WORST CASE ATTACKS AGAINST BINARY PROBABILISTIC TRAITOR TRACING CODES

Teddy Furon, Luis Pérez-Freire

WORST CASE ATTACKS AGAINST BINARY PROBABILISTIC TRAITOR TRACING CODES

T. Furon

Thomson
Security Labs
Cesson-Sévigné, France

L. Pérez-Freire *

Gradiant
ETSI Telecom.
Vigo, Spain

ABSTRACT

This article deals with traitor tracing which is also known as active fingerprinting, content serialization, or user forensics. We study the impact of worst case attacks on the well-known Tardos binary probabilistic traitor tracing code, and especially its optimum setups recently advised by Amiri and Tardos, and by Huang and Moulin. This paper assesses that these optimum setups are robust in the sense that a discrepancy between the foreseen numbers of colluders and the its actual value doesn't spoil the achievable rate of a joint decoder. On the other hand, this discrepancy might have a dramatic impact on a single decoder. Since the complexity of the today's joint decoder is prohibitive, this paper mitigates the interest of the optimum setups.

Index Terms— traitor tracing, worst case attacks

1. INTRODUCTION

The beginning of the year 2009 witnessed a major breakthrough in probabilistic traitor tracing codes showing how to achieve the optimum performances against a collusion of c dishonest users. In 2003, Gabor Tardos proposed a family of provably good codes. Two very recents and independents works show how to fine-tune this code to achieve the fingerprinting capacity [1, 2]. The main difficulty is that the collusion has an infinite number of attacks, ie. ways of mixing their contents in order to forge the pirated copy. Therefore, the performances must be guaranteed even for the worst case attack (WCA). On the other hand, the designer of the code can tune the time-sharing parameter distribution in order to maximize the performances of the code. The two recent works cast this problem in the game theory field where the pay-off function is the rate of a joint decoder, ie. the mutual information between the pirated sequence and the group of c colluders' codewords. The major result is that there is indeed a saddle point defined by the equilibrium attack and time-sharing parameter. The collusion and the designer of the code have no interest to divert from this equilibrium point. The pay-off function at the equilibrium is by definition the fingerprinting capacity. Moreover, Amiri and Tardos proposed a decoding algorithm taking full advantage of this optimum rate [1].

The goal of this paper is to mitigate these results. Our first argument is that the pay-off function used in these works is the capacity bounding the performances of a joint decoder. A joint decoder analyzes groups of c users, as the one proposed by Amiri and Tardos. Its complexity is simply unaffordable. From the number of the basic operations to process one group given in [1, Sec. 7], and assuming that there are $c = 4$ pirates, the decoder would require some months to process $n = 100,000$ users or several dozens of centuries to process one million of users when run on a IBM RoadRunner, which is the most powerful computer on Earth today! As far as we know, the only affordable accusation process is the single decoder, whose performances are bounded by the mutual information between the pirated sequence and one codeword. In other word, we propose to change the pay-off function.

The second argument is that the capacity achieving setup given in [1, 2] is strongly dependent on the size of collusion. In other words, the designer of the code must foresee in advance the number of pirates, or at least commit to a maximum number. This paper shows three important points:

- For the joint decoder, a mismatch between the foreseen collusion size and the real number of colluders is not at all a matter. The optimum setup is in a way robust.
- For the the single decoder, with the new pay-off function, this optimum setup can be very dangerous if the expected number of colluders is wrong. In certain circumstances, the rate cancels so that any single decoder will fail in reliably accusing.
- On the other hand, the Tardos and the flat distributions have an achievable rate closed to the capacity wrt joint decoding, and a gracefully decreasing achievable rate wrt to single decoding.

The demonstration of these three points is based on the study of the worst case attack (WCA) against joint (Sec. 3) and single (Sec. 4) decoders for a given setup. We now start by introducing the notations and the mathematical model needed to derive the WCA.

2. MATHEMATICAL MODEL

Random variables and their realizations are denoted by capital and lowercase letters, respectively. Boldface letters denote column vectors. Calligraphic letters are reserved for sets. $\Pr_X[x]$ is the probability that the discrete random variable X takes the value x. The shorthand $[m]$ will be used to denote the sequence of indices $\{1, \dots, m\}$. $H(.)$ is the entropy of a discrete random variable. $h_b(x) = -x\log(x) - (1-x)\log(1-x)$ is the binary entropy function. $D_{KL}(\Pr_X || \Pr_Y)$ is the Kullback-Leibler divergence or relative entropy between the random variables X and Y. All logarithms are to the base 2, so all rates and entropies are given in bits.

2.1. Binary probabilistic code with time-sharing

We briefly remind how the Tardos code is designed. The binary code $\mathcal{X}$ is composed of n sequences of m bits. The sequence $\mathbf{X}_j = (X(j,1), \cdots, X(j,m))^T$ identifying user j is composed of m independent binary symbols, with $\Pr_{X(j,i)}[1] = p_i, \forall i \in [m]$. The auxiliary random variables $\{P_i\}_{i=1}^m$ are independent and identically distributed in the range $[0, 1]$ according to the probability density

*The second author performed the work while at Thomson Security Labs

function f: $P_i \sim f$. Both the code $\mathcal{X}$ and the time-sharing sequence $\mathbf{p} = (p_1, \ldots, p_m)^T$ must remain as secret parameters. The rate of the code is defined by $R = \log(n)/m$.

This pdf f is of utmost importance. Tardos originally proposed $f_T(p) = (\pi\sqrt{p(1-p)})^{-1}$, whereas [1, 2] showed that, against a collusion of size c, the capacity achieving $f_c^\star(p)$ depends on c and is indeed a probability mass function:

$$f_c^\star(p) = \sum_{k \in [|\Pi(c)|]} w_{c,k}\delta(p - \pi_{c,k}). \quad (1)$$

The auxiliary variables are thus discrete and belong to the set $\Pi(c) = \{\pi_{c,k}\}_k$, a.k.a the support of $f_c^\star$.

2.2. Collusion process

Denote the subset of colluder indices by $\mathcal{C} = \{j_1, \cdots, j_c\}$, and $\mathcal{X}_\mathcal{C} = \{\mathbf{X}_{j_1}, \ldots, \mathbf{X}_{j_c}\}$ the restriction of the code to this subset. The collusion attack is the process of taking sequences in $\mathbf{X}_\mathcal{C}$ as inputs and yielding the pirated sequence $\mathbf{Y}$ as an output.

The marking assumption [3] states that, in its narrow-sense version, whatever the strategy of the collusion $\mathcal{C}$, we have $Y(i) \in \{X(j_1, i), \cdots, X(j_c, i)\}$. In words, colluders forge the pirated copy by assembling chunks from their personal copies. It implies that if, at index i, the colluders' symbols are identical, then this symbol value is decoded at the i-th chunk of the pirated copy.

The usual mathematical model of the collusion is essentially based on four main assumptions: the collusion attack is memoryless, stationary, possibly random, and permutation invariant (a.k.a. symmetric). The collusion attack is thus fully described by the following parameter vector: $\boldsymbol{\theta} = (\theta_0, \ldots, \theta_c)^T$, with $\theta_\sigma = \Pr_Y[1|\Sigma = \sigma]$, where the random variable $\Sigma \in \{0, \ldots, c\}$ denotes the number of symbol '1' in the colluders' copies at a given index. The marking assumption enforces that $\theta_0 = 0$ and $\theta_c = 1$. The authors of [1] also speak about 'eligible channel'. We denote $\Theta(c)$ the set of attacks for a collusion of size c, which is a hypercube of dimension $c - 1$.

2.3. Decoding families

The study of traitor tracing codes from an achievable rate standpoint largely decouples their performances from any particular decoding algorithm. However, there exist two different families of decoders: the simple decoder [4, Sec. 4] and the joint decoder [4, Sec. 5]. The simple decoder calculates a score from a user codeword and the pirated sequence, whereas the joint decoder calculates a score from a group of c user codewords and the pirated sequence. Due to their different nature, the two families have different achievable rates. Briefly, the joint decoder represents what the accusation side could do in an ideal world where complexity is not a matter, and it has been shown to be capacity-achieving. However, it has to tackle $\binom{n}{c}$ groups which seems hardly affordable for large n.

2.3.1. Joint decoder

The achievable rate for the joint decoder against a given collusion attack is based on the mutual information between Y, a symbol of the pirated sequence, and $X_\mathcal{C}$, the symbols of the colluders' code sequences [4, Sec. 5]. This holds for any index thanks to the symbol independence, and this is taken in expectation over the time-sharing random variable P:

$$\begin{aligned} R_{\text{joint}}(f, \boldsymbol{\theta}) &= \frac{1}{c}\mathbb{E}_P\left[I(Y; X_\mathcal{C}|p, \boldsymbol{\theta})\right] \\ &= \frac{1}{c}\left(\mathbb{E}_P\left[H(Y|p, \boldsymbol{\theta})\right] - \mathbb{E}_P\left[H(Y|\Sigma, p, \boldsymbol{\theta})\right]\right) \quad (2) \end{aligned}$$

Plugging the collusion model introduced in Sec. 2.2, we have:

$$\begin{aligned} \Pr_Y[1|p, \boldsymbol{\theta}] &= \sum_{\sigma=0}^{c} \theta_\sigma \Pr_\Sigma[\sigma|p], \quad (3) \\ \Pr_Y[1|\sigma, p, \boldsymbol{\theta}] &= \theta_\sigma, \quad (4) \end{aligned}$$

with $\Pr_\Sigma[\sigma|p] = \binom{c}{\sigma} p^\sigma (1-p)^{(c-\sigma)}$, known as the Bernstein polynomials [5]. Therefore, Eq. (2) can be rewritten as:

$$R_{\text{joint}}(f, \boldsymbol{\theta}) = \frac{1}{c}\mathbb{E}_P\left[h_b\left(\Pr_Y[1|p, \boldsymbol{\theta}]\right) - \sum_{\sigma=0}^{c} \Pr_\Sigma[\sigma|p]\right] h_b(\theta_\sigma). \quad (5)$$

2.3.2. Simple decoder

The achievable rate for the simple decoder against a given collusion attack is given in [4, Sec. 4]:

$$\begin{aligned} R_{\text{simple}}(f, \boldsymbol{\theta}) &= \mathbb{E}_P\left[I(Y; X|p, \boldsymbol{\theta})\right] \\ &= \mathbb{E}_P\left[H(Y|p, \boldsymbol{\theta})\right] - \mathbb{E}_P\left[H(Y|X, p, \boldsymbol{\theta})\right] \quad (6) \\ &= \mathbb{E}_P\left[D_{KL}(\Pr_{X,Y}||\Pr_X \Pr_Y|p, \boldsymbol{\theta})\right] \quad (7) \end{aligned}$$

This links the notion of rate to the inherent capability of distinguishing two hypothesis:

- $\mathcal{H}_0$: User j is innocent, and his codeword is independent of Y: $\Pr_{Y,X|\mathcal{H}_0} = \Pr_{Y|\boldsymbol{\theta}} \Pr_X$,
- $\mathcal{H}_1$: User j is guilty and Y has been created from his codeword: $\Pr_{Y,X|\mathcal{H}_1} = \Pr_{Y|X,\boldsymbol{\theta}} \Pr_X$.

The calculation of the rate needs the expressions of the conditional probabilities induced by the collusion model:

$$\begin{aligned} \Pr_Y[1|X = 1, p, \boldsymbol{\theta}] &= \sum_{k=1}^{c} \theta_k \binom{c-1}{k-1} p^{k-1}(1-p)^{(c-k)}, \quad (8) \\ \Pr_Y[1|X = 0, p, \boldsymbol{\theta}] &= \sum_{k=0}^{c-1} \theta_k \binom{c-1}{k} p^{k}(1-p)^{(c-k-1)}. \quad (9) \end{aligned}$$

3. WCA AGAINST JOINT DECODERS

The recent works [1, 2] were able to find the capacity solving the following game:

$$C(c) = \max_{f(p)} \min_{\boldsymbol{\theta} \in \Theta(c)} R_{\text{joint}}(f, \boldsymbol{\theta}).$$

The pay-off function R_{joint} is linear in f and convex wrt $\boldsymbol{\theta}$. For a given collusion size, the WCA against the optimum $f_c^\star$ was found thanks to some specific properties only holding at the equilibrium of the game. This section addresses a slightly different problem: for any given pdf f and collusion size c, what is the WCA $\boldsymbol{\theta}_c^\star$ against a joint decoder? The WCA is defined as follows:

$$\boldsymbol{\theta}_c^\star = \arg\min_{\boldsymbol{\theta} \in \Theta(c)} R_{\text{joint}}(f, \boldsymbol{\theta}) \quad (10)$$

$R_{\text{joint}}(f, \boldsymbol{\theta}_c^\star)$ is called the achievable rate in the sense that the code designer can rely on a rate greater or equal to it, whatever the collusion attack.

3.1. A Blahut-Arimoto algorithm

We resort now to the expression (2) of the joint decoding rate. The problem of minimizing this function can be rewritten as a double minimization, exactly like the Blahut-Arimoto algorithm for the computation of the rate-distortion function [6]. The main difference is that (10) corresponds to a degenerate problem because the only distortion constraint is that $\boldsymbol{\theta} \in \Theta(c)$. The reader is referred to [6] or [7, Chap. 13] for a detailed presentation of the Blahut-Arimoto algorithm as we only explain its application to our model.

In a slight abuse of notation, let us denote the rhs of (5) by $r(\mathrm{Pr}_Y, \boldsymbol{\theta})$. The WCA is disclosed by iteratively minimizing over each argument of this function, keeping the other constant. Thus, each iteration is comprised of two steps:

In the first step of the k-th iteration, for a fixed law $q^{(k-1)}(p) = \mathrm{Pr}_Y[1|p, \boldsymbol{\theta}^{(k-1)}]$, we minimize $r(q^{(k-1)}, \boldsymbol{\theta})$ over $\boldsymbol{\theta}$. Thanks to its convexity, the minimization amounts to canceling the $(c-1)$ partial derivatives (θ_0 and θ_σ are already fixed to 0 and 1, respectively):

$$\frac{\partial r(q^{(k-1)}, \boldsymbol{\theta})}{\partial \theta_\sigma} \propto \mathbb{E}_P \left[\mathrm{Pr}_\Sigma[\sigma|p] \log \frac{\theta_\sigma (1 - q^{(k-1)}(p))}{q^{(k-1)}(p)(1-\theta_\sigma)} \right]. \quad (11)$$

By setting the last expression to 0, we obtain

$$\theta_\sigma^{(k)} = \frac{1}{1 + B^{(k)}(\sigma)}, \ \sigma = 1, \ldots, c-1, \quad (12)$$

with

$$B^{(k)}(\sigma) = \exp \left(\frac{\mathbb{E}_P \left[\mathrm{Pr}_\Sigma[\sigma|p] \log \frac{1-q^{(k-1)}(p)}{q^{(k-1)}(p)} \right]}{\mathbb{E}_P \left[\mathrm{Pr}_\Sigma[\sigma|p] \right]} \right). \quad (13)$$

$B^{(k)}(\sigma)$ is well defined because $q^{(k-1)}(p) = 0$ only for $p = 0$ (resp. $q^{(k-1)}(p) = 1$ only for $p = 1$) where the polynomial $\mathrm{Pr}_\Sigma[\sigma|p]$ also goes to zero. The denominator is not null because there exists a $p \in]0, 1[$ such that $f(p) > 0$ (else, the code generation is void). Finally, Eq. (12) is always between 0 and 1, showing that the constraint $\boldsymbol{\theta} \in \Theta(c)$ is actually inactive.

The second step of the k-th iteration consists in updating the function $q^{(k)}(p)$ with respect to the new collusion model $\boldsymbol{\theta}^{(k)}$ found in the first step. This is done by finding the function $q^{(k)}$ minimizing the functional $r(q, \boldsymbol{\theta}^{(k)})$. We create an extension of the derivative of this functional in q by a Taylor expansion of the difference

$$r(q + \epsilon, \boldsymbol{\theta}^{(k)}) - r(q, \boldsymbol{\theta}^{(k)}) = \mathbb{E}_P \left[\left. \frac{\partial r}{\partial q} \right|_{q(p)} \epsilon(p) \right] + \mathbb{E}_P \left[o(\epsilon(p)) \right].$$

The minimum is reached for a function $q^{(k)}$ such that any perturbation ϵ doesn't change the value of the functional at least up to the first order. In other words, it cancels $\left. \frac{\partial r}{\partial q} \right|_{q(p)}$. This leads to the following update:

$$q^{(k)}(p) = \sum_{\sigma=0}^{c} \theta_\sigma^{(k)} \mathrm{Pr}_\Sigma[\sigma|p]. \quad (14)$$

Very much like for the Blahut-Arimoto algorithm, convergence to the WCA is monotonic, i.e. every step decreases the objective function. Since the optimization problem is convex, convergence to the worst $\boldsymbol{\theta}_c^\star$ is assured. We observe two surprising facts exposed in the following propositions, whose proofs are in our journal article [8].

Proposition 1. *For a symmetric f (being it a continuous pdf or a discrete pmf), the WCA is symmetric, i.e. $\theta_\sigma^* = 1 - \theta_{c-\sigma}^*$, $\forall \sigma \in [c]$.*

Fig. 1. WCA against the joint decoder and Tardos pdf.

Table 1. Achievable rates for the joint decoder (in % wrt capacity).

c	2	3	4	5	6	7
$C(c)$	0.250	0.097	0.054	0.034	0.023	0.017
$f_2^\star$	100	85.4	57.3	37.1	22.6	13.3
$f_3^\star$	88.3	100	97.6	94.7	87.7	78.6
$f_4^\star$	84.4	98.9	100	99.4	94.6	87.0
$f_5^\star$	81.7	96.3	98.1	100	98.5	95.4
$f_6^\star$	78.8	93.2	95.5	99.1	100	99.7
$f_7^\star$	77.5	91.5	93.6	97.7	99.4	100
f_F	77.6	88.4	87.8	89.3	89.2	88.7
f_T	61.5	72.8	74.9	78.4	80.4	81.8

Therefore, the WCA for 2 colluders is $\boldsymbol{\theta}_2^\star = (0, 0.5, 1)^T$.

Proposition 2. *The WCA is asymptotically the uniform (a.k.a interleaving) attack as the number of colluders increases, for any continuous density function f taking strictly positive values over $]0, 1[$:*

$$\arg \min_{\boldsymbol{\theta} \in \Theta(c)} R_{joint}(f, \boldsymbol{\theta}) \overset{c \to +\infty}{=} (0, 1/c, 2/c, \ldots, 1)^T.$$

Figure 1 illustrates the difference $\theta_\sigma^* - \sigma/c$ for the Tardos pdf. The WCA is quickly very close to the uniform attack, which is the most simple collusion process: the colluders roll an unbiased c-face dice to decider which symbol is paste in the pirated copy. Note that Prop. 2 does not tackle discrete pmf f like those proposed in [1, 2]. However, in the discrete case, the same convergence has been conjectured in [2].

3.2. Numerical results

This algorithm allows to calculate the achievable rate when there is a mismatch between the collusion size $\hat{c}$ expected by the designer and the real number c of colluders. Tab. 1 shows the achievable rates as a percentage of the capacity (for a given collusion size per column). We also calculates the achievable rate given by two pdf not requiring any bet on the collusion size: the Tardos pdf f_T and the flat pdf f_F. We verify that $f_{\hat{c}}^\star$ yields the biggest achievable rate, ie. 100% of the capacity, when $c = \hat{c}$. Except for $f_2^\star$, these distributions are quite robust: in case of mismatch, the loss compared to the optimum is surprisingly small. As proven in [2], the achievable rates for the Tardos pdf slowly converge to capacity as c increases, but the difference is substantial for small collusion. The flat pdf indeed gives a better trade-off.

4. WCA AGAINST SIMPLE DECODERS

The problem of the simple decoder is much harder because the payoff function (7) is no longer convex in $\boldsymbol{\theta}$. As far as we know, there is no result about the optimum pdf $f^\star$. This is not the aim of this section. We are rather interested in the WCA for given pdf and collusion size. However, this problem is still too difficult and we have only partial results.

4.1. Continuous time-sharing distribution

We were only able to find the WCA thanks to a simulated annealing minimization algorithm whose complexity is affordable when the

Fig. 2. plot of $\Pr_Y[1|p,\boldsymbol{\theta}_c^\star]$, simple decoder, Tardos pdf.

collusion size is not too big: $c \leq 15$. This allows us to formulate the following conjectures.

Conjecture 1. *For a symmetric f, the WCA is indeed symmetric, i.e. $\theta_\sigma^* = 1 - \theta_{c-\sigma}^*$, $\forall \sigma \in [c]$.*

We can prove this conjecture only for $c = 2$: the WCA for 2 colluders is $\boldsymbol{\theta}_2^\star = (0, 0.5, 1)^T$. This is the same attack as the WCA against a joint decoder.

Conjecture 2. *For the Tardos pdf f_T, $Pr_Y[1|p,\boldsymbol{\theta}_c^\star]$ converges to $q^{conv}(p) = (\arcsin(2p-1))/\pi + 1/2$, as c increases. More specifically, $Pr_Y[1|p,\boldsymbol{\theta}_c^\star]$ is the orthogonal projection of $q^{conv}(p)$ over the affine subspace spanned by the Bernstein polynomials $\{Pr_\Sigma[\sigma|p]\}_{\sigma\in[c-1]}$ and containing $Pr_\Sigma[c|p]$:*

$$\int_0^1 (Pr_Y[1|p,\boldsymbol{\theta}_c^\star] - q^{conv}(p)) Pr_\Sigma[\sigma|p] dp = 0,\ \forall \sigma \in [c-1].$$

We have to perform the projection of $q^{\text{conv}}(p) - \Pr_\Sigma[c|p]$ onto the linear subspace spanned by the Bernstein polynomials. The Durrmeyer-Sevy algorithm is an elegant way to perform this orthogonal projection [5, Th. 2]. Fig. 2 shows the convergence of $\Pr_Y[1|p,\boldsymbol{\theta}_c^\star]$ as c increases.

4.2. Discrete time-sharing distribution

We first tackle the case of $c = 2$ colluders. The pay-off function $R_{\text{simple}}(f,\boldsymbol{\theta}_2^\star)$ reaches its maximum in $p = 1/2$. If the optimum pdf for a single decoder is symmetric, then $f_2^\star(p) = \delta(p - 1/2)$ is the optimum and $\boldsymbol{\theta}_2^\star$ the WCA for both decoding families. This only holds for the special case of two colluders, but this brings a theoretical support to the code recently proposed by Nuida [9], which is a fully randomized code with $\Pr_{X_{i,j}}[1] = 1/2$, ie. a Tardos code tuned on $f_2^\star$.

According to (7), the achievable rate of a single decoder is the weighted sum of mutual informations knowing p in the support $\Pi(c)$. We first study when the colluders can cancel the summand $I(Y;X|p,\boldsymbol{\theta})$. This goal is reached by setting $\Pr_Y[1|X=1,p] = \Pr_Y[1|X=0,p]$. Since these probabilities are linear with $\boldsymbol{\theta}$, it amounts to finding a collusion attack $\boldsymbol{\theta} \in \Theta(c)$ such that

$$\boldsymbol{\theta}^T(\mathbf{q}_{\Sigma 1}(p) - \mathbf{q}_{\Sigma 0}(p)) = 0, \tag{15}$$

with

$$\mathbf{q}_{\Sigma 1}(p) = (\Pr_\Sigma[0|X=1,p],\ldots,\Pr_\Sigma[c|X=1,p])^T \tag{16}$$
$$\mathbf{q}_{\Sigma 0}(p) = (\Pr_\Sigma[0|X=0,p],\ldots,\Pr_\Sigma[c|X=0,p])^T. \tag{17}$$

Proposition 3. *c colluders cannot cancel $I(Y;X|p,\boldsymbol{\theta})$ if $p \notin [\eta_c, 1-\eta_c]$, with $1/c < \eta_c < 2/c$ the smallest real root of the following polynomial*

$$(1-p)^{c-2}(1-cp) + p^{c-1}. \tag{18}$$

Proof. Since the scalar product is linear, $\boldsymbol{\theta}^T(\mathbf{q}_{\Sigma 1} - \mathbf{q}_{\Sigma 0})$ can be written as a convex combination of the scalar products $\rho_i(p) = \mathbf{e}_{i+1}^T(\mathbf{q}_{\Sigma 1} - \mathbf{q}_{\Sigma 0})$, with $\mathbf{e}_{i+1}$ the $(i+1)$-th canonical vector:

$$\rho_i(p) = \tbinom{c}{i} p^{i-1}(1-p)^{c-i-1}(i/c - p), \forall i \in [c].$$

Note that $\rho_1(p)$ is the only one producing negative values over the interval $[1/c, 2/c]$. Therefore, on this interval, we have:

$$\rho_1(p) + \rho_c(p) \leq \boldsymbol{\theta}^T(\mathbf{q}_{\Sigma 1} - \mathbf{q}_{\Sigma 0}),$$

with equality if $\boldsymbol{\theta} = (0,1,0,\ldots,0,1)^T$.

For $c = 3$, $\rho_1(p) + \rho_3(p) = (2p-1)^2 \geq 0$. It is not possible to find any vector $\boldsymbol{\theta} \in \Theta(3)$, except for $p = 1/2$ with $\boldsymbol{\theta} = (0,1,0,1)^T$.

For $c > 3$, the lower bound $\rho_1(p) + \rho_c(p) = (1-p)^{c-2}(1-cp) + p^{c-1}$ is strictly positive for $p \in [0, 1/c]$ and negative for $p \in [2/c, 1/2]$. Therefore, there exists some $\eta_c \in [1/c, 2/c]$ such that, for $p < \eta_c$, it is impossible to cancel $I(Y;X|p,\boldsymbol{\theta})$.

The same rationale holds on the interval $[1-2/c, 1-1/c]$, where all the scalar products have negative values except $\rho_{c-1}(p)$, hence a lower bound is:

$$\sum_{i=1}^{c-2} \rho_i(p) + \rho_c(p) \leq \boldsymbol{\theta}^T(\mathbf{q}_{\Sigma 1} - \mathbf{q}_{\Sigma 0})$$

We can simplify the lower bound into: $p^{c-2}(1-c(1-p)) + (1-p)^{c-1}$, which is the symmetric version of the first bound $\rho_1(p) + \rho_c(p)$. For $p > 1 - \eta_c$, the mutual information cannot be canceled. □

Although it is not possible to obtain analytically the exact value of η_c, it can be approximated by $\eta_c \approx 1/c$. This approximation is asymptotically tight as c is increased.

A corollary of this proposition is that, for $c \geq 3$, the achievable rate for the pmf $f_2^\star(p) = \delta(p - 1/2)$ might be null. It is indeed the case for $c = 3$ and a minority vote. The following proposition shows this propagates as c increases.

Proposition 4. *If c colluders can cancel the achievable rate with the attack $\boldsymbol{\theta}_c$, then $c+1$ colluders can achieve the same goal with the following attack $\boldsymbol{\theta}_{c+1}$:*

$$\theta_{c+1,\sigma} = \frac{\sigma}{c+1}\theta_{c,\sigma-1} + \frac{c+1-\sigma}{c+1}\theta_{c,\sigma},\ \forall \sigma \in [c]. \tag{19}$$

Proof. We only give the sketch of the proof. The attack of the c colluders cancel the rate, thus it cancels the mutual informations $I(Y;X|\pi,\boldsymbol{\theta}^{(c)}), \forall \pi \in \Pi(c)$. Since, in general,

$$\begin{aligned}\Pr_Y[1|X=1,p,\boldsymbol{\theta}] &= \Pr_Y[1|p,\boldsymbol{\theta}] + \frac{(1-p)}{c}\frac{\partial}{\partial p}\Pr_Y[1|p,\boldsymbol{\theta}] \\ \Pr_Y[1|X=0,p] &= \Pr_Y[1|p,\boldsymbol{\theta}] - \frac{p}{c}\frac{\partial}{\partial p}\Pr_Y[1|p,\boldsymbol{\theta}],\end{aligned} \tag{20}$$

therefore this attack sets $\frac{\partial}{\partial p}\Pr_Y[1|\pi,\boldsymbol{\theta}_c] = 0$, $\forall \pi \in \Pi(c)$. Some trivial math shows that (19) leads to:

$$\frac{\partial}{\partial p}\Pr_Y[1|\pi,\boldsymbol{\theta}_{c+1}] = \frac{\partial}{\partial p}\Pr_Y[1|\pi,\boldsymbol{\theta}_c] = 0$$

□

The interpretation of (19) is quite easy: The $c+1$ colluders uniformly pick up and exclude one of their symbols and they lead the attack $\boldsymbol{\theta}_c$ with the remaining c symbols. If they had σ '1' over $c+1$ symbols, the probability that there remain $\sigma - 1$ '1' (resp. σ) over c equals $\sigma/(c+1)$ (resp. $1 - \sigma/(c+1)$). This proposition explains the series of null rates in Tab. 2.

Table 2. Achievable rates for the single decoder (in %).

c	2	3	4	5	6	7
$R(c)$	0.189	0.059	0.034	0.016	0.011	0.008
$f_2^\star$	100	0	0	0	0	0
$f_3^\star$	88.5	50.1	30.0	9.5	0.0	0
$f_4^\star$	84.6	77.2	67.2	24.3	1.0	0
$f_5^\star$	82.1	95.3	94.1	54.2	16.0	6.0
$f_6^\star$	79.4	100	100	84.5	48.1	29.5
$f_7^\star$	78.3	98.3	98.8	100	77.5	55.6
f_F	78.2	76.1	76.5	86.7	86.3	84.6
f_T	78.2	80.7	81.5	97.2	100	100

$\Pi(\hat{c}) \subset [\eta_c, 1-\eta_c]$ is a necessary condition for canceling the rate, but it is not sufficient. This goal is indeed achieved if the intersection between the definition set $\Theta(c)$ and the hyperplane defined by (15) taken in all $p \in \Pi(\hat{c})$ is not the emptyset. For instance, $\Pi(3)$ and $\Pi(4)$ have only 2 elements and $\Theta(3)$ has only two degrees of freedom. Therefore the hyperplane for $c = 3$ is just a point (a full rank system of 2 equations and 2 unknowns). It appears that this point is not in $\Theta(3)$. The hyperplane needs a bigger dimension, ie. more colluders, to finally intersects with $\Theta(3)$ and to cancel the rate.

4.3. Achievable rates

To be consistent with Sec. 3, we denote $\bar{R}(c)$ the maximum achievable rate against c colluders over the 8 tested pdf of Table 1. Table 2 shows the achievable rates as a percentage of $\bar{R}(c)$. Remember that $f_{\hat{c}}^\star$ is the optimal pdf for the *joint* decoder. As far as we know, the optimal pdf for the single decoder hasn't been discovered except for 2 colluders. Contrary to the joint decoder, a mismatch between the expected number of colluder and its actual value is a big issue. The single decoder sees his rate vanishing very fast as c increases if f is a pmf. Note, however, that a big number of foreseen colluders $\hat{c}$ implies a pmf $f_{\hat{c}}^\star$ with a large support. Amiri and Tardos give the lower bound: $|\Pi(\hat{c})| \geq \sqrt{\frac{\hat{c}}{4 \ln 2 \log \hat{c}}}$. Therefore, even more colluders are needed to cancel the single decoder rate. From $c = 6$ colluders, the best pdf we tested is the choice originally made by Tardos.

5. CONCLUSION

As a final remark, we would like to stress that a joint decoder has a higher but also more stable rate than the single decoder. Therefore, the importance of the capacity achieving function $f_{\hat{c}}^\star$ is strictly conditioned on the existence of a joint decoder with affordable complexity. A more realistic goal would be to trade high and stable rates against less computing power. If this is not possible, then continuous pdf such as the flat or the Tardos pdf seems to be more secure.

6. REFERENCES

[1] E. Amiri and G. Tardos, "High rate fingerprinting codes and the fingerprinting capacity," in *Proc. of 20th Annual ACM-SIAM Symposium on Discrete Algorithms (SODA 2009)*, jan 2009.

[2] Y.-W. Huang and P. Moulin, "Saddle-point solution of the fingerprinting capacity game under the marking assumption," in *accepted to ISIT 2009*, jun 2009.

[3] D. Boneh and J. Shaw, "Collusion-secure fingerprinting for digital data," *IEEE Trans. Inform. Theory*, vol. 44, pp. 1897–1905, September 1998.

[4] P. Moulin, "Universal fingerprinting: capacity and random-coding exponents," *IEEE Transactions on Information Theory*, January 2008, submitted. Preprint available at `http://arxiv.org/abs/0801.3837`.

[5] J. C. Sevy, "Lagrange and least-squares polynomials as limits of linear combinations of iterates of Bernstein and Durrmeyer polynomials," *Journal of Approximation Theory*, vol. 80, pp. 267–271, 1995.

[6] R. E. Blahut, "Computation of channel capacity and rate-distortion functions," *IEEE Transactions on Information Theory*, vol. 18, no. 4, pp. 460–473, July 1972.

[7] T. M. Cover and J. A. Thomas, *Elements of Information Theory*. Wiley series in Telecommunications, 1991.

[8] T. Furon and L. Pérez-Freire, "Worst case attack against binary probabilistic traitor tracing codes," *submitted to IEEE Trans. on Inf. Forensics and Security*, p. preprint available at http://arxiv.org/abs/0903.3480, 2009.

[9] K. Nuida, "An improvement of short 2-secure fingerprint codes strongly avoiding false-positive," in *to appear in proc. of 11th Information Hiding workshop*, 2009.