Tuple decoders for traitor tracing schemes

Jan-Jaap Oosterwijk[a,b], Jeroen Doumen[b], Thijs Laarhoven[a]

[a]Eindhoven University of Technology, P.O. Box 513, 5600 MB Eindhoven, The Netherlands;
[b]Irdeto B.V., P.O. Box 3047, 2130 KA Hoofddorp, The Netherlands

ABSTRACT

In the field of collusion-resistant traitor tracing, Oosterwijk et al. recently determined the optimal suspicion function for simple decoders. Earlier, Moulin also considered another type of decoder: the generic joint decoder that compares all possible coalitions, and showed that usually the generic joint decoder outperforms the simple decoder. Both Amiri and Tardos, and Meerwald and Furon described constructions that assign suspicion levels to c-tuples, where c is the number of colluders. We investigate a novel idea: the tuple decoder, assigning a suspicion level to tuples of a fixed size. In contrast to earlier work, we use this in a novel accusation algorithm to decide for each distinct user whether or not to accuse him. We expect such a scheme to outperform simple decoders while not being as computationally intensive as the generic joint decoder. In this paper we generalize the optimal suspicion functions to tuples, and describe a family of accusation algorithms in this setting that accuses individual users using this tuple-based information.

Keywords: Collusion resistance, traitor tracing.

1. INTRODUCTION

1.1 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 serial number. Once an unauthorized copy of the content is found, the identities of those users who participated in its creation can be determined. A tracing algorithm outputs a list of suspicious users.

The most powerful attacks against watermarking are *collusion attacks*, in which multiple attackers (the 'coalition') combine their differently watermarked versions of the same content; the observed differences point to the locations of the hidden marks.

In the past two decades several types of collusion-resistant codes have been developed. The most popular type in the recent literature is the class of *bias-based* codes. These were introduced[1] by Tardos in 2003. The original paper was followed by a flurry of activity, e.g. improved analyses,[2–7] code modifications,[8–10] decoder modifications[11–13] and various generalizations.[14–17] The advantage of bias-based versus deterministic codes is that they can achieve a code length ℓ as short as $\ell \propto c^2$, i.e. quadratic in the coalition size c.

Two kinds of tracing algorithms were previously considered: (i) *simple decoders*, where the decision whether to accuse a user is based only on this user's codeword, and (ii) *joint decoders*,[11–13, 18] where this decision may also depend on codewords of other users. These joint decoders generally accuse the complete coalition (a c-tuple). The more practical constructions employ a simple decoder as a bootstrapping step, and then build from there to find the complete coalition.

Tardos' scheme worked with a binary code and a simple decoder. Its 'suspicion function' for computing a level of suspicion for single users was improved[15] and the scheme was generalized to q-ary alphabets. However, it turns out[19] that the suspicion function yields sub-optimal fingerprinting rates, i.e. rather far below the fingerprinting capacity[20–22] and far below the best achieved dynamic code rate.[23, 24] When the colluder strategy

Further author information: (Send correspondence to J.O.)
J.O.: E-mail: J.Oosterwijk@tue.nl
J.D.: E-mail: JDoumen@Irdeto.com
T.L.: E-mail: mail@thijs.com

is known (or can be estimated), one can specifically optimize the suspicion functions.[26] It turns out that the optimal suspicion function against the interleaving attack asymptotically achieves capacity,[25] whatever strategy the colluders employ.

1.2 Contributions

The capacity for joint decoders is generally larger than that for simple decoders.[21] However, there are many capacities in between that have been largely unstudied. In this work we turn our attention to tuple decoders which offer a trade-off between the short code length and high complexity of the joint decoder, and the longer code length and linear complexity of the simple decoder.

Our contributions in this work are threefold:

- We generalize our earlier work[26] and optimize suspicion functions that assign suspicion levels to t-tuples. Using functional derivation methods we obtain suspicion functions that for large c maximize the expected score for the coalition, allowing the tracer to distinguish best between them and the innocent users. We present results for the Combined-Digit Model and the Restricted-Digit Model.
- We consider a set of often-considered attack strategies. We substitute these attacks into the generic formulas and obtain closed-form expressions for optimal suspicion functions associated with these attacks.
- We propose a new manner of *deploying* t-tuple suspicion functions. Traditionally, these have only been used iteratively (with the exception of c-tuple suspicion functions), in order to build up to "full" c-tuples with which one then hopes to catch the complete coalition at once. We argue that they are a more powerful tool and employ them to decide whether a specific user should be accused or not. Based on the suspicion levels of *all* t-tuples the specific user is part of, we decide whether to accuse him or not.

1.3 Outline

We start our paper by introducing our notation and a few key concepts in Section 2. In Section 3 we introduce the performance indicator(s) for a t-tuple decoder, and find the suspicion function that optimizes this for a given collusion attack under the Gaussian assumption. In Section 4 we obtain closed-form formulas for this optimal suspicion function against common attacks. In Section 5 we introduce our novel accusation algorithm. Finally, we discuss our results in Section 6.

2. PRELIMINARIES

Simple decoders attribute a level of suspicion to each user individually, based on the similarity of the user ID to that embedded in the pirated document. Very suspicious users, whose level exceeds some threshold, are accused.

A good traitor tracing system accuses the right people: the probability of falsely accusing an innocent user should stay within bounds and the probability of not accusing any guilty users as well. Setting the threshold is a delicate task: set it too low, and the probability of falsely accusing innocent users rises. Set it too high, and the probability of not accusing any guilty users rises.

Let $\mu_{t,g}$ denote the expected average level of suspicion attributed to tuples of t users of whom g are guilty and $t-g$ are innocent and let $\sigma_{t,g}^2$ be its variance. We derive, for each tuple size t and number of guilty users g in the tuple, the optimal function that assigns an level of suspicion to tuples of size t, such that the expected average level of suspicion assigned to t-tuples that consist entirely of guilty users is maximized, while keeping the average level of suspicion assigned to tuples that consist entirely of innocent users centered (with zero mean) and normalized (with unit variance).

2.1 A note on notation

We denote sets by calligraphic letters. In particular, we reserve $\mathcal{A}$ for the discrete alphabet of q symbols, sometimes indexed explicitly as $\mathcal{A} = \{0, \ldots, q-1\}$. We define $[\ell] := \{1, \ldots, \ell\}$. We will use the shorthand notation of $[g^k i^{t-k}]$ for a t-tuple consisting of k guilty users and $t-k$ innocents.

We denote tuples by letters in boldface. We stress the fact that $\boldsymbol{x}$ is a tuple (and no longer a singleton as in our previous work) by writing $\vec{\boldsymbol{x}}$. We use multi-index notation, e.g. for the q-tuple $\boldsymbol{m}$ we define the sum $|\boldsymbol{m}| = \sum_{\alpha \in \mathcal{A}} m_\alpha$ of components, and for the vector $\boldsymbol{p}$ of identical dimension we define the product $\boldsymbol{p}^{\boldsymbol{m}} := \prod_{\alpha \in \mathcal{A}} p_\alpha^{m_\alpha}$ of component-wise powers, and the multinomial coefficient $\binom{c}{\boldsymbol{m}} := c!/\prod_{\alpha \in \mathcal{A}} m_\alpha!$.

We denote random variables by capital letters and their realizations in lower case. For probability mass or density functions we use abbreviated notation of the form $f_{y|\boldsymbol{p}} := f_{Y|\boldsymbol{P}}(y|\boldsymbol{p})$. We abbreviate conditional expectations as $\mathbb{E}_{\boldsymbol{M}|\boldsymbol{p}}[\ldots] = \mathbb{E}_{\boldsymbol{M}}[\ldots|\boldsymbol{P} = \boldsymbol{p}]$. An expectation taken over *all* probabilistic degrees of freedom is written as an $\mathbb{E}[\ldots]$ without subscripts.

We use the Pochhammer symbol $(c)_g$ to denote the falling factorial $(c)_g := \prod_{k=1}^{g}(c-k+1) = c(c-1)\cdots(c-g+1)$.

We use the Kronecker delta $\delta_{x,y}$ to denote the function which evaluates to 1 when $x = y$ and to 0 when $x \neq y$.

When taking a partial derivative of a function, any dependencies among its variables are not enforced until *after* differentiation. In particular, to calculate $\frac{\partial f(y|\boldsymbol{p})}{\partial p_x}|_{|\boldsymbol{p}|=1}$, the fact that the $|\boldsymbol{p}| = 1$ is not enforced until after differentiation.

2.2 Bias-based tracing; simple decoder

The content contains ℓ abstract 'locations' into which a q-ary symbol can be embedded. For each location $i \in [\ell]$ independently, the tracer draws a bias vector $\boldsymbol{P}_i = (P_{i,\alpha})_{\alpha \in \mathcal{A}}$ from a distribution $f_{\boldsymbol{P}}$. The biases satisfy $P_{i,\alpha} \geq 0$ and $|\boldsymbol{P}_i| = 1$. A symmetric Dirichlet distribution was taken,[15] with concentration parameter $\kappa > 0$,

$$f_{\boldsymbol{p}} = \boldsymbol{p}^{\kappa-1}\Gamma(q\kappa)/[\Gamma(\kappa)]^q. \tag{1}$$

For $q = 2$ it is customary to set $\kappa = \frac{1}{2}$, turning (1) into the arcsine distribution for the component p_1. However, in that case the support has to be reduced to $p_1 \in [\delta, 1-\delta]$, with cutoff parameter $\delta > 0$, in order to avoid statistical problems due to extremely unlikely events. The probability density function then becomes

$$f_{p_1} = \frac{1}{2\arcsin(1-2\delta)} \frac{1}{\sqrt{p_1(1-p_1)}}. \tag{2}$$

As the cutoff parameter is typically chosen so small that it vanishes, we will neglect it in our analysis. The number of users is n. For each $i \in [\ell]$ and each $j \in [n]$, the tracer draws a random symbol $X_{i,j} \in \mathcal{A}$ according to the categorical distribution with parameter $\boldsymbol{P}_i$, i.e. $\mathbb{P}[X_{i,j} = \alpha | \boldsymbol{P}_i = \boldsymbol{p}_i] = p_{i,\alpha}$ independent of j. The symbol $X_{i,j}$ is embedded into the content of user j in location i.

The coalition of attackers is denoted as $\mathcal{C} \subset [n]$, with $|\mathcal{C}| = c$. In some attack models, e.g. the Combined-Digit Model (Section 2.3), they are allowed to do signal processing attacks such as introducing noise and fusing symbols. In the Restricted-Digit Model (RDM) they are only allowed to select one colluder's symbol (denoted as y_i) in location i. In the *simple decoder* approach, the tracer determines a score S_j for each user j by adding independently computed sub-scores $S_{i,j}$ for each location i; these are based on $\boldsymbol{p}_i$, $X_{i,j}$ and the colluders' output in location i. If the score exceeds a threshold, user j is accused.

Tardos[1] introduced a (simple decoder) score system for the RDM at $q = 2$ that was later[15] symmetrized and generalized to $q > 2$. The sub-scores for each location are computed using a 'suspicion function' g as $S_{i,j} = g(x_{i,j}, y_i, \boldsymbol{p}_i)$ with

$$g(x, y, \boldsymbol{p}) = \begin{cases} \sqrt{(1-p_y)/p_y} & \text{if } x = y \\ -\sqrt{p_y/(1-p_y)} & \text{if } x \neq y. \end{cases} \tag{3}$$

It has the special property that the $S_{i,j}$ of innocent users has expectation 0 and variance 1.

Given the symmetries present in the code generation and accusation algorithm, it is usually assumed that the attackers apply a strategy that acts at every location independently. Furthermore, we assume that the colluders take equal risks. In such an attack model, the colluders' decision in location i depends only on the tallies $M_{i,\alpha} = |\{j \in \mathcal{C} | X_{i,j} = \alpha\}|$ (with $\alpha \in \mathcal{A}$). The tallies satisfy $|\boldsymbol{M}_i| = c$, and they are multinomially distributed, with density $f_{\boldsymbol{m}|\boldsymbol{p}} = \binom{c}{\boldsymbol{m}}\boldsymbol{p}^{\boldsymbol{m}}$. The attack strategy may be probabilistic.

2.3 Combined-Digit Model (CDM)

The CDM[16] allows colluders to mix symbols and to introduce noise (see Figure 1). In each location, the symbols that are mixed are assumed to have equal power. The set of symbols that the colluders choose to mix is denoted as $\boldsymbol{\Psi} \subseteq \mathcal{A}$ with $m_\alpha > 0$ for each $\alpha \in \boldsymbol{\Psi}$. The attack strategy is parametrized by a set of probabilities $f_{\boldsymbol{\psi}|\boldsymbol{m}}$. The tracer has a detector that outputs a set $\boldsymbol{\Phi} \subseteq \mathcal{A}$ of observed symbols. The joint effects of the noise and the mixing lead to probability distributions $f_{\boldsymbol{\Phi}|\boldsymbol{\Psi}}$, where it is possible that the noise introduces symbols in $\boldsymbol{\Phi}$ that are absent in $\boldsymbol{\Psi}$. Simple-decoder score systems were introduced in.[16,17]

$$\begin{tikzcd}
\boldsymbol{P} \arrow[r, "\text{code generation}", "f_{\boldsymbol{M}|\boldsymbol{P}}"'] & \boldsymbol{M} \arrow[r, "\text{colluder mix}", "f_{\boldsymbol{\Psi}|\boldsymbol{M}}"'] & \boldsymbol{\Psi} \arrow[r, "\text{tracer detection}", "f_{\boldsymbol{\Phi}|\boldsymbol{\Psi}}"'] & \boldsymbol{\Phi}
\end{tikzcd}$$

Figure 1. *A schematic depiction of the CDM.*

The CDM reduces to the RDM when the noise strength is sent to zero and the detector unerringly observes $\boldsymbol{\Phi} = \boldsymbol{\Psi}$, forcing the colluders to output a single symbol, $\boldsymbol{\Psi} = \{Y\}$. For the RDM, a strategy is parametrized by a set of probabilities $f_{y|\boldsymbol{m}}$.

3. MEASURING PERFORMANCE UNDER THE GAUSSIAN ASSUMPTION

Tracing large coalitions requires relatively long codes. Since the level of suspicion of a tuple is calculated by adding (independent and identically distributed) intermediate values over all positions of the code, it generally tends to a Gaussian distribution (for a fixed tuple size t), assuming that the distribution of these intermediate values does not drastically change when c or n is increased. Under this Gaussian assumption, the distribution of user scores is characterized by just two parameters: the mean and the variance. For large coalitions, it therefore suffices to know the expectation and variance of the level of suspicion to estimate the performance of bias-based traitor tracing schemes.

3.1 Expected average suspicion of various types of tuples

A tuple of t distinct users consists either entirely of innocent users, entirely of guilty users, or is a mixture of both. Specifically, it consists of any number of guilty users between 0 and t. Given a suspicion function h_t, each of these $t+1$ types of tuples has a different expected average level of suspicion.

Let us first consider the case of t-tuples that consist entirely of innocent users. In the combined-digit model, the expected level of suspicion assigned to such tuples and its variance are simply

$$\mu_{t,0} = \mathbb{E}[h_t] = \mathbb{E}_{\boldsymbol{P}}\mathbb{E}_{\boldsymbol{M}|\boldsymbol{P}}\mathbb{E}_{\boldsymbol{\Phi}|\boldsymbol{M}}\mathbb{E}_{\vec{\boldsymbol{X}}|\boldsymbol{P}}[h_t(\vec{\boldsymbol{X}}, \boldsymbol{\Phi}, \boldsymbol{P})] = \mathbb{E}_{\boldsymbol{P}}\mathbb{E}_{\boldsymbol{\Phi}|\boldsymbol{P}}\mathbb{E}_{\vec{\boldsymbol{X}}|\boldsymbol{P}}[h_t(\vec{\boldsymbol{X}}, \boldsymbol{\Phi}, \boldsymbol{P})]; \tag{4}$$

$$\sigma_{t,0}^2 = \mathbb{E}[(h_t - \mu_{t,0})^2] = \mathbb{E}[h_t^2] - \mu_{t,0}^2 = \mathbb{E}_{\boldsymbol{P}}\mathbb{E}_{\boldsymbol{\Phi}|\boldsymbol{P}}\mathbb{E}_{\vec{\boldsymbol{X}}|\boldsymbol{P}}[h_t^2(\vec{\boldsymbol{X}}, \boldsymbol{\Phi}, \boldsymbol{P})] - \mu_{t,0}^2. \tag{5}$$

Next, we consider the other extreme: t-tuples that consist entirely of guilty users. There are $(c)_t$ such t-tuples. In any specific position where symbols were distributed according to bias vector $\boldsymbol{p} = (p_x)_{x\in\mathcal{A}}$, where the coalition received symbols according to tally vector $\boldsymbol{m} = (m_x)_{x\in\mathcal{A}}$, and where the detector recognized symbols according to vector $\boldsymbol{\phi} = (\phi_x)_{x\in\mathcal{A}}$, the level of suspicion $h_t(\vec{\boldsymbol{x}}, \boldsymbol{\phi}, \boldsymbol{p})$ is assigned to $\prod_{k=1}^{t}\left(m_{x_k} - \sum_{j<k}\delta_{x_j,x_k}\right)$ distinct tuples. If a guilty user has been selected to contribute the first symbol x_1 to the tuple, then there are not m_{x_1} but

$m_{x_1} - 1$ *other* guilty users left with that same symbol, and so on. The average level of suspicion assigned to t-tuples consisting entirely of guilty users is thus

$$\frac{1}{(c)_t} \sum_{\vec{\boldsymbol{x}} \in \mathcal{A}^t} \left(\prod_{k=1}^{t} \Big(m_{x_k} - \sum_{j<k} \delta_{x_j, x_k} \Big) \right) h_t(\vec{\boldsymbol{x}}, \boldsymbol{\phi}, \boldsymbol{p}) \tag{6}$$

and the expected average

$$\mu_{t,t} = \mathbb{E}_{\boldsymbol{P}} \mathbb{E}_{\boldsymbol{M}|\boldsymbol{P}} \mathbb{E}_{\boldsymbol{\Phi}|\boldsymbol{M}} \left[\frac{1}{(c)_t} \sum_{\vec{\boldsymbol{x}} \in \mathcal{A}^t} \left(\prod_{k=1}^{t} \Big(M_{x_k} - \sum_{j<k} \delta_{x_j, x_k} \Big) \right) h_t(\vec{\boldsymbol{x}}, \boldsymbol{\Phi}, \boldsymbol{P}) \right]. \tag{7}$$

Finally, we consider the intermediate cases of t-tuples that consist of g guilty and $t - g$ innocent users, with $0 < g < t$. As long as the suspicion function h_t is symmetric (invariant under permutations of $\vec{\boldsymbol{x}}$), the expected average $\mu_{t,g}$ taken over the unordered symbol-tuple $\boldsymbol{x}$ is equal to that taken over tuples with some specific ordering. We consider ordered tuples in which the guilty users are placed in front and the innocents at the end. Assume for a moment that the $t - g$ innocent users have already been selected. Then there are $(c)_g$ ways (permutations) to select the remaining g guilty users from the coalition of c colluders. The expected average level of suspicion assigned to t-tuples that consist of g guilty and $t - g$ innocent users is thus

$$\begin{aligned} \mu_{t,g} = {} & \mathbb{E}_{\boldsymbol{P}} \mathbb{E}_{\boldsymbol{M}|\boldsymbol{P}} \mathbb{E}_{\boldsymbol{\Phi}|\boldsymbol{M}} \mathbb{E}_{X_{g+1}|\boldsymbol{P}} \cdots \mathbb{E}_{X_t|\boldsymbol{P}} \\ & \left[\frac{1}{(c)_g} \sum_{x_1 \in \mathcal{A}} \cdots \sum_{x_g \in \mathcal{A}} \left(\prod_{k=1}^{g} \Big(M_{x_k} - \sum_{j<k} \delta_{x_j, x_k} \Big) \right) h_t((x_1, \ldots, x_g, X_{g+1}, \ldots, X_t), \boldsymbol{\Phi}, \boldsymbol{P}) \right]. \end{aligned} \tag{8}$$

3.2 Standardized expressions for the expected averages

Each type of tuple has a different expected average level $\mu_{t,g}$ of suspicion. In particular, the expectations are of different forms: all are taken over the bias vector $\boldsymbol{P}$ and the detection vector $\boldsymbol{\Phi}$, but where guilty users are involved ($g > 0$), the expectation is also taken over the tally vector $\boldsymbol{M}$. Moreover, for each of the $t - g$ innocent users involved, the expectation is also taken over the symbols $X_{g+1}, \ldots, X_t$ received by those users.

In this subsection we standardize the form of the expected averages by rewriting the expectation $\mu_{t,g}$ to the same form of that of $\mu_{t,0}$. In the next subsection, where we phrase our objective mathematically as a constrained functional optimization, this will allow us to apply the method of Lagrange multipliers to instantly derive the suspicion function $\hat{h}_{t,g}$ that maximizes the expected average $\mu_{t,g}$ for t-tuples with g distinct guilty users in front, while keeping the level of suspicion assigned to t-tuples that consist entirely of innocent users centered ($\mu_{t,0} = 0$) and normalized ($\sigma_{t,0}^2 = 1$).

3.2.1 The combined-digit model

In the combined-digit model, the random variables involved are the vector $\boldsymbol{P}$ of biases, the tuple $\boldsymbol{M}$ of tallies, the tuple $\boldsymbol{\Psi}$ of symbols output by the coalition, the tuple $\boldsymbol{\Phi}$ of detected symbols and the tuple $\vec{\boldsymbol{X}}$ of symbols received by the t-tuple of users. The most general suspicion function we study is therefore of the form $h_t(\vec{\boldsymbol{x}}, \boldsymbol{\phi}, \boldsymbol{\psi}, \boldsymbol{m}, \boldsymbol{p})$, a function of the realizations of all random variables involved.

Lemma 3.1. *An optimal suspicion function of the form $h_t(\vec{\boldsymbol{x}}, \boldsymbol{\phi}, \boldsymbol{\psi}, \boldsymbol{p})$ does not depend on $\boldsymbol{\phi}$. An optimal suspicion function of the form $h_t(\vec{\boldsymbol{x}}, \boldsymbol{\phi}, \boldsymbol{\psi}, \boldsymbol{m}, \boldsymbol{p})$ depends neither on $\boldsymbol{\phi}$ nor $\boldsymbol{\psi}$.*

Proof. The set $\boldsymbol{\psi}$ contains more information about the attacks than the set $\boldsymbol{\phi}$. Likewise, the tallies $\boldsymbol{m}$ contain more information than $\boldsymbol{\psi}$. □

To determine the optimal suspicion functions of the increasingly general form $h_t(\vec{\boldsymbol{x}}, \boldsymbol{\phi}, \boldsymbol{p})$, $h_t(\vec{\boldsymbol{x}}, \boldsymbol{\phi}, \boldsymbol{\psi}, \boldsymbol{p})$, and $h_t(\vec{\boldsymbol{x}}, \boldsymbol{\phi}, \boldsymbol{\psi}, \boldsymbol{m}, \boldsymbol{p})$, it suffices to study the forms $h_t(\vec{\boldsymbol{x}}, \boldsymbol{\phi}, \boldsymbol{p})$, $h_t(\vec{\boldsymbol{x}}, \boldsymbol{\psi}, \boldsymbol{p})$, and $h_t(\vec{\boldsymbol{x}}, \boldsymbol{m}, \boldsymbol{p})$, respectively. Since the last form $h_t(\vec{\boldsymbol{x}}, \boldsymbol{m}, \boldsymbol{p})$ no longer depends on the collusion attack (and the digit model, for that matter), we postpone its discussion to a separate subsection on what we call the Tally Model.

Lemma 3.2. *Let the suspicion function h_t be of the form $h_t(\vec{\boldsymbol{x}}, \boldsymbol{\phi}, \boldsymbol{p})$. Then the expected average suspicion $\mu_{t,g}$ can be expressed as a weighted expectation of h_t,*

$$\mu_{t,g} = \mathbb{E}[w_g \cdot h_t] \tag{9}$$

with weight function

$$w_g((x_1, \dots, x_g), \boldsymbol{\phi}, \boldsymbol{p}) := \frac{\mathbb{E}_{\boldsymbol{M}|\boldsymbol{p}}\left[f_{\boldsymbol{\phi}|\boldsymbol{M}} \prod_{k=1}^{g}\Big(M_{x_k} - \sum_{j<k} \delta_{x_j,x_k}\Big)\right]}{f_{\boldsymbol{\phi}|\boldsymbol{p}} \cdot (c)_g \cdot p_{x_1} \cdots p_{x_g}} = \frac{1}{f_{\boldsymbol{\phi}|\boldsymbol{p}} \cdot (c)_g} \left.\frac{\partial^g(|\boldsymbol{p}|^c f_{\boldsymbol{\phi}|\boldsymbol{p}})}{\partial p_{x_1} \cdots \partial p_{x_g}}\right|_{|\boldsymbol{p}|=1} \tag{10}$$

that has unit mean

$$\mathbb{E}_{X_1|\boldsymbol{p}} \cdots \mathbb{E}_{X_g|\boldsymbol{p}}[w_g((X_1, \dots, X_g), \boldsymbol{\phi}, \boldsymbol{p})] = \mathbb{E}_{\boldsymbol{\Phi}|\boldsymbol{p}}[w_g((x_1, \dots, x_g), \boldsymbol{\Phi}, \boldsymbol{p})] = 1. \tag{11}$$

Proof. We rewrite the expected average (8) as a weighted expectation of h_t:

$$\begin{aligned}
\mu_{t,g} &= \mathbb{E}_{\boldsymbol{P}}\mathbb{E}_{\boldsymbol{M}|\boldsymbol{P}}\mathbb{E}_{\boldsymbol{\Phi}|\boldsymbol{M}}\mathbb{E}_{X_{g+1}|\boldsymbol{P}} \cdots \mathbb{E}_{X_t|\boldsymbol{P}} \\
&\qquad \left[\frac{1}{(c)_g} \sum_{x_1 \in \mathcal{A}} \cdots \sum_{x_g \in \mathcal{A}} \left(\prod_{k=1}^{g}\Big(M_{x_k} - \sum_{j<k} \delta_{x_j,x_k}\Big)\right) h_t((x_1, \dots, x_g, X_{g+1}, \dots, X_t), \boldsymbol{\Phi}, \boldsymbol{P})\right] && (12) \\
&= \mathbb{E}_{\boldsymbol{P}}\mathbb{E}_{\boldsymbol{M}|\boldsymbol{P}}\mathbb{E}_{\boldsymbol{\Phi}|\boldsymbol{P}}\left[\frac{f_{\boldsymbol{\Phi}|\boldsymbol{M}}}{f_{\boldsymbol{\Phi}|\boldsymbol{P}}}\mathbb{E}_{X_{g+1}|\boldsymbol{P}} \cdots \mathbb{E}_{X_t|\boldsymbol{P}}\right. \\
&\qquad \left.\left[\frac{1}{(c)_g}\mathbb{E}_{X_1|\boldsymbol{P}} \cdots \mathbb{E}_{X_g|\boldsymbol{P}}\left[\frac{\prod_{k=1}^{g}(M_{X_k} - \sum_{j<k} \delta_{X_j,X_k})}{P_{X_1} \cdots P_{X_g}} h_t(\vec{\boldsymbol{X}}, \boldsymbol{\Phi}, \boldsymbol{P})\right]\right]\right] && (13) \\
&= \mathbb{E}_{\boldsymbol{P}}\mathbb{E}_{\boldsymbol{\Phi}|\boldsymbol{P}}\mathbb{E}_{\vec{\boldsymbol{X}}|\boldsymbol{P}}\left[\frac{\mathbb{E}_{\boldsymbol{M}|\boldsymbol{P}}\left[f_{\boldsymbol{\Phi}|\boldsymbol{M}} \prod_{k=1}^{g}\Big(M_{X_k} - \sum_{j<k} \delta_{X_j,X_k}\Big)\right]}{f_{\boldsymbol{\Phi}|\boldsymbol{P}} \cdot (c)_g \cdot P_{X_1} \cdots P_{X_g}} h_t(\vec{\boldsymbol{X}}, \boldsymbol{\Phi}, \boldsymbol{P})\right] && (14) \\
&= \mathbb{E}[w_g \cdot h_t]. && (15)
\end{aligned}$$

When taking the partial derivative $\frac{\partial^g(|\boldsymbol{p}|^c f_{\boldsymbol{\phi}|\boldsymbol{p}})}{\partial p_{x_1} \cdots \partial p_{x_g}}$, the fact that $|\boldsymbol{p}| = 1$ is not enforced until after differentiation. Since

$$f_{\boldsymbol{\phi}|\boldsymbol{p}} = \mathbb{E}_{\boldsymbol{M}|\boldsymbol{p}}[f_{\boldsymbol{\phi}|\boldsymbol{M}}] = \frac{1}{|\boldsymbol{p}|^c} \sum_{\boldsymbol{m}} \binom{c}{\boldsymbol{m}} \boldsymbol{p}^{\boldsymbol{m}} f_{\boldsymbol{\phi}|\boldsymbol{m}}, \tag{16}$$

we find that

$$\begin{aligned}
\frac{\partial^g(|\boldsymbol{p}|^c f_{\boldsymbol{\phi}|\boldsymbol{p}})}{\partial p_{x_1} \cdots \partial p_{x_g}} &= \sum_{\boldsymbol{m}} \binom{c}{\boldsymbol{m}} \left(\prod_{k=1}^{g} \frac{m_{x_k} - \sum_{j<k} \delta_{x_j,x_k}}{p_{x_k}}\right) \boldsymbol{p}^{\boldsymbol{m}} f_{\boldsymbol{\phi}|\boldsymbol{m}} && (17) \\
&= \frac{|\boldsymbol{p}|^c \cdot \mathbb{E}_{\boldsymbol{M}|\boldsymbol{p}}\left[f_{\boldsymbol{\phi}|\boldsymbol{M}} \prod_{k=1}^{g}\Big(M_{x_k} - \sum_{j<k} \delta_{x_j,x_k}\Big)\right]}{p_{x_1} \cdots p_{x_g}} && (18)
\end{aligned}$$

so

$$w_g((x_1, \dots, x_g), \boldsymbol{\phi}, \boldsymbol{p}) := \frac{\mathbb{E}_{\boldsymbol{M}|\boldsymbol{p}}\left[f_{\boldsymbol{\phi}|\boldsymbol{M}} \prod_{k=1}^{g}\Big(M_{x_k} - \sum_{j<k} \delta_{x_j,x_k}\Big)\right]}{f_{\boldsymbol{\phi}|\boldsymbol{p}} \cdot (c)_g \cdot p_{x_1} \cdots p_{x_g}} = \frac{1}{f_{\boldsymbol{\phi}|\boldsymbol{p}} \cdot (c)_g} \left.\frac{\partial^g(|\boldsymbol{p}|^c f_{\boldsymbol{\phi}|\boldsymbol{p}})}{\partial p_{x_1} \cdots \partial p_{x_g}}\right|_{|\boldsymbol{p}|=1}. \tag{19}$$

The weight function has unit expectation, both taken over the symbols

$$\begin{aligned}
&\mathbb{E}_{X_1|\boldsymbol{p}}\cdots\mathbb{E}_{X_g|\boldsymbol{p}}[w_g((X_1,\ldots,X_g),\boldsymbol{\phi},\boldsymbol{p})]\\
&= \frac{1}{f_{\boldsymbol{\phi}|\boldsymbol{p}}\cdot(c)_g}\mathbb{E}_{\boldsymbol{M}|\boldsymbol{p}}\left[f_{\boldsymbol{\phi}|\boldsymbol{M}}\mathbb{E}_{X_1|\boldsymbol{p}}\cdots\mathbb{E}_{X_g|\boldsymbol{p}}\left[\frac{\prod_{k=1}^{g}\left(M_{X_k}-\sum_{j<k}\delta_{X_j,X_k}\right)}{p_{X_1}\cdots p_{X_g}}\right]\right] && (20)\\
&= \frac{1}{f_{\boldsymbol{\phi}|\boldsymbol{p}}\cdot(c)_g}\mathbb{E}_{\boldsymbol{M}|\boldsymbol{p}}\left[f_{\boldsymbol{\phi}|\boldsymbol{M}}\sum_{x_1\in\mathcal{A}}\cdots\sum_{x_g\in\mathcal{A}}\prod_{k=1}^{g}\left(M_{x_k}-\sum_{j<k}\delta_{x_j,x_k}\right)\right] && (21)\\
&= \frac{1}{f_{\boldsymbol{\phi}|\boldsymbol{p}}\cdot(c)_g}\mathbb{E}_{\boldsymbol{M}|\boldsymbol{p}}\left[f_{\boldsymbol{\phi}|\boldsymbol{M}}\sum_{x_1\in\mathcal{A}}\cdots\sum_{x_{g-1}\in\mathcal{A}}\left(\prod_{k=1}^{g-1}\left(M_{x_k}-\sum_{j<k}\delta_{x_j,x_k}\right)\right)\sum_{x_g\in\mathcal{A}}\left(M_{x_g}-\sum_{j<g}\delta_{x_j,x_k}\right)\right] && (22)\\
&= \frac{1}{f_{\boldsymbol{\phi}|\boldsymbol{p}}\cdot(c)_g}\mathbb{E}_{\boldsymbol{M}|\boldsymbol{p}}\left[f_{\boldsymbol{\phi}|\boldsymbol{M}}\sum_{x_1\in\mathcal{A}}\cdots\sum_{x_{g-1}\in\mathcal{A}}\left(\prod_{k=1}^{g-1}\left(M_{x_k}-\sum_{j<k}\delta_{x_j,x_k}\right)\right)(c-g+1)\right] && (23)\\
&= \frac{1}{f_{\boldsymbol{\phi}|\boldsymbol{p}}\cdot(c)_{g-1}}\mathbb{E}_{\boldsymbol{M}|\boldsymbol{p}}\left[f_{\boldsymbol{\phi}|\boldsymbol{M}}\sum_{x_1\in\mathcal{A}}\cdots\sum_{x_{g-1}\in\mathcal{A}}\prod_{k=1}^{g-1}\left(M_{x_k}-\sum_{j<k}\delta_{x_j,x_k}\right)\right] && (24)\\
&= \frac{1}{f_{\boldsymbol{\phi}|\boldsymbol{p}}}\mathbb{E}_{\boldsymbol{M}|\boldsymbol{p}}\left[f_{\boldsymbol{\phi}|\boldsymbol{M}}\right]=1 && (25)
\end{aligned}$$

as well as over the detected symbol

$$\begin{aligned}
\mathbb{E}_{\boldsymbol{\Phi}|\boldsymbol{p}}[w_g((x_1,\ldots,x_g),\boldsymbol{\Phi},\boldsymbol{p})] &= \sum_{\boldsymbol{\phi}} f_{\boldsymbol{\phi}|\boldsymbol{p}}\frac{\mathbb{E}_{\boldsymbol{M}|\boldsymbol{p}}\left[f_{\boldsymbol{\phi}|\boldsymbol{M}}\prod_{k=1}^{g}\left(M_{x_k}-\sum_{j<k}\delta_{x_j,x_k}\right)\right]}{f_{\boldsymbol{\phi}|\boldsymbol{p}}\cdot(c)_g\cdot p_{x_1}\cdots p_{x_g}} && (26)\\
&= \sum_{\boldsymbol{\phi}}\frac{\mathbb{E}_{\boldsymbol{M}|\boldsymbol{p}}\left[f_{\boldsymbol{\phi}|\boldsymbol{M}}\prod_{k=1}^{g}\left(M_{x_k}-\sum_{j<k}\delta_{x_j,x_k}\right)\right]}{(c)_g\cdot p_{x_1}\cdots p_{x_g}} && (27)\\
&= \frac{\mathbb{E}_{\boldsymbol{M}|\boldsymbol{p}}\left[\prod_{k=1}^{g}\left(M_{x_k}-\sum_{j<k}\delta_{x_j,x_k}\right)\right]}{(c)_g\cdot p_{x_1}\cdots p_{x_g}} && (28)\\
&= \frac{\sum_{\boldsymbol{m}}\binom{c}{\boldsymbol{m}}\boldsymbol{p}^{\boldsymbol{m}}\prod_{k=1}^{g}\left(m_{x_k}-\sum_{j<k}\delta_{x_j,x_k}\right)}{(c)_g\cdot p_{x_1}\cdots p_{x_g}} && (29)\\
&= \frac{1}{(c)_g}\sum_{\boldsymbol{m}}\binom{c}{\boldsymbol{m}}\boldsymbol{p}^{\boldsymbol{m}}\prod_{k=1}^{g}\frac{m_{x_k}-\sum_{j<k}\delta_{x_j,x_k}}{p_{x_k}} && (30)\\
&= \frac{1}{(c)_g}\;\frac{\partial^g}{\partial p_{x_1}\cdots\partial p_{x_g}}\sum_{\boldsymbol{m}}\binom{c}{\boldsymbol{m}}\boldsymbol{p}^{\boldsymbol{m}}\bigg|_{|\boldsymbol{p}|=1} && (31)\\
&= \frac{1}{(c)_g}\;\frac{\partial^g|\boldsymbol{p}|^c}{\partial p_{x_1}\cdots\partial p_{x_g}}\bigg|_{|\boldsymbol{p}|=1} = \frac{(c)_g|\boldsymbol{p}|^{c-g}}{(c)_g}\bigg|_{|\boldsymbol{p}|=1} = 1. && (32)
\end{aligned}$$

□

When the colluders' output ψ is known, we have the following result:

LEMMA 3.3. *Let h_t be of the form $h_t(\vec{\boldsymbol{x}},\boldsymbol{\psi},\boldsymbol{p})$. Then the expected average suspicion $\mu_{t,g}$ can be expressed as a weighted expectation of h_t,*

$$\mu_{t,g} = \mathbb{E}[w_g\cdot h_t] \tag{33}$$

with weight function

$$w_g((x_1,\ldots,x_g),\boldsymbol{\psi},\boldsymbol{p}) := \frac{\mathbb{E}_{\boldsymbol{M}|\boldsymbol{p}}\left[f_{\boldsymbol{\psi}|\boldsymbol{M}}\prod_{k=1}^{g}\Big(M_{x_k}-\sum_{j<k}\delta_{x_j,x_k}\Big)\right]}{f_{\boldsymbol{\psi}|\boldsymbol{p}}\cdot(c)_g\cdot p_{x_1}\cdots p_{x_g}} = \frac{1}{f_{\boldsymbol{\psi}|\boldsymbol{p}}\cdot(c)_g}\left.\frac{\partial^g(|\boldsymbol{p}|^c f_{\boldsymbol{\psi}|\boldsymbol{p}})}{\partial p_{x_1}\cdots\partial p_{x_g}}\right|_{|\boldsymbol{p}|=1} \tag{34}$$

that has unit mean

$$\mathbb{E}_{X_1|\boldsymbol{p}}\cdots\mathbb{E}_{X_g|\boldsymbol{p}}[w_g((X_1,\ldots,X_g),\boldsymbol{\psi},\boldsymbol{p})] = \mathbb{E}_{\boldsymbol{\Psi}|\boldsymbol{p}}[w_g((x_1,\ldots,x_g),\boldsymbol{\Psi},\boldsymbol{p})] = 1. \tag{35}$$

Proof. For suspicion functions h_t of the form $h_t(\vec{\boldsymbol{x}},\boldsymbol{\psi},\boldsymbol{p})$, the expected average is

$$\begin{aligned}\mu_{t,g} = {} & \mathbb{E}_{\boldsymbol{P}}\mathbb{E}_{\boldsymbol{M}|\boldsymbol{P}}\mathbb{E}_{\boldsymbol{\Psi}|\boldsymbol{M}}\mathbb{E}_{X_{g+1}|\boldsymbol{P}}\cdots\mathbb{E}_{X_t|\boldsymbol{P}} \\ & \left[\frac{1}{(c)_g}\sum_{x_1\in\mathcal{A}}\cdots\sum_{x_g\in\mathcal{A}}\left(\prod_{k=1}^{g}\Big(M_{x_k}-\sum_{j<k}\delta_{x_j,x_k}\Big)\right)h_t((x_1,\ldots,x_g,X_{g+1},\ldots,X_t),\boldsymbol{\Psi},\boldsymbol{P})\right].\end{aligned} \tag{36}$$

Note the similarity between equations (36) and (12). The proof proceeds analogously with $\boldsymbol{\Psi}$ instead of $\boldsymbol{\Phi}$. □

3.2.2 The restricted-digit model

The restricted-digit model is a special case of the combined-digit model.

Corollary 1. *Let h_t be of the form $h_t(\vec{\boldsymbol{x}},y,\boldsymbol{p})$. Then the expected average suspicion $\mu_{t,g}$ can be expressed as a weighted expectation of h_t,*

$$\mu_{t,g} = \mathbb{E}[w_g\cdot h_t] \tag{37}$$

with weight function

$$w_g((x_1,\ldots,x_g),y,\boldsymbol{p}) := \frac{\mathbb{E}_{\boldsymbol{M}|\boldsymbol{p}}\left[f_{y|\boldsymbol{M}}\prod_{k=1}^{g}\Big(M_{x_k}-\sum_{j<k}\delta_{x_j,x_k}\Big)\right]}{f_{y|\boldsymbol{p}}\cdot(c)_g\cdot p_{x_1}\cdots p_{x_g}} = \frac{1}{f_{y|\boldsymbol{p}}\cdot(c)_g}\left.\frac{\partial^g(|\boldsymbol{p}|^c f_{y|\boldsymbol{p}})}{\partial p_{x_1}\cdots\partial p_{x_g}}\right|_{|\boldsymbol{p}|=1} \tag{38}$$

that has unit mean

$$\mathbb{E}_{X_1|\boldsymbol{p}}\cdots\mathbb{E}_{X_g|\boldsymbol{p}}[w_g((X_1,\ldots,X_g),y,\boldsymbol{p})] = \mathbb{E}_{Y|\boldsymbol{p}}[w_g((x_1,\ldots,x_g),Y,\boldsymbol{p})] = 1. \tag{39}$$

Proof. Follows directly from Lemma 3.3 when reducing the tuple $\boldsymbol{\psi}$ of symbols output by the coalition to the singleton $\boldsymbol{\psi}=(y)$. □

3.2.3 The tally model

When even the tallies $\boldsymbol{m}$ of the symbols received by the coalition are known, we have the following result:

Lemma 3.4. *Let h_t be of the form $h_t(\vec{\boldsymbol{x}},\boldsymbol{m},\boldsymbol{p})$. Then the expected average suspicion $\mu_{t,g}$ can be expressed as a weighted expectation of h_t,*

$$\mu_{t,g} = \mathbb{E}[w_g\cdot h_t] \tag{40}$$

with weight function

$$\begin{aligned} w_g((x_1,\ldots,x_g),\boldsymbol{m},\boldsymbol{p}) := \frac{\prod_{k=1}^{g}\Big(m_{x_k}-\sum_{j<k}\delta_{x_j,x_k}\Big)}{(c)_g\cdot p_{x_1}\cdots p_{x_g}} &= \frac{1}{f_{\boldsymbol{m}|\boldsymbol{p}}\cdot(c)_g}\left.\frac{\partial^g(|\boldsymbol{p}|^c f_{\boldsymbol{m}|\boldsymbol{p}})}{\partial p_{x_1}\cdots\partial p_{x_g}}\right|_{|\boldsymbol{p}|=1} & (41)\\ &= \frac{1}{\boldsymbol{p}^{\boldsymbol{m}}\cdot(c)_g}\left.\frac{\partial^g\boldsymbol{p}^{\boldsymbol{m}}}{\partial p_{x_1}\cdots\partial p_{x_g}}\right|_{|\boldsymbol{p}|=1} & (42)\end{aligned}$$

that has unit mean

$$\mathbb{E}_{X_1|\boldsymbol{p}}\cdots\mathbb{E}_{X_g|\boldsymbol{p}}[w_g((X_1,\ldots,X_g),\boldsymbol{m},\boldsymbol{p})] = \mathbb{E}_{\boldsymbol{M}|\boldsymbol{p}}[w_g((x_1,\ldots,x_g),\boldsymbol{M},\boldsymbol{p})] = 1. \tag{43}$$

Proof. For suspicion functions h_t of the form $h_t(\vec{\boldsymbol{x}},\boldsymbol{m},\boldsymbol{p})$, the expected average is

$$\begin{aligned}\mu_{t,g} = {} & \mathbb{E}_{\boldsymbol{P}}\mathbb{E}_{\boldsymbol{M}|\boldsymbol{P}}\mathbb{E}_{X_{g+1}|\boldsymbol{P}}\cdots\mathbb{E}_{X_t|\boldsymbol{P}} \\ & \left[\frac{1}{(c)_g}\sum_{x_1\in\mathcal{A}}\cdots\sum_{x_g\in\mathcal{A}}\left(\prod_{k=1}^{g}\left(M_{x_k}-\sum_{j<k}\delta_{x_j,x_k}\right)\right)h_t((x_1,\ldots,x_g,X_{g+1},\ldots,X_t),\boldsymbol{M},\boldsymbol{P})\right].\end{aligned} \tag{44}$$

We rewrite it as a weighted expectation of h_t

$$\begin{aligned}\mu_{t,g} &= \mathbb{E}_{\boldsymbol{P}}\mathbb{E}_{\boldsymbol{M}|\boldsymbol{P}}\mathbb{E}_{X_{g+1}|\boldsymbol{P}}\cdots\mathbb{E}_{X_t|\boldsymbol{P}}\left[\frac{1}{(c)_g}\mathbb{E}_{X_1|\boldsymbol{P}}\cdots\mathbb{E}_{X_g|\boldsymbol{P}}\left[\frac{\prod_{k=1}^{g}\left(M_{X_k}-\sum_{j<k}\delta_{X_j,X_k}\right)}{P_{X_1}\cdots P_{X_g}}h_t(\vec{\boldsymbol{X}},\boldsymbol{M},\boldsymbol{P})\right]\right] && (45)\\ &= \mathbb{E}_{\boldsymbol{P}}\mathbb{E}_{\boldsymbol{M}|\boldsymbol{P}}\mathbb{E}_{\vec{\boldsymbol{X}}|\boldsymbol{P}}\left[\frac{\prod_{k=1}^{g}\left(M_{X_k}-\sum_{j<k}\delta_{X_j,X_k}\right)}{(c)_g P_{X_1}\cdots P_{X_g}}h_t(\vec{\boldsymbol{X}},\boldsymbol{M},\boldsymbol{P})\right] && (46)\\ &= \mathbb{E}[w_g\cdot h_t]. && (47)\end{aligned}$$

Since

$$f_{\boldsymbol{m}|\boldsymbol{p}} = \frac{1}{|\boldsymbol{p}|^c}\binom{c}{\boldsymbol{m}}\boldsymbol{p}^{\boldsymbol{m}}, \tag{48}$$

we find that

$$\left.\frac{\partial^g(|\boldsymbol{p}|^c f_{\boldsymbol{m}|\boldsymbol{p}})}{\partial p_{x_1}\cdots\partial p_{x_g}}\right|_{|\boldsymbol{p}|=1} = \binom{c}{\boldsymbol{m}}\left(\prod_{k=1}^{g}\frac{m_{x_k}-\sum_{j<k}\delta_{x_j,x_k}}{p_{x_k}}\right)\boldsymbol{p}^{\boldsymbol{m}} = \frac{|\boldsymbol{p}|^c\cdot f_{\boldsymbol{m}|\boldsymbol{p}}\prod_{k=1}^{g}\left(m_{x_k}-\sum_{j<k}\delta_{x_j,x_k}\right)}{p_{x_1}\cdots p_{x_g}} \tag{49}$$

and

$$\left.\frac{\partial^g\boldsymbol{p}^{\boldsymbol{m}}}{\partial p_{x_1}\cdots\partial p_{x_g}}\right|_{|\boldsymbol{p}|=1} = \left(\prod_{k=1}^{g}\frac{m_{x_k}-\sum_{j<k}\delta_{x_j,x_k}}{p_{x_k}}\right)\boldsymbol{p}^{\boldsymbol{m}} = \frac{\boldsymbol{p}^{\boldsymbol{m}}\prod_{k=1}^{g}\left(m_{x_k}-\sum_{j<k}\delta_{x_j,x_k}\right)}{p_{x_1}\cdots p_{x_g}} \tag{50}$$

so

$$\begin{aligned}w_g((x_1,\ldots,x_g),\boldsymbol{m},\boldsymbol{p}) := \frac{\prod_{k=1}^{g}\left(m_{x_k}-\sum_{j<k}\delta_{x_j,x_k}\right)}{(c)_g\cdot p_{x_1}\cdots p_{x_g}} &= \frac{1}{f_{\boldsymbol{m}|\boldsymbol{p}}\cdot(c)_g}\left.\frac{\partial^g(|\boldsymbol{p}|^c f_{\boldsymbol{m}|\boldsymbol{p}})}{\partial p_{x_1}\cdots\partial p_{x_g}}\right|_{|\boldsymbol{p}|=1} && (51)\\ &= \frac{1}{\boldsymbol{p}^{\boldsymbol{m}}\cdot(c)_g}\left.\frac{\partial^g\boldsymbol{p}^{\boldsymbol{m}}}{\partial p_{x_1}\cdots\partial p_{x_g}}\right|_{|\boldsymbol{p}|=1}. && (52)\end{aligned}$$

The weight function has unit expectation, both taken over the symbols

$$\begin{align}
&\mathbb{E}_{X_1|\boldsymbol{p}}\cdots\mathbb{E}_{X_g|\boldsymbol{p}}[w_g((X_1,\ldots,X_g),\boldsymbol{m},\boldsymbol{p})] \notag\\
&= \frac{1}{(c)_g}\mathbb{E}_{X_1|\boldsymbol{p}}\cdots\mathbb{E}_{X_g|\boldsymbol{p}}\left[\frac{\prod_{k=1}^{g}\left(M_{X_k}-\sum_{j<k}\delta_{X_j,X_k}\right)}{p_{X_1}\cdots p_{X_g}}\right] \tag{53}\\
&= \frac{1}{(c)_g}\sum_{x_1\in\mathcal{A}}\cdots\sum_{x_g\in\mathcal{A}}\prod_{k=1}^{g}\left(M_{x_k}-\sum_{j<k}\delta_{x_j,x_k}\right) \tag{54}\\
&= \frac{1}{(c)_g}\sum_{x_1\in\mathcal{A}}\cdots\sum_{x_{g-1}\in\mathcal{A}}\left(\prod_{k=1}^{g-1}\left(M_{x_k}-\sum_{j<k}\delta_{x_j,x_k}\right)\right)\sum_{x_g\in\mathcal{A}}\left(M_{x_g}-\sum_{j<g}\delta_{x_j,x_k}\right) \tag{55}\\
&= \frac{1}{(c)_g}\sum_{x_1\in\mathcal{A}}\cdots\sum_{x_{g-1}\in\mathcal{A}}\left(\prod_{k=1}^{g-1}\left(M_{x_k}-\sum_{j<k}\delta_{x_j,x_k}\right)\right)(c-g+1) \tag{56}\\
&= \frac{1}{(c)_{g-1}}\sum_{x_1\in\mathcal{A}}\cdots\sum_{x_{g-1}\in\mathcal{A}}\prod_{k=1}^{g-1}\left(M_{x_k}-\sum_{j<k}\delta_{x_j,x_k}\right) \tag{57}\\
&= 1 \tag{58}
\end{align}$$

as well as taken over the tally vector

$$\begin{align}
\mathbb{E}_{\boldsymbol{M}|\boldsymbol{p}}[w_g((x_1,\ldots,x_g),\boldsymbol{M},\boldsymbol{p})] &= \frac{\mathbb{E}_{\boldsymbol{M}|\boldsymbol{p}}\left[\prod_{k=1}^{g}\left(M_{x_k}-\sum_{j<k}\delta_{x_j,x_k}\right)\right]}{(c)_g\cdot p_{x_1}\cdots p_{x_g}} \tag{59}\\
&= \frac{\sum_{\boldsymbol{m}}\binom{c}{\boldsymbol{m}}\boldsymbol{p}^{\boldsymbol{m}}\prod_{k=1}^{g}\left(m_{x_k}-\sum_{j<k}\delta_{x_j,x_k}\right)}{(c)_g\cdot p_{x_1}\cdots p_{x_g}} \tag{60}\\
&= \frac{1}{(c)_g}\sum_{\boldsymbol{m}}\binom{c}{\boldsymbol{m}}\boldsymbol{p}^{\boldsymbol{m}}\prod_{k=1}^{g}\frac{m_{x_k}-\sum_{j<k}\delta_{x_j,x_k}}{p_{x_k}} \tag{61}\\
&= \frac{1}{(c)_g}\;\frac{\partial^g}{\partial p_{x_1}\cdots\partial p_{x_g}}\sum_{\boldsymbol{m}}\binom{c}{\boldsymbol{m}}\boldsymbol{p}^{\boldsymbol{m}}\Bigg|_{|\boldsymbol{p}|=1} \tag{62}\\
&= \frac{1}{(c)_g}\;\frac{\partial^g|\boldsymbol{p}|^c}{\partial p_{x_1}\cdots\partial p_{x_g}}\Bigg|_{|\boldsymbol{p}|=1} \tag{63}\\
&= \frac{(c)_g|\boldsymbol{p}|^{c-g}}{(c)_g}\Bigg|_{|\boldsymbol{p}|=1} \tag{64}\\
&= 1. \tag{65}
\end{align}$$

□

3.3 Suspicion functions that maximize the expected suspicion

THEOREM 3.5. *In each of the cases above (we use $\Box$ as a placeholder for y, ϕ or ψ), the suspicion function $\hat{h}_{t,g}$ that maximizes the expected average suspicion $\mu_{t,g}$ is the centered and normalized weight function*

$$\hat{h}_{t,g}(\vec{\boldsymbol{x}},\Box,\boldsymbol{p}) = \frac{w_g((x_1,\ldots,x_g),\Box,\boldsymbol{p})-1}{\sqrt{\mathrm{Var}[w_g]}} \tag{66}$$

using only the symbols of the first g users in the t-tuple. With this suspicion function, the maximum expected average suspicion is

$$\hat{\mu}_{t,g} = \sqrt{\mathrm{Var}[w_g]}. \tag{67}$$

Proof. To find the suspicion function $\hat{h}_{t,g}$ that maximizes the expected average suspicion $\mu_{t,g}$ of tuples that consist of g guilty and $t-g$ innocent users, under the constraints that tuples of innocent users have a suspicion with zero mean $\mu_{t,0} = 0$ and unit variance $\sigma_{t,0}^2 = 1$, we define the Lagrangian

$$L(h, \lambda_1, \lambda_2) := \mathbb{E}[w_g \cdot h_t] - \lambda_1 \mathbb{E}[h_t] - \tfrac{1}{2}\lambda_2(\mathbb{E}[h^2] - 1) \tag{68}$$

with Lagrange multipliers λ_1 and λ_2. Let $\hat{h}_{t,g}$ be such that $\frac{\delta L}{\delta \hat{h}_{t,g}} = 0$. Then $D(w_g - \lambda_1 - \lambda_2 \hat{h}_{t,g}) = 0$, where D is the product of the probability densities of the random variables. So $\hat{h}_{t,g} = \frac{w_g - \lambda_1}{\lambda_2}$. The first constraint, $\mathbb{E}[\hat{h}_{t,g}] = 0$, implies that $\lambda_1 = \mathbb{E}[w_g] = 1$ and the second constraint, $\mathbb{E}[\hat{h}_{t,g}^2] = 1$, implies that $\lambda_2^2 = \mathbb{E}[(w_g - \lambda_1)^2] = \mathrm{Var}[w_g]$.

From the previous lemmas, we conclude that

$$\hat{\mu}_{t,g} = \mathbb{E}[w_g \cdot \hat{h}_{t,g}] = \frac{\mathbb{E}[(w_g)^2 - 1]}{\sqrt{\mathrm{Var}[w_g]}} = \frac{\mathrm{Var}[w_g]}{\sqrt{\mathrm{Var}[w_g]}} = \sqrt{\mathrm{Var}[w_g]}. \tag{69}$$

□

Corollary 2. *The maximum expected average suspicion $\hat{\mu}_{t,g}$ of t-tuples that consist of g guilty and $t-g$ innocent users, attained by the suspicion function $\hat{h}_{t,g}$, is equal to the maximum expected average suspicion $\hat{\mu}_{g,g}$ of g-tuples that consist entirely of g guilty users, attained by the suspicion function $\hat{h}_{g,g}$. In a formula, $\hat{\mu}_{t,g} = \hat{\mu}_{g,g}$.*

Proof. The maximum expected average suspicion $\hat{\mu}_{t,g}$ depends on $g \leq t$, but not explicitly on t itself. So

$$\hat{\mu}_{t,g} = \sqrt{\mathrm{Var}[w_g]} = \hat{\mu}_{g,g}. \tag{70}$$

□

Note that this corollary requires our choice of using *ordered* tuples, and thus on the possibility of having an asymmetric suspicion function h. In other words, by optimizing $\hat{\mu}_{t,g}$ we specifically chose to optimize the average score of a [$\mathrm{g}^g\mathrm{i}^{t-g}$]-tuple, with all colluders in front. Since all possible orderings of the t-tuple are assumed to be calculated, this will give the maximum average score. Intuitively, otherwise one would have to consider a symmetrized version, decreasing the average suspicion $\mu_{t,g}$ by concurrently optimizing all permutations of [$\mathrm{g}^g\mathrm{i}^{t-g}$]-tuples. This difference is illustrated in Figure 2 for pairs.

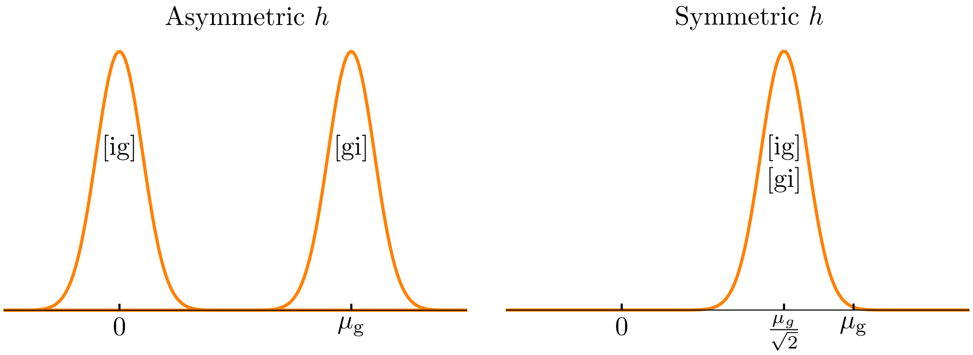

Figure 2. The average suspicion $\mu_{t,g}$ for [gi] and [ig] pairs with an asymmetric suspicion function h (left) and with a symmetric suspicion function h (right), compared to the simple decoders average μ_g.

We therefore restrict our attention to optimal suspicion functions of the form $h_{t,t}$.

Lemma 3.6. *For $\hat{h}_{t,t}$,*

$$\mu_{t,g} = \frac{\hat{\mu}_{g,g}^2}{\hat{\mu}_{t,t}} \tag{71}$$

Proof. Since

$$\begin{aligned}
&\mathbb{E}_{X_{g+1}|\boldsymbol{P}} \cdots \mathbb{E}_{X_t|\boldsymbol{P}} \left[w_t((x_1, \ldots, x_g, X_{g+1}, \ldots, X_t), \boldsymbol{\Phi}, \boldsymbol{P}) \right] && (72)\\
&= \frac{1}{f_{\boldsymbol{\phi}|\boldsymbol{p}} \cdot (c)_t \cdot p_{x_1} \cdots p_{x_g}} \mathbb{E}_{\boldsymbol{M}|\boldsymbol{p}} \left[f_{\boldsymbol{\phi}|\boldsymbol{M}} \mathbb{E}_{X_{g+1}|\boldsymbol{p}} \cdots \mathbb{E}_{X_t|\boldsymbol{p}} \left[\frac{\prod_{k=1}^t (M_{X_k} - \sum_{j<k} \delta_{X_j,X_k})}{p_{X_{g+1}} \cdots p_{X_t}} \right] \right] && (73)\\
&= \frac{1}{f_{\boldsymbol{\phi}|\boldsymbol{p}} \cdot (c)_t \cdot p_{x_1} \cdots p_{x_g}} \mathbb{E}_{\boldsymbol{M}|\boldsymbol{p}} \left[f_{\boldsymbol{\phi}|\boldsymbol{M}} \sum_{x_{g+1} \in \mathcal{A}} \cdots \sum_{x_t \in \mathcal{A}} \prod_{k=1}^t (M_{x_k} - \sum_{j<k} \delta_{x_j,x_k}) \right] && (74)\\
&= \frac{1}{f_{\boldsymbol{\phi}|\boldsymbol{p}} \cdot (c)_t \cdot p_{x_1} \cdots p_{x_g}} \mathbb{E}_{\boldsymbol{M}|\boldsymbol{p}} \left[f_{\boldsymbol{\phi}|\boldsymbol{M}} \sum_{x_{g+1} \in \mathcal{A}} \cdots \sum_{x_{t-1} \in \mathcal{A}} \left(\prod_{k=1}^{t-1} (M_{x_k} - \sum_{j<k} \delta_{x_j,x_k}) \right) \sum_{x_t \in \mathcal{A}} (M_{x_t} - \sum_{j<t} \delta_{x_j,x_k}) \right] && (75)\\
&= \frac{1}{f_{\boldsymbol{\phi}|\boldsymbol{p}} \cdot (c)_t \cdot p_{x_1} \cdots p_{x_g}} \mathbb{E}_{\boldsymbol{M}|\boldsymbol{p}} \left[f_{\boldsymbol{\phi}|\boldsymbol{M}} \sum_{x_{g+1} \in \mathcal{A}} \cdots \sum_{x_{t-1} \in \mathcal{A}} \left(\prod_{k=1}^{t-1} (M_{x_k} - \sum_{j<k} \delta_{x_j,x_k}) \right) (c - t + 1) \right] && (76)\\
&= \frac{\mathbb{E}_{\boldsymbol{M}|\boldsymbol{p}} \left[f_{\boldsymbol{\phi}|\boldsymbol{M}} \prod_{k=1}^g (M_{x_k} - \sum_{j<k} \delta_{x_j,x_k}) \right]}{f_{\boldsymbol{\phi}|\boldsymbol{p}} \cdot (c)_g \cdot p_{x_1} \cdots p_{x_g}} = w_g((x_1, \ldots, x_g), \boldsymbol{\phi}, \boldsymbol{p}) && (77)
\end{aligned}$$

we find that

$$\begin{aligned}
\mathbb{E}[w_g \cdot w_t] &= \mathbb{E}_{\boldsymbol{P}} \mathbb{E}_{\boldsymbol{\Phi}|\boldsymbol{P}} \mathbb{E}_{X_1|\boldsymbol{P}} \cdots \mathbb{E}_{X_t|\boldsymbol{P}} \left[w_g((X_1, \ldots, X_g), \boldsymbol{\Phi}, \boldsymbol{P}) \cdot w_t(\vec{\boldsymbol{X}}, \boldsymbol{\Phi}, \boldsymbol{P}) \right] && (78)\\
&= \mathbb{E}_{\boldsymbol{P}} \mathbb{E}_{\boldsymbol{\Phi}|\boldsymbol{P}} \mathbb{E}_{X_1|\boldsymbol{P}} \cdots \mathbb{E}_{X_g|\boldsymbol{P}} \left[w_g((X_1, \ldots, X_g), \boldsymbol{\Phi}, \boldsymbol{P}) \mathbb{E}_{X_{g+1}|\boldsymbol{P}} \cdots \mathbb{E}_{X_t|\boldsymbol{P}} \left[w_t(\vec{\boldsymbol{X}}, \boldsymbol{\Phi}, \boldsymbol{P}) \right] \right] && (79)\\
&= \mathbb{E}_{\boldsymbol{P}} \mathbb{E}_{\boldsymbol{\Phi}|\boldsymbol{P}} \mathbb{E}_{X_1|\boldsymbol{P}} \cdots \mathbb{E}_{X_g|\boldsymbol{P}} \left[w_g^2((X_1, \ldots, X_g), \boldsymbol{\Phi}, \boldsymbol{P}) \right] = \mathbb{E}[w_g^2] && (80)
\end{aligned}$$

and thus

$$\mu_{t,g} = \mathbb{E}[w_g \cdot \hat{h}_{t,t}] = \frac{\mathbb{E}[w_g \cdot w_t] - \mathbb{E}[w_g \cdot 1]}{\sqrt{\mathrm{Var}[w_t]}} = \frac{\mathbb{E}[w_g \cdot w_t] - 1}{\sqrt{\mathrm{Var}[w_t]}} = \frac{\mathbb{E}[w_g^2] - (\mathbb{E}[w_g])^2}{\sqrt{\mathrm{Var}[w_t]}} = \frac{\mathrm{Var}[w_g]}{\sqrt{\mathrm{Var}[w_t]}} = \frac{\hat{\mu}_{g,g}^2}{\hat{\mu}_{t,t}}. \tag{81}$$

□

Lemma 3.7. *For $\hat{h}_{t,t}$ and any $0 \leq s \leq t$, we have that*

$$\hat{\mu}_{t,t} \geq \frac{\hat{\mu}_{s,s} + \hat{\mu}_{t-s,t-s}}{\sqrt{2}}. \tag{82}$$

Proof. Consider the (normalized) suspicion function

$$h_{t,t}((x_1, \ldots, x_t), \boldsymbol{\phi}, \boldsymbol{p}) = \frac{1}{\sqrt{2}} \left(\hat{h}_{s,s}((x_1, \ldots, x_s), \boldsymbol{\phi}, \boldsymbol{p}) + \hat{h}_{t-s,t-s}((x_{s+1}, \cdots, x_t), \boldsymbol{\phi}, \boldsymbol{p}) \right). \tag{83}$$

Taking expectations on both sides yields the claimed statement. □

In particular, the previous lemma can be used to relate the optimal average score to that of the simple decoder $\hat{\mu}_{t,t} \geq \sqrt{t}\ \hat{\mu}_{1,1}$. If we assume that $\hat{\mu}_{t,t} = \alpha \hat{\mu}_{1,1}$, with $\alpha \geq \sqrt{t}$, then from Lemma 3.6 we obtain that $\mu_{t,1} = \frac{1}{\alpha} \hat{\mu}_{1,1}$ when using the optimal suspicion function $\hat{h}_{t,t}$. In other words, the better the suspicion function works against [g^t]-tuples, the worse it performs against tuples with fewer colluders. This is schematically illustrated below in Figure 3.

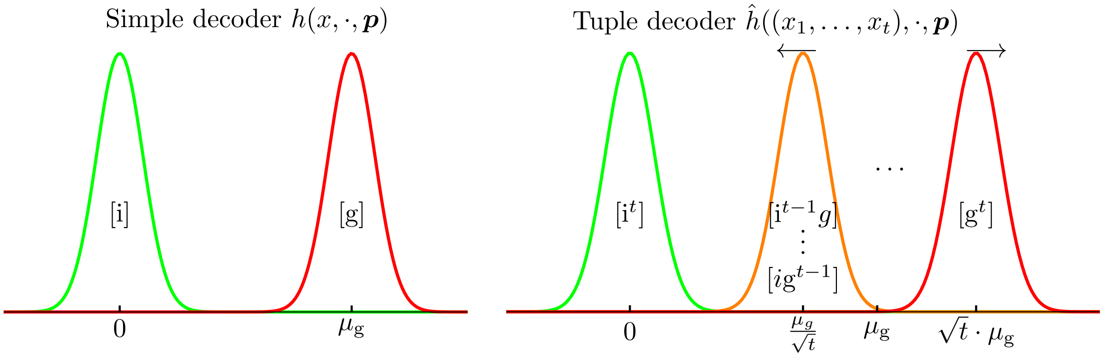

Figure 3. Improvements by moving to a tuple decoder (right) from a simple decoder (left). The more the $[\mathrm{g}^t]$ peak lies to the right, the more the inner peaks shift towards 0 by Lemma 3.6.

4. DEFENDING AGAINST COMMON COLLUSION STRATEGIES

1. The *interleaving attack* randomly selects an attacker and outputs his symbol.
2. The *all-high attack* is special as it breaks the symbol-symmetry. It assumes that the alphabet can be ordered in some meaningful way, and outputs the largest received symbol. For a binary alphabet ($q = 2$), this attack is known as the all-1 attack, as it will output a 1 if the coalition has received one.
3. The *random-symbol attack* randomly selects a received symbol, irrespective of the tally vector $\boldsymbol{m}$, and outputs it. For a binary alphabet ($q = 2$), this attack is known as the coin-flip attack.
4. The *majority voting attack* outputs the symbol that was received most often by the coalition. In case multiple symbols are received equally often, a random symbol is chosen among them.
5. The *minority voting attack* outputs the symbol that was received least often (but at least once) by the coalition. When multiple symbols are received equally often, a random symbol is chosen among them.

The first three attacks will be treated in detail below. The other two are listed for completeness.

4.1 The interleaving defense

The interleaving attack is a collusion strategy where the coalition randomly selects an attacker and outputs his symbol, such that the probability that symbol y is output, given that the coalition had received symbols according to the tally vector $\boldsymbol{m}$, $f_{y|\boldsymbol{m}} = \frac{m_y}{c}$.

Proposition 1 (Prop. 7).[26] *Against the interleaving attack, for singletons the weight function is given by*

$$w_1(x,y,\boldsymbol{p}) = 1 + \frac{1}{c}\left(\frac{\delta_{x,y}}{p_y} - 1\right) \tag{84}$$

and the optimal suspicion function is

$$h_{1,1}(x,y,\boldsymbol{p}) = \frac{1}{\sqrt{q-1}}\left(\frac{\delta_{x,y}}{p_y} - 1\right). \tag{85}$$

Proposition 2. *Against the interleaving attack, the weight function for t-tuples is*

$$w_t(\vec{\boldsymbol{x}},y,\boldsymbol{p}) = 1 + \frac{1}{c}\sum_{k=1}^{t}\left(\frac{\delta_{x_k,y}}{p_y} - 1\right) = 1 + \sum_{k=1}^{t}\left(w_1(x_k,y,\boldsymbol{p}) - 1\right). \tag{86}$$

and the optimal suspicion function for t-tuples is the normalized sum of the optimal suspicion function for the t singletons

$$\hat{h}_{t,t}(\vec{\boldsymbol{x}}, y, \boldsymbol{p}) = \frac{1}{\sqrt{t}} \sum_{k=1}^{t} \hat{h}_{1,1}(x_k, y, \boldsymbol{p}) = \frac{1}{\sqrt{t(q-1)}} \left(\frac{\sum_{k=1}^{t} \delta_{x_k,y}}{p_y} - t \right) \tag{87}$$

Proof. We find

$$|\boldsymbol{p}|^c f_{y|\boldsymbol{p}} = \frac{1}{c} \sum_{\boldsymbol{m}} \binom{c}{\boldsymbol{m}} \boldsymbol{p}^{\boldsymbol{m}} m_y = \frac{p_y}{c} \frac{\partial}{\partial p_y} \sum_{\boldsymbol{m}} \binom{c}{\boldsymbol{m}} \boldsymbol{p}^{\boldsymbol{m}} = \frac{p_y}{c} \frac{\partial |\boldsymbol{p}|^c}{\partial p_y} = p_y |\boldsymbol{p}|^{c-1}. \tag{88}$$

so

$$\frac{\partial \left(|\boldsymbol{p}|^c f_{y|\boldsymbol{p}} \right)}{\partial p_{x_1}} = \delta_{x_1,y} |\boldsymbol{p}|^{c-1} + (c-1) p_y |\boldsymbol{p}|^{c-2} \tag{89}$$

Thus

$$\frac{\partial^t \left(|\boldsymbol{p}|^c f_{y|\boldsymbol{p}} \right)}{\partial p_{x_1} \cdots p_{x_t}} = \frac{(c)_t}{c} \sum_{k=1}^{t} \delta_{x_k,y} |\boldsymbol{p}|^{c-t} + \frac{(c)_{t+1} \cdot p_y}{c} |\boldsymbol{p}|^{c-t-1} \tag{90}$$

is true for $t = 1$. If (90) is true for some t, then

$$\frac{\partial^{t+1} \left(|\boldsymbol{p}|^c f_{y|\boldsymbol{p}} \right)}{\partial p_{x_1} \cdots p_{x_{t+1}}} = \frac{(c)_{t+1}}{c} \sum_{k=1}^{t} \delta_{x_k,y} |\boldsymbol{p}|^{c-t-1} + \frac{(c)_{t+1} \delta_{x_{t+1},y}}{c} |\boldsymbol{p}|^{c-t-1} + \frac{(c)_{t+2} \cdot p_y}{c} |\boldsymbol{p}|^{c-t-2} \tag{91}$$

$$= \frac{(c)_{t+1}}{c} \sum_{k=1}^{t+1} \delta_{x_k,y} |\boldsymbol{p}|^{c-t-1} + \frac{(c)_{t+2} \cdot p_y}{c} |\boldsymbol{p}|^{c-t-2} \tag{92}$$

so (90) also holds for $t + 1$. We have thus shown by mathematical induction that (90) holds for all values of t.

So the centered weight function for t-tuples is the sum of t centered weight functions for singletons:

$$w_t(\vec{\boldsymbol{x}}, y, \boldsymbol{p}) = \frac{1}{f_{y|\boldsymbol{p}} \cdot (c)_t} \left. \frac{\partial^t (|\boldsymbol{p}|^c f_{y|\boldsymbol{p}})}{\partial p_{x_1} \cdots \partial p_{x_t}} \right|_{|\boldsymbol{p}|=1} = \frac{\sum_{k=1}^{t} \delta_{x_k,y}}{c p_y} + \frac{c-t}{c} = 1 + \frac{1}{c} \sum_{k=1}^{t} \left(\frac{\delta_{x_k,y}}{p_y} - 1 \right) \tag{93}$$

$$= 1 + \sum_{k=1}^{t} \left(w_1(x_k, y, \boldsymbol{p}) - 1 \right). \tag{94}$$

Thus

$$\mathrm{Var}[w_t] = \mathrm{Var}\left[\sum_{k=1}^{t} w_1(x_k, y, \boldsymbol{p}) \right] = t \cdot \mathrm{Var}[w_1] = \frac{t(q-1)}{c^2} = \sum_{k=1}^{t} \mathrm{Var}[w_1(x_k, y, \boldsymbol{p})]. \tag{95}$$

and hence

$$\hat{h}_{t,t}(\vec{\boldsymbol{x}}, y, \boldsymbol{p}) = \frac{1}{\sqrt{t}} \sum_{k=1}^{t} \hat{h}_{1,1}(x_k, y, \boldsymbol{p}) = \frac{1}{\sqrt{t(q-1)}} \left(\frac{\sum_{k=1}^{t} \delta_{x_k,y}}{p_y} - t \right) \tag{96}$$

□

Proposition 3. *When the interleaving attack is used against the interleaving defense, then $\mu_{t,t} = \frac{1}{c}\sqrt{t(q-1)}$, achieving capacity for any bias distribution $f_{\boldsymbol{p}}$.*

Proof.

$$\mu_{t,t} = \sqrt{\mathrm{Var}[w_t]} = \frac{1}{c} \sqrt{t(q-1)} \tag{97}$$

□

Since the simple and joint capacities are equal for the interleaving attack, the last result is to be expected. Nothing can be gained here from going to a tuple decoder.

4.2 The all-high defense

The all-high attack

$$f_{y|\boldsymbol{m}} = \delta_{y,\max(\alpha\in\mathcal{A}:m_\alpha>0)} = \begin{cases} 1 & \text{if } m_y > 0 \text{ and } m_{y+1} = \cdots = m_{q-1} = 0 \\ 0 & \text{else} \end{cases} \tag{98}$$

outputs the highest symbol among those received by the coalition.

Note that this is the only attack we consider that breaks symbol symmetry and assumes an ordering of the alphabet. This is a special case of the so-called *preferred-sequence attack*, in which the colluders have a predetermined ranking of the symbols. The results below generalize to the preferred-sequence attack. Recall our shorthand notation $a_k := p_0 + \cdots p_k$.

Proposition 4. *Against the all-high attack, the optimal suspicion function is* $\hat{h}_{t,t} = (w_t - 1)/\sqrt{\mathrm{Var}[w_t]}$, *with*

$$w_t(\vec{\boldsymbol{x}}, y, \boldsymbol{p}) = \begin{cases} \left(a_y^{c-t} - a_{y-1}^{c-t}\right) / \left(a_y^c - a_{y-1}^c\right) & \textit{if } \max_{1\le k\le t}(x_k) < y \\ a_y^{c-t} / \left(a_y^c - a_{y-1}^c\right) & \textit{if } \max_{1\le k\le t}(x_k) = y \\ 0 & \textit{if } \max_{1\le k\le t}(x_k) > y. \end{cases} \tag{99}$$

Proof. We find

$$f_{y|\boldsymbol{p}} = \mathbb{E}_{\boldsymbol{M}|\boldsymbol{p}}[f_{y|\boldsymbol{M}}] = \mathbb{P}[M_y > 0, M_{y+1} = \cdots = M_{q-1} = 0] \tag{100}$$

$$= \mathbb{P}[M_{y+1} = \cdots = M_{q-1} = 0] - \mathbb{P}[M_y = \cdots = M_{q-1} = 0] = \frac{a_y^c}{|\boldsymbol{p}|^c} - \frac{a_{y-1}^c}{|\boldsymbol{p}|^c} \tag{101}$$

so

$$\frac{\partial^t \left(|\boldsymbol{p}|^c f_{y|\boldsymbol{p}}\right)}{\partial p_{x_1} \cdots p_{x_t}} = \begin{cases} (c)_t \left(a_y^{c-t} - a_{y-1}^{c-t}\right) & \text{if } \max_{1\le k\le t}(x_k) < y \\ (c)_t \cdot a_y^{c-t} & \text{if } \max_{1\le k\le t}(x_k) = y \\ 0 & \text{if } \max_{1\le k\le t}(x_k) > y. \end{cases} \tag{102}$$

So the weight function is

$$w_t(\vec{\boldsymbol{x}}, y, \boldsymbol{p}) = \frac{1}{f_{y|\boldsymbol{p}} \cdot (c)_t} \left.\frac{\partial^t (|\boldsymbol{p}|^c f_{y|\boldsymbol{p}})}{\partial p_{x_1} \cdots \partial p_{x_t}}\right|_{|\boldsymbol{p}|=1} = \begin{cases} \left(a_y^{c-t} - a_{y-1}^{c-t}\right) / \left(a_y^c - a_{y-1}^c\right) & \text{if } \max_{1\le k\le t}(x_k) < y \\ a_y^{c-t} / \left(a_y^c - a_{y-1}^c\right) & \text{if } \max_{1\le k\le t}(x_k) = y \\ 0 & \text{if } \max_{1\le k\le t}(x_k) > y. \end{cases} \tag{103}$$

$\square$

We will use the shorthand notation $a_{\mathcal{B}} = \sum_{\beta\in\mathcal{B}} p_\beta$ for $\mathcal{B} \subseteq \mathcal{A}$.

4.3 The random-symbol defense

The random symbol attack selects one of the received symbols uniformly at random. Tallies are disregarded, but a symbol can only be chosen it its tally is nonzero. The attack is parametrized by

$$f_{y|\boldsymbol{m}} = (1 - \delta_{m_y,0}) / |\{\alpha \in \mathcal{A} : m_\alpha > 0\}|. \tag{104}$$

Proposition 5. *For the random-symbol attack we find*

$$|\boldsymbol{p}|^c f_{y|\boldsymbol{p}} = \frac{a_{\mathcal{A}}^c - a_{\mathcal{A}\setminus\{y\}}^c}{q} + \sum_{\mathcal{B}\subsetneq\mathcal{A}:\, y\in\mathcal{B}} \frac{a_{\mathcal{B}}^c - a_{\mathcal{B}\setminus\{y\}}^c}{|\mathcal{B}|(|\mathcal{B}|+1)}. \tag{105}$$

Against the random-symbol attack, the optimal suspicion function is $\hat{h}_{t,t} = (w_t - 1)/\sqrt{\mathrm{Var}[w_t]}$*, with*

$$w_t(\vec{\boldsymbol{x}}, y, \boldsymbol{p}) = \begin{cases} \dfrac{1}{f_{y|\boldsymbol{p}}} \left(\dfrac{1}{q} + \displaystyle\sum_{\substack{\mathcal{B} \subsetneq \mathcal{A} \\ x_1,\ldots,x_t,y \in \mathcal{B}}} \dfrac{a_{\mathcal{B}}^{c-t}}{|\mathcal{B}|(|\mathcal{B}|+1)} \right) & \textit{if } \exists k : x_k = y \\ \dfrac{1}{f_{y|\boldsymbol{p}}} \left(\dfrac{1-(1-p_y)^{c-t}}{q} + \displaystyle\sum_{\substack{\mathcal{B} \subsetneq \mathcal{A} \\ x_1,\ldots,x_t,y \in \mathcal{B}}} \dfrac{a_{\mathcal{B}}^{c-t} - a_{\mathcal{B}\setminus\{y\}}^{c-t}}{|\mathcal{B}|(|\mathcal{B}|+1)} \right) & \textit{if } \forall k : x_k \neq y \end{cases} \tag{106}$$

Proof. For the random-symbol attack, the probability $f_{y|\boldsymbol{m}}$ that the symbol y is produced, is 0 if $m_y = 0$. It is $\frac{1}{q}$ if for all $\alpha \in \mathcal{A}$, $m_\alpha > 0$. It is $\frac{1}{q-1}$ if $m_y > 0$ and there is exactly one symbol $\alpha_1 \in \mathcal{A}$ for which $m_{\alpha_1} = 0$. It is $\frac{1}{q-2}$ if $m_y > 0$ and there are exactly two distinct symbols $\alpha_1, \alpha_2 \in \mathcal{A}$ for which $m_{\alpha_1} = m_{\alpha_2} = 0$, etc. This can be written in additive form using indicator functions:

$$\begin{aligned} & f_{y|\boldsymbol{m}} = \tfrac{1}{q}\mathbf{1}_{\{m_y>0\}} \\ & + \left(\tfrac{1}{q-1} - \tfrac{1}{q}\right)\mathbf{1}_{\{m_y>0\}}\mathbf{1}_{\{\exists\alpha_1:m_{\alpha_1}=0\}} \\ & + \left(\tfrac{1}{q-2} - \tfrac{1}{q-1}\right)\mathbf{1}_{\{m_y>0\}}\mathbf{1}_{\{\exists\alpha_1:m_{\alpha_1}=0\}}\mathbf{1}_{\{\exists\alpha_2\neq\alpha_1:m_{\alpha_2}=0\}} \\ & + \cdots + \left(1 - \tfrac{1}{2}\right)\mathbf{1}_{\{m_y>0\}} \\ & \cdot \mathbf{1}_{\{\exists\alpha_1:m_{\alpha_1}=0\}} \cdots \mathbf{1}_{\{\exists\alpha_{q-1}\neq\alpha_1,\ldots\alpha_{q-2}:m_{\alpha_{q-1}}=0\}}. \end{aligned} \tag{107}$$

Note that

$$\mathbb{P}[M_y > 0] = 1 - \mathbb{P}[M_y = 0] = \frac{|\boldsymbol{p}|^c - a_{\mathcal{A}\setminus\{y\}}^c}{|\boldsymbol{p}|^c} \tag{108}$$

and for each proper subset $\mathcal{B} \subsetneq \mathcal{A}$ with $y \in \mathcal{B}$, it holds that

$$\begin{aligned} \mathbb{P}[M_y > 0 \text{ and } \forall\alpha \notin \mathcal{B}, M_\alpha = 0] &= \mathbb{P}[\forall\alpha \notin \mathcal{B}, M_\alpha = 0] - \mathbb{P}[M_y = 0 \text{ and } \forall\alpha \notin \mathcal{B}, M_\alpha = 0] && (109) \\ &= \frac{a_{\mathcal{B}}^c - a_{\mathcal{B}\setminus\{y\}}^c}{|\boldsymbol{p}|^c}. && (110) \end{aligned}$$

Since $f_{y|\boldsymbol{p}} = \mathbb{E}_{\boldsymbol{M}|\boldsymbol{p}}[f_{y|\boldsymbol{M}}]$, and for all sets $\mathcal{V}, \mathcal{W}$, it holds that $\mathbf{1}_{\mathcal{V}}\mathbf{1}_{\mathcal{W}} = \mathbf{1}_{\mathcal{V}\cap\mathcal{W}}$, and $\mathbb{E}[\mathbf{1}_{\mathcal{V}}] = \mathbb{P}[\mathcal{V}]$, we find

$$|\boldsymbol{p}|^c f_{y|\boldsymbol{p}} = \frac{a_{\mathcal{A}}^c - a_{\mathcal{A}\setminus\{y\}}^c}{q} + \sum_{\mathcal{B}\subsetneq\mathcal{A}:\, y\in\mathcal{B}} \left(\frac{1}{|\mathcal{B}|} - \frac{1}{|\mathcal{B}|+1}\right)\left(a_{\mathcal{B}}^c - a_{\mathcal{B}\setminus\{y\}}^c\right).$$

which simplifies to equation (105). Differentiating t times yields the weight function. $\square$

5. ACCUSATION

We describe a family of accusation algorithms that decides for each individual user, whether or not to accuse him to be part of the coalition. We use the tuple decoder not to accuse entire tuples, but to better accuse single users.

For the simple decoder ($t = 1$), any user whose level of suspicion exceeded some fixed threshold was accused. Once we start looking at pairs, there are three types, consisting of two, one or no innocent users. Distinguishing an innocent from a guilty user requires determining whether he is in $N - c - 1$ $[i^2]$ pairs and c $[gi]$ pairs, or in $N - c$ $[gi]$ pairs and $c - 1$ $[g^2]$ pairs. This is depicted in Figure 4.

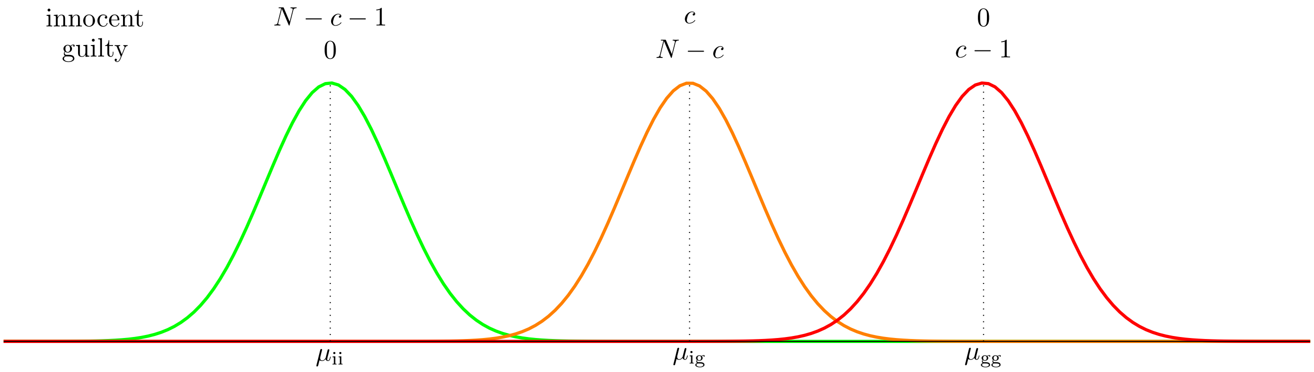

Figure 4. Distinguishing innocent from guilty users in a pair decoder.

For the joint decoder ($t \geq 2$), we thus start out with two thresholds: one called $z_{t,0}$, less than and close to $\mu_{t,0} = 0$ to determine if many tuples lie in the $[i^t]$ range, and another called $z_{t,t}$, close to $\mu_{t,t}$, to check if a tuple lies in the $[g^t]$ range. We accuse any user who is part of some t-tuple with a suspicion that exceeds $z_{t,t}$, but is part of no t-tuple with a suspicion below $z_{t,0}$. We bound the probabilities of two types of error: a false positive error FP, where an innocent user is falsely accused, and a false negative error FN, where no colluder is accused.

The picture above generalizes: an innocent user is part of many tuples of type $[i^t]$ and in none of type $[g^t]$. A guilty user is in none of type $[i^t]$ and in $c-1$ tuples of type $[g^t]$.

Let $\hat{h}_{t,t}^{(i)}(\boldsymbol{u})$ be the level of suspicion attributed to tuple $\boldsymbol{u}$ in segment i. For a code of ℓ segments, let $s_t(\boldsymbol{u}) := \sum_{i=1}^{\ell} \hat{h}_{t,t}^{(i)}(\boldsymbol{u})$ be the total score of that tuple over all segments.

We repeatedly make use of the fact that, for any two sets $\mathcal{S}$ and $\mathcal{T}$,

$$\mathbb{P}[\mathcal{S} \cup \mathcal{T}] = \mathbb{P}[\mathcal{S}] + \mathbb{P}[\mathcal{T}] - \mathbb{P}[\mathcal{S} \cap \mathcal{T}] \leq \mathbb{P}[\mathcal{S}] + \mathbb{P}[\mathcal{T}] \tag{111}$$

and

$$\mathbb{P}[\mathcal{S} \cap \mathcal{T}] = \mathbb{P}[\mathcal{S} | \mathcal{T}] \cdot \mathbb{P}[\mathcal{T}] \leq \mathbb{P}[\mathcal{S}] \cdot \mathbb{P}[\mathcal{T}] \tag{112}$$

and thus for a tuple $(\mathcal{S}_i)_{1 \leq i \leq t}$ of t sets,

$$\mathbb{P}\left[\bigcup_{i=1}^{t} \mathcal{S}_i\right] \leq \sum_{i=1}^{t} \mathbb{P}[\mathcal{S}_i] \quad \text{and} \quad \mathbb{P}\left[\bigcap_{i=1}^{t} \mathcal{S}_i\right] \leq \prod_{i=1}^{t} \mathbb{P}[\mathcal{S}_i]. \tag{113}$$

Phrased in terms of events E and F, this means that

$$\mathbb{P}[E \text{ or } F] \leq \mathbb{P}[E] + \mathbb{P}[F] \quad \text{and} \quad \mathbb{P}[E \text{ and } F] \leq \mathbb{P}[E] \cdot \mathbb{P}[F] \tag{114}$$

and for a tuple $(E_i)_{1 \leq i \leq t}$ of events

$$\mathbb{P}\left[\exists (1 \leq i \leq t) : E_i\right] \leq \sum_{i=1}^{t} \mathbb{P}[E_i] \quad \text{and} \quad \mathbb{P}\left[\forall (1 \leq i \leq t) : E_i\right] \leq \prod_{i=1}^{t} \mathbb{P}[E_i]. \tag{115}$$

The false positive probability of accusing innocent user j is

$$\mathbb{P}[\text{FP}] = \mathbb{P}\left[\left(\forall \text{ tuple}_j : S > z_{t,0}\right) \text{ and } \left(\exists \text{ tuple}_j : S > z_{t,t}\right)\right] \leq \mathbb{P}\left[\forall \text{ tuple}_j : S > z_{t,0}\right] \tag{116}$$

$$\leq \prod_{k=0}^{t-1} \left(\mathbb{P}[S_{[g^k j i^{t-k-1}]} > z_{t,0}]\right)^{(c)_k (n-c-1)_{t-k-1}}. \tag{117}$$

where tuple_j indicates a t-tuple containing user j. We can use the Markov bound and write, for any $\alpha > 0$,

$$\begin{aligned}
\mathbb{P}[S_{[g^kji^{t-k-1}]} > z_{t,0}] = \mathbb{P}\left[\exp\left(\alpha S_{[g^kji^{t-k-1}]}\right) > \exp(\alpha z_{t,0})\right] &\leq \frac{\mathbb{E}\left[\exp\left(\alpha S_{[g^kji^{t-k-1}]}\right)\right]}{\exp(\alpha z_{t,0})} &(118)\\
= \frac{\mathbb{E}\left[\exp\left(\alpha \sum_{d=1}^{\ell} h^{(d)}_{[g^kji^{t-k-1}]}\right)\right]}{\exp(\alpha z_{t,0})} &= \frac{\mathbb{E}\left[\prod_{d=1}^{\ell} \exp\left(\alpha h^{(d)}_{[g^kji^{t-k-1}]}\right)\right]}{\exp(\alpha z_{t,0})} &(119)\\
= \frac{\left(\mathbb{E}\left[\exp\left(\alpha h_{[g^kji^{t-k-1}]}\right)\right]\right)^{\ell}}{\exp(\alpha z_{t,0})}. & &(120)
\end{aligned}$$

We choose r_1 such that for x small enough,

$$1 + x \leq \exp(x) \leq 1 + x + r_1 x^2 \tag{121}$$

and bound

$$\begin{aligned}
\frac{\left(\mathbb{E}\left[\exp\left(\alpha h_{[g^kji^{t-k-1}]}\right)\right]\right)^{\ell}}{\exp(\alpha z_{t,0})} &\leq \frac{\left(1 + \alpha \mathbb{E}\left[h_{[g^kji^{t-k-1}]}\right] + r_1 \alpha^2 \mathbb{E}\left[h^2_{[g^kji^{t-k-1}]}\right]\right)^{\ell}}{\exp(\alpha z_{t,0})} &(122)\\
&= \frac{\left(1 + \alpha \mu_{t,k} + r_1 \alpha^2 (\sigma^2_{t,k} + \mu^2_{t,k})\right)^{\ell}}{\exp(\alpha z_{t,0})}. &(123)
\end{aligned}$$

To make the bound as tight as possible, we choose the parameters r_1 and α that satisfy equation (121) and minimize this above expression. Knowing the moments, this gives us one relation between the code length ℓ and the threshold $z_{t,0}$ as a function of a chosen maximum false positive probability.

Concerning the false negative error, for any guilty user j, the probability he is not accused is

$$\mathbb{P}[\text{FN}] = \mathbb{P}[\left(\exists \text{ tuple}_j : S < z_{t,0}\right) \text{ or } \left(\forall \text{ tuple}_j : S < z_{t,t}\right)] \leq \mathbb{P}[\exists \text{ tuple}_j : S < z_{t,0}] + \mathbb{P}[\forall \text{ tuple}_j : S < z_{t,t}] \tag{124}$$

where

$$\mathbb{P}[\exists \text{ tuple}_j : S < z_{t,0}] \leq \sum_{k=0}^{t-1} \mathbb{P}[\exists [g^k j i^{t-k-1}] : S < z_{t,0}] \leq \sum_{k=0}^{t-1} (c)_k (n-c)_{t-k-1} \mathbb{P}[S_{[g^kji^{t-k-1}]} < z_{t,0}] \tag{125}$$

and

$$\mathbb{P}\left[\forall \text{ tuple}_j : S < z_{t,t}\right] \leq \mathbb{P}\left[\forall [g^{t-1} j] : S < z_{t,t}\right] \leq \left(\mathbb{P}[S_{[g^{t-1}j]} < z_{t,t}]\right)^{(c-1)_{t-1}}. \tag{126}$$

In equation (125), we can use the Markov bound and write, for any $\beta > 0$,

$$\begin{aligned}
\mathbb{P}[S_{[g^kji^{t-k-1}]} < z_{t,0}] = \mathbb{P}\left[\exp\left(-\beta S_{[g^kji^{t-k-1}]}\right) > \exp(-\beta z_{t,0})\right] &\leq \frac{\mathbb{E}\left[\exp\left(-\beta S_{[g^kji^{t-k-1}]}\right)\right]}{\exp(-\beta z_{t,0})} &(127)\\
= \frac{\mathbb{E}\left[\exp\left(-\beta \sum_{d=1}^{\ell} h^{(d)}_{[g^kji^{t-k-1}]}\right)\right]}{\exp(-\beta z_{t,0})} &= \frac{\mathbb{E}\left[\prod_{d=1}^{\ell} \exp\left(-\beta h^{(d)}_{[g^kji^{t-k-1}]}\right)\right]}{\exp(-\beta z_{t,0})} &(128)\\
\leq \frac{\left(\mathbb{E}\left[\exp\left(-\beta h_{[g^kji^{t-k-1}]}\right)\right]\right)^{\ell}}{\exp(-\beta z_{t,0})}. & &(129)
\end{aligned}$$

We choose r_2 such that

$$1 + x \leq \exp(x) \leq 1 + x + r_2 x^2 \tag{130}$$

and bound

$$\frac{\left(\mathbb{E}\left[\exp\left(-\beta h_{[g^k j i^{t-k-1}]}\right)\right]\right)^\ell}{\exp(-\beta z_{t,0})} \leq \frac{\left(1-\beta\mathbb{E}\left[h_{[g^k j i^{t-k-1}]}\right]+r_2\beta^2\mathbb{E}\left[h^2_{[g^k j i^{t-k-1}]}\right]\right)^\ell}{\exp(-\beta z_{t,0})} \tag{131}$$

$$= \frac{\left(1-\beta\mu_{t,k+1}+r_2\beta^2(\sigma^2_{t,k+1}+\mu^2_{t,k+1})\right)^\ell}{\exp(\beta z_{t,0})}. \tag{132}$$

To make the bound as tight as possible, we choose the parameters r_2 and β that satisfy equation (130) and minimize this above expression.

In equation (126), we can again use the Markov bound and write, for any $\gamma > 0$,

$$\mathbb{P}[S_{[g^{t-1}j]} < z_{t,t}] = \mathbb{P}\left[\exp\left(-\gamma S_{[g^{t-1}j]}\right) > \exp(-\gamma z_{t,t})\right] \leq \frac{\mathbb{E}\left[\exp\left(-\gamma S_{[g^{t-1}j]}\right)\right]}{\exp(-\gamma z_{t,t})} \tag{133}$$

$$= \frac{\mathbb{E}\left[\exp\left(-\gamma\sum_{d=1}^{\ell} h^{(d)}_{[g^{t-1}j]}\right)\right]}{\exp(-\gamma z_{t,t})} = \frac{\mathbb{E}\left[\prod_{d=1}^{\ell}\exp\left(-\gamma h^{(d)}_{[g^{t-1}j]}\right)\right]}{\exp(-\gamma z_{t,t})} \tag{134}$$

$$\leq \frac{\left(\mathbb{E}\left[\exp\left(-\gamma h_{[g^{t-1}j]}\right)\right]\right)^\ell}{\exp(-\gamma z_{t,t})}. \tag{135}$$

We choose r_3 such that

$$1+x \leq \exp(x) \leq 1+x+r_3x^2 \tag{136}$$

and bound

$$\frac{\left(\mathbb{E}\left[\exp\left(-\gamma h_{[g^{t-1}j]}\right)\right]\right)^\ell}{\exp(-\gamma z_{t,t})} \leq \frac{\left(1-\gamma\mathbb{E}\left[h_{[g^{t-1}j]}\right]+r_3\gamma^2\mathbb{E}\left[h^2_{[g^{t-1}j]}\right]\right)^\ell}{\exp(-\gamma z_{t,t})} \tag{137}$$

$$= \frac{\left(1-\gamma\mu_{t,t}+r_3\gamma^2(\sigma^2_{t,t}+\mu^2_{t,t})\right)^\ell}{\exp(\gamma z_{t,t})}. \tag{138}$$

To make the bound as tight as possible, we choose the parameters r_3 and γ that satisfy equation (136) and minimize this above expression.

Knowing the moments, this gives us another relation between the code length ℓ and the thresholds $z_{t,0}$ and $z_{t,t}$ as a function of a chosen maximum false negative probability.

Note that we now have a system of equations with more degrees of freedom. This will potentially allow the selection of better accusation algorithm parameters.

5.1 Generalizing the accusation algorithm

The previously described accusation algorithm uses two thresholds: one to check that no tuple containing user j lies in the $[i^t]$ range, and another to check that at least one lies in the $[g^t]$ range. Potentially one could introduce other thresholds as well, for example $z_{t,t-1}$ to check whether enough tuples lie in the $[ig^{t-1}]$ plus the $[g^t]$ ranges.

6. DISCUSSION

We have investigated the optimization of the performance indicator $\mu_{t,t}$ for bias-based traitor tracing in the joint-decoder setting. A straightforward Lagrangian approach under the Gaussian assumption yields a simple expression (Theorem 3.5) for the optimal t-tuple suspicion function in a wide variety of contexts, e.g. CDM and RDM, binary and q-ary, and any tuple size t. While we usually assume that $t \leq c$, the formulas easily generalize to larger tuples as well. The result is again a Neyman-Pearson score for the hypothesis $j \in \mathcal{C}$ against the hypothesis $j \notin \mathcal{C}$, based on single segments of the code word.

The h function we obtain with the Lagrangian method depends either on the collusion strategy or on the coalition's symbol tallies $\boldsymbol{m}$. These quantities are usually unknown to the tracer. Our optimization approach does not allow for deriving suspicion functions that are based purely on data known to the tracer.

In Section 3 we speculated on the use of the $\boldsymbol{m}$-dependent suspicion function in the EM algorithm or as a consistency check for candidate coalitions. Further exploration is left for future work.

For several q-ary attacks in the RDM we have derived the optimal suspicion function. We have investigated the performance indicator $\mu_{t,t}$ in many combinations of suspicion function and attack strategy. In some cases analytic results are obtained.

Our other main contribution in this paper is a novel accusation algorithm. We propose to use a t-tuple decoder to decide whether to accuse a *single* user, and describe the first family of constructions that achieves this. In the style of the original proofs of Tardos, we show how to obtain guarantees on the false positive and false negative probabilities for specific collusion attacks.

Future work will focus on (a) investigating what code length improvements can be achieved against the various listed attacks; (b) constructing better accusation algorithms in the spirit of Section 5; (c) investigating the theoretical capacities of a t-tuple decoder given a certain strategy; (d) simulations using these tuple decoders; (e) iterative joint decoders employing the m-dependent suspicion functions as consistency check.

ACKNOWLEDGMENTS

We thank Boris Škorić and Benne de Weger for valuable discussions. This research is partly supported by the Dutch Technology Foundation STW, which is part of the Netherlands Organisation for Scientific Research (NWO), and which is partly funded by the Ministry of Economic Affairs.

REFERENCES

[1] Tardos, G., "Optimal Probabilistic Fingerprint Codes," in [*35th Annual ACM Symposium on Theory of Computing (STOC)*], 116–125 (2003).

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

[3] Furon, T., Guyader, A., and Cérou, F., "On the design and optimization of Tardos probabilistic fingerprinting codes," in [*Information Hiding (IH)*], *LNCS* **5284**, 341–356, Springer (2008).

[4] Furon, T., Pérez-Freire, L., Guyader, A., and Cérou, F., "Estimating the minimal length of Tardos code," in [*Information Hiding (IH)*], *LNCS* **5806**, 176–190 (2009).

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

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

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

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

[9] Nuida, K., "Short collusion-secure fingerprint codes against three pirates," in [*Information Hiding (IH)*], *LNCS* **6387**, 86–102 (2010).

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

[11] Amiri, E. and Tardos, G., "High rate fingerprinting codes and the fingerprinting capacity," in [*20th Annual ACM-SIAM Symposium on Discrete Algorithms (SODA)*], 336–345 (2009).

[12] Charpentier, A., Xie, F., Fontaine, C., and Furon, T., "Expectation maximization decoding of Tardos probabilistic fingerprinting code," in [*SPIE Electronic Imaging/Media Forensics and Security*], *SPIE Proceedings* **7254**, 72540 (2009).

[13] Meerwald, P. and Furon, T., "Towards joint Tardos decoding: the 'Don Quixote' algorithm," in [*Information Hiding (IH)*], *LNCS* **6958**, 28–42 (2011).

[14] Charpentier, A., Fontaine, C., Furon, T., and Cox, I., "An asymmetric fingerprinting scheme based on Tardos codes," in [*Information Hiding (IH)*], *LNCS* **6958**, 43–58 (2011).

[15] Škorić, B., Katzenbeisser, S., and Celik, M. U., "Symmetric Tardos Fingerprinting Codes for Arbitrary Alphabet Sizes," *Designs, Codes and Cryptography* **46**(2), 137–166 (2008).

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

[17] Xie, F., Furon, T., and Fontaine, C., "On-off keying modulation and Tardos fingerprinting," in [*10th ACM Workshop on Multimedia and Security (MMSec)*], 101–106 (2008).

[18] Moulin, P., "Universal fingerprinting: Capacity and random-coding exponents," arXiv:`0801.3837v3` `[cs.IT]` (2011).

[19] Škorić, B. and Oosterwijk, J.-J., "Binary and q-ary Tardos codes, revisited." Cryptology ePrint Archive, Report 2012/249 (2012).

[20] Boesten, D. and Škorić, B., "Asymptotic fingerprinting capacity for non-binary alphabets," in [*Information Hiding (IH)*], *LNCS* **6958**, 1–13 (2011).

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

[22] Huang, Y.-W. and Moulin, P., "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).

[23] Laarhoven, T., Doumen, J., Roelse, P., Škorić, B., and de Weger, B., "Dynamic Tardos traitor tracing schemes," *IEEE Transactions on Information Theory* **59**(7), 4230–4242 (2013).

[24] Laarhoven, T., Oosterwijk, J.-J., and Doumen, J., "Dynamic traitor tracing for arbitrary alphabets: Divide and conquer," in [*IEEE International Workshop on Information Forensics and Security (WIFS)*], 240 –245 (2012).

[25] Oosterwijk, J.-J., Škorić, B., and Doumen, J., "A capacity-achieving simple decoder for bias-based traitor tracing schemes." Cryptology ePrint Archive, Report 2013/389 (2013).

[26] Oosterwijk, J.-J., Škorić, B., and Doumen, J., "Optimal suspicion functions for Tardos traitor tracing schemes," in [*1st ACM workshop on Information hiding and Multimedia Security (IHMMSec)*], 19–28 (2013).