BENCHMARKING OF SCORING FUNCTIONS FOR BIAS-BASED FINGERPRINTING CODE

Minoru Kuribayashi

Okayama University
Graduate School of Natural Science and Technology
3-1-1, Tsushima-naka, Kita-ku, Okayama, 700-8530 Japan.
kminoru@okayama-u.ac.jp

ABSTRACT

The study of universal detector for fingerprinting code is strongly dependent on the design of scoring function. The best detector is known as the MAP detector that calculates an optimal correlation score, but the number of colluders and their collusion strategy are inevitable. Although there are some scoring functions under some collusion strategies and asymptotic analyses, their numerical evaluation has not been done. In this study, their performance is evaluated for some typical collusion strategies using a discretized bias-based binary fingerprinting code. We also propose a simple but efficient scoring function based on a heuristic observation.

Index Terms— fingerprinting code, scoring function, collusion attack

1. INTRODUCTION

Fingerprinting technique is a means for tracing illegal users from a pirated copy. At the distribution of a copy, digital content is watermarked uniquely by embedding fingerprint information. The main threat of the technique is the collusion attack such that a coalition of users compare their copies and modify/delete the embedded information. One of the most important breakthroughs is the invention of bias-based fingerprinting code such as Tardos code [1] and Nuida code [2].

The Tardos' scoring function [1] which computes a level of suspicion for each user was improved by Škorić et al. [3] considering the symmetric characteristic of code generation process. If the score exceeds a certain threshold, the corresponding user is regarded as guilty. The detector of fingerprinting code includes the scoring function and the decision function with a false-positive probability as input. The Tardos' and Škorić's detectors provide a stable performance independent of the collusion strategy. The decoder does not take the collusion strategy into account at the scoring function.

If the number c of colluders and the collusion strategy $\boldsymbol{\theta_c}$ are known in advance, the MAP detector [4] is optimal. The difficulty in the algorithm is how to estimate the number of colluders and their collusion strategy from a given pirated codeword. In case of mismatch of estimated parameters, the results may differ drastically. Oosterwijk et al. [5] studied the impacts of mismatching cases on the score, and claimed that the interleave-defense is the better choice among some typical collusion strategies. Related to the study, the design of universal detector gets much attention. The universal detector implies that for arbitrary collusion strategy the performance is better than the stable detector like the Škorić's symmetric scoring function.

In this paper, we review conventionally proposed scoring functions and investigate the performance using the discretized bias-based code. In the comparison, we assume the detection strategy as catch-many type which try to catch as many collders as possible under a constant false-positive. Different from the conventional works, the bias equalizer [6] utilizes the characteristics of discretized bias distribution. In a bias-based binary fingerprinting codewords, the number of symbols "1" in a codeword is expected to be equal to that of symbols "0", though the probability that a symbol at a certain element is biased in a codeword. After a collusion attack, the number of symbols is not always balanced in a pirated codeword, and hence, the information about the collusion attack can be derived from the observation of symbols. Such biases of symbols are exploited to calculate weights for correlation scores in the bias equalizer.

We also study the scoring function of bias equalizer and propose a classification method of attack strategy in a heuristic approach. From the observation of effects on scoring function, we discover that some weighting parameters in the scoring function should be excluded or be replaced with better ones according to collusion strategies. The performance is also compared with the conventional scoring functions.

2. FINGERPRINTING CODE

Since Tardos [1] introduced a new construction of fingerprinting code, the class of bias-based code becomes popular. In this section, we review the construction and the related topic on the bias-based code.

A binary codeword of j-th user is denoted by $X_{j,i} \in \{0,1\}$, $(1 \leq i \leq L)$, where $X_{j,i}$ is generated from an in-

dependently and identically distributed random number with a probability p_i such that $\Pr[X_{j,i} = 1] = p_i$ and $\Pr[X_{j,i} = 0] = 1 - p_i$. This probability p_i in the Tardos code follows a continuous distribution $\mathcal{P}$ over an open unit interval $(0, 1)$, which is called *bias distribution*. Nuida et al. [7] discretized the distribution $\mathcal{P}$ to obtain the optimal one for arbitrary collusion size. In essence, the construction of the codeword is similar to that of Tardos code and the same detector can be applied for a given probability p_i. For both cases, the maximum number of colluders should be determined at the setup of code. For convenience, we denote by c_{max} the number in this paper. In the operational mode, the parameters p_i and $X_{j,i}$ and c_{max} have been already determined.

Suppose that a pirated codeword $y_i,\ 1 \le i \le L$ is produced by a certain collusion strategy from c colluders. The correlation score of j-th user is calculated by the sum of each piece $S_{j,i}$ for each element c_i of codeword with length L, namely, $S_j = \sum_{i=1}^{L} S_{j,i}$. The original tracing algorithm only use a half of information from a pirated copy because the value of the score $S_{j,i}$ becomes zero when $y_i = 0$. In order to utilize the whole information, Škorić et al. [3] proposed a symmetric version of the scoring function.

$$S_{j,i}^{sym} = \begin{cases} \sqrt{\frac{p_i}{1-p_i}} & \text{if } X_{j,i} = y_i = 0 \\ -\sqrt{\frac{p_i}{1-p_i}} & \text{if } X_{j,i} = 0, y_i = 1 \\ -\sqrt{\frac{1-p_i}{p_i}} & \text{if } X_{j,i} = 1, y_i = 0 \\ \sqrt{\frac{1-p_i}{p_i}} & \text{if } X_{j,i} = y_i = 1 \end{cases} \quad (1)$$

A single detector calculates each score to each user based on the available data. The performance of the detector strongly depends on the scoring function whether it can distinguish between two cases: user j is guilty or innocent. The scoring function should consider the trade-off between the false-positive error (accusing an innocent user) and the false-negative error (not accusing a guilty user).

According to the Neyman-Peason lemma, the optimal scoring function is given by the following log-likelihood ratio:

$$S_{j,i}^{MAP} = \log\left(\frac{\Pr[y_i|X_{j,i}, p_i, \boldsymbol{\theta_c}]}{\Pr[y_i|p_i, \boldsymbol{\theta_c}]}\right). \quad (2)$$

3. UNIVERSAL SCORING FUNCTION

The analysis in this paper targets for a single detector such that calculates a score per user. There are some methods to calculate the score $S_{j,i}$ from p_i, $X_{j,i}$, y_i, and some other parameters. In this section, we briefly review the methods.

3.1. Oosterwijk's Method

The universality and optimality of scoring function tailored against some attacks has been investigated in [5]. When it is tailored against the *interleaving* attack, the min-max game for the asymptotic code rate has a saddle-point. So, Oosterwijk proposed the following scoring function.

$$S_{j,i}^{Oos} = \begin{cases} \frac{1}{1-p_i} - 1 & \text{if } X_{j,i} = y_i = 0 \\ -1 & \text{if } X_{j,i} \neq y_i \\ \frac{1}{p_i} - 1 & \text{if } X_{j,i} = y_i = 1 \end{cases} \quad (3)$$

In order to bound its amplitude, the cut-off parameter of the bias probability should be given.

3.2. Laarhoven's Method

Laarhoven [8] showed that the detector designed against the *interleave* attack is a universal detector, and achieves the fingerprinting capacity under the condition such that the collusion strategy is not given. In order to get rid of the cut-off parameter, *i.e.* $\mathcal{P} = (0, 1)$, in the Oosterwijk's method, the scoring function is modified by strictly analyzing the effects of interleave attack.

$$S_{j,i}^{Laa} = \begin{cases} \log\left(1 + \frac{p_i}{c(1-p_i)}\right) & \text{if } X_{j,i} = y_i = 0 \\ \log\left(1 - \frac{1}{c}\right) & \text{if } X_{j,i} \neq y_i \\ \log\left(1 + \frac{1-p_i}{cp_i}\right) & \text{if } X_{j,i} = y_i = 1 \end{cases} \quad (4)$$

The advantage of the scoring function is to get rid of the cut-off parameter. However, it needs to know c.

3.3. Meerwald's Method

Under the assumption that the real collusion size c is less than or equal to c_{max}, the correlation score in [9] is respectively calculated for c selected from [1, c_{max}], where the scoring function is based on the MAP detector designed for *WCA* $\boldsymbol{\theta_t^{WCA}}, 1 \le t \le c_{max}$. The score $S_{j,i}^{Mee}$ is determined one of the c_{max} candidates which value becomes maximum.

$$S_{j,i}^{Mee} = \max_{1 \le t \le c_{max}} \left\{ \log\left(\frac{\Pr[y_i|X_{j,i}, p_i, \boldsymbol{\theta_t^{WCA}}]}{\Pr[y_i|p_i, \boldsymbol{\theta_t^{WCA}}]}\right) \right\} \quad (5)$$

3.4. Desoubeaux's Method

Similar to the Meerwald's method, Desoubeaux [10] customized the scoring function for *coin flip* attack defense and aggregate c_{max} candidates in the following manner.

$$S_{j,i}^{Des} = \log\left(\sum_{t=1}^{c_{max}} t \cdot \left(\frac{\Pr[y_i|X_{j,i}, p_i, \boldsymbol{\theta_t^{coin}}]}{\Pr[y_i|p_i, \boldsymbol{\theta_t^{coin}}]}\right)\right) \quad (6)$$

3.5. Bias Equalizer

In binary fingerprinting codes, the number of symbols "0" and "1" is generally balanced because of the design of the codeword. After a collusion attack, the number of symbols is not always balanced in a pirated codeword. Certain information

about the collusion attack can be derived from the statistical analysis of symbols. The bias of symbols is utilized to calculate weights for correlation scores in [6].

Suppose that there are n candidates for the probability p_i in a discrete version of bias distribution. Hence, $p_i, (1 \le i \le L)$ can be classified into n groups. The number of elements in each group is expected to be $q_\xi L, (1 \le \xi \le n)$, where q_ξ is the emerging probability of p_i in the group. For convenience, the number of elements is denoted by ℓ_ξ where $\ell_\xi \ge 0$ and $\sum_{\xi=1}^{n} \ell_\xi = L$, and the numbers of symbols "0" and "1" are denoted by $\ell_{\xi,0}$ and $\ell_{\xi,1}$, respectively.

Without loss of generality, we classify the elements y_i of a pirated codeword into n groups according to the probability p_i. At ξ-th group, the correlation score $S^{Bias}_{j,i,\xi}$ is calculated as follows.

$$S^{Bias}_{j,i,\xi} = \begin{cases} \frac{\ell_{\xi,1}}{\ell_\xi}\sqrt{\frac{p_i}{1-p_i}} & \text{if } X_{j,i} = y_i = 0 \\ -\frac{\ell_{\xi,0}}{\ell_\xi}\sqrt{\frac{p_i}{1-p_i}} & \text{if } X_{j,i} = 0, y_i = 1 \\ -\frac{\ell_{\xi,1}}{\ell_\xi}\sqrt{\frac{1-p_i}{p_i}} & \text{if } X_{j,i} = 1, y_i = 0 \\ \frac{\ell_{\xi,0}}{\ell_\xi}\sqrt{\frac{1-p_i}{p_i}} & \text{if } X_{j,i} = y_i = 1 \end{cases} \tag{7}$$

Let $\mathcal{I}_\xi$ be the set of indices i which probability p_i belongs to ξ-th group. Then, the total score S_j is calculated by

$$S_j = \sum_{\xi=1}^{n} \sum_{i \in \mathcal{I}_\xi} S^{Bias}_{j,i,\xi}. \tag{8}$$

4. PROPOSED METHOD

It is experimentally reported in [6] that the bias equalizer improves the performance of symmetric detector for some typical collusion strategies. However, there are considerable gaps for the all-0, all-1, minority, and coin-flip attacks.

When all-0 or all-1 attack is performed, the following relationships are observed.

$$\begin{cases} \ell_{\xi,0} \approx \ell_\xi, & \text{if } p_i < 0.5 \text{ holds for all } \xi \\ \ell_{\xi,1} \approx \ell_\xi, & \text{if } p_i > 0.5 \text{ holds for all } \xi \end{cases} \tag{9}$$

In the above cases, we empirically adjust the scoring function to maximize the total performance.

$$S^{Bias^\dagger}_{j,i,\xi} = \begin{cases} \frac{\ell_{\xi,0}\ell_{\xi,1}}{L_0}\sqrt{\frac{p_i}{1-p_i}} & \text{if } X_{j,i} = y_i = 0 \\ -\frac{\ell_{\xi,0}\ell_{\xi,1}}{L_1}\sqrt{\frac{p_i}{1-p_i}} & \text{if } X_{j,i} = 0, y_i = 1 \\ -\frac{\ell_{\xi,0}\ell_{\xi,1}}{L_0}\sqrt{\frac{1-p_i}{p_i}} & \text{if } X_{j,i} = 1, y_i = 0 \\ \frac{\ell_{\xi,0}\ell_{\xi,1}}{L_1}\sqrt{\frac{1-p_i}{p_i}} & \text{if } X_{j,i} = y_i = 1 \end{cases} \tag{10}$$

where L_0 and L_1, respectively, stand for the number of "0" and "1" elements in a pirated codeword $\boldsymbol{y}$. For the classification, we introduce a threshold $T^\dagger$ to check the following two cases:

- $\ell_{\xi,0}/\ell_\xi > T^\dagger$ holds for all ξ that satisfies $p_i < 0.5$
- $\ell_{\xi,1}/\ell_\xi > T^\dagger$ holds for all ξ that satisfies $p_i > 0.5$.

If one of the above two conditions is satisfied, the correlation score $S^{Bias^\dagger}_{j,i,\xi}$ is used instead of $S^{Bias}_{j,i,\xi}$ in Eq.(8).

When the minority or coin-flip attack is performed, the following relationships are observed for ξ-th group.

$$\begin{cases} \frac{\ell_{\xi,0}}{\ell_{\xi,1}} < \sqrt{\frac{1-p_i}{p_i}}, & \text{if } p_i < 0.5 \\ \frac{\ell_{\xi,1}}{\ell_{\xi,0}} < \sqrt{\frac{p_i}{1-p_i}}, & \text{if } p_i > 0.5 \end{cases} \tag{11}$$

When p_i or $1 - p_i$ is not close to 0, their scores $S^{Bias}_{j,i,\xi}$ affect the total sum in a negative way as an interference. Hence, these scores are excluded in the proposed method. We also introduce a threshold $T^\ddagger$ for the classification of such a case. Let $n^\ddagger$ be the number of the groups ξ that satisfy Eq.(11). If $n^\ddagger \ge n/2$, then the correlation score is calculated by

$$S^{Bias^\ddagger}_{j,i,\xi} = \begin{cases} 0 & \text{if } T^\ddagger < p_i < 1 - T^\ddagger \\ S^{Bias}_{j,i,\xi} & \text{otherwise} \end{cases}, \tag{12}$$

and it is used instead of $S^{Bias}_{j,i,\xi}$ in Eq.(8).

5. EXPERIMENTAL RESULTS

For the comparison of the performance of scoring functions enumerated in the previous section, we perform the Monte Carlo simulation such that pirated codewords are produced by collusion attack on randomly selected 10^3 combinations of c colluders. The number of users was $N = 10^6$, and the false-positive probability was $\epsilon_1 = 10^{-10}$, namely, $\eta \approx 10^{-4}$. We used Nuida code with $c_{max} = 8$ and $L = \{1024, 2048\}$. The thresholds in a proposed method are fixed by $T^\dagger = 0.95$ and $T^\dagger = 0.1$ through the experiments, and is calculated by rare event simulator[1].

Tables 1 shows the sum of detected colluders for $2 \le c \le 10$, where the maximum is $54 = \sum_{c=2}^{10} c$. In case of the symmetric scoring function [3], the obtained results indicate that the "minority" strategy is the most damaging one, though it is supposed to be stable against collusion strategies. The fluctuation may come from the parameters for encoding codewords and the threshold calculated by the rare event simulator.

It is noticed from the results that the worst strategy is "minority" or "WCA" under this condition. Because some universal detectors neglect the "minority" strategy, the attack happens to become the worst case. It is remarkable that the universal detector proposed by Laarhoven [8] is defeated by the "minority" attack in a sense that the performance is below the symmetric scoring function. From the table, it can be said that the Meerwald's and Desoubeaux's methods are superior because the worst score is the maximum among these methods and is almost coincident with the MAP method. For the

[1] We extracted it from the source code downloaded from `http://www.irisa.fr/texmex/people/furon/fp.zip`

Table 1. Sum of detected colluders for $2 \leq c \leq 10$, where the values are at most $54 = \sum_{c=2}^{10} c$ and the values represented by bold font are the worst case.

(a) $L = 1024$

	majority	minority	coin-flip	all-0	all-1	interleave	WCA	total
symmetric [3]	7.16	**6.32**	6.75	6.73	6.72	6.93	6.72	47.33
MAP	21.26	53.70	9.37	30.30	30.62	9.94	**8.65**	163.84
Oosterwijk [5]	17.17	8.19	7.81	7.87	7.76	8.62	**7.45**	64.87
Laarhoven [8]	16.54	**5.82**	7.80	7.85	7.76	9.96	7.77	63.50
Meerwald [9]	9.83	11.11	9.00	9.08	9.02	8.73	**8.64**	65.41
Desoubeaux [10]	9.76	10.90	9.01	9.08	9.01	8.68	**8.64**	65.08
Bias Equalizer [6]	21.14	**6.69**	7.72	18.58	18.65	9.70	7.69	90.17
Proposed	21.14	32.72	7.70	24.65	24.76	9.70	**7.39**	128.06

(b) $L = 2048$

	majority	minority	coin-flip	all-0	all-1	interleave	WCA	total
symmetric [3]	14.65	**13.39**	13.88	13.91	13.94	14.33	14.05	98.15
MAP	45.33	54.00	23.16	53.83	53.85	21.88	**17.97**	270.03
Oosterwijk [5]	36.84	19.43	17.47	17.30	17.44	20.06	**15.89**	144.43
Laarhoven [8]	35.32	**12.49**	16.12	15.98	16.10	21.91	16.62	134.54
Meerwald [9]	18.89	23.07	20.32	20.16	20.18	18.70	**17.97**	139.29
Desoubeaux [10]	18.74	22.91	20.08	19.93	19.96	19.10	**17.95**	138.67
Bias Equalizer [6]	45.09	**15.02**	15.71	43.60	43.57	21.40	16.45	200.84
Proposed	45.09	53.82	19.18	52.40	52.37	21.40	**16.39**	260.65

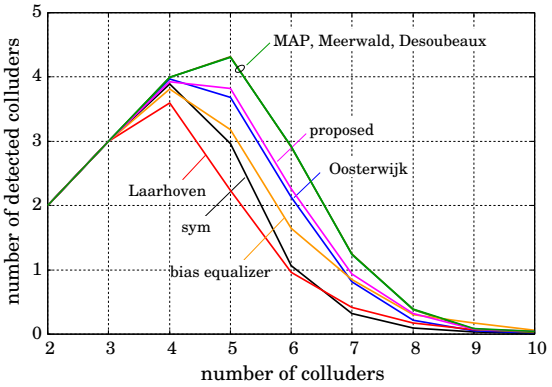

Fig. 1. Comparison of the worst case when $L = 2048$.

comparison, the number of detected colluders versus the number of colluders is depicted in Fig.1 when each worst attack is performed for each scoring function. It is observed that the performance of Meerwald's and Desoubeaux's methods are very close to that of MAP method while the Laarhoven's method becomes lower than the symmetric scoring function.

From a different point of view, the bias equalizer and the proposed method show the better total balance against the collusion tolerance. Especially for the proposed method, the total sum is very close to the optimal (MAP) detector when $L = 2048$. As for the "minority", "all-0", and "all-1" strategies, it is remarkable that the proposed method is much better than the others. Therefore, the advantage of the proposed method is the total balance of collusion tolerance for some possible collusion strategies.

6. CONCLUSION

In this paper, we studied the characteristic of scoring functions and experimentally evaluated their performance. If the estimation of collusion strategies and the number of colluders is failed for the MAP detector, the degradation of the performance is not negligible. Hence, the accurate estimation of c and $\boldsymbol{\theta}_{\boldsymbol{c}}$ is essential.

Among some universal detectors, we discovered that the total performance of the bias equalizer is better than the others. The proposed method classifies collusion strategies using two thresholds $T^{\dagger}$ and $T^{\ddagger}$ into three types, and customizes the bias equalizer. Though the proposed method estimate the attack strategy from the observation of a pirated codeword, the total performance against some typical collusion strategies are much better than the conventional universal detectors. One of our future works is to investigate the theoretical analysis for the bias equalizer.

7. REFERENCES

[1] Gábor Tardos, "Optimal probabilistic fingerprint codes," *J. ACM*, vol. 55, no. 2, pp. 1–24, 2008.

[2] K. Nuida, M. Hagiwara, H. Watanabe, and H. Imai, "Optimization of Tardos's fingerprinting codes in a viewpoint of memory amount," in *IH 2007*. 2008, vol. 4567 of *LNCS*, pp. 279–293, Springer, Heidelberg.

[3] B. Škorić, S. Katzenbeisser, and M. Celik, "Symmetric Tardos fingerprinting codes for arbitrary alphabet sizes," *Designs, Codes and Cryptography*, vol. 46, no. 2, pp. 137–166, 2008.

[4] T. Furon and L. P. Preire, "EM decoding of Tardos traitor tracing codes," in *ACM Multimedia and Security*, 2009, pp. 99–106.

[5] J. J. Oosterwijk, B. Škorić, and J. Doumen, "A capacity-achieving simple decoder for bias-based traitor tracing schemes," *IEEE Trans. Inform. Theory*, vol. 61, no. 7, pp. 3882–3900, 2015.

[6] M. Kuribayashi, "Bias equalizer for binary probabilistic fingerprinting codes," in *IH 2012*. 2012, vol. 7692 of *LNCS*, pp. 269–283, Springer, Heidelberg.

[7] K. Nuida, S. Fujitu, M. Hagiwara, T. Kitagawa, H. Watanabe, K. Ogawa, and H. Imai, "An improvement of discrete Tardos fingerprinting codes," *Designs, Codes and Cryptography*, vol. 52, no. 3, pp. 339–362, 2009.

[8] T Laarhoven, "Capacities and capacity-achieving decoders for various fingerprinting games," in *Proc. IH&MMSec2014*, 2014, pp. 123–134.

[9] P. Meerwald and T. Furon, "Towards practical joint decoding of binary Tardos fingerprinting codes," *IEEE Trans. Inform. Forensics and Security*, vol. 7, no. 4, pp. 1168–1180, 2012.

[10] M. Desoubeaux, C. Herzet, W. Puech, and G. Le Guelvouit, "Enhanced blind decoding of Tardos codes with new MAP-based functions," in *Proc. MMSP*, 2013, pp. 283–288.