Tally-based simple decoders for traitor tracing and group testing

Boris Škorić

***Abstract*—The topic of this paper is collusion resistant watermarking, a.k.a. traitor tracing, in particular bias-based traitor tracing codes as introduced by G. Tardos in 2003. The past years have seen an ongoing effort to construct efficient high-performance decoders for these codes.**
In this paper we construct a score system from the Neyman-Pearson hypothesis test (which is known to be the most powerful test possible) into which we feed all the evidence available to the tracer, in particular the codewords of *all* users. As far as we know, until now simple decoders using Neyman-Pearson have taken into consideration only the codeword of a single user, namely the user under scrutiny.
The Neyman-Pearson score needs as input the attack strategy of the colluders, which typically is not known to the tracer. We insert the Interleaving attack, which plays a very special role in the theory of bias-based traitor tracing by virtue of being part of the asymptotic (i.e. large coalition size) saddlepoint solution. The score system obtained in this way is universal: effective not only against the Interleaving attack, but against all other attack strategies as well. Our score function for one user depends on the other users' codewords in a very simple way: through the symbol tallies, which are easily computed.
We present bounds on the False Positive probability and show ROC curves obtained from simulations. We investigate the probability distribution of the score. Finally we apply our construction to the area of (medical) Group Testing, which is related to traitor tracing.

***Index Terms*—traitor tracing, Tardos code, collusion, watermarking, group testing.**

I. Introduction

A. Collusion attacks on watermarking

Forensic watermarking is a means for tracing the origin and distribution of digital content. Before distribution, the content is modified by embedding an imperceptible watermark, which plays the role of a personalized identifier. Once an unauthorized copy of the content is found, the watermark present in this copy can be used to reveal the identities of those users who participated in its creation. A tracing algorithm or 'decoder' outputs a list of suspicious users. This procedure is also known as 'traitor tracing'.

The most powerful attacks against watermarking are *collusion attacks*: multiple attackers (the 'coalition') combine their differently watermarked versions of the same content; the observed differences point to the locations of the hidden marks and allow for a targeted attack.

Several types of collusion-resistant codes have been developed. The most popular type is the class of *bias-based* codes, introduced by G. Tardos in 2003. The original paper [36], [37] was followed by a lot of activity, e.g. improved analyses [4], [13], [14], [22], [32], [41], [40], code modifications [16], [28], [29], decoder modifications [2], [8], [24], [30], [12] and various generalizations [7], [38], [39], [42]. The advantage of bias-based versus deterministic codes is that they can achieve the asymptotically optimal relationship $\ell \propto c^2$ between the sufficient code length ℓ and the coalition size c.

Two types of tracing algorithm can be distinguished: *simple decoders*, which assign a level of suspicion to single users, and *joint decoders* [2], [8], [24], which look at sets of users. One of the main advances in recent years was finding [16], [18] the *saddlepoint* of the information-theoretic max-min game (see Section II-E) in the case of joint decoding. Knowing the location of the saddlepoint makes it easier for the tracer to build a decoder that works optimally against the worst-case attack and that works well against all other attacks too.

B. Contributions and outline

We consider the non-asymptotic regime, i.e. coalitions of arbitrary finite size. We do something that has somehow been overlooked: we determine the Neyman-Pearson score [27] aimed against the collusion attack in the asympotic saddle-point (i.e. the Interleaving attack), but contrary to previous approaches (such as [21] for a binary alphabet), we take *all* available information as evidence in the Neyman-Pearson hypothesis test. More precisely, in order to determine if a user j is suspicious, a hypothesis test is done taking as evidence not only *his* codeword, but all the other codewords as well. The result is a simple decoder which, when the Interleaving attack is inserted, miraculously simplifies to an easy-to-compute score for a single user; the score depends on all the other users' codewords merely through symbol tallies.

- In Section II we give some background on traitor tracing.
- In Section III we derive our new tally-based score function. We first present a general result valid in the Combined Digit Model and then narrow it down to the Restricted Digit Model. For alphabet size 2, in the limit of many users our score reduces to the log-likelihood score of Laarhoven [21]; for larger alphabets the limit of many users yields the non-binary generalisation of the Laarhoven score. In the large c limit our score further reduces to the asymptotic-capacity-achieving score of Oosterwijk et al. [30].
- In Section IV we compute the probability that there exist one or more *infinite colluder scores*. Such an occurence allows for errorless decoding. Then we upper-bound the False Positive error rate using an approach similar to the 'operational mode' that was recently proposed by Furon and Desoubeaux [12]. We show that there is quite a large

gap between this bound and error rates in simulations. We provide ROC curves from simulations; they show that the performance of out new score is mostly the same as the Laarhoven score, except in special cases such as the Minority Voting attack.
- In Section V we derive the single-position probability distribution for the scores of Oosterwijk et al. and Laarhoven as well as our new score. The Oosterwijk et al. score has a power-law tail with a finite 2nd moment but in general with an infinite 3rd moment. The generalized Laarhoven score has an exponential tail. Our new score is discrete; it has a probability mass function rather than a density function.
- In Section VI we briefly comment on hypothesis tests for Group Testing. There is a link between Group Testing on the one hand and on the other hand binary traitor tracing where the colluders employ the All-1 attack. We evaluate the Neyman-Pearson hypothesis test in the case of the All-1 attack and obtain a new, tally-dependent, score function for Group Testing.

II. Preliminaries

A. General notation and terminology

Random variables are written as capitals, and their realisations in lower-case. Sets are written in calligraphic font. (E.g. random variable X with realisations $x \in \mathcal{X}$.) The probability of an event A is denoted as $\Pr[A]$, and the expectation over a random variable X is denoted as $\mathbb{E}_X[f(X)] \stackrel{\text{def}}{=} \sum_{x\in\mathcal{X}} \Pr[X=x] f(x)$.

The notation $[n]$ stands for $\{1,\ldots,n\}$. The Kronecker delta is written as δ_{xy}, the Dirac delta function as $\delta(\cdot)$. The step function is denoted as $\Theta(\cdot)$. Vectors are written in boldface. The 1-norm of a vector $\boldsymbol{v}$ is denoted as $|\boldsymbol{v}| = \sum_\alpha v_\alpha$.

The number of users is n. The length of the code is ℓ. The alphabet is $\mathcal{Q}$, with size $|\mathcal{Q}| = q$. The number of colluders is c. The set of colluders is denoted as $\mathcal{C} \subset [n]$ with $|\mathcal{C}| = c$. The coalition size that the code is built to withstand is c_0. We will use the term 'asymptotically' meaning 'in the limit of large c_0'.

B. Code generation

The bias vector in position i is denoted as $\boldsymbol{p}_i = (p_{i\alpha})_{\alpha\in\mathcal{Q}}$, and it satisfies $|\boldsymbol{p}_i| \stackrel{\text{def}}{=} \sum_{\alpha\in\mathcal{Q}} p_{i\alpha} = 1$. The bias vectors $\boldsymbol{p}_i$ are drawn independently from a probability density F. The asymptotically optimal F is given by the following Dirichlet distribution (multivariate Beta distribution): $F(\boldsymbol{p}) = \Gamma(\frac{q}{2})[\Gamma(\frac{1}{2})]^{-q} \prod_{\alpha\in\mathcal{Q}} p_\alpha^{-1/2}$. We use the 'bar' notation to indicate a quantity in all positions, e.g. $\bar{\boldsymbol{p}} \stackrel{\text{def}}{=} (\boldsymbol{p}_i)_{i\in[\ell]}$.

The code matrix is a matrix $x \in \mathcal{Q}^{n\times\ell}$; the matrix rows are the codewords. The j'th row is denoted as $\bar{x}_j \stackrel{\text{def}}{=} (x_{ji})_{i\in[\ell]}$. The entries of x are generated column-wise from the bias vectors: in position i, the probability distribution for user j's symbol is given by $\Pr[X_{ji} = \alpha | \boldsymbol{P}_i = \boldsymbol{p}_i] = p_{i\alpha}$.

C. Collusion attack

For $i \in [\ell]$, $\alpha \in \mathcal{Q}$ we introduce tally variables as follows,

$$\begin{aligned} t_{i\alpha} &\stackrel{\text{def}}{=} |\{j \in [n] : x_{ji} = \alpha\}| \\ m_{i\alpha} &\stackrel{\text{def}}{=} |\{j \in \mathcal{C} : x_{ji} = \alpha\}|. \end{aligned} \tag{1}$$

In words: $t_{i\alpha}$ is the number of users who have symbol α in the i'th position of their codeword; $m_{i\alpha}$ is the number of *colluders* who have symbol α in the i'th position of their codeword. We write $\boldsymbol{t}_i = (t_{i\alpha})_{\alpha\in\mathcal{Q}}$ and $\boldsymbol{m}_i = (m_{i\alpha})_{\alpha\in\mathcal{Q}}$. They satisfy $|\boldsymbol{t}_i| = n$ and $|\boldsymbol{m}_i| = c$. In the remainder of this paper, the position index i will sometimes be omitted when it is clear that a single position is studied.

In the Combined Digit Model (CDM) [39], the attackers have to decide which symbol, or combination of averaged ('fused') symbols, to choose in each content position $i \in [\ell]$. This set of symbols is denoted as $\psi_i \subseteq \mathcal{Q}$, with $\psi_i \neq \emptyset$. According to the Marking Assumption, ψ_i may only contain symbols for which the colluder tally is nonzero. In addition, the colluders may add noise. The effect of the attack on the content is nondeterministic, and causes the tracer to detect of a set of symbols $\varphi_i \subseteq \mathcal{Q}$ that does not necessarily match ψ_i. This is modelled as a set of transition probabilities $P_{\varphi|\psi}$ which depend on $|\psi|$, the amount of noise etc. For more details on the CDM we refer to [39].

In the Restricted Digit Model (RDM) the colluders are allowed to select only a single symbol (usually denoted as $y \in \mathcal{Q}$) with nonzero tally, which then gets detected with 100% fidelity by the tracer.

As is customary in the literature on traitor tracing, we will assume that the attackers equally share the risk. This leads to "colluder symmetry", i.e. the attack is invariant under permutation of the colluder identities. Furthermore we assume that there is no natural ordering on the alphabet $\mathcal{Q}$, i.e. everything is invariant under permutation of the alphabet. Given these two symmetries, the attack depends only on $\bar{\boldsymbol{m}}$, the set of colluder tallies. Any attack strategy can then be fully characterized by a set of probabilities $\theta_{\bar{\psi}|\bar{\boldsymbol{m}}}$. In the case of the RDM this reduces to $\theta_{\bar{\boldsymbol{y}}|\bar{\boldsymbol{m}}}$.

The process of generating the matrix x as well as tracing the colluders is fully position-symmetric, i.e. invariant under permutations of the columns of x (the content positions). However, that does not guarantee that the optimal collusion strategy is position-symmetric as well, since the realisation of x itself breaks the symmetry. Asymptotically the symmetry is restored (due to $\ell \to \infty$); the attack strategy can then be parametrized more compactly as a set of probabilities $\theta_{\psi|\boldsymbol{m}}$ applied in each position independently. In the RDM the asymptotically optimal attack [17], [18] is the Interleaving attack: a colluder is selected uniformly at random and his symbol is output.

D. Decoders

The process of tracing colluders based on $\bar{\boldsymbol{p}}$, x and $\bar{y}$ is referred to as 'decoding'. The decoder outputs a list $\mathcal{L} \subset [n]$ of suspicious users. The literature distinguishes between two types of decoder: *simple* and *joint*. A simple decoder computes

a score for each user $j \in [n]$. A joint decoder, on the other hand, investigates tuples of users. The runtime of a simple decoder is linear in n, whereas a joint decoder typically takes much more time (polynomial in n) because it e.g. has to check all possible user tuples up to a certain size.

Examples of simple decoders are the original Tardos score function [36], [37], its symmetrized generalization [38], the empirical mutual information score [34], [26], and the score function [30] targeted against the Interleaving attack. Examples of joint decoders are the Expectation Maximization algorithm [8], the decoder of Amiri and Tardos [2] and the Don Quixote algorithm [24].

Often a threshold is used in the accusation procedure: if the score of a user/tuple exceeds the threshold, he/they are accused. In this scenario a decoder can make two kinds of mistake: (i) Accusation of one or more innocent users, known as False Positive (FP); (ii) Not finding any of the colluders, known as False Negative (FN).

The error probabilities of the decoder are $P_{\rm FP} = \Pr[\mathcal{L}\backslash\mathcal{C} \neq \emptyset]$ and $P_{\rm FN} = \Pr[\mathcal{L}\cap\mathcal{C} = \emptyset]$. In the literature on Tardos codes one is often interested in the one-user false accusation probability $P_{\rm FP1} \stackrel{\rm def}{=} \Pr[j \in \mathcal{L} | j \in [n] \setminus \mathcal{C}]$ for some fixed innocent user j; this is for proof-technical reasons. For bias-based codes it holds [40] that $P_{\rm FP} \approx (n-c)P_{\rm FP1}$ if $P_{\rm FP} \ll 1$.

E. Joint decoder saddlepoint

The fingerprinting rate is defined as $R \stackrel{\rm def}{=} (\log_q n)/\ell$. This is the number of q-ary symbols needed to specify a single user in $[n]$ (the message part of the codeword), divided by the actual number of symbols used by the code in order to convey this message.

The maximum achievable fingerprinting rate at which the error probabilities can be kept under control is called the *fingerprinting capacity*. We consider the most general case, the joint decoder, in which case the capacity is denoted as $C_{\rm joint}$. Shannon's channel coding theorem (see e.g. [10]) gives a bound on the decoding error probability $P_{\rm err}$ of an error-correcting code (for $\ell \to \infty$), $P_{\rm err} \leq q^{-\ell(C-R)}$. From this it follows that, in the limit of large n, the sufficient code length $\ell_{\rm suff}$ for resisting c_0 colluders at some given error probability is given by

$$\ell_{\rm suff} = \frac{\ln(n/P_{\rm FP})}{C_{\rm joint}(q,c_0)\ln q}. \tag{2}$$

Here the FP error appears because it is usually dominant (more critical than FN) in audio-video watermarking. Computing the capacity as a function of q and c_0 is a nontrivial exercise. It is necessary to find the saddlepoint of a max-min game with payoff function $\frac{1}{c}I(\Phi;\boldsymbol{M}|\boldsymbol{P})$, where $I(\cdot;\cdot)$ stands for mutual information. In the max-min game, the tracer controls the bias distribution F and tries to maximize the mutual information. The colluders know F. They control the attack strategy and try to minimize the mutual information. There is a saddlepoint, a special combination of F and strategy such that it is bad for both parties to stray from that point. The value of the payoff function in the saddlepoint is the capacity. The asymptotic (large c) capacity in the RDM was found [2], [5], [17] to be $C_{\rm joint}^{\rm RDM,asym} = (q-1)/(2c^2\ln q)$, leading to a sufficient code length $\ell_{\rm suff}^{\rm RDM,asym} = \frac{2}{q-1}c_0^2\ln(n/P_{\rm FP})$. In the asymptotic saddlepoint [18] the bias distribution is the Dirichlet distribution as specified in Section II-B, and the attack strategy is the Interleaving attack applied independently in each content position. For non-asymptotic c_0 only numerical results are available (except at $c_0 = 2$). There are also numerical results for the asymptotics in the case of attack models like the CDM [6]. It turns out [17] that the optimal attack quickly converges to Interleaving with increasing c.

F. Universal score function

Based on the work of Abbe and Zheng [1], Meerwald and Furon [25] pointed out that a *universal* decoder for traitor tracing is obtained by evaluating a Neyman-Pearson score [27] in the saddlepoint of the mutual-information-game. The term 'universal' means that the decoder is effective not only against the saddlepoint value of the attack, but also all other attacks. This is a very important point. Usually one can check the effectiveness of a decoder only against a small set of attacks and then hope that the decoder will work against other attacks as well; a universal decoder is *guaranteed* to work against all attacks.

The general formula for the Neyman-Pearson score yields a result that depends on the attack strategy, which is not known to the tracer. Hence the existence of a universal decoder is very important.

Laarhoven [21] showed for the binary case that the asymptotic-capacity-achieving score function of Oosterwijk et al. [30] is asymptotically equivalent to such a Neyman-Pearson score evaluated for the Interleaving attack.

G. The multivariate hypergeometric distribution

Consider a single column of the matrix x. Let $\boldsymbol{T}$ be the total tally vector and $\boldsymbol{M}$ the colluders' tally vector, as defined in (1). If a coalition of c users is selected uniformly at random out of the n users, the probability $L_{\boldsymbol{m}|\boldsymbol{t}}$ that colluder tally $\boldsymbol{m}$ occurs, for given $\boldsymbol{t}$, is

$$L_{\boldsymbol{m}|\boldsymbol{t}} \stackrel{\rm def}{=} \Pr[\boldsymbol{M}=\boldsymbol{m}|\boldsymbol{T}=\boldsymbol{t}] = \frac{1}{\binom{n}{c}}\prod_{\alpha\in\mathcal{Q}}\binom{t_\alpha}{m_\alpha}. \tag{3}$$

(For each symbol α, a number m_α of users have to be selected out of the t_α users who have that symbol). Eq. (3) is known as the multivariate hypergeometric distribution. Its first and second moment are

$$\mathbb{E}_{\boldsymbol{M}|\boldsymbol{t}}[\boldsymbol{M}] = \frac{c}{n}\boldsymbol{t} \tag{4}$$

$$\mathbb{E}_{\boldsymbol{M}|\boldsymbol{t}}[M_\alpha M_\beta] - \frac{c^2}{n^2}t_\alpha t_\beta = c\frac{n-c}{n-1}[\delta_{\alpha\beta}\frac{t_\alpha}{n} - \frac{t_\alpha t_\beta}{n^2}]. \tag{5}$$

H. Useful Lemmas

Lemma 1 (Bernstein's inequality [3]): Let $V_1, \cdots, V_\ell$ be independent zero-mean random variables, with $|V_i| \leq a$ for all i. Let $\zeta \geq 0$. Then

$$\Pr\left[\sum_{i=1}^{\ell} V_i > \zeta\right] \leq \exp\left(-\frac{\zeta^2/2}{\sum_i \mathbb{E}[V_i^2] + a\zeta/3}\right).$$

Lemma 2 (Beta function): The Beta function is defined as $B(u,v) = \Gamma(u)\Gamma(v)/\Gamma(u+v)$ and for $u, v > 0$ it has the integral representation

$$B(u,v) = \int_0^1 p^{-1+u}(1-p)^{-1+v}\,\mathrm{d}p. \tag{6}$$

III. Tally-based universal score function

Motivated by the fact that with increasing c the saddlepoint value of the attack strategy quickly converges to Interleaving, we construct a Neyman-Pearson score against Interleaving. However, instead of taking as the evidence the detected symbols $\bar{\varphi}$, the biases $\bar{\boldsymbol{p}}$ and a single user's codeword, as was done before, we include the *whole matrix* x. This is an obvious step, but as far as we know it has not been done before in a simple decoder.

Theorem 1: Let the biases $\bar{\boldsymbol{p}}$, the matrix x and the detected symbols (or symbol fusions) $\bar{\varphi}$ be known to the tracer. Let the attack be position-symmetric, parametrized by the probabilities $\theta_{\psi|\boldsymbol{m}}$. Consider a tracer who has no a priori suspicions about the users. His a priori knowledge about the coalition is that it is a uniformly random tuple of c users from $[n]$. For him the most powerful hypothesis test to decide if a certain user $j \in [n]$ is a colluder or not is to use the score

$$\sum_{i=1}^{\ell} \ln \frac{\sum_{\boldsymbol{m}} L_{\boldsymbol{m}|\boldsymbol{t}_i} P_{\varphi_i|\boldsymbol{m}} m_{x_{ji}}}{\sum_{\boldsymbol{m}} L_{\boldsymbol{m}|\boldsymbol{t}_i} P_{\varphi_i|\boldsymbol{m}} (t_{ix_{ji}} - m_{x_{ji}})} \tag{7}$$

where we have used the notations $P_{\varphi|\boldsymbol{m}}$, $L_{\boldsymbol{m}|\boldsymbol{t}}$, m and t as defined in the Preliminaries section.

Proof: The most powerful test to decide between two hypotheses is to see if the Neyman-Pearson score exceeds a certain threshold. We consider the hypothesis $H_j = (j \in \mathcal{C})$. The Neyman-Pearson score in favour of this hypothesis is the ratio $\Pr[H_j|\text{evidence}]/\Pr[\neg H_j|\text{evidence}]$, which can be rewritten as $\frac{\Pr[H_j]}{\Pr[\neg H_j]} \cdot \frac{\Pr[\text{evidence}|H_j]}{\Pr[\text{evidence}|\neg H_j]}$. We have $\Pr[H_j] = \frac{c}{n}$ and $\Pr[\neg H_j] = 1 - \frac{c}{n}$ since the a priori distribution of colluders over the users is uniform. We discard[1] the constant factor $\frac{\Pr[H_j]}{\Pr[\neg H_j]}$ and study the expression $\frac{\Pr[\text{evidence}|H_j]}{\Pr[\text{evidence}|\neg H_j]}$. The evidence is given by $\bar{\boldsymbol{p}}, x, \bar{\varphi}$. Using symbol symmetry and colluder symmetry we have

$$\begin{aligned} R_j &\stackrel{\text{def}}{=} \frac{\Pr[\bar{\boldsymbol{p}}, x, \bar{\varphi}|H_j]}{\Pr[\bar{\boldsymbol{p}}, x, \bar{\varphi}|\neg H_j]} \\ &= \frac{\Pr[\bar{\boldsymbol{p}}]\Pr[x|\bar{\boldsymbol{p}}]\sum_{\bar{\boldsymbol{m}}}\Pr[\bar{\boldsymbol{m}}|x,H_j]\Pr[\bar{\varphi}|\bar{\boldsymbol{m}}]}{\Pr[\bar{\boldsymbol{p}}]\Pr[x|\bar{\boldsymbol{p}}]\sum_{\bar{\boldsymbol{m}}}\Pr[\bar{\boldsymbol{m}}|x,\neg H_j]\Pr[\bar{\varphi}|\bar{\boldsymbol{m}}]} \\ &= \frac{\sum_{\bar{\boldsymbol{m}}}\Pr[\bar{\boldsymbol{m}}|x,H_j]\Pr[\bar{\varphi}|\bar{\boldsymbol{m}}]}{\sum_{\bar{\boldsymbol{m}}}\Pr[\bar{\boldsymbol{m}}|x,\neg H_j]\Pr[\bar{\varphi}|\bar{\boldsymbol{m}}]}. \end{aligned} \tag{8}$$

In applying the chain rule for probabilities (2nd line) we have used $\Pr[\bar{\boldsymbol{m}}|x\bar{\boldsymbol{p}}] = \Pr[\bar{\boldsymbol{m}}|x]$, which is due to the fact that $\bar{\boldsymbol{m}}$ is created directly from x. We have also used $\Pr[\bar{\varphi}|\bar{\boldsymbol{p}}x\bar{\boldsymbol{m}}H_j] = \Pr[\bar{\varphi}|\bar{\boldsymbol{m}}]$ (and similarly for $\neg H_j$) since $\bar{\varphi}$ is created directly from $\bar{\psi}$ which is a function of $\bar{\boldsymbol{m}}$ only.

Note that the randomness of the coalition causes $\bar{\boldsymbol{M}}|x$ to be a random variable. Due to the position symmetry of the attack, R_j reduces to a factorized expression,

$$\begin{aligned} R_j &= \prod_{i=1}^{\ell} \frac{\sum_{\boldsymbol{m}_i}\Pr[\boldsymbol{m}_i|x,H_j]P_{\varphi_i|\boldsymbol{m}_i}}{\sum_{\boldsymbol{m}_i}\Pr[\boldsymbol{m}_i|x,\neg H_j]P_{\varphi_i|\boldsymbol{m}_i}} \\ &= \prod_{i=1}^{\ell} \frac{\sum_{\boldsymbol{m}_i}\Pr[\boldsymbol{m}_i|\boldsymbol{t}_i,H_j]P_{\varphi_i|\boldsymbol{m}_i}}{\sum_{\boldsymbol{m}_i}\Pr[\boldsymbol{m}_i|\boldsymbol{t}_i,\neg H_j]P_{\varphi_i|\boldsymbol{m}_i}}. \end{aligned} \tag{9}$$

Next we write

$$\begin{aligned} \Pr[\boldsymbol{m}_i|\boldsymbol{t}_i,H_j] &= \frac{1}{\binom{n-1}{c-1}}\prod_{\alpha\in\mathcal{Q}}\binom{t_{i\alpha}-\delta_{\alpha,x_{ji}}}{m_{i\alpha}-\delta_{\alpha,x_{ji}}} \\ &= \frac{n}{c}\cdot\frac{m_{ix_{ji}}}{t_{ix_{ji}}}L_{\boldsymbol{m}_i|\boldsymbol{t}_i} \end{aligned} \tag{10}$$

$$\begin{aligned} \Pr[\boldsymbol{m}_i|\boldsymbol{t}_i,\neg H_j] &= \frac{1}{\binom{n-1}{c}}\prod_{\alpha\in\mathcal{Q}}\binom{t_{i\alpha}-\delta_{\alpha,x_{ji}}}{m_{i\alpha}} \\ &= \frac{n}{n-c}\frac{t_{ix_{ji}}-m_{ix_{ji}}}{t_{ix_{ji}}}L_{\boldsymbol{m}_i|\boldsymbol{t}_i}. \end{aligned} \tag{11}$$

Substitution of (10),(11) into (9) yields

$$R_j = \prod_{i=1}^{\ell}\frac{n-c}{c}\cdot\frac{\sum_{\boldsymbol{m}_i} m_{ix_{ji}}L_{\boldsymbol{m}_i|\boldsymbol{t}_i}P_{\varphi_i|\boldsymbol{m}_i}}{\sum_{\boldsymbol{m}_i}(t_{ix_{ji}}-m_{ix_{ji}})L_{\boldsymbol{m}_i|\boldsymbol{t}_i}P_{\varphi_i|\boldsymbol{m}_i}}. \tag{12}$$

We discard the constant factor $(\frac{n-c}{n})^{\ell}$. We drop the index i on the summation variable $\boldsymbol{m}_i$. Finally we take the logarithm; this is allowed since applying a monotonic function to a Neyman-Pearson score leads to an equivalent score system. ∎

We note a number of interesting properties of the score (7):

- The $\bar{\boldsymbol{p}}$ has disappeared from the score. This is not surprising because x contains more evidence than $\bar{\boldsymbol{p}}$. (The x is generated from $\bar{\boldsymbol{p}}$ and after that all further events depend directly on x.)
- The score for user j depends on the tallies $\bar{\boldsymbol{t}}$, i.e. on the codewords of all the other users. This is unusual. In other simple decoders only the codeword of user j is considered.

In the case of the RDM, the $\bar{\varphi}$ reduces to $\bar{y}$, and $P_{\varphi_i|\boldsymbol{m}_i}$ reduces to $\theta_{y_i|\boldsymbol{m}_i}$. Note that (7) depends on the attack strategy, which is in general not known to the tracer. The colluder tallies $\boldsymbol{m}$ are also not known to the tracer, but these are averaged over, hence (7) does *not* depend on the unknown colluder tallies.

Theorem 2: In the case of the Restricted Digit Model and the Interleaving attack, the score function of Theorem 1 reduces to

$$\sum_{i=1}^{\ell}\left(\ln\frac{c}{n-c} + \ln\left[1+\frac{1}{c}\left\{\delta_{x_{ji}y_i}\frac{1-1/n}{t_{iy_i}/n-1/n}-1\right\}\right]\right) \tag{13}$$

which is equivalent to

$$\sum_{i=1}^{\ell}\ln\left[1+\frac{1}{c}\left\{\delta_{x_{ji}y_i}\frac{1-1/n}{t_{iy_i}/n-1/n}-1\right\}\right]. \tag{14}$$

Proof: We omit the indices i and j for notational brevity. In the case of the RDM and Interleaving, the $P_{\varphi|\boldsymbol{m}}$ in (7) reduces to $\theta_{y|\boldsymbol{m}} = m_y/c$. With the use of (4),(5) we obtain

$$\sum_{\boldsymbol{m}} L_{\boldsymbol{m}|\boldsymbol{t}} m_y m_x = c^2\frac{t_x t_y}{n^2} + c\frac{n-c}{n-1}[\delta_{xy}\frac{t_y}{n} - \frac{t_x t_y}{n^2}] \tag{15}$$

[1] This is allowed. Score systems that differ in a constant factor are equivalent.

and

$$\sum_{\boldsymbol{m}} L_{\boldsymbol{m}|\boldsymbol{t}} m_y (t_x - m_x) = t_x \sum_{\boldsymbol{m}} L_{\boldsymbol{m}|\boldsymbol{t}} m_y - \sum_{\boldsymbol{m}} L_{\boldsymbol{m}|\boldsymbol{t}} m_y m_x$$
$$= (\frac{c}{n} - \frac{c^2}{n^2}) t_x t_y - c \frac{n-c}{n-1} [\delta_{xy} \frac{t_y}{n} - \frac{t_x t_y}{n^2}]. \quad (16)$$

We have two cases, $\delta_{xy} = 0$ and $\delta_{xy} = 1$, which after some algebra can be simplified to

$$x \neq y: \quad \frac{(15)}{(16)} = \frac{c-1}{n-c}$$
$$x = y: \quad \frac{(15)}{(16)} = \frac{c-1}{n-c} + \frac{1}{n-c} \cdot \frac{1 - 1/n}{t_y/n - 1/n}. \quad (17)$$

Together this can again be written compactly as

$$\frac{(15)}{(16)} = \frac{c-1}{n-c} [1 + \frac{\delta_{xy}}{c-1} \cdot \frac{1-1/n}{t_y/n - 1/n}]$$
$$= \frac{c}{n-c} [1 + \frac{1}{c} \{\delta_{xy} \frac{1-1/n}{t_y/n - 1/n} - 1\}]. \quad (18)$$

The result (13) follows by substituting (18) into (7) and finally taking the logarithm. ∎

We mention a number of interesting points about the score function (14):

- If for any $i \in [\ell]$ it occurs that $\delta_{x_{ji} y_i} = 1$ and $t_{iy_i} = 1$, then user j's score is *infinite*. This makes perfect sense: he is the only user who received symbol y_i in position i, which makes it possible to accuse him with 100% certainty.
- For large n the expression (14) approaches $\sum_i \ln(1 + c^{-1}[\delta_{x_{ji} y_i} \frac{1}{\hat{p}_{y_i}} - 1]) \approx \sum_i \ln(1 + c^{-1}[\delta_{x_{ji} y_i} \frac{1}{p_{y_i}} - 1])$. The latter form was already obtained by Laarhoven [21] in the case of binary alphabets.
- If c is large as well, then the score may be approximated by its first order Taylor expansion, yielding $c^{-1} \sum_i [\delta_{x_{ji} y_i} \frac{1}{p_{y_i}} - 1]$. This is (up to the unimportant constant c^{-1}) precisely the asymptotic-capacity-achieving simple decoder of Oosterwijk et al. [30].
- For given $\boldsymbol{p}_i$, the tally $\boldsymbol{t}_i$ is multinomial-distributed with parameters n and $\boldsymbol{p}_i$. The first moment and variance are given by $\mathbb{E}_{\boldsymbol{T}_i|\boldsymbol{P}_i=\boldsymbol{p}_i}[\boldsymbol{T}_i] = n\boldsymbol{p}_i$ and $\mathbb{E}_{\boldsymbol{T}_i|\boldsymbol{P}_i=\boldsymbol{p}_i}[T_{i\alpha}^2] - (np_{i\alpha}^2) = np_{i\alpha}(1 - p_{i\alpha})$. Thus the expression t_{iy_i}/n that appears in the score function is an estimator for p_{iy_i} that becomes more accurate with increasing n. We will use the shorthand notation $\hat{p}_{i\alpha} \stackrel{\text{def}}{=} t_{i\alpha}/n$. The typical deviation $|\hat{p}_{i\alpha} - p_{i\alpha}|$ scales as $1/\sqrt{n}$. If n is not very large, or if p_{iy_i} is small, then $\hat{p}_{iy_i}$ is noticeably different from p_{iy_i}, which yields a score noticeably different from [21].
- The parameter c appears in the score function, even though it is not known to the tracer. The tracer has to use a parameter c_0 instead, indicating the maximum coalition size that can be traced given the code length ℓ and alphabet size q. Alternatively, he can use several score systems, each with a different c_0, in parallel.

Due to $c < \infty$ there is of course a mismatch between the strategy that the Neyman-Pearson score is aimed against (Interleaving) and the actual saddlepoint strategy. Hence (14) is not completely optimal. However, it is guaranteed to give a low FP error probability even when the coalition is much larger than expected.

We investigate the performance of our score function in Section IV.

IV. Performance of the tally-based score function

A. Setting the scene

We first define a version of the score that is shifted by a constant $\ln(1 - 1/c)$, such that a symbol $x_{ji} \neq y_i$ incurs zero score. Furthermore we replace the unknown c by c_0.

$$s_j = \frac{1}{\ell} \sum_{i=1}^{\ell} s_{ji} \quad (19)$$

$$s_{ji} \stackrel{\text{def}}{=} \ln \left(1 + \frac{\delta_{x_{ji} y_i}}{c_0 - 1} \cdot \frac{n-1}{t_{iy_i} - 1}\right)$$
$$= \delta_{x_{ji} y_i} \ln \left(1 + \frac{1}{c_0 - 1} \cdot \frac{n-1}{t_{iy_i} - 1}\right). \quad (20)$$

Most scores in the literature are balanced such that an innocent user's expected score (at fixed $\bar{\boldsymbol{p}}$) is zero. However, here we cannot achieve this with a constant shift, because an innocent's score depends on the coalition's actions in a complicated way. The tracer uses a threshold Z that may in principle depend on all the knowledge he has, namely $\bar{\boldsymbol{p}}$, x and $\bar{y}$. In contrast to e.g. the Tardos score function [36], [38] a constant Z will not work. We analyze this more complicated situation by considering the following sequence of experiments.

- **Experiment 0**. Randomly generate $\bar{\boldsymbol{p}}$ according to the distribution F. Then, using $\bar{\boldsymbol{p}}$, generate the codewords of the colluders, i.e. the $\bar{x}_j$ for all $j \in \mathcal{C}$. Finally generate $\bar{y}$ based on $\bar{\boldsymbol{m}}$. (The $\bar{\boldsymbol{m}}$ follows from the colluders' codewords.)
- **Experiment 1**. The $\bar{p}$, $(\bar{x}_j)_{j \in \mathcal{C}}$ and $\bar{y}$ are fixed. Now randomly generate the codewords of the innocent users. (Note: the innocent user symbols at all the positions $i \in [\ell]$ are *independent* random variables, even if the attack strategy breaks position symmetry!)

This approach is similar to the 'operational mode' of Furon and Desoubeaux [12].

B. Infinite scores

As mentioned in Section III, it can occur that one or more colluders have an infinite score, in which case it is possible to accuse them with 100% certainty. (Innocent users can never get an infinite score.) For some cases we can provide a simple analytic expression for the probability that such an infinite score occurs.

Definition 1: Consider a single position. For $b \in \{1, \ldots, c\}$ we define $G_b \in [0, 1]$ as

$$G_b \stackrel{\text{def}}{=} \Pr[M_Y = b]. \quad (21)$$

In words: G_b is a parameter that depends on the colluder strategy, and it indicates the probability that the output symbol

y was seen by exactly b colluders. From the Marking Assumption it follows that G_c does not depend on the attack strategy. For the investigation of infinite scores we are of course interested in G_1. In a number of cases we can obtain simple expressions for G_b.

Lemma 3: For ($q = 2$, $c \geq 3$, Majority Voting) it holds that $G_1 = 0$. For ($q = 2$, $c \geq 3$, Minority Voting) it holds that

$$G_1^{\mathrm{MinV}} = \frac{\Gamma(c - \frac{1}{2})}{\sqrt{\pi}\,\Gamma(c)}. \tag{22}$$

For ($q = 2$, Coin Flip) it holds that

$$G_1^{\mathrm{CoinFlip}} = \frac{\Gamma(c - \frac{1}{2})}{2\sqrt{\pi}\,\Gamma(c)}. \tag{23}$$

In the case of the Interleaving attack, the parameter G_b is given by

$$\begin{aligned} G_b^{\mathrm{Int}} &= q\binom{c-1}{b-1}\frac{B(\frac{1}{2}+b, \frac{q-1}{2}+c-b)}{B(\frac{1}{2}, \frac{q-1}{2})} \qquad (24)\\ &= \frac{q}{\sqrt{\pi}}\frac{\Gamma(c)}{\Gamma(c+\frac{q}{2})}\frac{\Gamma(b+\frac{1}{2})}{\Gamma(b)}\frac{\Gamma(c-b+\frac{q-1}{2})}{\Gamma(c-b+1)}\frac{\Gamma(\frac{q}{2})}{\Gamma(\frac{q-1}{2})}. \end{aligned}$$

Furthermore, independent of the strategy, G_c is given by

$$G_c = \frac{q}{\sqrt{\pi}}\frac{\Gamma(c+\frac{1}{2})\Gamma(\frac{q}{2})}{\Gamma(c+\frac{q}{2})}. \tag{25}$$

Proof: The G_1 for Majority Voting is trivial, since $\theta_{\alpha|\boldsymbol{m}} = 0$ under the given circumstances.

The m_α is binomial-distributed, with $\Pr[M_\alpha = m_\alpha|\boldsymbol{p}] = \binom{c}{m_\alpha} p_\alpha^{m_\alpha}(1-p_\alpha)^{c-m_\alpha}$. For fixed α we introduce the notation $\boldsymbol{m} = (m_\alpha, \boldsymbol{m}_{\setminus\alpha})$, where $\boldsymbol{m}_{\setminus\alpha}$ is the $(q-1)$-component vector consisting of the tallies $(m_\beta)_{\beta\in\mathcal{Q}\setminus\{\alpha\}}$. We write

$$\begin{aligned} G_b &= \sum_{\alpha\in\mathcal{Q}} \mathbb{E}_{\boldsymbol{p}}\mathbb{E}_{\boldsymbol{m}|\boldsymbol{p}}\delta_{m_\alpha,b}\theta_{\alpha|\boldsymbol{m}}\\ &= \sum_{\alpha\in\mathcal{Q}} \mathbb{E}_{\boldsymbol{p}} \sum_{\boldsymbol{m}_{\setminus\alpha}} \binom{c}{b}\binom{c-b}{\boldsymbol{m}_{\setminus\alpha}} p_\alpha^b \boldsymbol{p}_{\setminus\alpha}^{\boldsymbol{m}_{\setminus\alpha}}\theta_{\alpha|\boldsymbol{m}}\\ &= \binom{c}{b}\sum_{\alpha\in\mathcal{Q}} \mathbb{E}_{\boldsymbol{p}} p_\alpha^b \sum_{\boldsymbol{m}_{\setminus\alpha}} \binom{c-b}{\boldsymbol{m}_{\setminus\alpha}} \boldsymbol{p}_{\setminus\alpha}^{\boldsymbol{m}_{\setminus\alpha}}\theta_{\alpha|(b,\boldsymbol{m}_{\setminus\alpha})} \qquad (26) \end{aligned}$$

In the case of $c \geq 3$, $q = 2$, $b = 1$ and Minority Voting we have $\theta = 1$. Eq. (26) then reduces to $c\sum_\alpha \mathbb{E}_{\boldsymbol{p}} p_\alpha(1-p_\alpha)^{c-1}$, which can be evaluated using the marginal probability density function $F(p_\alpha) = p_\alpha^{-\frac{1}{2}}(1-p_\alpha)^{-1+\frac{q-1}{2}}/B(\frac{1}{2}, \frac{q-1}{2})$ [32] and Lemma 2, yielding (23). (The $\sum_\alpha$ reduces to a factor 2).

For $q = 2$ the analysis for the Coin Flip attack differs from Minority Voting only be a factor $1/2$, since $\theta_{\alpha|\boldsymbol{m}}$ equals $1/2$ as long as m_α is not 0 or c.

In the case of the Interleaving attack for general b, we substitute $\theta_{\alpha|(b,\boldsymbol{m}_{\setminus\alpha})} = b/c$ in (26), which allows us to evaluate the multinomial sum over $\boldsymbol{m}_{\setminus\alpha}$,

$$\begin{aligned} G_b^{\mathrm{Int}} &= \frac{b}{c}\binom{c}{b}\sum_{\alpha\in\mathcal{Q}} \mathbb{E}_{\boldsymbol{p}} p_\alpha^b(1-p_\alpha)^{c-b}\\ &= \frac{b}{c}\binom{c}{b}\sum_{\alpha\in\mathcal{Q}} \frac{B(b+\frac{1}{2}, c-b+\frac{q-1}{2})}{B(\frac{1}{2}, \frac{q-1}{2})}. \qquad (27) \end{aligned}$$

In the last line we used the marginal pdf $F(p_\alpha)$ and Lemma 2. The $\sum_\alpha$ reduces to a factor q. Writing the Beta functions and $\binom{c}{b}$ in terms of Gamma functions gives the second expression for G_b^{Int} in the lemma.

Finally, it follows directly from the Marking Assumption that G_c cannot depend on the colluder strategy, since $\theta_{\alpha|\boldsymbol{m}}$ equals 1 when $m_\alpha = c$. The G_c is obtained e.g. by setting $b = c$ in (24). ∎

Lemma 4: Let $\alpha \in \mathcal{Q}$. Consider the bias $\boldsymbol{p}$ in a single position. The marginal distribution of p_α, given tally m_α, is

$$F(p_\alpha|m_a) = \frac{p_\alpha^{-\frac{1}{2}+m_\alpha}(1-p_\alpha)^{-1+c-m_\alpha+\frac{q-1}{2}}}{B(m_\alpha+\frac{1}{2}, c-m_\alpha+\frac{q-1}{2})}. \tag{28}$$

Proof: We start from the joint probability $F(p_\alpha, m_\alpha) = F(p_\alpha)\binom{c}{m_\alpha}p_\alpha^{m_\alpha}(1-p_\alpha)^{c-m_\alpha}$, that follows from the $F(\boldsymbol{p})$ given in Section II-B. We divide $F(p_\alpha, m_\alpha)$ by the marginal distribution of m_α, which does not depend on p_α. This yields $F(p_\alpha|m_a) \propto p_\alpha^{-\frac{1}{2}+m_\alpha}(1-p_\alpha)^{-1+c-m_\alpha+\frac{q-1}{2}}$. The Beta function in (28) is a normalization constant. ∎

Lemma 5: Consider a single position. The probability that an infinite colluder score occurs in this position is given by

$$\Pr[T_Y = 1] = G_1\frac{\Gamma(c+\frac{q}{2})\Gamma(n+\frac{q}{2}-\frac{5}{2})}{\Gamma(c+\frac{q}{2}-\frac{3}{2})\Gamma(n+\frac{q}{2}-1)}. \tag{29}$$

Proof: $\Pr[T_Y = 1] = \Pr[M_Y = 1]\Pr[T_Y = 1|M_Y = 1] = G_1\int_0^1 F(p_y|1)(1-p_y)^{n-c}\mathrm{d}p_y$, where $F(p_y|1)$ is the conditional marginal of Lemma 4. The factor $(1-p_y)^{n-c}$ is the probability that none of the innocents receive symbol y, for given p_y. The integral is evaluated using Lemma 2. ∎

Theorem 3: Let the colluder strategy be position symmetric. The probability that at least one colluder has infinite score is given by

$$\Pr[\exists j \in \mathcal{C}:\ s_j = \infty] = 1 - (1 - \Pr[T_Y = 1])^\ell \tag{30}$$

where $\Pr[T_Y = 1]$ stands for the single-position probability given in Lemma 5.

Proof: The probability of having *no* infinite score overall is the product of the probabilities of *not* having $t_y = 1$ in the ℓ separate positions. ∎

For $n \gg q$, $n \gg c^2$ the probability (30) can be approximated by its first order Taylor term,

$$\Pr[\exists j \in \mathcal{C}:\ s_j = \infty] \approx \ell G_1\frac{\Gamma(c+\frac{q}{2})}{\Gamma(c+\frac{q}{2}-\frac{3}{2})}n^{-\frac{3}{2}}. \tag{31}$$

We see that for large n the probability of having errorless accusation quickly dwindles.

C. False Positive bound using Bernstein's inequality

For Experiment 1 we want to investigate the probability $\Pr[S_j > Z]$ for arbitrary innocent user $j \notin \mathcal{C}$. We want to use Bernstein's inequality (Lemma 1). However, our S_{ji} does not have zero mean, so we first have to shift it. We define

$$U_{ji} \stackrel{\mathrm{def}}{=} S_{ji} - \mathbb{E}_{X_{\mathrm{innocents}}|\bar{p}\bar{\boldsymbol{m}}\bar{y}}[S_{ji}] \quad \text{for } j \notin \mathcal{C}. \tag{32}$$

We stress that U_{ji} is defined only for *innocent* users. In order to do the '$X_{\text{innocents}}$' average we introduce a tally variable $\boldsymbol{K}$ for the set of innocent users minus user j,

$$k_{i\alpha} \stackrel{\text{def}}{=} |\{v \in ([n] \setminus \mathcal{C}) \setminus \{j\} : \; x_{vi} = \alpha\}|. \tag{33}$$

For all $i \in [\ell]$ it holds that $\sum_{\alpha \in \mathcal{Q}} k_{i\alpha} = n - c - 1$. The dependence of $\boldsymbol{k}_i$ on j is not made explicit in the notation, since $\boldsymbol{k}_i$ has the interpretation 'the tally of a set of $n - c - 1$ randomly generated innocent users'. The tally $\boldsymbol{k}_i$ is multinomial-distributed, with parameters $\boldsymbol{p}_i$ and $n - c - 1$. This notation allows us to express U_{ji} more precisely,

$$U_{ji} \stackrel{\text{def}}{=} S_{ji} - \mathbb{E}_{X_{ji}\boldsymbol{K}_i|\bar{p}\bar{\boldsymbol{m}}\bar{y}}[S_{ji}] \quad \text{for } j \notin \mathcal{C}. \tag{34}$$

We write $T_{i\alpha} = m_{i\alpha} + \delta_{X_{ji}\alpha} + K_{i\alpha}$, which yields

$$S_{ji} = \delta_{X_{ji}y_i} \ln(1 + \frac{1}{c_0 - 1} \cdot \frac{n-1}{m_{iy_i} + K_{iy_i}}) \quad \text{for } j \notin \mathcal{C}. \tag{35}$$

The X_{ji} and $\boldsymbol{K}_i$ are independent random variables. Hence the $\mathbb{E}_{X_{ji}\boldsymbol{K}_i|\bar{p}\bar{\boldsymbol{m}}\bar{y}} S_{ji}$ factorizes into $(\mathbb{E}_{X_{ji}|\bar{p}\bar{\boldsymbol{m}}\bar{y}}\delta_{X_{ji}y_i}) \cdot \mathbb{E}_{\boldsymbol{K}_i|\bar{p}\bar{\boldsymbol{m}}\bar{y}} \ln(\cdots)$ and we get

$$\begin{aligned} U_{ji} &= \delta_{X_{ji}y_i} \ln(1 + \frac{1}{c_0 - 1} \cdot \frac{n-1}{m_{iy_i} + K_{iy_i}}) \\ &\quad - p_{iy_i} J_1(p_{iy_i}, m_{iy_i}) \end{aligned} \tag{36}$$

$$J_a(p_{iy_i}, m_{iy_i}) \stackrel{\text{def}}{=} \mathbb{E}_{\boldsymbol{K}_i|\boldsymbol{p}_i} \ln^a(1 + \frac{1}{c_0 - 1} \cdot \frac{n-1}{m_{iy_i} + K_{iy_i}}).$$

We furthermore define

$$\begin{aligned} U_{\max} \stackrel{\text{def}}{=} \max_i \max\Big[&\ln(1 + \frac{n-1}{(c_0-1)m_{iy_i}}) - p_{iy_i} J_1(p_{iy_i}, m_{iy_i}), \\ &p_{iy_i} J_1(p_{iy_i}, m_{iy_i})\Big] \end{aligned} \tag{37}$$

as the maximum absolute value of the score that could possibly occur, and

$$\nu(\bar{\boldsymbol{p}}, \bar{\boldsymbol{m}}, \bar{y}) \stackrel{\text{def}}{=} \frac{1}{\ell} \sum_{i=1}^{\ell} p_{iy_i} J_1(p_{iy_i}, m_{iy_i}) \tag{38}$$

$$\zeta \stackrel{\text{def}}{=} Z - \nu \tag{39}$$

$$\sigma^2(\bar{\boldsymbol{p}}, \bar{\boldsymbol{m}}, \bar{y}) \stackrel{\text{def}}{=} \frac{1}{\ell} \sum_{i=1}^{\ell} [p_{iy_i} J_2(p_{iy_i}, m_{iy_i}) - p_{iy_i}^2 J_1^2(p_{iy_i}, m_{iy_i})] \tag{40}$$

We are now ready to invoke Bernstein's inequality.

Theorem 4: Let $j \in [n] \setminus \mathcal{C}$ be an arbitrary innocent user. Let the score S_{ji} be defined as in (35) and let the threshold Z be parametrized as $Z = \nu + \zeta$. Then in Experiment 1 the one-user false accusation probability $P_{\text{FP1}} \stackrel{\text{def}}{=} \Pr[\frac{1}{\ell}\sum_{i\in[\ell]} S_{ji} > Z | j \notin \mathcal{C}]$ can be bounded as

$$P_{\text{FP1}}^{\text{Exp.1}} \leq \exp\left[-\ell \frac{\zeta^2}{2\sigma^2 + \frac{2}{3}\zeta U_{\max}}\right]. \tag{41}$$

where $U_{\max}$ and σ^2 are defined as in (37), (40).

Proof: We have $\Pr[\frac{1}{\ell}\sum_{i\in[\ell]} S_{ji} > Z] = \Pr[\frac{1}{\ell}\sum_{i\in[\ell]} U_{ji} > \zeta]$. The U_{ji} are zero-mean, independent random variables, and ζ does not depend on these variables. We write $V_i = U_{ji}/\ell$ in Bernstein's inequality (Lemma 1). The absolute value $|U_{ji}|$ cannot exceed $U_{\max}$. Hence we can set $a = U_{\max}/\ell$ in Bernstein's inequality. Finally we need to evaluate $\mathbb{E}[U_{ji}^2]$. We have $\sum_i \mathbb{E}[U_{ji}^2] = \sum_i \mathbb{E}[S_{ji}^2 - 2p_{iy_i}S_{ji}J_1 + p_{iy_i}^2 J_1^2] = \sum_i [p_{iy_i}J_2 - 2p_{iy_i}^2 J_1^2 + p_{iy_i}^2 J_1^2] = \ell\sigma^2$. Substitution of all these elements into Lemma 1 yields (41). ∎

Even though we cannot analytically evaluate the expressions J_2 and J_1, they are straightforward to compute numerically, and hence Theorem 4 gives a recipe for setting the accusation threshold.

Theorem 5: Let the tracer use the score function (20) and set the accusation threshold as

$$\begin{aligned} Z_* &= \nu + \zeta_* \\ \zeta_* &= \frac{1}{3\ell} U_{\max} \ln\frac{1}{\varepsilon_1} + \sqrt{(\frac{1}{3\ell} U_{\max} \ln\frac{1}{\varepsilon_1})^2 + \frac{2}{\ell}\sigma^2 \ln\frac{1}{\varepsilon_1}}. \end{aligned} \tag{42}$$

Then in Experiment 1 it holds that $P_{\text{FP1}} \leq \varepsilon_1$.

Proof: According to Theorem 4, it is sufficient for the tracer to set ζ such that $\exp[-\ell \cdot \frac{1}{2}\zeta^2/(\sigma^2 + \frac{1}{3}U_{\max}\zeta)] = \varepsilon_1$. This yields a quadratic equation in ζ, namely $\frac{1}{2}\ell\zeta^2 - \frac{1}{3}U_{\max}\ln\frac{1}{\varepsilon_1}\zeta - \sigma^2 \ln\frac{1}{\varepsilon_1} = 0$, whose positive solution ζ_* is precisely the expression given in Theorem 5. Hence the tracer may set the threshold Z at $\nu + \zeta_*$ or larger, and then it is guaranteed that $P_{\text{FP1}} \leq \varepsilon_1$. ∎

The result (42) makes intuitive sense. The part ν corresponds to the observed average of all the user scores. The σ^2 under the square root corresponds to the score variance. Its magnitude compared to the $(\frac{1}{3}\cdots)^2$ term under the square root depends on the collusion strategy. If the variance term dominates, then Z tends to the form "$\nu + \sigma\ell^{-1/2}\sqrt{2\ln(1/\varepsilon_1)}$", which is approximately where one would put the threshold if the score were Gaussian-distributed.

Note that the tracer does not know the colluder tallies $\bar{\boldsymbol{m}}$; hence the above result is not immediately practical. Below we derive a practical recipe for placing the threshold.

Lemma 6: Let $U_{\max}$ be defined as in (37). For $n \gg c$ it then holds that

$$U_{\max} < U_{\max}^{\text{pract}} \stackrel{\text{def}}{=} \ln[1 + \frac{n-1}{c_0 - 1}]. \tag{43}$$

Proof: For $n \gg c$, the expression $\ln[1 + \frac{n-1}{(c_0-1)m_{iy_i}}]$ in (37) dominates the expressions containing J_1. This yields $U_{\max} = \max_i \Big[\ln(1 + \frac{n-1}{(c_0-1)m_{iy_i}}) - p_{iy_i}J_1(p_{iy_i}, m_{iy_i})\Big] < \max_i \ln(1 + \frac{n-1}{(c_0-1)m_{iy_i}}) \leq \ln(1 + \frac{n-1}{c_0-1})$. ∎

Lemma 7: Let σ^2 be defined as in (40). Let $2 \leq c \leq c_0$. Then

$$\sigma^2 < \sigma^2_{\text{pract}} \stackrel{\text{def}}{=} \frac{1}{\ell}\sum_{i=1}^{\ell} [p_{iy_i} J_2(p_{iy_i}, 1) - p_{iy_i}^2 J_1^2(p_{iy_i}, c_0)]. \tag{44}$$

Proof: We use $1 \leq m_{iy_i} \leq c$. We have $J_2(p_{iy_i}, m_{iy_i}) \leq J_2(p_{iy_i}, 1)$ and $J_1(p_{iy_i}, m_{iy_i}) \geq J_1(p_{iy_i}, c) \geq J_1(p_{iy_i}, c_0)$. Substitution of these inequalities into (40) yields the right-hand side of (44). Since m_{iy_i} cannot be simultaneously equal to 1 and to c_0, the σ^2 cannot equal σ^2_{pract}. ∎

Lemma 8: Let ν be defined as in (38). Then

$$\nu \leq \nu_{\text{pract}} \stackrel{\text{def}}{=} \frac{1}{\ell}\sum_{i=1}^{\ell} p_{iy_i} J_1(p_{iy_i}, 1). \tag{45}$$

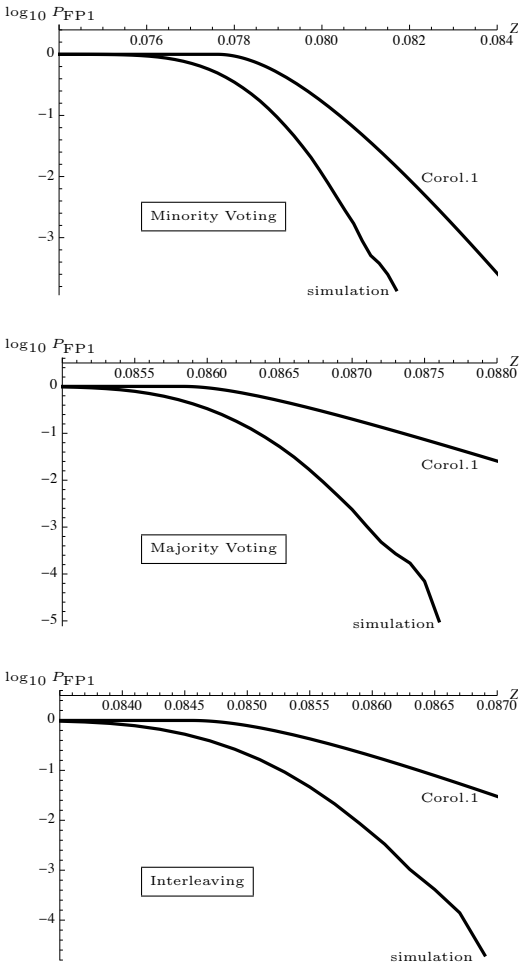

Fig. 1. *The one-user false positive probability $P_{\rm FP1}$ as a function of the threshold Z, for Experiment 1 with parameters $n = 100000, c = 12, \ell = 14000$ and use of the score function (20). In the simulation the $P_{\rm FP1}$ was estimated by doing a single run of Experiment 0 and Experiment 1 and then counting how many innocent users had a score exceeding Z.*

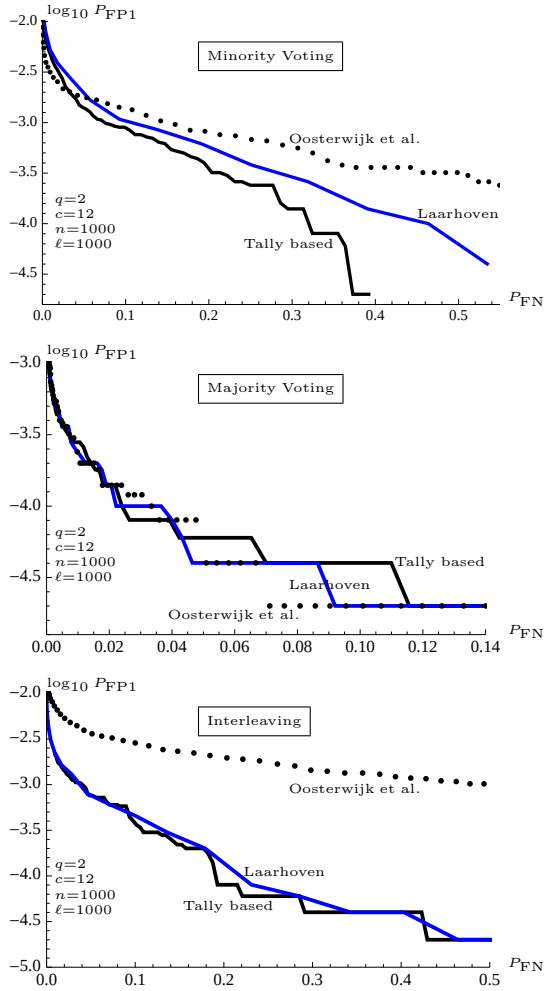

Fig. 2. *ROC curves for the Oosterwijk et al. score function $-1 + \delta_{xy}/p_y$, the Laarhoven score function $\ln(1+\frac{1}{c_0}[-1+\delta_{xy}/p_y])$ with $c = c_0$, and the new tally based score function (20) with $c = c_0$. The simulations consisted of 50000 repetitions of the steps {Experiment 0; then make one innocent user codeword and ℓ tallies k_{iy_i} for the rest of the innocent users}. No cutoff was used on the $\mathbf{p}$. The error probabilities were obtained by counting the number of events with an FN or FP1 error. Three different attacks are shown. The jumps at low $P_{\rm FP1}$ are numerical artifacts caused by the finite number of runs (50000).*

Proof: We use $J_1(p_{iy_i}, m_{iy_i}) \le J_1(p_{iy_i}, 1)$. ∎

For $n \gg c_0 \ge c$ the 'practical' parameters do not differ much from the original ones.

Corollary 1: Let the threshold in Experiment 1 be set as $Z = \nu_{\rm pract} + \zeta$. Then

$$P_{\rm FP1}^{\rm Exp.1} < \exp\left[-\ell \frac{\zeta^2/2}{\sigma^2_{\rm pract} + \frac{1}{3}\zeta U^{\rm pract}_{\max}}\right]. \qquad (46)$$

For obtaining $P_{\rm FP1}^{\rm Exp.1} \le \varepsilon_1$ it suffices to set

$$\zeta = \frac{1}{3\ell} U^{\rm pract}_{\max} \ln\frac{1}{\varepsilon_1} + \sqrt{(\frac{1}{3\ell} U^{\rm pract}_{\max} \ln\frac{1}{\varepsilon_1})^2 + \frac{2}{\ell}\sigma^2_{\rm pract} \ln\frac{1}{\varepsilon_1}}. \qquad (47)$$

Proof: We have $Z > \nu + \zeta$, which implies that the FP error probability is smaller than in Thorem 5. Into Theorem 5 we substitute $\sigma^2 < \sigma^2_{\rm pract}$ and $U_{\max} < U^{\rm pract}_{\max}$ (Lemmas 7 and 6). This yields (46). Finally (47) follows by demanding that the right-hand side of (46) equals ε_1 and then solving for ζ. ∎

Corollary 1 is a recipe that contains only quantities known to the tracer.

It is of course possible to derive bounds using other techniques. In the Appendix we present an analysis using Markov's inequality instead of Bernstein's inequality. The tracer can set the threshold to the value prescribed by Corollary 1 or Theorem 11, whichever is smaller (usually Corollary 1).

It is worth noting that the analytic bounds obtained in this way are far from tight in non-asymptotic cases. Fig. 1 shows that the gap between the the bound on $P_{\rm FP1}$ and the actual $P_{\rm FP1}$ can be orders of magnitude.

D. ROC curves

Obtaining analytic bounds for the False *Negative* probability is far more complicated. It is also far less interesting. In the context of audio-video content tracing, a deterring effect is achieved even with very large FN probabilities, e.g. $P_{\rm FN} \approx 0.5$. It is entirely feasible to accurately determine such high probabilities by doing simulations. (On the other hand,

an accurate estimate of $P_{\rm FP}$, which may be as small as 10^{-6}, takes millions of simulation runs of Experiments 0 and 1.)

In Fig. 2 we show an example of Receiver Operating Characteristic (ROC) curves[2] obtained from simulations. Even at $n = 1000$, a rather small number of users, we see that there is little performance difference between the score (20) proposed in this paper and the Laarhoven score. An exception is the case of the Minority Voting attack, which favours low p_y values; at low p_y the statistical fluctuations in t_y are more pronounced than at large p_y, as already mentioned in Section III.

For the Interleaving attack there is a large performance gap between the Oosterwijk et al. score on the one hand, and the Laarhoven and tally-based score on the other. This is hardly surprising, since the Oosterwijk et al. score is designed to work at asymptotically large c.

It is important to note that *the ROC curves shown here are based on an accusation procedure that does not exploit the existence of infinite scores*. When an infinite score is detected, the decoder should in fact re-set the threshold; this was not done in Fig. 2, even though the Minority Voting experiments at $n = 1000$ had such events occurring 19% of the time and Interleaving 2%. Analysis of such an improved decoder, as well as more exhaustive numerics for different combinations of q, c, c_0, n and ℓ, are the subject of future work.

V. Probability distribution of the single-position score

In this section we study the probability mass function of the single-position tally-based score S_{ji} (20) for an innocent user j, for repetitions of Experiments 0 and 1. As far as we are aware, this kind of analysis has not yet been done for the Oosterwijk et al. score and the Laarhoven score. Therefore we first present the analysis for these scores.

The single-position distribution is of interest for several reasons. (i) Knowing the distribution allows one to use the method of Simone et al. [32] to obtain the probability distribution of the entire score (i.e. added over all positions). (ii) By looking at the moments of the single-position score distribution, especially the third moment, one can determine how Gaussian the entire score is. A Gaussian distribution allows for simpler analysis.

A. Probability density for the Oosterwijk et al. score

The generalized Laarhoven score is given by

$$w_j \stackrel{\rm def}{=} \frac{1}{\ell}\sum_{i=1}^{\ell} w_{ji} \;;\; w_{ji} \stackrel{\rm def}{=} \delta_{x_{ji}y_i} \ln(1+\frac{1}{(c_0-1)p_{iy_i}}), \quad (48)$$

We do our analysis by first looking at Oosterwijk et al.'s score function h,

$$h(x,y,\boldsymbol{p}) \stackrel{\rm def}{=} \frac{\delta_{xy}}{p_y} - 1, \quad (49)$$

and then applying a change of variables,

$$w_{ji} = \ln[1+\frac{1}{c_0}h(x_{ji},y_i,\boldsymbol{p}_i)] - \ln[1-\frac{1}{c_0}]. \quad (50)$$

[2]Actually we represent the axes in a slightly different way. We plot the FN instead of the True Positive probability.

We derive the distribution of the score h in a couple of small steps.

Lemma 9: [See e.g. [15].] Let $f : \mathbb{R} \to \mathbb{R}$ be a monotonous function with inverse function $f^{\rm inv}$. Let δ denote the Dirac delta function. Then $\delta(u - f(p)) = \frac{\delta(p - f^{\rm inv}(u))}{|f'(p)|}$.

Corollary 2: Let $h_1(p) \stackrel{\rm def}{=} 1/p - 1$. It holds that

$$\delta(u - h_1(p)) = p^2\delta(p - \frac{1}{u+1}) = \frac{\delta(p-\frac{1}{u+1})}{(u+1)^2}. \quad (51)$$

Proof: We use Lemma 9 with $f = h_1$. We have $h_1^{\rm inv}(u) = 1/(u+1)$ and $h_1'(p) = -p^{-2}$. ∎

Lemma 10: For a user $j \notin \mathcal{C}$, the probability density of the score h in a single position, with given p_y, is

$$\varphi_h(u|p_y) = (1-p_y)\delta(u+1) + p_y\delta(u - h_1(p_y)). \quad (52)$$

Proof: With probability $1-p_y$, an innocent user gets score $u = -1$; with probability p_y he gets $u = h_1(p_y)$. ∎

At this point we assume position symmetry of the attack.

Lemma 11: Let the colluders use a position-symmetric strategy. The probability density for the variable P_Y is given by $\rho(p_y) = \sum_{b=1}^{c} G_b F(p_y|b)$.

Proof: If m_y is known, then the probability density for P_Y is given by $F(p_y|m_y)$ as defined in (28). Taking the expectation over M_Y yields the $\sum_b$ expression in Lemma 11. ∎

Theorem 6: Let the colluders use a position-symmetric strategy. For a user $j \notin \mathcal{C}$, the probability density of the Oosterwijk et al. score h in a single position is

$$\begin{aligned}\varphi_h(u) &= \sum_{b=1}^{c} G_b \left\{ \delta(u+1)\frac{c-b+\frac{q-1}{2}}{c+\frac{q}{2}} \right. \quad (53)\\ &+ \left. \frac{\Theta(u)(1+u)^{-\frac{5}{2}-b}}{B(b+\frac{1}{2}, c-b+\frac{q-1}{2})}(\frac{u}{1+u})^{-1+c-b+\frac{q-1}{2}} \right\}.\end{aligned}$$

Proof: We have $\varphi_h(u) = \mathbb{E}_{p_y}\varphi_h(u|p_y)$. Using Lemma 10 and Corollary 2 we get $\varphi_h(u) = \delta(u+1)\mathbb{E}_{p_y}(1-p_y) + (u+1)^{-3}\mathbb{E}_{p_y}\delta(p_y - \frac{1}{u+1})$. The expectations are evaluated using the $\rho(p_y)$ from Lemma 11,

$$\begin{aligned}\mathbb{E}_{p_y}(1-p_y) &= \sum_{b=1}^{c} G_b \int_0^1 {\rm d}p\; F(p|b)(1-p)\\ &= \sum_{b=1}^{c} G_b \frac{c-b+\frac{q-1}{2}}{c+\frac{q}{2}} \quad (54)\end{aligned}$$

$$\mathbb{E}_{p_y}\delta(p_y - \frac{1}{u+1}) = \Theta(u)\sum_{b=1}^{c} G_b\; F(\frac{1}{u+1}|b). \quad (55)$$

The step function $\Theta(u)$ in (55) occurs because for $u < 0$ the delta function $\delta(p_y - \frac{1}{u+1})$, with $p_y \le 1$, vanishes. ∎

From Theorem 6 we see that the density at $u \gg 1$ is proportional to $(\frac{1}{u})^{5/2+b}$, with $b \ge 1$.

- The Minority Voting strategy will cause the largest possible G_1 and thereby put maximal probability mass in the tail. Note that the Majority Voting attack for $c > 2$ has $G_1 = 0$.
- The 2nd moment of the distribution always exists, but in general ($G_1 > 0$) not the 3rd moment. The nonexistence of the 3rd moment implies that the distribution of the

overall score (all positions added) has 'fat tails', i.e. the distribution converges to Gaussian everywhere except in the tails, where the power law from the single-position distribution is inherited.

Theorem 7: Let the coalition use the Interleaving attack. Then for a user $j \notin \mathcal{C}$, the probability density of the Oosterwijk et al. score h in a single position is

$$\varphi_h^{\rm Int}(u) = \delta(u+1)\frac{q-1}{2+q} + \Theta(u)\frac{q(1+u)^{-\frac{7}{2}}}{B(\frac{1}{2},\frac{q-1}{2})}(\frac{u}{1+u})^{-1+\frac{q-1}{2}}. \tag{56}$$

Proof: We follow the proof of Theorem 6, but now the expectations (54) and (55) can be easily computed using $\Pr[Y=y|\boldsymbol{P}=\boldsymbol{p}] = p_y$,

$$\begin{aligned}
\mathbb{E}_{p_y}(1-p_y) &= \sum_y \mathbb{E}_{\boldsymbol{p}} p_y(1-p_y) = 1 - \sum_y \mathbb{E}_{\boldsymbol{p}} p_y^2 \\
&= 1 - \sum_y \frac{B(\frac{1}{2}\mathbf{1}_q + 2\boldsymbol{e}_y)}{B(\frac{1}{2}\mathbf{1}_q)} \\
&= 1 - q\frac{\Gamma(\frac{5}{2})\Gamma(\frac{q}{2})}{\Gamma(\frac{1}{2})\Gamma(2+\frac{q}{2})} \\
&= \frac{q-1}{2+q} \qquad (57)\\
\mathbb{E}_{p_y}\delta(p_y - \frac{1}{u+1}) &= \sum_y \mathbb{E}_{\boldsymbol{p}} p_y \delta(p_y - \frac{1}{u+1}) \\
&= q\,[pF(p)]_{p=\frac{1}{u+1}} \\
&= \frac{q}{u+1}F(\frac{1}{u+1}). \qquad (58)
\end{aligned}$$

Here $\mathbf{1}_q$ is the vector $(1,1,\ldots,1)$ of length q, and $\boldsymbol{e}_y$ is a q-component vector with $(\boldsymbol{e}_y)_\alpha = \delta_{y\alpha}$. The 'B' is the generalized Beta function. ∎

B. Probability density for the generalized Laarhoven score

Lemma 12: Let $X \sim \rho_X$ and $Y \sim \rho_Y$, with $Y = \lambda(X)$, where λ is a monotonous function. Then $\rho_Y(y) = \rho_X(x)/|\lambda'(x)| = \rho_X(\lambda^{\rm inv}(y)) \,/\, |\lambda'(\lambda^{\rm inv}(y))|$.

For a proof, see any book on probability theory.

Theorem 8: Let the colluders use a position-symmetric strategy. For a user $j \notin \mathcal{C}$, the probability density of the generalized Laarhoven score w_{ji} (48) in a single position is

$$\begin{aligned}
\varphi_w(\alpha) &= \sum_{b=1}^{c} G_b \left\{ \delta(\alpha)\frac{c-b+\frac{q-1}{2}}{c+q/2} \right. \\
&\quad + \Theta(\alpha - \ln\tfrac{c_0}{c_0-1})\frac{(c_0-1)^{-\frac{3}{2}-b}}{B(b+\frac{1}{2}, c-b+\frac{q-1}{2})} \cdot \\
&\quad \left. \cdot \frac{e^\alpha (e^\alpha - \frac{c_0}{c_0-1})^{-1+\frac{q-1}{2}+c-b}}{(e^\alpha-1)^{1+c+q/2}} \right\}. \qquad (59)
\end{aligned}$$

Proof: We use Lemma 12 with $\rho_X \to \varphi_h$; $\rho_Y \to \varphi_w$; $\alpha = \lambda(u) = \ln\frac{c_0+u}{c_0-1}$; $u = \lambda^{\rm inv}(\alpha) = (c_0-1)(e^\alpha - \frac{c_0}{c_0-1})$; $u+1 = (c_0-1)(e^\alpha-1)$; $1/\lambda'(u) = c_0 + u = (c_0-1)e^\alpha$, and then simplify. We use that $\delta(u+1) = e^{-\alpha}(c_0-1)^{-1}\delta(\alpha)$ and $\Theta(u) = \Theta(\alpha - \ln\frac{c_0}{c_0-1})$. ∎

Note that (59) contains c_0 as well as c. Also note that for $e^\alpha \gg 1$ the b'th term is proportional to $e^{-[\frac{3}{2}+b]\alpha}$, i.e. the tail *exponentially decreases*, with dominant contribution $\propto e^{-\frac{5}{2}\alpha}$ if $G_1 > 0$. When random variables with an exponential tail are summed, the result quickly converges to a Gaussian-distributed random variable. Without showing the data we mention that we observed Gaussian distributions in the (limited) simulations we performed.

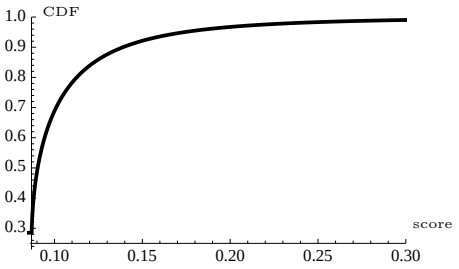

Fig. 3. *Cumulative Distribution Function for an innocent user's score in a single position, obtained from (59) and (61) for $q = 2$, $c = c_0 = 12$, $n = 1000$, Interleaving attack. The two curves completely overlap.*

C. Probability mass function for the tally-based score

For an innocent user $j \notin \mathcal{C}$, the possible values of the score s_{ji} (20) are either $s_{ji} = 0$ (the case $x \neq y$) or $s_{ji} = \ln[1 + \frac{1}{c_0-1}\cdot\frac{n-1}{t_y-1}]$, with $t_y \in \{2,\ldots,n\}$. In the second case we have $t_y \geq 2$ since $m_y \geq 1$ and the innocent user has symbol $x = y$. Whereas the Oosterwijk et al. score and the Laarhoven score are *continuous* random variables with a probability *density* function, the tally-based score is *discrete* and has a probability *mass* function. As such, the innocent user score does not have any complications such as infinite moments.

Theorem 9: Let the colluders use a position-symmetric strategy. Let $t \in \{2,\ldots,n\}$. For a user $j \notin \mathcal{C}$, the probability mass function of the tally-based score s_{ji} (20) in a single position i is given by

$$\Pr[S_{ji} = 0] = \sum_{b=1}^{c} G_b \frac{c-b+\frac{q-1}{2}}{c+\frac{q}{2}}, \tag{60}$$

$$\begin{aligned}
\Pr[S_{ji} = \ln(1 + \frac{1}{c_0-1}\frac{n-1}{t-1})] \qquad (61)\\
= \sum_{b=1}^{\min(t-1,c)} G_b \binom{n-c-1}{t-b-1} \frac{B(\frac{1}{2}+t, \frac{q-1}{2}+n-t)}{B(\frac{1}{2}+b, \frac{q-1}{2}+c-b)}.
\end{aligned}$$

Proof: We have $\Pr[S_{ji}=0] = \Pr[X \neq Y]$ which can be written as $\sum_b G_b \int_0^1 F(p_y|b)(1-p_y)\mathrm{d}p_y$. The integral is evaluated using Lemma 2. For proving (61) we have to compute the probability $\Pr[X = Y \wedge T_Y = t]$, which we can express as $\sum_b G_b \int_0^1 F(p_y|b) p_y \Pr[T_Y = t | M_Y = b, X = Y, P_Y = p_y]\mathrm{d}p_y$, where b cannot exceed $t-1$. The last factor is a binomial, $\binom{n-c-1}{t-b-1} p_y^{t-b-1}(1-p_y)^{n-c-t+b}$. The integration is again evaluated using Lemma 2. ∎

The probability mass function (61) is illustrated in Fig. 3 in the form of a cumulative distribution. The graph also shows the Laarhoven density function (59); it is indistinguishable for the given choice of parameters.

VI. Group Testing

There is a well known link [35], [9], [23], [19] between on the one hand Traitor Tracing in the RDM with the 'All-1' attack,

and on the other hand (non-adaptive) Group Testing [11]. The Group Testing scenario is as follows. There is a population of n people, of which c are infected. Medical tests are expensive, and there is money to do only ℓ tests, with $\ell \ll n$. Furthermore the tests take a long time, so they are done non-adaptively, in parallel. An efficient way has to be devised to find out who is infected. Luckily it is possible to combine samples (e.g. blood samples) from multiple people and run a single test on the combination; if one or more of the individual samples come from an infected person, the medical test is positive.

The analogy with Traitor Tracing is straightforward. The user symbol $x_{ji} \in \{0,1\}$ indicates whether person j'th blood is included in the i'th test. The result of the i'th test is $y_i \in \{0,1\}$. The way the combined test works exactly matches the All-1 strategy: $\theta_{1|m_1}$ equals 1 if $m_1 \geq 1$ and 0 if $m_1 = 0$.

We derive the most powerful hypothesis test for the hypothesis 'person j is infected'.

Theorem 10: In the case of the Restricted Digit Model, $q = 2$, and the All-1 collusion strategy, the score (7) reduces to

$$\begin{aligned}
y_i = 0,\, x_{ji} = 0: &\qquad \ln c - \ln(t_{i0} - c) \\
y_i = 0,\, x_{ji} = 1: &\qquad -\infty \\
y_i = 1,\, x_{ji} = 0: &\qquad -\ln \frac{\binom{n-1}{c} - \binom{t_{i0}-1}{c-1}}{\binom{n-1}{c-1} - \binom{t_{i0}-1}{c-1}} \\
y_i = 1,\, x_{ji} = 1: &\qquad -\ln \frac{n-c}{c} - \ln[1 - \frac{\binom{t_{i0}}{c}}{\binom{n}{c}}]. \qquad (62)
\end{aligned}$$

Proof: We omit indices i and j. For $q = 2$ the colluder tally vector reduces to $(c - m_1, m_1)$ and we can sum over a single variable $m_1 \in \{0, \ldots, c\}$. We will write m instead of m_1. The strategy parameters can be written as $\theta_{y|m} = \delta_{y1}(1 - \delta_{m0}) + \delta_{y0}\delta_{m0}$. We go case by case.

For $y = 0, x = 0$ the enumerator in (7) reduces to $\sum_m L_{m|t_1}\theta_{0|m}(c-m) = L_{0|t_1}c$ and the denominator reduces to $\sum_m L_{m|t_1}\theta_{0|m}(t_0 - m_0) = L_{0|t_1}(t_0 - c)$.

For $y = 0, x = 1$ the enumerator reduces to zero, while the denominator is nonzero. The logarithm of zero is $-\infty$.

For $y = 1, x = 0$ the enumerator reduces to $c(1-L_{0|t_1}) - \frac{c}{n}t_1$, while the denominator becomes $(t_0 - c)(1 - L_{0|t_1}) + \frac{c}{n}t_1$. Then we use $t_1 = n - t_0$ and $L_{0|t_1} = \binom{t_0}{c}/\binom{n}{c}$, followed by some laborious rewriting.

For $y = 1, x = 1$ the enumerator reduces to $\frac{c}{n}t_1$ and the denominator to $t_1(1 - L_{0|t_1}) - \frac{c}{n}t_1$. ∎

We note the following about Theorem 10,

- The '$-\infty$' for $x_{ji} = 1$, $y_i = 0$ makes perfect sense: if person j is included in the i'th test and this test gives a negative result, then person j is definitely not infected.
- In the case $y_i = 0$, $x_{ji} = 0$ we see that the score increases when t_{i0} decreases. This is intuitively correct: At decreasing t_{i0} the event $Y_i = 0$ becomes more and more 'special' in the sense of condemning person j, since the tested group becomes bigger and bigger without yielding a detection. In the extreme case $t_{i0} = c$, the outcome $y_i = 0$ immediately implies that all the people excluded from the test, including j, are infected. Indeed the score becomes $-\ln 0 = +\infty$. (Note that $t_0 < c$ automatically causes $y = 1$; Eq. (62) never gets a negative argument in a logarithm.)
- It may look strange that in the case $x_{ji} = 0$ (user j not included in the i'th test) the score actually depends on y_i. This dependence is caused by the fact that the result y_i does say something about the number of infected people *outside* the tested set.

In group testing there is no adversary and hence no max-min game. Instead of using a bias distribution $F(\boldsymbol{p})$ it is optimal to take a constant $\boldsymbol{p}$ for each test, with $p_1 = (\ln 2)/c + \mathcal{O}(c^{-2})$ [20]. This means that typically $t_1 = \mathcal{O}(n/c)$ and $t_0 = n - \mathcal{O}(n/c)$. Hence the fraction $\binom{t_0}{c}/\binom{n}{c}$ typically is not much smaller than 1.

Lemma 13: For $n \gg c$ we can approximate the score (62) as

$$\begin{aligned}
y_i = 0,\, x_{ji} = 0: &\qquad -\ln \frac{n}{c} - \ln \frac{t_{i0}}{n} + \mathcal{O}(\frac{c}{n}) \\
y_i = 0,\, x_{ji} = 1: &\qquad -\infty \\
y_i = 1,\, x_{ji} = 0: &\qquad -\ln \frac{n}{c} + \ln[1 - (\frac{t_{i0}}{n})^{c-1}] + \mathcal{O}(\frac{c}{n}) \\
y_i = 1,\, x_{ji} = 1: &\qquad -\ln \frac{n}{c} - \ln[1 - (\frac{t_{i0}}{n})^{c}] + \mathcal{O}(\frac{c}{n}). \qquad (63)
\end{aligned}$$

Proof: (*sketch*) We asked Wolfram Mathematica for a series expansion in the limit $n \to \infty$ for finite c. ∎

Note that we can add $\ln \frac{n}{c}$ to all the expressions in (63) to obtain an equivalent score that does not depend so heavily on the (possibly unknown) parameter c.

VII. Summary

We have written down a Neyman-Pearson hypothesis test for the hypothesis "user j is part of the coalition", and as evidence we have taken *all* the information available to the tracer, including the codewords of all the other users. This results in Theorem 1, which is very general. Motivated by the closeness of the Saddlepoint attack to Interleaving, we have substituted into our test the Interleaving attack, in order to obtain a 'universal' decoder. This procedure yields the score (20) for user j, which depends on the 'guilty symbol' tallies $(t_{iy_i})_{i=1}^{\ell}$ of the whole population.

In the limit $n \to \infty$ the score function reduces to (the q-ary generalization of) the $\boldsymbol{p}$-dependent log-likelihood Laarhoven score [21], which in turn reduces to Oosterwijk et al.'s score [30] for $c_0 \to \infty$.

We have given a first analysis of the error probabilities. Corollary 1 shows a threshold setting sufficiently high to ensure that the single-user FP error probability stays below ε_1. The threshold depends on the observed $\bar{y}$ and $\bar{\boldsymbol{p}}$. For non-asymptotic c_0 there is a large gap between the bound and the actual performance of the scheme. ROC curves for $q = 2$, obtained from a limited set of simulations, show that the new score is very close to the Laarhoven score except for attack strategies that favour low p_y values, such as Minority Voting; there the new score clearly performs better.

In the case of position-symmetric attacks, the statistical behaviour of a score system can be understood by studying the probability distribution of single-position scores [32], [31], [33]. To this end we have derived the innocent-user single-position distribution for the Oosterwijk et al. score, the Laarhoven score and our new score. The results are given

in Theorems 6, 8 and 9. The strategy dependence is entirely contained in the parameters G_b.

Finally we have applied our Neyman-Pearson test (7) to the field of Group Testing and obtained a new score function (Theorem 10) that may improve the state of the art.

We see various open questions for future work. (i) Investigate how much performance difference there is between (20) and the score that would be obtained if the finite-c saddlepoint is substituted into Theorem 1; (ii) More elaborate simulations (for many combinations of q, c, c_0, ℓ, n, and attack strategy) to study the difference between the various decoders; (iii) Get a tighter bound on the FP, e.g. using techniques from [12]; (iv) Use the method of Simone et al. [32] to determine the full probability distribution of the score (48); (v) See if (62) yields an improvement over previously known group testing 'decoders'. (vi) Study various noise models and generalizations for group testing, using Theorem 1 as a starting point. (vii) Study decoders that exploit the occasional occurrence of infinite colluders scores.

ACKNOWLEDGMENT

Thijs Laarhoven, Jeroen Doumen, Jan-Jaap Oosterwijk and Benne de Weger are thankfully acknowledged for useful discussions. We thank the anonymous reviewers for their helpful suggestions.

REFERENCES

[1] E. Abbe and L. Zheng. Linear universal decoding for compound channels. *IEEE Transactions on Information Theory*, 56(12):5999–6013, 2010.

[2] E. Amiri and G. Tardos. High rate fingerprinting codes and the fingerprinting capacity. In *SODA 2009*, pages 336–345, 2009.

[3] S.N. Bernstein. *Theory of Probability*. Nauka, 1927.

[4] O. Blayer and T. Tassa. Improved versions of Tardos' fingerprinting scheme. *Designs, Codes and Cryptography*, 48(1):79–103, 2008.

[5] D. Boesten and B. Škorić. Asymptotic fingerprinting capacity for non-binary alphabets. In *Information Hiding 2011*, volume 6958 of *Lecture Notes in Computer Science*, pages 1–13. Springer, 2011.

[6] D. Boesten and B. Škorić. Asymptotic fingerprinting capacity in the Combined Digit Model. In *Information Hiding 2012*, pages 255–268. Springer, 2012. LNCS Vol. 7692.

[7] A. Charpentier, C. Fontaine, T. Furon, and I.J. Cox. An asymmetric fingerprinting scheme based on Tardos codes. In *Information Hiding 2011*, volume 6958 of *LNCS*, pages 43–58. Springer, 2011.

[8] A. Charpentier, F. Xie, C. Fontaine, and T. Furon. Expectation maximization decoding of Tardos probabilistic fingerprinting code. In *SPIE Media Forensics and Security 2009*, page 72540, 2009.

[9] C.J. Colbourn, D. Horsley, and V.R. Syrotiuk. Frameproof codes and compressive sensing. In *48th Allerton Conference on Communication, Control, and Computing*, pages 985–990, 2010.

[10] T.M. Cover and J.A. Thomas. *Elements of information theory, 2nd edition*. Wiley, 2006.

[11] R. Dorfman. The detection of defective members of large populations. *The Annals of Mathematical Statistics*, 14(4):436–440, 1943.

[12] T. Furon and M. Desoubeaux. Tardos codes for real. In *IEEE Workshop on Information Forensics and Security (WIFS) 2014*, 2014.

[13] T. Furon, A. Guyader, and F. Cérou. On the design and optimization of Tardos probabilistic fingerprinting codes. In *Information Hiding 2008*, volume 5284 of *LNCS*, pages 341–356. Springer, 2008.

[14] T. Furon, L. Pérez-Freire, A. Guyader, and F. Cérou. Estimating the minimal length of Tardos code. In *Information Hiding 2009*, volume 5806 of *LNCS*, pages 176–190, 2009.

[15] R.F. Hoskins. *Delta Functions, 2nd edition*. Woodhead Publishing, 2009.

[16] Y.-W. Huang and P. Moulin. Capacity-achieving fingerprint decoding. In *IEEE Workshop on Information Forensics and Security (WIFS) 2009*, pages 51–55, 2009.

[17] Y.-W. Huang and P. Moulin. On the saddle-point solution and the large-coalition asymptotics of fingerprinting games. *IEEE Transactions on Information Forensics and Security*, 7(1):160–175, 2012.

[18] Ye.-W. Huang and P. Moulin. On fingerprinting capacity games for arbitrary alphabets and their asymptotics. In *IEEE International Symposium on Information Theory (ISIT) 2012*, pages 2571–2575, 2012.

[19] T. Laarhoven. Efficient probabilistic group testing based on traitor tracing. In *51st Allerton Conference on Communication, Control and Computing*, pages 1458–1465, 2013.

[20] T. Laarhoven. Asymptotics of fingerprinting and group testing: Tight bounds from channel capacities. http://arxiv.org/abs/1404.2576, 2014.

[21] T. Laarhoven. Capacities and capacity-achieving decoders for various fingerprinting games. In *ACM Information Hiding and Multimedia Security Workshop (IH&MMSec) 2014*, pages 123–134, 2014.

[22] T. Laarhoven and B. de Weger. Optimal symmetric Tardos traitor tracing schemes. *Designs, Codes and Cryptography*, pages 1–21, 2012.

[23] P. Meerwald and T. Furon. Group testing meets traitor tracing. In *IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP) 2011*, pages 4204–4207, 2011.

[24] P. Meerwald and T. Furon. Towards Joint Tardos Decoding: The 'Don Quixote' Algorithm. In *Information Hiding 2011*, pages 28–42, 2011.

[25] P. Meerwald and T. Furon. Toward Practical Joint Decoding of Binary Tardos Fingerprinting Codes. *IEEE Transactions on Information Forensics and Security*, 7(4):1168–1180, 2012.

[26] P. Moulin. Universal fingerprinting: Capacity and random-coding exponents. In *Preprint arXiv:0801.3837v2*, 2008.

[27] J. Neyman and E.S. Pearson. On the problem of the most efficient tests of statistical hypotheses. *Philosophical Transactions of the Royal Society A: Mathematical, Physical and Engineering Sciences*, 231:694–706, 1933.

[28] K. Nuida. Short collusion-secure fingerprint codes against three pirates. In *Information Hiding 2010*, volume 6387 of *LNCS*, pages 86–102. Springer, 2010.

[29] K. Nuida, S. Fujitsu, M. Hagiwara, T. Kitagawa, H. Watanabe, K. Ogawa, and H. Imai. An improvement of discrete Tardos fingerprinting codes. *Designs, Codes, and Cryptography*, 52(3):339–362, 2009.

[30] J.-J. Oosterwijk, B. Škorić, and J. Doumen. Optimal suspicion functions for Tardos traitor tracing schemes. In *ACM Information Hiding and Multimedia Security Workshop (IH&MMSec) 2013*, pages 19–28, 2013.

[31] A. Simone and B. Škorić. Asymptotically false-positive-maximizing attack on non-binary Tardos codes. In *Information Hiding 2011*, pages 14–27, 2011.

[32] A. Simone and B. Škorić. Accusation probabilities in Tardos codes: beyond the Gaussian approximation. *Designs, Codes and Cryptography*, 63(3):379–412, 2012.

[33] A. Simone and B. Škorić. False Positive probabilities in q-ary Tardos codes: comparison of attacks. *Designs, Codes and Cryptography*, Feb 2014.

[34] A. Somekh-Baruch and N. Merhav. On the capacity game of private fingerprinting systems under collusion attacks. *IEEE Transactions on Information Theory*, 51(3):884–899, 2005.

[35] D.R. Stinson, T. van Trung, and R. Wei. Secure frameproof codes, key distribution patterns, group testing algorithms and related structures. *Journal of Statistical Planning and Inference*, 86(2):595–617, 2000.

[36] G. Tardos. Optimal probabilistic fingerprint codes. In *ACM Symposium on Theory of Computing (STOC) 2003*, pages 116–125, 2003.

[37] G. Tardos. Optimal probabilistic fingerprint codes. *J. ACM*, 55(2):1–24, 2008.

[38] B. Škorić, S. Katzenbeisser, and M.U. Celik. Symmetric Tardos fingerprinting codes for arbitrary alphabet sizes. *Designs, Codes and Cryptography*, 46(2):137–166, 2008.

[39] B. Škorić, S. Katzenbeisser, H.G. Schaathun, and M.U. Celik. Tardos Fingerprinting Codes in the Combined Digit Model. *IEEE Transactions on Information Forensics and Security*, 6(3):906–919, 2011.

[40] B. Škorić and J.-J. Oosterwijk. Binary and q-ary Tardos codes, revisited. *Designs, Codes, and Cryptography*, July 2013.

[41] B. Škorić, T.U. Vladimirova, M.U. Celik, and J.C. Talstra. Tardos Fingerprinting is Better Than We Thought. *IEEE Transactions on Information Theory*, 54(8):3663–3676, 2008.

[42] F. Xie, T. Furon, and C. Fontaine. On-off keying modulation and Tardos fingerprinting. In *Multimedia & Security (MM&Sec) 2008*, pages 101–106. ACM, 2008.

APPENDIX

A False Positive bound using Markov's inequality

We again look at the FP error probability in Experiment 1, but now we use Markov's inequality (Lemma 14).

Lemma 14 (Markov's inequality): Let X be a nonnegative random variable, and let $a > 0$. Then $\Pr[X \geq a] \leq a^{-1}\mathbb{E}[X]$.

Lemma 15: Let A be a (N, p)-binomial-distributed random variable. Then $\mathbb{E}\frac{1}{1+A} = \frac{1-(1-p)^{N+1}}{(N+1)p}$.

Proof: $\sum_{a=0}^{N} \frac{1}{1+a}\binom{N}{a}p^a(1-p)^{N-a} = \frac{1}{(N+1)p}\sum_{a=0}^{N}\binom{N+1}{a+1}p^{a+1}(1-p)^{N+1-(a+1)} = \frac{1}{(N+1)p}\sum_{a'=1}^{N}\binom{N+1}{a'}p^{a'}(1-p)^{N+1-a'}$. The summation consists of the full binomial sum $\sum_{a'=0}^{N}$ minus the $a' = 0$ term. ∎

Theorem 11: Let $c \leq c_0$. Let the tracer use the score function (20) and set the accusation threshold as

$$Z_1 = \frac{1}{\ell}\ln\frac{1}{\varepsilon_1} + \frac{1}{\ell}\sum_{i\in[\ell]}\ln\left[1 + \frac{n-1}{n-c_0}\cdot\frac{1-(1-p_{iy_i})^{n-c_0}}{c_0-1}\right]. \tag{64}$$

Then in Experiment 1 it holds that $P_{\mathrm{FP1}} \leq \varepsilon_1$.

Proof: For arbitrary innocent user j, we write $P_{\mathrm{FP1}} = \Pr[S_j > Z] \leq \Pr[S_j \geq Z] = \Pr[e^{\ell S_j} \geq e^{\ell Z}]$. Then we use Markov's inequality to get $\Pr[e^{\ell S_j} \geq e^{\ell Z}] \leq e^{-\ell Z}\mathbb{E}[e^{\ell S_j}]$, where the expectation is over the 'innocent' part of the matrix x. We write S_{ji} as in (35). This allows us to write $P_{\mathrm{FP1}} \leq e^{-\ell Z}\prod_i \mathbb{E}_{\boldsymbol{K}_i|\boldsymbol{p}_i}\mathbb{E}_{X_{ji}|\boldsymbol{p}_i}e^{S_{ji}}$. Next we have

$$\begin{aligned}\mathbb{E}_{X_{ji}|\boldsymbol{p}_i}e^{S_{ji}} &= (1-p_{iy_i})e^0 + p_{iy_i}(1 + \frac{n-1}{c_0-1}\cdot\frac{1}{m_{iy_i}+K_{iy_i}}) \\ &\leq 1 - p_{iy_i} + p_{iy_i}(1 + \frac{n-1}{c_0-1}\cdot\frac{1}{1+K_{iy_i}}) \\ &= 1 + p_{iy_i}\frac{n-1}{c_0-1}\cdot\frac{1}{1+K_{iy_i}}. \end{aligned}\tag{65}$$

Next we evaluate the expectation $\mathbb{E}_{\boldsymbol{K}_i|\boldsymbol{p}_i}$ using Lemma 15 where K_{iy_i} is the binomial variable and we substitute $N \to n-c-1$ and $p \to p_{iy_i}$. This yields

$$\begin{aligned}\mathbb{E}_{\boldsymbol{K}_i|\boldsymbol{p}_i}\mathbb{E}_{X_{ji}|\boldsymbol{p}_i}e^{S_{ji}} &\leq 1 + p_{iy_i}\frac{n-1}{c_0-1}\cdot\frac{1-(1-p_{iy_i})^{n-c}}{p_{iy_i}(n-c)} \\ &= 1 + \frac{n-1}{c_0-1}\cdot\frac{1-(1-p_{iy_i})^{n-c}}{n-c} \\ &\leq 1 + \frac{n-1}{c_0-1}\cdot\frac{1-(1-p_{iy_i})^{n-c_0}}{n-c_0}. \end{aligned}\tag{66}$$

In the last step we used $c \leq c_0$ and the fact that $(1–u^x)/x$, with $u \in (0,1)$, is a decreasing function of x. Thus we have established that $P_{\mathrm{FP1}} \leq e^{-\ell Z}\exp\sum_i \ln[1 + \frac{n-1}{n-c_0}\cdot\frac{1-(1-p_{iy_i})^{n-c_0}}{c_0-1}]$. Setting the threshold according to (64) achieves $P_{\mathrm{FP1}} \leq \varepsilon_1$. ∎

A more simple, $\bar{\boldsymbol{p}}$-independent, expression can be obtained if we sacrifice a little bit of tightness.

Corollary 3: Let $c \leq c_0$. Let the tracer use the score function (20) and set the accusation threshold as

$$Z_2 = \frac{1}{\ell}\ln\frac{1}{\varepsilon_1} + \ln\left[1 + \frac{n-1}{n-c_0}\cdot\frac{1}{c_0-1}\right]. \tag{67}$$

Then $P_{\mathrm{FP1}} \leq \varepsilon_1$.

Proof: In the proof of Theorem 11, at the end, we use $1 - (1-p_{iy_i})^{n-c} \leq 1$. The $\sum_i$ reduces to a factor ℓ. ∎