiments with FPGA

A. Assessment with our lab's setup

In the simulation, we confirmed that the Wasserstein distance captures the difference between distributions. Before comparing it with the t-test and χ^2-test, we verify whether the same results (i.e., the distance changes according to noise levels) can be obtained in an actual experimental environment. In this section, we use a SAKURA-G board, which is designed for a side-channel analysis evaluation [21].

1) Setup: We implemented a simple circuit consisting of an 8-bit three shares register with a keep_hierarchy option on the Xilinx ISE development tool. Table II outlines the equipment used in the experiments, and Figure 6 shows our lab setup. We measure 800,000 traces (400,000 fixed data and 400,000 random data) of the power consumption at 6.25 GS/s sampling rate using an oscilloscope with a 12 bit analog-to-digital converter (ADC). Moreover, we supplied a slow 3 MHz clock signal to obtain clear traces.

For data communication, we used MATLAB to send six

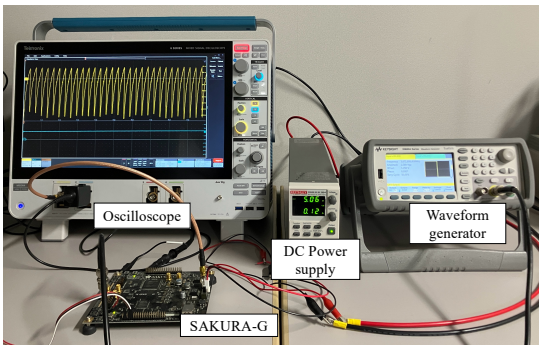
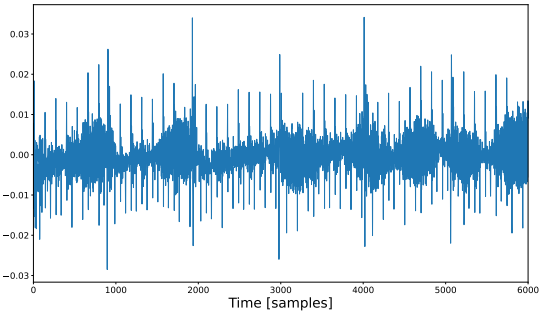
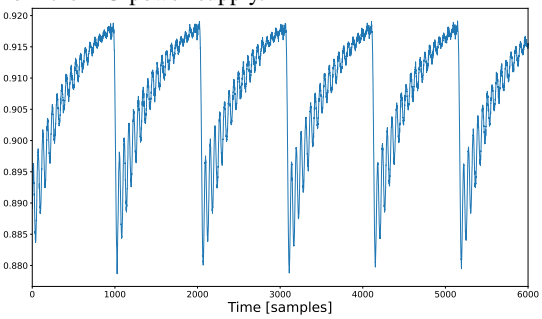


Fig. 6: Experimental setup of our lab.



(a) Trace of the registers-only circuit. The signal consists of a lot of noise from the DC power supply.



(b) Trace of the circuit with Keccak-f800.

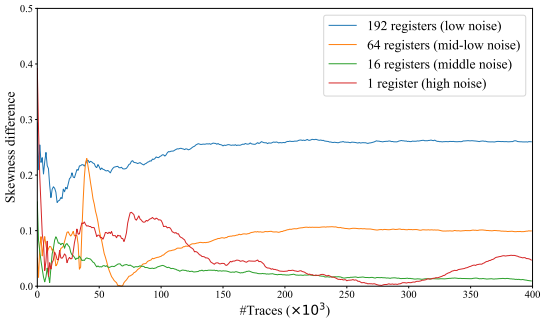
Fig. 7: Power consumption traces for the circuit with and without Keccak-f800 circuit.

bytes of data, i.e., $(x_0, x_1, x_2, y_0, y_1, y_2)$, to SAKURA-G, with x_0, x_1, y_0, and y_1 being randomly generated by MATLAB's built-in *randi* function.

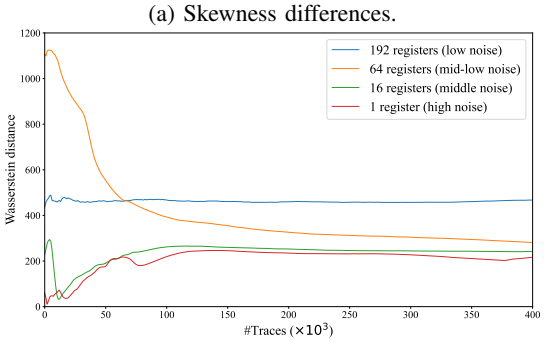
Figure 7 shows the power consumption traces of the implemented circuits with registers both with and without a pseudo-random number generator (PRNG). As shown in Figure 7(a), the trace of the register-only circuit consists of noise from the power supply, making it difficult to screen the measured traces. Therefore, we added a PRNG (Keccak-f800 [22]) to the circuit to amplify the power consumption, resulting in clearer power traces as shown in Figure 7(b).

Because of the difficulty of controlling the SNR of the practical experiment environment, we duplicated the registers as follows:

- 192 registers for low noise.



(a) Skewness differences.



(b) Wasserstein distances.

Fig. 8: Skewness differences the Wasserstein distances for 192 regs., 64 regs., 16 regs. and 1 register circuit.

- 64 registers for mid-low noise.
- 16 registers for middle noise.
- 1 register (default) for high noise.

In other words, we enhanced a numerator of the SNR, i.e., the total power consumption, instead of reducing or increasing the noise (a denominator of the SNR).

Before computing the Wasserstein distance, we conducted a correlation analysis with known plaintexts. The correlation coefficients for 192 registers, 64 registers, 16 registers, and 1 register were 0.91, 0.51, 0.11, and 0.04, respectively.

2) Explanation of the process at each clock cycle: The trace shown in Figure 7(b) contains six clock cycles, and the processing at each cycle is described below:

First clock: Do nothing.
Second clock: Reset registers to zero, i.e., $\mathbf{0} = (0, 0, 0)$.
Third clock. Do nothing.
Fourth clock: Transitioning from $\mathbf{0} = (0, 0, 0)$ to $\mathbf{x} = \mathbf{0}$.
Fifth clock: Transitioning from $\mathbf{x}$ to $\mathbf{y}$.
Sixth clock: Transitioning from $\mathbf{y}$ to $\mathbf{0} = (0, 0, 0)$.

Therefore, we focus on the fifth clock cycle (at around 4,100 sample points), where the transition from data $\mathbf{x}$ to $\mathbf{y}$ occurs. Following a fast computation methodology by making a histogram [23], we used raw values obtained from the oscilloscope's ADC. We note again that these data $\mathbf{x}$ and $\mathbf{y}$ are shared-form values.

3) Results: To confirm that the Wasserstein distance indicates the difficulty of attacks using a higher-order statistical moment, we calculate the difference in skewness instead of the t-statistics used in Welch's t-test.

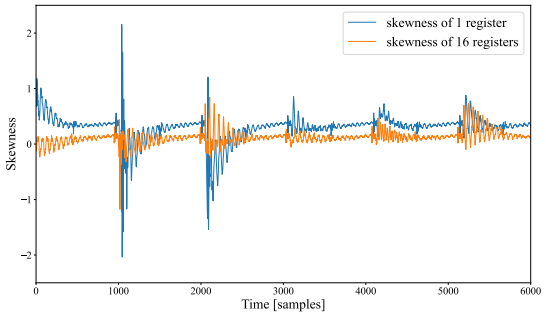


Fig. 9: Skewness differences of 1 and 16 registers.

As shown in Figure 8, there is a correspondence between the number of registers (i.e., SNR) and the difference in skewness. Additionally, it can be seen that the Wasserstein distance fluctuates with the skewness difference. Notably, the skewness difference for 1 register is greater than for 16 registers in the latter part of the graph. This is because the power consumption signal of the register is negligible compared to the noise, so that the skewness values for 1 register are higher than those for 16 registers as shown in Figure 9. Nonetheless, as shown in Figure 8(b), the Wasserstein distance decreases as noise increases, indicating that it effectively reflects the difficulty of the attack, i.e., the similarity of the given distributions. Furthermore, the distances stabilize with 400,000 traces, which is consistent with the simulation results.

These results suggest that using the Wasserstein distance for evaluation applies to actual cryptographic circuits and can be conducted with fewer traces compared to traditional hypothesis testing. This means the reduction in the number of traces required to demonstrate security evaluation leads to a decrease in experimental time and cost.

B. Assessment with open dateset

AAA

VI. Conclusion and Future Works

This paper has presented an analytical methodology for leakage evaluation using the Wasserstein distance. In the simulated experiments, we confirmed that the Wasserstein distance depends on the noise level and decreases as the difference between distributions becomes smaller. Additionally, we verified that the Wasserstein distance can detect differences between distributions with a smaller number of traces. In the case of high noise, the distribution difference can be observed with less than half of the number of traces required by the t-test or χ^2-test to detect leakage. We then conducted practical experiments using an FPGA. Similar to the simulation results, the Wasserstein distance in the practical experiments also varied with the noise levels. Furthermore, we observed that the Wasserstein distance decreases as the skewness difference (i.e., the difficulty of the attack) decreases. From these results, we believe that the Wasserstein distance can indicate the difficulty of attacks. Furthermore, since the metric stabilizes with a small number of traces, the evaluation is feasible in a shorter time compared to hypothesis tests.

In future work, we will investigate the relationship between the Wasserstein distance and the difficulty of attacks, such as the number of traces required for key recovery. We will also experimentally confirm the metric for higher-order shared implementations. Furthermore, we will evaluate the effectiveness of the Wasserstein distance on masked cryptographic circuits such as the AES cipher.

Acknowledgment

This work was supported by JST CREST (grant number JPMJCR23M2) and JSPS Bilateral Joint Research Projects (grant number JPJSBP120242301).

References

[1] P. C. Kocher, "Timing attacks on implementations of Diffie-Hellman, RSA, DSS, and other systems," in *Advances in Cryptology — CRYPTO '96*. Springer, 1996, pp. 104–113.

[2] P. Kocher, J. Jaffe, and B. Jun, "Differential power analysis," in *Advances in Cryptology — CRYPTO' 99*, ser. LNCS, vol. 1666. Springer, 1999, pp. 388–397.

[3] E. Brier, C. Clavier, and F. Olivier, "Correlation power analysis with a leakage model," in *Cryptographic Hardware and Embedded Systems - CHES 2004: 6th International Workshop Cambridge, MA, USA, August 11-13, 2004. Proceedings*, ser. Lecture Notes in Computer Science, M. Joye and J. Quisquater, Eds., vol. 3156. Springer, 2004, pp. 16–29. [Online]. Available: https://doi.org/10.1007/978-3-540-28632-5_2

[4] J.-J. Quisquater and D. Samyde, "Electromagnetic analysis (EMA): Measures and counter-measures for smart cards," in *Smart Card Programming and Security*. Springer, 2001, pp. 200–210.

[5] Y. Ishai, A. Sahai, and D. Wagner, "Private circuits: Securing hardware against probing attacks," in *Advances in Cryptology — CRYPTO 2003*. Springer, 2003, pp. 463–481.

[6] G. T. Becker, J. Cooper, E. K. DeMulder, G. Goodwill, J. Jaffe, G. Kenworthy, T. Kouzminov, A. J. Leiserson, M. E. Marson, P. Rohatgi, and S. Saab, "Test vector leakage assessment (TVLA) methodology in practice," in *International Cryptographic Module Conference*, 2013.

[7] A. Moradi, B. Richter, T. Schneider, and F.-X. Standaert, "Leakage detection with the x2-test," *IACR Transactions on Cryptographic Hardware and Embedded Systems*, vol. 2018, no. 1, p. 209–237, Feb. 2018.

[8] X. Zhou, K. Qiao, and C. Ou, "Leakage detection with kolmogorov-smirnov test," Cryptology ePrint Archive, Paper 2019/1478, 2019.

[9] M. Arjovsky, S. Chintala, and L. Bottou, "Wasserstein generative adversarial networks," in *Proceedings of the 34th International Conference on Machine Learning*, ser. Proceedings of Machine Learning Research, vol. 70. PMLR, 06–11 Aug 2017, pp. 214–223.

[10] T. Schneider and A. Moradi, "Leakage assessment methodology," in *Cryptographic Hardware and Embedded Systems – CHES 2015*. Berlin, Heidelberg: Springer-Verlag, 2022, p. 495–513.

[11] A. Moradi and F.-X. Standaert, "Moments-correlating DPA," in *Proceedings of the 2016 ACM Workshop on Theory of Implementation Security*, ser. TIS '16. Association for Computing Machinery, 2016, pp. 5–15.

[12] C. Ou, Z. Wang, D. Sun, and X. Zhou, "Profiling good leakage models for masked implementations," Cryptology ePrint Archive, Paper 2017/660, 2017.

[13] Y. Rubner, C. Tomasi, and L. Guibas, "A metric for distributions with applications to image databases," in *Sixth International Conference on Computer Vision (IEEE Cat. No.98CH36271)*, 1998, pp. 59–66.

[14] V. M. Panaretos and Y. Zemel, "Statistical aspects of wasserstein distances," *Annual Review of Statistics and Its Application*, vol. 6, no. 1, p. 405–431, Mar. 2019.

[15] Z. Wang, D. Zhou, M. Yang, Y. Zhang, C. Rao, and H. Wu, "Robust document distance with wasserstein-fisher-rao metric," in *Asian Conference on Machine Learning*. PMLR, 2020, pp. 721–736.

[16] A. Abdelwahab and N. Landwehr, "Deep distributional sequence embeddings based on a wasserstein loss," *Neural Processing Letters*, vol. 54, pp. 3749–3769, 2022.

[17] T. Mosavirik, P. Schaumont, and S. Tajik, "Impedanceverif: On-chip impedance sensing for system-level tampering detection," *IACR Transactions on Cryptographic Hardware and Embedded Systems*, vol. 2023, no. 1, p. 301–325, Nov. 2022.

[18] X. Li, X. Ren, L. Ning, and C. Ou, "Fusion channel attack with POI learning encoder," Cryptology ePrint Archive, Paper 2024/1092, 2024.

[19] S. Mangard, E. Oswald, and T. Popp, *Power Analysis Attacks: Revealing the Secrets of Smart Cards*, 1st ed. Springer Publishing Company, Incorporated, 2010.

[20] A. Battistello, J.-S. Coron, E. Prouff, and R. Zeitoun, "Horizontal side-channel attacks and countermeasures on the isw masking scheme," in *Cryptographic Hardware and Embedded Systems – CHES 2016*, B. Gierlichs and A. Y. Poschmann, Eds. Berlin, Heidelberg: Springer Berlin Heidelberg, 2016, pp. 23–39.

[21] H. Guntur, J. Ishii, and A. Satoh, "Side-channel attack user reference architecture board sakura-g," in *2014 IEEE 3rd Global Conference on Consumer Electronics (GCCE)*, 2014, pp. 271–274.

[22] G. Bertoni, J. Daemen, M. Peeters, and G. Van Assche, "The keccak reference," 2011.

[23] O. Reparaz, B. Gierlichs, and I. Verbauwhede, "Fast leakage assessment," in *Cryptographic Hardware and Embedded Systems – CHES 2017*, ser. Lecture Notes in Computer Science, vol. 10529. Springer, 2017, pp. 387–399.