Symmetric Tardos fingerprinting codes for arbitrary alphabet sizes

B. Škorić, S. Katzenbeisser, M.U. Celik

Abstract

Fingerprinting provides a means of tracing unauthorized redistribution of digital data by individually marking each authorized copy with a personalized serial number. In order to prevent a group of users from collectively escaping identification, collusion-secure fingerprinting codes have been proposed. In this paper, we introduce a new construction of a collusion-secure fingerprinting code which is similar to a recent construction by Tardos but achieves shorter code lengths and allows for codes over arbitrary alphabets.

For binary alphabets, n users and a false accusation probability of η, a code length of $m \approx \pi^2 c_0^2 \ln(n/\eta)$ is provably sufficient to withstand collusion attacks of at most c_0 colluders. This improves Tardos' construction by a factor of 10. Furthermore, invoking the Central Limit Theorem we show that even a code length of $m \approx \frac{1}{2}\pi^2 c_0^2 \ln(n/\eta)$ is sufficient in most cases. For q-ary alphabets, assuming the restricted digit model, the code size can be further reduced. Numerical results show that a reduction of 35% is achievable for $q = 3$ and 80% for $q = 10$.

1 Introduction

1.1 Digital fingerprinting

Fingerprinting, or forensic watermarking, provides a means of tracing the unauthorized redistribution of digital data, such as entertainment content (i.e. music or movie clips), digital records or software. Before authorized distribution, the distributor imperceptibly embeds a *fingerprint*, which plays the role of a personalized serial number, directly into the content. This is done using a digital watermarking algorithm. If the fingerprint is different for each recipient (also called 'user'), the distributor can extract the embedded fingerprint from an unauthorized copy of the content and trace the recipient who leaked it.

Mathematically speaking, a fingerprint is a finite string over some q-ary alphabet Σ; the set of all fingerprints is called a *fingerprinting code*. Throughout this paper we will denote by n the number of users and by m the length of the fingerprint. In order to mark a piece of content before distribution, the distributor picks a fingerprint from the code and imperceptibly embeds each symbol of the fingerprint into different segments of the content, such as in different scenes of a movie. In addition, he stores in a database the association of a fingerprint with the identity of the user who received the personalized copy. In case an unauthorized copy of the content is found, the distributor can perform watermark detection on the segments of the content to read out its fingerprint. Once the fingerprint is retrieved, he can compare it with his database of fingerprints to identify the guilty user. Current watermarking schemes provide a considerable level of robustness that allows correct reconstruction of the fingerprint even if the content has suffered heavy distortions.

1.2 Collusion resistance

Fingerprinting schemes need to be robust against *collusion attacks*, where several users pool different individualized versions of the same content. By looking at the differences between these versions, the colluding users (also referred to as 'colluders' or 'the coalition') try to produce an untraceable version of the content, from which the distributor cannot identify any of the colluders. A segment of the content is called a *detectable position* if the colluders have at least two differently marked versions of that segment available.

A code is called collusion-resistant against a coalition of size c_0, if any set of $c \leq c_0$ colluders is unable to produce an untraceable copy. The construction of collusion-resistant codes has been an active research topic since the late 1990s (see e.g. [5, 8, 3, 6, 9]). The constructions and the achieved results depend strongly on various assumptions which restrict the type of manipulations the attackers are allowed to perform. One often made assumption is the *marking condition*, stating that the colluders are able to change fingerprint symbols only in detectable positions. Throughout this paper we will assume that the marking condition holds. Furthermore, several attack models have been introduced in the literature:

- The *restricted digit model* or *narrow-case model* allows the colluders only to 'mix and match' their copies of the content, i.e. to replace a segment in a detectable position by any other segment they have available in that position. On the fingerprinting code level, this means that in the unauthorized copy the symbol at each position can only be one of the symbols that they have available in that position.
- The *unreadable digit model* allows for slightly stronger attacks. Besides mixing the content segments, the attackers can also erase the embedded fingerprint at detectable positions. At the code level, we denote this by a special erasure symbol $? \notin \Sigma$.
- The *arbitrary digit model* allows for even stronger attacks: the attackers can put an arbitrary q-ary symbol from Σ (but not the erasure symbol '?') in detectable positions.
- The *general digit model* allows the attackers to put any symbol, including '?', in detectable positions.

Note that in the case of a binary alphabet all four attack models are equivalent in terms of traceability. (For $q = 2$ it is detrimental for the colluders to use '?', since it gives the distributor more information than a '0' or '1', namely that the position is a detectable position for the coalition).

The main parameters of a fingerprinting code are the *codeword length*, the *False Positive* (FP) error probability and the *False Negative* (FN) error probability. The codeword length influences to a great extent the practical usability of a fingerprinting scheme, as the number of segments m that can be used to embed a fingerprint symbol is severely constrained; typical video watermarking algorithms for instance can only embed 7 bits of information in a robust manner in one minute of a video clip. Furthermore, the amount of information that can be embedded per segment is limited; hence the alphabet size q must be small (typically $q \leq 16$). Obviously, distributors are interested in the shortest possible codes that are secure against a large number of colluders, while accommodating a huge number n of users (of the order of $n \approx 10^6$ or even $n \approx 10^9$).

Low error probabilities are another central requirement. The most important type of error is the FP, where an innocent user gets accused. The probability of such an event must be extremely small; otherwise the distributor's accusations would be questionable, making the whole fingerprinting scheme unworkable. We will denote by ε_1 the probability that one specific user gets falsely accused, while η denotes the probability that there are innocent users among the accused. The second type of error is the FN, where the scheme fails to accuse any of the colluders. The FN probability will be denoted as ε_2. In practical situations, fairly large values of ε_2 can be tolerated. Often the objective of fingerprinting is to *deter* unauthorized distribution rather than to prosecute all those responsible for it. Even a mere 50% probability of getting caught is a significant deterrent for colluders.

1.3 Related work

For the restricted digit model, 'deterministic' fingerprinting codes have been proposed. Here 'deterministic' means that the error probabilities ε_1 and ε_2 are zero. Identifiable Parent Property

(IPP) codes, introduced in [5], allow the distributor to identify at least one member of the coalition with certainty, without the danger of accusing innocent people. However, the schemes proposed in [5] are not resistant against more than two colluders. In [8] the existence was proved of a deterministic fingerprinting code resistant against c_0 colluders, having code length $m = c_0^2 \log_q(n)$. However, the alphabet size is impractically large, requiring $q \geq n - 1$.

More efficient fingerprinting schemes are possible if nonzero error probabilities η and ε_2 are tolerated. In [3] Boneh and Shaw presented a binary scheme ($q = 2$) with code length $m = \mathcal{O}(c_0^4 \log \frac{n}{\eta} \log \frac{1}{\eta})$. Their scheme uses concatenation of a partly randomized inner code with an outer code. They also proved, for binary alphabets, a lower bound on the code length required for resistance against c_0 colluders: $m > \mathcal{O}(c_0 \log \frac{1}{c_0 \eta})$. In [6] Peikert et al. proved a tighter lower bound of $m > \mathcal{O}(c_0^2 \log \frac{1}{c_0 \eta})$.

In [9] Tardos further tightened the lower bound to $m > \mathcal{O}(c_0^2 \log \frac{n}{\eta})$. This bound is valid for arbitrary alphabets in the arbitrary digit model and the unreadable digit model. In the same paper, he described a fully randomized binary fingerprinting code achieving this lower bound. The code has length $m = 100 c_0^2 \ln \frac{n}{\eta}$; a construction was given only for the binary alphabet. In [7] Tardos' construction was further analyzed. It was shown that, without changing the scheme, the constant '100' can be reduced to $4\pi^2$. In the same paper it was shown that an important quantity in the scheme (the 'accusation sum', see Section 2.1), resulting from the summation of many i.i.d. stochastic variables, has a Gaussian distribution, up to correction terms that vanish for large c_0. Without changing the Tardos scheme in any way, but assuming a Gaussian distribution, the code length m was further reduced to $m = 2\pi^2 c_0^2 \ln \frac{n}{\eta}$.

1.4 Contributions and outline

In this paper, we propose a new construction of a fingerprinting code, which is similar in spirit to Tardos' original code, but allows for codes over arbitrary-size alphabets. For binary alphabets the new scheme allows for codes that are a factor 4 shorter than the construction given by [7] (and thus a factor 10 shorter than the scheme given in [9]). In the restricted digit attack model, moving from a binary to a q-ary alphabet allows for even shorter fingerprinting codes. The key contributions of the paper are summarized as follows:

- In Section 2 we review Tardos' binary fingerprinting scheme [9] and propose a different construction, which is symmetric and which can be used for arbitrary alphabets Σ. The construction is different from Tardos' code even for binary alphabets.

- In Section 4 we study the collusion resistance of the symmetric code. We apply the methods of [9] to rigorously prove a lower bound on the code length m, such that the desired error rates are achieved. The bound is given by $m > 4\tilde{\mu}^{-2} c_0^2 \ln \frac{n}{\eta}$, where the quantity $\tilde{\mu}$ is the expectation value of the coalition's collective 'suspiciousness'.

- In Section 5 we compute the expectation value $\tilde{\mu}$ in the restricted digit model. In the case of a binary alphabet we have $\tilde{\mu} = 2/\pi$. This corresponds to a bound on the code length of $m > \pi^2 c_0^2 \ln \frac{n}{\eta}$, which is a factor 4 shorter than the bound obtained for the Tardos scheme in [7] and a factor ≈10 shorter than the bound given in [9]. For q-ary alphabets we compute $\tilde{\mu}$ numerically. The code length m is further reduced (with respect to the binary symmetric scheme) by 40% for $q = 3$ and by 80% for $q = 10$.

- In Section 6 we make use of the Central Limit Theorem to show that an important quantity in the scheme, the accusation sum of an innocent user, has a probability density that is almost Gaussian. Convergence to the normal form improves with increasing c_0. Approximation of the distribution by a Gaussian is accurate starting from a value of c_0 between 10 and 20. Assuming a perfect normal distribution, we show that the desired error rates are achieved for $m > 2\tilde{\mu}^{-2} c_0^2 \ln \frac{n}{\eta}$. This is a factor 2 shorter than the code length derived in Section 4 without any assumptions.

2 Symmetric Tardos fingerprinting for arbitrary alphabet sizes

In this section we first introduce Tardos' initial binary fingerprinting code [9] and then provide a generalization for arbitrary alphabets.

2.1 The Tardos fingerprinting scheme

Let n be the number of users to be accomodated in the system. The Tardos fingerprinting scheme distributes a binary codeword of length m to each user; the length m is a system parameter chosen by the distributor. It affects the FP and FN error rates. The distributed codewords can be arranged as an $n \times m$ matrix $\mathbf{X}$, where the j-th row corresponds to the fingerprint given to the j-th user. Let C be a set of colluding users. We denote by c the number of colluders and by $\mathbf{X}_C$ the $c \times m$ matrix of codewords distributed to the colluders. The colluders use a (possibly nondeterministic) strategy ρ to create an unauthorized copy of the content from their personalized copies. The unauthorized copy carries a fingerprint $y \in \{0,1\}^m$ which depends on both the strategy and the received codewords, i.e. $y = \rho(\mathbf{X}_C)$.

Fingerprint code generation. The distributor generates the matrix $\mathbf{X}$ in two randomized steps. In the first step, he chooses m random variables $\{p_i\}_{i=1}^m$ over the interval $p_i \in [t, 1-t]$, where t is a fixed small parameter satisfying $c_0 t \ll 1$. The variables p_i are independent and identically distributed according to the probability density function f. The function $f(p)$ is symmetric[1] around $p = 1/2$ and heavily biased towards values of p close to t and $1-t$,

$$f(p) = \frac{1}{2\arcsin(1-2t)} \frac{1}{\sqrt{p(1-p)}}. \tag{1}$$

In the second step, the distributor fills the columns of the matrix $\mathbf{X}$ by independently drawing random bits $X_{ji} \in \{0,1\}$ according to $\mathbb{P}[X_{ji} = 1] = p_i$.

Fingerprint embedding. Before the content is realeased to customer j, it is watermarked with the j-th row of the matrix $\mathbf{X}$.

Accusation. Having spotted an unauthorized copy with embedded watermark y, the content owner wants to identify at least one colluder. To achieve this, he computes for each user $1 \le j \le n$ an accusation sum S_j as

$$S_j = \sum_{i=1}^{m} y_i \, U(X_{ji}, p_i), \quad \text{with} \quad U(X_{ji}, p_i) = \begin{cases} g_1(p_i) & \text{if } X_{ji} = 1 \\ g_0(p_i) & \text{if } X_{ji} = 0, \end{cases} \tag{2}$$

where g_1 and g_0 are the 'accusation functions'

$$g_1(p) = \sqrt{\frac{1-p}{p}} \quad \text{and} \quad g_0(p) = -\sqrt{\frac{p}{1-p}}. \tag{3}$$

The distributor decides that user j is guilty if $S_j > Z$. The parameter Z is called the 'accusation threshold'. The threshold is a system parameter chosen by the distributor.

In words, the accusation sum S_j is computed by summing over all symbol positions i in y. All positions with $y_i = 0$ are ignored. For each position where $y_i = 1$, the accusation sum S_j is either increased or decreased, depending on how much suspicion arises from that position: if user j has a '1' in that position, then the accusation is increased by a positive amount $g_1(p_i)$. Note that the suspicion decreases with higher probability p_i, since g_1 is a positive monotonically decreasing function. If user j has a '0', the accusation is corrected by the negative amount $g_0(p_i)$, which gets more pronounced for large values of p_i, as g_0 is negative and monotonically decreasing.

Tardos chose the specific form (3) for the functions g_1 and g_0 because it has nice properties: For fixed p_i, the accusation $U(X_{ji}, p_i)$ in (2) has zero mean and unit variance. Especially the fact that the variance does not depend on p_i greatly simplifies the analysis of the scheme. It was

[1] In [9] the parametrization $p_i = \sin^2 r_i$ is used, and the density function for r_i is specified.

shown in [7] that for Tardos' scheme the choice (1) for f is optimal, and that the choice (3) for the accusation functions is optimal within the class of functions of the form $p^{z_1}(1-p)^{z_2}$, where z_1 and z_2 are constants.

Tardos chose the system parameters m and Z as follows:

$$m = Ac_0^2 \lceil \ln \varepsilon_1^{-1} \rceil \quad ; \quad Z = Bc_0 \lceil \ln \varepsilon_1^{-1} \rceil, \tag{4}$$

with $A = 100$ and $B = 20$. Recall from Section 1.2 that the parameter ε_1 is a re-scaled version of the false positive error parameter η. It represents the probability that a *specific* innocent user j gets accused. The relation between η and ε_1 is $\eta = 1-(1-\varepsilon_1)^{n-c}$. For $\varepsilon_1 \ll 1$ and $c \ll n$ this becomes $\eta \approx (n-c)\varepsilon_1 \approx n\varepsilon_1$.

A False Negative (FN) is defined as the event where none of the colluders are accused. Tardos proved in [9] that his scheme achieves FP and FN error rates smaller than ε_1 and ε_2, respectively, against coalitions of size $c \le c_0$, for $\varepsilon_2 = \varepsilon_1^{c_0/4}$. In [7] the Tardos scheme was further analyzed and the following results were obtained for $\varepsilon_2 \gg \varepsilon_1$ (a far more reasonable choice of parameters, see Section 1.2): *(i)* the code length parameter A in (4) can be reduced to $4\pi^2$. *(ii)* The accusation sum S_j has an almost Gaussian probability density function, with corrections that vanish in the limit $c_0 \to \infty$. *(iii)* Assuming a perfect Gaussian distribution for S_j, the parameter A can be reduced to $2\pi^2$. Hence, for sufficiently large c_0, the code length m can be set to $m = 2\pi^2 c_0^2 \lceil \ln \varepsilon_1^{-1} \rceil$ without any modification of the code construction, embedding or accusation method.

2.2 Proposed symmetric fingerprinting scheme

The scheme presented in Section 2.1 has two drawbacks. First, the computation of Tardos' accusation sum (2) is asymmmetric in the sense that only those codeword positions i contribute where $y_i = 1$, while all the others are discarded. This is an inefficient way of exploiting the information present in the unauthorized copy, because the $y_i = 0$ positions carry as much information about the colluders as the $y_i = 1$ positions. Second, due to this asymmetry, the construction cannot be directly applied to nonbinary alphabets.

We apply two modifications to Tardos' construction. The first modification is a straightforward generalization of the fingerprint generation step to produce a random q-ary code. Instead of bits we have $\mathbf{X}_{ji} \in \Sigma$, with $\Sigma = \{0, 1, \cdots, q-1\}$. Instead of scalar random variables p_i we have, independently for each column, a q-component random vector $\mathbf{p}^{(i)} = (p_0^{(i)}, \cdots, p_{q-1}^{(i)})$, with $\sum_{\alpha=0}^{q-1} p_\alpha^{(i)} = 1$. The vectors $\mathbf{p}^{(i)}$ have the probability density function $F(\mathbf{p})$, which replaces $f(p)$. While f is invariant under the mapping $p \to 1-p$, our function F is invariant under any permutation of the symbols $\alpha \in \Sigma$. Thus our construction is symmetric in all symbols $\alpha \in \Sigma$. In the i-th column of $\mathbf{X}$, random symbols are drawn with probabilities dictated by $\mathbf{p}^{(i)}$. The colluders create an unauthorized copy $y = \rho(\mathbf{X}_C) \in \Sigma^m$ according to a (possibly non-deterministic) strategy ρ. We will always assume that this strategy does not depend on the column index, i.e. the same strategy ρ is applied to each column of $\mathbf{X}_C$. We can make this assumption without loss of generality; due to the column symmetry of $\mathbf{p}$ and $\mathbf{X}$, the best colluder strategy is column-symmetric.

The second modification lies in the computation of the accusation sum. In contrast to Tardos' scheme, we let every fingerprint symbol in the unauthorized copy give rise to accusations. The accusation for a certain user at a certain symbol location is positive if he has the same symbol as the unauthorized copy; otherwise it is negative. The magnitude of the accusation depends on the likelihood of the symbol that appears in the unauthorized copy. In full detail the proposed construction is as follows:

Fingerprint code generation. As in the original Tardos construction, the distributor produces an $n \times m$ matrix $\mathbf{X}$ of q-ary symbols; the rows of the matrix correspond to the fingerprints for the individual users. We parametrize m as in (4); the value of the parameter A is the subject of Sections 4, 5 and 6. Again, the distributor uses a two-step procedure:

1. He generates m independent random vectors $\mathbf{p}^{(i)} = (p_0^{(i)}, \cdots, p_{q-1}^{(i)})$ for $1 \le i \le m$, where

the components satisfy $p_\alpha^{(i)} \in [t/(q-1), 1-t]$ and $\sum_{\alpha=0}^{q-1} p_\alpha^{(i)} = 1$. We call t the 'cutoff parameter' or the 'cutoff'. It satisfies $0 < t \ll 1$; we parametrize it as $t = Tc_0^{-a}$, with $T > 0$ and $a \in (0, 2)$. The random variables have a probability density function that is symmetric in all the components p_α. In our construction, we use a class of functions that are a special case of the Dirichlet distribution (see e.g. [4]),

$$F_{q\kappa t}(\mathbf{p}) = \mathcal{N}_{q\kappa t}^{-1} \prod_{\alpha=0}^{q-1} p_\alpha^{-1+\kappa} \qquad \text{with } \kappa > 0. \tag{5}$$

Here $\mathcal{N}_{q\kappa t}$ is a normalising constant ensuring that $\int_{J(t,q)} \mathrm{d}^q\mathbf{p}\; F_{q\kappa t}(\mathbf{p}) = 1$. The expression $\int_{J(t,q)} \mathrm{d}^q\mathbf{p}$ stands for $\int_{\frac{t}{q-1}}^{1-t} \mathrm{d}p_0 \cdots \int_{\frac{t}{q-1}}^{1-t} \mathrm{d}p_{q-1} \delta(1 - \sum_{\beta=0}^{q-1} p_\beta)$, where $\delta(\cdot)$ is the Dirac delta function. The delta function ensures that the integration is done only over $\mathbf{p}$ such that $\sum_\beta p_\beta = 1$. The parameter κ determines the steepness of $F_{q\kappa t}$. For $q = 2$, $\kappa = \frac{1}{2}$ the function $F_{q\kappa t}$ reduces to Tardos' density function (1).

2. The distributor generates the columns of $\mathbf{X}$ independently. In the i-th column, the vector $\mathbf{p}^{(i)}$ determines the probabilities of generating each specific symbol in the alphabet:

$$\mathbb{P}[X_{ji} = \alpha] = p_\alpha^{(i)}. \tag{6}$$

Fingerprint embedding. Before the content is realeased to customer j, it is watermarked with the j-th row of the matrix $\mathbf{X}$.
Accusation. The distributor extracts the fingerprint y from the unauthorized copy. For each user j, the distributor computes the 'accusation sum' $\mathcal{A}_j$ from $\mathbf{X}$, $\mathbf{p}$ and y. He decides that the user j is guilty if $\mathcal{A}_j > Z$, where Z is referred to as the 'accusation threshold'. We parametrize Z as in (4), with the constant B as yet left undetermined. The list of accused users is denoted as $\sigma(\mathbf{p}, \mathbf{X}, y)$. The accusation sum $\mathcal{A}_j$ is given by

$$\mathcal{A}_j(\mathbf{p}, \mathbf{X}, y) = \sum_{i=1}^{m} \mathcal{A}_j^{(i)} \quad ; \quad \mathcal{A}_j^{(i)} := \delta_{y_i, X_{ji}} g_1(p_{y_i}^{(i)}) + [1 - \delta_{y_i, X_{ji}}] g_0(p_{y_i}^{(i)}), \tag{7}$$

where $\delta_{x,y}$ denotes the Kronecker delta. We have chosen the same functions $g_0(p) = \sqrt{(1-p)/p}$, $g_1(p) = -\sqrt{p/(1-p)}$ as Tardos. There is no guarantee that this choice is optimal for $q > 2$. The choice is motivated by the zero-mean, unit-variance property mentioned in Section 2.1; this property leads to a substantial simplification of the analysis in the coming sections.

In words, the accusation (7) is computed as follows. If user j has the same symbol in position i as the unauthorized copy, then he is accused by a positive amount $g_1(p_{y_i}^{(i)})$, where the accusation decreases with growing likelihood of the symbol. If user j has a different symbol than the unauthorized copy, then he is accused by a negative amount $g_0(p_{y_i}^{(i)})$, which has the largest effect when the symbol y_i is likely to occur.

Note that (7) is fully symmetric in the symbols and that it differs from Tardos' construction even for $q = 2$. Note further that the Kronecker deltas in (7) reduce the symbol space into two classes: $X_{ji} = y_i$ and $X_{ji} \neq y_i$. In the latter case the accusation does not depend on the actual value of X_{ji}.
Attack model. As mentioned in Section 1.2, we use the marking assumption and we assume that the restricted digit model holds. In addition, we make two assumptions on the attack strategy of the colluders. First, we assume that all members of the coalition are equivalent. Hence, they base their decisions only on the number of symbols they receive, and not on the identity of the members who receive them. (Any deviation from this strategy will make it easier for the distributor to identify a colluder). Second, we assume that the colluders' strategy applies to each watermark position independently. This is not a restrictive assumption, since the columns of $\mathbf{X}$ are independent.

3 Preliminaries

In order to facilitate our work in Sections 4 and 5, we introduce some notation and state a number of lemmas.

3.1 Normalisation constant

The value of the normalisation constant $\mathcal{N}_{q\kappa t}$ in (5) is easily computed for $t = 0$, using the following lemma (see e.g. [1]):

Lemma 1: *Let* $\mathbf{v}$ *be a vector of length* q *with* $v_\alpha > 0$ *for* $0 \le \alpha \le q-1$*. Then*

$$\int_{J(0,q)} \mathrm{d}^q\mathbf{p} \prod_{\beta=0}^{q-1} p_\beta^{-1+v_\beta} = B(\mathbf{v}) := \frac{\prod_{\alpha=0}^{q-1}\Gamma(v_\alpha)}{\Gamma(\sum_{\beta=0}^{q-1} v_\beta)}.$$

The function B is the generalized Beta function, also referred to as the multinomial Beta function or Dirichlet integral.
Proof sketch: For two components ($q = 2$) the lemma is true, as the integral yields the ordinary Beta function. For higher q the lemma can be proved by induction. □
For $t \neq 0$, $q = 2$ the integral yields the so-called incomplete Beta function.

Applying Lemma 1 to the definition of $F_{q\kappa t}$ in (5), we compute the normalisation factor $\mathcal{N}_{q\kappa t}$ for $t = 0$ to be

$$\mathcal{N}_{q\kappa 0} = \frac{[\Gamma(\kappa)]^q}{\Gamma(\kappa q)}. \tag{8}$$

Remark: The difference between $\mathcal{N}_{q\kappa t}$ and $\mathcal{N}_{q\kappa 0}$ is small. This is seen as follows. The integrand in $\mathcal{N}_{q\kappa t}$ is of the form $\prod_\beta p_\beta^{-1+\kappa}$ with $\kappa > 0$. The primitive function near a pole at $p_\alpha = 0$ scales as p_α^κ. Hence the contributions from the poles, present in $\mathcal{N}_{q\kappa 0}$ and absent in $\mathcal{N}_{q\kappa t}$, are of order t^κ. If κ is not extremely close to 0, then $t^\kappa \ll 1$.

3.2 Collective accusation sum

Let C be the set of colluding users and $\mathbf{X}_C$ the restriction of $\mathbf{X}$ to the rows received by the colluders. From (7) we define a useful quantity: the 'collective accusation sum' $\mathcal{A}_C$, being the sum of all individual accusation sums of the coalition members,

$$\mathcal{A}_C = \sum_{j\in C}\mathcal{A}_j = \sum_{i=1}^{m}\mathcal{A}_C^{(i)} \quad ; \quad \mathcal{A}_C^{(i)} := b_{y_i}^{(i)} g_1(p_{y_i}^{(i)}) + [c - b_{y_i}^{(i)}] g_0(p_{y_i}^{(i)}). \tag{9}$$

Here $b_\alpha^{(i)}$ stands for the number of occurrences of the symbol α in column i of $\mathbf{X}_C$. These numbers satisfy the constraint $\sum_{\alpha=0}^{q-1} b_\alpha^{(i)} = c$. The sum $\mathcal{A}_C$ plays an important role in the FN error rate.

3.3 Definition of averages

There are three stochastic processes involved in the creation of the fingerprinting codewords and the unauthorized copy: The distributor's choice of vectors $\mathbf{p}^{(i)}$, his process of generating the columns of $\mathbf{X}$, and the coalition's choice of symbols y_i. For each process we define a separate expectation value. Averaging over $\mathbf{p}$ is denoted as $\mathbb{E}_p$. Within the i-th column this is defined as

$$\mathbb{E}_p[\zeta(\mathbf{p}^{(i)})] := \int_{J(t,q)} \mathrm{d}^q\mathbf{p}\ \zeta(\mathbf{p}) F_{q\kappa t}(\mathbf{p}), \tag{10}$$

for an arbitrary function ζ. Here $F_{q\kappa t}$ is the probability density function (5).

We remind the reader that all the vectors $\mathbf{p}^{(i)}$ are independent. We denote the codeword received by user j as X_j. For given $\mathbf{p}$, averaging over X_j is denoted as $\mathbb{E}_{X_j}$. We define

$$\mathbb{E}_{X_j}[\zeta(X_{ji})] := \sum_{\alpha=0}^{q-1} \zeta(\alpha) p_\alpha^{(i)}. \tag{11}$$

In particular we have, for an innocent user j,

$$\mathbb{E}_{X_j}[\delta_{y_i, X_{ji}}] = p_{y_i}^{(i)}. \tag{12}$$

For fixed $\mathbf{p}$, averaging over $\mathbf{X}_C$ is equivalent to averaging over the integers b (see Eq.9). The $b_\alpha^{(i)}$ are distributed according to a multinomial distribution. We have

$$\mathbb{E}_b[\zeta(\mathbf{b}^{(i)})] := \sum_{\mathbf{b}} \zeta(\mathbf{b}) \binom{c}{\mathbf{b}} \prod_{\alpha=0}^{q-1} [p_\alpha^{(i)}]^{b_\alpha}. \tag{13}$$

The notation $\binom{c}{\mathbf{b}}$ stands for the multinomial $c!/(b_0! \cdots b_{q-1}!)$. The sum $\sum_{\mathbf{b}}$ stands for summation over all q components of $\mathbf{b}$, with the condition $\sum_\alpha b_\alpha = c$ implicitly assumed,

$$\sum_{\mathbf{b}} \zeta(\mathbf{b}) = \sum_{b_0=0}^{c} \cdots \sum_{b_{q-1}=0}^{c} \delta_{c, b_0+b_1+\cdots+b_{q-1}} \zeta(\mathbf{b}). \tag{14}$$

Finally we have to deal with the stochastic strategy of the coalition. We introduce the notation $P_{\mathbf{b}}(\alpha)$ for the probability that the colluders output the symbol $y = \alpha$ in a certain position, given that they received symbols according to $\mathbf{b}$. Averaging over y is denoted as $\mathbb{E}_y$,

$$\mathbb{E}_y[\zeta(y_i)] := \sum_{\alpha=0}^{q-1} \zeta(\alpha) P_{\mathbf{b}^{(i)}}(\alpha). \tag{15}$$

The expectation value taken over all stochastic degrees of freedom is denoted as $\mathbb{E}_{yXp}$. It can be computed by first taking the expectation value $\mathbb{E}_y$ (15) for fixed $\mathbf{b}$, then for fixed $\mathbf{p}$ taking $\mathbb{E}_b$ (13) and $\mathbb{E}_{X_j}$ (11) for all innocent users j, and finally $\mathbb{E}_p$ (10). Note that several orderings are possible. For instance, the expectation $\mathbb{E}_{X_j}$ (for innocent j) can be taken before $\mathbb{E}_y$ and $\mathbb{E}_b$, since y and $\mathbf{b}$ do not depend on the codewords given to innocent users.

3.4 Statistical properties of the accusation sums

To facilitate the analysis in the coming sections we introduce 'scaled' averages and variances, defined such that they do not depend on m. For an innocent user j we define

$$\tilde{\mu}_j = \frac{\mathbb{E}_{yXp}[\mathcal{A}_j]}{m} = \mathbb{E}_{yXp}[\mathcal{A}_j^{(i)}] \quad ; \quad \tilde{\sigma}_j^2 = \frac{\mathbb{E}_{yXp}[\mathcal{A}_j^2] - \mathbb{E}_{yXp}^2[\mathcal{A}_j]}{m}. \tag{16}$$

For the collective accusation we define

$$\tilde{\mu} = \frac{\mathbb{E}_{yXp}[\mathcal{A}_C]}{m} = \mathbb{E}_{yXp}[\mathcal{A}_C^{(i)}] \quad ; \quad \tilde{\sigma}^2 = \frac{\mathbb{E}_{yXp}[\mathcal{A}_C^2] - \mathbb{E}_{yXp}^2[\mathcal{A}_C]}{m}. \tag{17}$$

The column index i in $\mathcal{A}_C^{(i)}$ in (17) and $\mathcal{A}_j^{(i)}$ in (16) can be chosen arbitrarily; the result does not depend on i. The quantites $\tilde{\mu}_j$, $\tilde{\sigma}_j$ and $\tilde{\sigma}$ are discussed below, whereas Section 5 is devoted to computing $\tilde{\mu}$.

Lemma 2: *For an innocent user j we have $\tilde{\mu}_j = 0$.*

Proof: We evaluate the expectation $\mathbb{E}_{yXp}$ by first computing the expectation $\mathbb{E}_{X_j}$. We apply (12) to the definition of $\mathcal{A}_j^{(i)}$ (7). This gives $\mathbb{E}_{X_j}[\mathcal{A}_j^{(i)}] = p_{y_i}^{(i)} g_1(p_{y_i}^{(i)}) + (1-p_{y_i}^{(i)}) g_0(p_{y_i}^{(i)}) = 0$. The last equality follows from the definition (3) of g_1 and g_0. From $\mathbb{E}_{X_j}[\mathcal{A}_j^{(i)}] = 0$ it follows that $\mathbb{E}_{yXp}[\mathcal{A}_j^{(i)}] = 0$. □

Lemma 3: *For an innocent user j we have $\tilde{\sigma}_j = 1$.*

Proof: Using the idempotency of the Kronecker deltas in the definition of $\mathcal{A}_j^{(i)}$ in (7) we write

$$\mathcal{A}_j^2 = \sum_{i=1}^{m} \left\{ \delta_{y_i, X_{ji}} \frac{1-p_{y_i}^{(i)}}{p_{y_i}^{(i)}} + (1-\delta_{y_i,X_{ji}}) \frac{p_{y_i}^{(i)}}{1-p_{y_i}^{(i)}} \right\} + \sum_{\substack{1\le i,k\le m \\ i\neq k}} \mathcal{A}_j^{(i)} \mathcal{A}_j^{(k)}. \tag{18}$$

We evaluate $\mathbb{E}_{yXp}$ by first computing the expectation $\mathbb{E}_{X_j}$. Using property (12) and independence of the columns of $\mathbf{X}$, we get

$$\mathbb{E}_{X_j}[\mathcal{A}_j^2] = \sum_{i=1}^{m} \mathbb{E}_{X_j}[1] + \sum_{i,k;i\neq k} \mathbb{E}_{X_j}[\mathcal{A}_j^{(i)}] \mathbb{E}_{X_j}[\mathcal{A}_j^{(k)}] = m + 0. \tag{19}$$

Here we have made use of the property $\mathbb{E}_{X_j}[\mathcal{A}_j^{(i)}] = 0$ (see proof of Lemma 2). From $\mathbb{E}_{X_j}[\mathcal{A}_j^2] = m$ it follows that $\mathbb{E}_{yXp}[\mathcal{A}_j^2] = m$. The definition of $\tilde{\sigma}_j$ in (16) can be rewritten as $\tilde{\sigma}_j = (1/m)\mathbb{E}_{yXp}[\mathcal{A}_j^2] - m\tilde{\mu}_j^2$. Substitution of $\mathbb{E}_{yXp}[\mathcal{A}_j^2] = m$ into this expression and application of Lemma 2 gives $\tilde{\sigma}_j = 1$. □

Lemma 4: *The mean $\tilde{\mu}$ and variance $\tilde{\sigma}$ satisfy*

$$\tilde{\mu}^2 + \tilde{\sigma}^2 < qc. \tag{20}$$

Proof: From the definitions of $\tilde{\mu}$, $\tilde{\sigma}$ (17) and $\mathcal{A}_C$, $\mathcal{A}_C^{(i)}$ (9) it follows that

$$\begin{aligned} \tilde{\sigma}^2 &= m^{-1}\mathbb{E}_{yXp}[\mathcal{A}_C^2] - m\tilde{\mu}^2 \\ &= m^{-1}\left(\sum_{i=1}^{m} \mathbb{E}_{yXp}[\{\mathcal{A}_j^{(i)}\}^2] + \sum_{i\neq j} \mathbb{E}_{yXp}[\mathcal{A}_j^{(i)}]\mathbb{E}_{yXp}[\mathcal{A}_j^{(k)}] \right) - m\tilde{\mu}^2 \\ &= \mathbb{E}_{yXp}[\{\mathcal{A}_C^{(i)}\}^2] - \tilde{\mu}^2. \end{aligned} \tag{21}$$

Using the idempotency of the Kronecker delta, we write

$$\{\mathcal{A}_C^{(i)}\}^2 = \sum_{\alpha=0}^{q-1} \delta_{\alpha y} [b_\alpha g_1(p_\alpha) + (c-b_\alpha) g_0(p_\alpha)]^2. \tag{22}$$

We apply the total average $\mathbb{E}_{yXp}$ as described in Section 3.3, by first performing $\mathbb{E}_y$, then $\mathbb{E}_X$ and finally $\mathbb{E}_p$. We get

$$\begin{aligned} \mathbb{E}_{yXp}[\{\mathcal{A}_C^{(i)}\}^2] = \sum_{\alpha=0}^{q-1} \sum_{b_\alpha=0}^{c} \binom{c}{b_\alpha} \times \\ \mathbb{E}_p \left[p_\alpha^{b_\alpha} (1-p_\alpha)^{c-b_\alpha} \mathbb{E}_{\mathbf{b}\setminus b_\alpha}[P_{\mathbf{b}}(\alpha)] \{ b_\alpha g_1(p_\alpha) + [c-b_\alpha] g_0(p_\alpha) \}^2 \right]. \end{aligned} \tag{23}$$

Here the notation $\mathbb{E}_{\mathbf{b}\setminus b_\alpha}$ indicates averaging over all degrees of freedom in $\mathbf{b}$ except b_α. Note that the expression in $\mathbb{E}_p[\cdots]$ is always nonnegative. So, using $\mathbb{E}_{\mathbf{b}\setminus b_\alpha}[P_{\mathbf{b}}(\alpha)] \le 1$ we can bound the r.h.s. of (23) by

$$\begin{aligned}\mathbb{E}_{yXp}[\{\mathcal{A}_C^{(i)}\}^2] &< \sum_{\alpha=0}^{q-1}\mathbb{E}_p\left[\sum_{b_\alpha=0}^{c}\binom{c}{b_\alpha}p_\alpha^{b_\alpha}(1-p_\alpha)^{c-b_\alpha}\left\{b_\alpha g_1(p_\alpha)+[c-b_\alpha]g_0(p_\alpha)\right\}^2\right] \\ &= \sum_{\alpha=0}^{q-1}\mathbb{E}_p[c] = qc. \end{aligned} \tag{24}$$

The first equality is obtained by observing, as in [9], that the b_α-sum represents the result of a random walk consisting of c steps, each of which has zero mean and unit variance. (This follows from Lemmas 2 and 3). □

Remark: In Section 5 it will be shown that $\tilde{\mu}$ does not increase as a function of c. Lemma 4 then shows that $\tilde{\sigma} = \mathcal{O}(\sqrt{c})$ for $c \to \infty$. This asymptotic behaviour of $\tilde{\sigma}$ will play an important role in Section 6.

4 Lower bound on the code length in the proposed symmetric scheme

Here we analyze the symmetric scheme described in Section 2.2. We provide a lower bound on the code length m, as a function of the maximum coalition size c_0 and the maximum tolerable FP and FN error probabilities ε_1, ε_2. We define the following two properties:

Property 1: *We say that a fingerprinting scheme that generates a list σ of accused users has Property 1 for a certain fixed value ε_1 if, for all innocent users j, all coalitions C with $j \notin C$, and all coalition strategies, the following holds:*

$$\mathbb{P}[\text{False Positive}] = \mathbb{P}[j \in \sigma] < \varepsilon_1. \tag{25}$$

Property 2: *We say that a fingerprinting scheme that generates a list σ of accused users has Property 2 for certain fixed values c_0, ε_2, if, for all coalitions C of size $c \leq c_0$, and all coalition strategies, it holds that*

$$\mathbb{P}[\text{False Negative}] = \mathbb{P}[C \cap \sigma = \emptyset] < \varepsilon_2. \tag{26}$$

Our bound on the code length is an asymptotic result for $c_0 \gg 1$. We formulate it as follows:

Theorem 1: *Let the code length m and the accusation threshold Z of our symmetric fingerprinting scheme be chosen as*

$$m = Ac_0^2\lceil \ln \varepsilon_1^{-1}\rceil \quad ; \quad Z = Bc_0\lceil \ln \varepsilon_1^{-1}\rceil \tag{27}$$

with $\varepsilon_1 \in (0,1]$ a fixed parameter and

$$A = 4\tilde{\mu}^{-2}(1+\delta)^2 \quad ; \quad B = 4\tilde{\mu}^{-1}(1+\delta), \tag{28}$$

where $\tilde{\mu}$ is defined in (17). Let $\varepsilon_2 \in (0,1]$ be a fixed parameter. For all $\delta > 0$ there exists a sufficiently large c_0 such that the symmetric fingerprinting scheme has Property 1 for parameter ε_1 and Property 2 for parameters c_0, ε_2.

Thus, according to Theorem 1, for large c_0 a code length

$$m > 4\tilde{\mu}^{-2}c_0^2\lceil \ln \varepsilon_1^{-1}\rceil \tag{29}$$

guarantees resistance against coalitions of size $c \leq c_0$.

In Sections 4.1, 4.2 and 4.3 we present a proof of Theorem 1 following the approach of [7], with minor modifications. First, conditions on A, B and c_0 are derived for achieving Properties 1 and 2. Then the lowest value of A is identified within the space of allowed parameters. The value of $\tilde{\mu}$ is determined in Section 5. At this point we already mention that $0 < \lim_{c_0\to\infty}\tilde{\mu} < \infty$. Hence (29) has the asymptotic behaviour $m = \mathcal{O}(c_0^2)$.

4.1 Conditions for satisfying Property 1

We consider a fixed innocent user j. We introduce an auxiliary variable $\alpha_1 > 0$ that allows us to use the Markov inequality,

$$\mathbb{P}[j \in \sigma] = \mathbb{P}[\mathcal{A}_j > Z] = \mathbb{P}[e^{\alpha_1 \mathcal{A}_j} > e^{\alpha_1 Z}] \leq \frac{\mathbb{E}_{X_j}[\exp(\alpha_1 \mathcal{A}_j)]}{\exp(\alpha_1 Z)}. \tag{30}$$

Due to the independence of the columns of $\mathbf{X}$ we can write $\mathbb{E}_{X_j}[\exp(\alpha_1 \mathcal{A}_j)] = \left\{\mathbb{E}_{X_j}[\exp(\alpha_1 \mathcal{A}_j^{(i)})]\right\}^m$. In what follows, we will always restrict α_1 such that $\alpha_1 \mathcal{A}_j^{(i)} \leq 1.7$. This allows us to use the following (easily verified) inequality

$$e^u < 1 + u + u^2 \quad \text{for } u \leq 1.7, \tag{31}$$

so that we can write

$$\mathbb{E}_{X_j}[e^{\alpha_1 \mathcal{A}_j^{(i)}}] < 1 + \alpha_1 \mathbb{E}_{X_j}[\mathcal{A}_j^{(i)}] + \alpha_1^2 \mathbb{E}_{X_j}[\{\mathcal{A}_j^{(i)}\}^2]. \tag{32}$$

We enforce the restriction $\alpha_1 \mathcal{A}_j^{(i)} \leq 1.7$ for all realisations of the stochastic $\mathbf{p}$, $\mathbf{X}$ and y. For negative $\mathcal{A}_j^{(i)}$ all $\alpha_1 > 0$ are allowed. For positive $\mathcal{A}_j^{(i)}$ we must have $\alpha_1 < 1.7/g_1(p_y)$. As g_1 is a monotonously decreasing function, the strongest restriction on α_1 occurs for $p_y = p_{\min} = t/(q-1)$. Hence we restrict α_1 to the interval $(0, 1.7/g_1(\frac{t}{q-1})]$.

From Lemmas 2 and 3 we know that $\mathbb{E}_{X_j}[\mathcal{A}_j^{(i)}] = 0$ and $\mathbb{E}_{X_j}[\{\mathcal{A}_j^{(i)}\}^2] = 1$ for innocent j; thus (32) yields $\mathbb{E}_{X_j}[e^{\alpha_1 \mathcal{A}_j^{(i)}}] < 1 + \alpha_1^2$. Next we apply the inequality

$$1 + u < e^u \quad \text{for } u \neq 0 \tag{33}$$

to write $\mathbb{E}_{X_j}[\exp(\alpha_1 \mathcal{A}_j)] < \exp(m\alpha_1^2)$. Substitution into (30) gives

$$\mathbb{P}[j \in \sigma] < \min_{\alpha_1 \in (0, 1.7/g_1(\frac{t}{q-1})]} e^{\alpha_1(m\alpha_1 - Z)}. \tag{34}$$

Filling in the explicit form for m and Z (27) into (34) we get

$$\mathbb{P}[j \in \sigma] < \min_{\alpha_1 \in (0, 1.7/g_1(\frac{t}{q-1})]} \varepsilon_1^{c_0 \alpha_1 (B - c_0 A \alpha_1)}. \tag{35}$$

The minimum lies at $\alpha_1^* = B/(2c_0 A)$, provided that the upper bound on α_1 is large enough. The condition $1.7/g_1(\frac{t}{q-1}) \geq \alpha_1^*$ can be rewritten as

$$c_0 \geq \left[\left(\frac{B}{3.4 \cdot A}\right)^2 \frac{q-1}{T}\right]^{\frac{1}{2-a}} \left[1 - \frac{T c_0^{-a}}{q-1}\right]^{\frac{1}{2-a}} \approx \left[\left(\frac{B}{3.4 \cdot A}\right)^2 \frac{q-1}{T}\right]^{\frac{1}{2-a}}, \tag{36}$$

where we have used the parametrisation $t = T c_0^{-a}$. Substituton of α_1^* into (35) gives

$$\mathbb{P}[j \in \sigma] < \varepsilon_1^{B^2/4A}. \tag{37}$$

Hence a sufficient condition for Property 1 to be satisfied is that (36) holds and that

$$B^2/4A \geq 1. \tag{38}$$

4.2 Conditions for satisfying Property 2

We start with a lemma that helps us to upper bound the FN error rate.

Lemma 5: *Let C be a coalition of size $c \leq c_0$. We have*

$$\mathbb{P}[C \cap \sigma = \emptyset] \leq \mathbb{P}[\mathcal{A}_C < cZ] \leq \mathbb{P}[\mathcal{A}_C < c_0 Z] \tag{39}$$

Proof: The event $C \cap \sigma = \emptyset$ implies $\mathcal{A}_C < cZ$. $\square$
Remark: $\mathcal{A}_C < cZ$ does not imply $C \cap \sigma = \emptyset$. It can happen that $\mathcal{A}_C < cZ$ while somebody in the coalition *does* get accused.

Next we introduce an auxiliary variable $\alpha_2 > 0$ that allows us to use the Markov inequality,

$$\mathbb{P}[\mathcal{A}_C < c_0 Z] = \mathbb{P}[e^{-\alpha_2 \mathcal{A}_C} > e^{-\alpha_2 c_0 Z}] < \frac{\mathbb{E}_{yXp}[\exp(-\alpha_2 \mathcal{A}_C)]}{\exp(-\alpha_2 c_0 Z)}. \tag{40}$$

The columns of $\mathbf{X}$ are independently generated, and the colluder strategy is the same for each column. This allows us to write $\mathbb{E}_{yXp}[\exp(-\alpha_2 \mathcal{A}_C)] = \{\mathbb{E}_{yXp}[\exp(-\alpha_2 \mathcal{A}_C^{(i)})]\}^m$. We restrict α_2 such that $-\alpha_2 \mathcal{A}_C^{(i)} \leq 1.7$, allowing us to apply inequality (31) to bound the exponential. This gives

$$\mathbb{E}_{yXp}[e^{-\alpha_2 \mathcal{A}_C^{(i)}}] < 1 + \alpha_2 \tilde{\mu} + \alpha_2^2 (\tilde{\mu}^2 + \tilde{\sigma}^2), \tag{41}$$

where we have used the definitions (17). The restriction $-\alpha_2 \mathcal{A}_C^{(i)} \leq 1.7$ holds for any realisation of $\mathbf{p}$ and $\mathbf{X}$. The smallest (most negative) achievable value of $\mathcal{A}_C^{(i)}$ is $c_0 g_0(p_y^{\max}) = c_0 g_0(1-t) = -c_0\sqrt{(1-t)/t}$. Hence the condition on α_2 is satisfied for

$$\alpha_2 \leq \alpha_2^{\max} = 1.7 c_0^{-1} \sqrt{t/(1-t)}. \tag{42}$$

From Lemma 4 we know that $\tilde{\mu}^2 + \tilde{\sigma}^2 < qc$. Thus we have from (41)

$$\mathbb{E}_{yXp}[e^{-\alpha_2 \mathcal{A}_C}] < (1 - \alpha_2 \tilde{\mu} + \alpha_2^2 q c_0)^m < e^{-m\alpha_2 \tilde{\mu}(1 - \alpha_2 c_0 q/\tilde{\mu})}. \tag{43}$$

In the last inequality we have made use of (33). Substitution of (43) into (40) and minimizing over α_2 gives

$$\mathbb{P}[\mathcal{A}_C < c_0 Z] < \min_{\alpha_2 \in (0, \alpha_2^{\max}]} e^{-\alpha_2 [m\tilde{\mu}(1-\alpha_2 c_0 q/\tilde{\mu}) - c_0 Z]}. \tag{44}$$

We choose m and Z such that $m\tilde{\mu}(1 - \alpha_2^{\max} c_0 q/\tilde{\mu}) > c_0 Z$. Hence the minimum in (44) occurs at $\alpha_2 = \alpha_2^{\max}$. Substitution of (27) into (44) and evaluation at $\alpha_2^{\max}$ gives

$$\mathbb{P}[\mathcal{A}_C < c_0 Z] < \varepsilon_1^{1.7 c_0 \sqrt{\frac{t}{1-t}} [A\tilde{\mu}(1-\psi_1) - B]}, \tag{45}$$

where we have introduced the notation $\psi_1 = 1.7\sqrt{\frac{t}{1-t}} q/\tilde{\mu}$. To satisfy Property 2, (45) must not be larger than ε_2. Hence Property 2 is satisfied if

$$A\tilde{\mu}(1-\psi_1) - B \geq \psi_2, \tag{46}$$

where we have defined

$$\psi_2 = \frac{\sqrt{1-t}}{1.7 c_0 \sqrt{t}} \cdot \frac{\ln \varepsilon_2}{\ln \varepsilon_1}. \tag{47}$$

Note that the parameters ψ_1 and ψ_2 go to zero for $c_0 \to \infty$.

4.3 Final step in the proof of Theorem 1

We use the results of Sections 4.1 and 4.2 to prove Theorem 1. The conditions (38) and (46) can be rewritten as an interval for A such that Properties 1 and 2 are both satisfied,

$$\frac{B+\psi_2}{\tilde{\mu}(1-\psi_1)} \leq A \leq \frac{B^2}{4}. \tag{48}$$

A solution exists only if the r.h.s. is not smaller than the l.h.s. in (48). We wish to identify the smallest value of A for with a solution exists. This occurs when the l.h.s. is equal to the r.h.s. Solving the quadratic equation in B gives

$$B = \frac{4}{\tilde{\mu}}(1+\theta) \quad \text{with} \quad \theta := \frac{1+\sqrt{1+\psi_2\tilde{\mu}(1-\psi_1)}}{2(1-\psi_1)} - 1 \tag{49}$$

$$A = \frac{B^2}{4} = \frac{4}{\tilde{\mu}^2}(1+\theta)^2. \tag{50}$$

Finally, Theorem 1 follows by setting the parameter δ in (28) equal to the expression θ in (49), which goes to zero in the limit $c_0 \to \infty$. □

5 The expectation of the collective accusation sum

As was shown in Section 4, the average collective accusation $\tilde{\mu}$ plays a central role in determining the code length m required for collusion resistance. In this section we compute the value of $\tilde{\mu}$ in the restricted digit model. (Other attack models are discussed in Section 7.2). Unfortunately the computations are tedious. We first derive a general result in Section 5.1, for all alphabet sizes q, all values of the steepness parameter κ and all colluder strategies. This result takes the form of a $(q-1)$-dimensional sum over all possible symbol frequencies $\mathbf{b}$ received by the colluders. Then, in Section 5.2 we investigate the special case ($q=2, \kappa=\frac{1}{2}$), precisely corresponding to the choice of parameters of Tardos [9] (but not the same accusation method). It turns out that our symmetric accusation method yields an improvement of a factor 4 in the code length. In Section 5.3 we study the case $q=2$ for arbitrary κ. It turns out that for $q=2$, the choice $\kappa=\frac{1}{2}$ is optimal, a result that was obtained for the original Tardos construction in [7]. Finally, in Section 5.4, we come back to the nonbinary case $q>2$.

5.1 Sum representation of $\tilde{\mu}$

According to the definition (17), $\tilde{\mu}$ is defined as the expectation value $\mathbb{E}_{yXp}[\mathcal{A}_C^{(i)}]$. We follow the procedure outlined in Section 3.3: We first compute the expectation value with respect to the colluder strategy, then w.r.t. the matrix $\mathbf{X}_C$ and finally w.r.t. the vectors $\mathbf{p}^{(i)}$. Since it is understood that the results are identical for each column of $\mathbf{X}_C$, we will omit the column index i on the quantities y, $\mathbf{p}$ and $\mathbf{b}$ for notational simplicity.

We regard y as a (possibly stochastic) strategy-dependent function of $\mathbf{b} = (b_0, \cdots, b_{q-1})$ only. The colluders' strategy ρ cannot depend on $\mathbf{p}$, since they do not know $\mathbf{p}$. We assume that ρ is not influenced by the colluders' identities, i.e. their decisions are purely based on how many instances of each symbol were received, not by whom they were received. Using the notation introduced in (15), we have

$$\mathbb{E}_y[\mathcal{A}_C^{(i)}] = \sum_{\alpha=0}^{q-1} P_{\mathbf{b}}(\alpha)\left\{b_\alpha g_1(p_\alpha) + [c-b_\alpha]g_0(p_\alpha)\right\} = \sum_{\alpha=0}^{q-1} P_{\mathbf{b}}(\alpha)\frac{b_\alpha - cp_\alpha}{\sqrt{p_\alpha(1-p_\alpha)}}. \tag{51}$$

Next we average over $\mathbf{b}$ and $\mathbf{p}$. Applying (13) and (10) to (51), we obtain

$$\tilde{\mu} = \sum_{\mathbf{b}} \binom{c}{\mathbf{b}} \sum_{\alpha=0}^{q-1} P_{\mathbf{b}}(\alpha) \int_{J(t,q)} \mathrm{d}^q\mathbf{p}\; F(\mathbf{p}) \prod_{\beta=0}^{q-1} p_\beta^{b_\beta} \frac{b_\alpha - cp_\alpha}{\sqrt{p_\alpha(1-p_\alpha)}}. \tag{52}$$

We further evaluate the integral $\int \mathrm{d}^q\mathbf{p}$ for $t = 0$. As discussed in Section 3.1, the error resulting from integration over $J(0, q)$ instead of $J(t, q)$ is small. Furthermore, we will see in Section 7.1 that setting $t = 0$ is allowed for $q \geq 3$ in the Gaussian approximation. First we split the integration into two parts: p_α and the remaining $q - 1$ components

$$\int_{J(0,q)} \mathrm{d}^q\mathbf{p} = \int_0^1 \mathrm{d}p_\alpha \int_0^{1-p_\alpha} \mathrm{d}^{q-1}\mathbf{p}\ \delta(1 - p_\alpha - \sum_{\beta\neq\alpha} p_\beta). \tag{53}$$

Note that the upper bound on the second integration interval is reduced from 1 to $1 - p_\alpha$. This prevents us from directly applying Lemma 1. For all $\gamma \neq \alpha$ we write $p_\gamma = (1 - p_\alpha)s_\gamma$, with $s_\gamma \in (0, 1)$ and $\sum_{\gamma\neq\alpha} s_\gamma = 1$. This substitution has the following effect,

$$\begin{aligned}
\int_{J(0,q)} \mathrm{d}^q\mathbf{p} &= \int_0^1 \mathrm{d}p_\alpha (1 - p_\alpha)^{q-2} \int_0^1 \mathrm{d}^{q-1}\mathbf{s}\ \delta(1 - \sum_{\gamma\neq\alpha} s_\gamma) \\
F(\mathbf{p}) &= \mathcal{N}_{q\kappa 0}^{-1}\, p_\alpha^{-1+\kappa} \cdot (1 - p_\alpha)^{(-1+\kappa)(q-1)} \prod_{\gamma\neq\alpha} s_\gamma^{-1+\kappa} \\
\prod_{\beta=0}^{q-1} p_\beta^{b_\beta} &= p_\alpha^{b_\alpha} (1 - p_\alpha)^{c-b_\alpha} \prod_{\beta\neq\alpha} s_\beta^{b_\beta}.
\end{aligned} \tag{54}$$

Here we have used the property $\delta(ax) = |a|^{-1}\delta(x)$ for constant $a \neq 0$. Substituting (54) into (52) and applying Lemma 1 to the $q - 1$ degrees of freedom s_γ we obtain

$$\begin{aligned}
\tilde{\mu} &= \mathcal{N}_{q\kappa 0}^{-1} \sum_{\mathbf{b}} \binom{c}{\mathbf{b}} \sum_{\alpha=0}^{q-1} P_{\mathbf{b}}(\alpha) \frac{\prod_{\gamma\neq\alpha} \Gamma(\kappa + b_\gamma)}{\Gamma(c - b_\alpha + \kappa[q - 1])} \\
&\quad \times \int_0^1 \mathrm{d}p_\alpha\, p_\alpha^{b_\alpha - \frac{3}{2} + \kappa} (1 - p_\alpha)^{c-b_\alpha-\frac{3}{2}+\kappa[q-1]} (b_\alpha - cp_\alpha).
\end{aligned} \tag{55}$$

Finally, the p_α-integral is evaluated as well, yielding ordinary Beta functions,

$$\begin{aligned}
\tilde{\mu} &= \frac{\Gamma(\kappa q)}{[\Gamma(\kappa)]^q} \frac{c \cdot c!}{\Gamma(c + \kappa q)} \sum_{\mathbf{b}} [\prod_{\gamma=0}^{q-1} \frac{\Gamma(\kappa + b_\gamma)}{\Gamma(1 + b_\gamma)}] \times \\
&\sum_{\alpha=0}^{q-1} P_{\mathbf{b}}(\alpha) \frac{\Gamma(b_\alpha - \frac{1}{2} + \kappa)}{\Gamma(b_\alpha + \kappa)} \frac{\Gamma(c - b_\alpha - \frac{1}{2} + \kappa[q - 1])}{\Gamma(c - b_\alpha + \kappa[q - 1])} \left\{\tfrac{1}{2} - \kappa - \frac{b_\alpha}{c}(1 - \kappa q)\right\}.
\end{aligned} \tag{56}$$

Here we have used (8) for the normalisation constant $\mathcal{N}_{q\kappa 0}$. Expression (56) is rather complicated. One property of (56) can be seen easily, however: For $c \gg 1$, the leading order terms of $\tilde{\mu}$ are of order 1, and do not depend on c. This is readily seen by writing $b_\gamma = c \cdot w_\gamma$, with $w_\gamma \in [0, 1]$, then applying the Stirling approximation $\Gamma(x + 1) \approx \sqrt{2\pi x}(x/e)^x$ to all Gamma functions and collecting powers of c. For the quotients of Gamma functions appearing in (56) we have the proportionality $\Gamma(b_\beta + v_1)/\Gamma(b_\beta + v_2) \propto c^{v_1-v_2}$ and $\Gamma(c - b_\beta + v_1)/\Gamma(c - b_\beta + v_2) \propto c^{v_1-v_2}$ for constants $v_1, v_2 \ll c$. The sum $\sum_{\mathbf{b}}$ gives rise to a factor c^{q-1}, since it can be approximated by an integral $\int_1^c \mathrm{d}^q\mathbf{b}\ \delta(c - \sum_\alpha b_\alpha) \approx c^{q-1} \int_0^1 \mathrm{d}^q\mathbf{w}\ \delta(1 - \sum_\alpha w_\alpha)$. The corrections arising from the summation terms where the condition $b_\gamma \gg 1$ does not hold are negligible, since the support is negligible compared to the full summation $\sum_{\mathbf{b}}$.

The fact that $\tilde{\mu}$ has a finite value in the limit $c \to \infty$ shows that the asymptotic behaviour of (27) is given by $m \propto c_0^2$, without further dependence on c_0 arising from $\tilde{\mu}$.

5.2 The case $q = 2$, $\kappa = \frac{1}{2}$

This case corresponds to the probability density function in the original Tardos construction, $F(p_0, p_1) \propto (p_0 p_1)^{-1/2}\delta(1 - p_0 - p_1)$. Note that for $q = 2$, $\kappa = \frac{1}{2}$ the factor between curly brackets in (56) vanishes. However, $\tilde{\mu}$ does not completely vanish, since for $(q=2, b_\alpha = c)$ the expression $\Gamma(c - b_\alpha - \frac{1}{2} + \kappa[q-1])$ is divergent in the limit $\kappa \to \frac{1}{2}$. We have

$$\lim_{\kappa\to 1/2}(-\tfrac{1}{2}+\kappa)\Gamma(-\tfrac{1}{2}+\kappa) = \lim_{\kappa\to 1/2}\Gamma(\tfrac{1}{2}+\kappa) = 1. \tag{57}$$

Hence, the only terms contributing in the $\mathbf{b}$-sum in (56) are those where $b_\alpha = c$. Because of the marking condition, $P_{\mathbf{b}}(\alpha) = 1$ for these terms, as the coalition only sees the symbol α. The complicated expression (56) reduces to a constant:

$$\tilde{\mu} = \frac{\Gamma(1)}{[\Gamma(\frac{1}{2})]^2}\sum_{\alpha=0}^{1} 1 = \frac{2}{\pi}. \tag{58}$$

Substitution into (27,28) gives the following asymptotic bound on the code length,

$$m > \pi^2 c_0^2 \lceil \ln \varepsilon_1^{-1} \rceil. \tag{59}$$

This bound is 4 times lower than the bound obtained in [7] and 10 times lower than the bound in [9].

5.3 The case $q = 2$, $\kappa \neq \frac{1}{2}$

Next we study how the symmetric binary scheme performs for $\kappa \neq \frac{1}{2}$. Substitution of $q = 2$ into (56) gives

$$\begin{aligned}\tilde{\mu} \;&=\; \frac{\Gamma(2\kappa)}{[\Gamma(\kappa)]^2}\frac{(\frac{1}{2}-\kappa)c}{c-1+2\kappa}\sum_{b_1=0}^{c}\binom{c}{b_1}B(b_1-\tfrac{1}{2}+\kappa, c-b_1-\tfrac{1}{2}+\kappa) \\ &\quad\times\left\{-1+\frac{2}{c}[b_1 P_{\mathbf{b}}(0)+(c-b_1)P_{\mathbf{b}}(1)]\right\},\end{aligned} \tag{60}$$

where B denotes the Beta function. From (60) we can identify which colluder strategy ρ forces the content owner to use the longest possible code. We denote this ‘extremal’ strategy as ρ_2^*. We remind the reader that $m \propto \tilde{\mu}^{-2}$. Hence, in order to maximize m, the strategy ρ_2^* has to minimize the summand in (60) for each $\mathbf{b}$. Note that the $b_1 = 0$ and $b_1 = c$ contributions to the summation are not affected by the strategy. For $1 \leq b_1 \leq c-1$ the Beta function in (60) is positive. Hence, the factor $(\frac{1}{2}-\kappa)$ in front of the summation determines the overall sign of the strategy-dependent contributions. For $\kappa < \frac{1}{2}$, this factor is positive, so the colluders wish to minimize the expression $[b_1 P_{\mathbf{b}}(0) + (c-b_1)P_{\mathbf{b}}(1)]$. They achieve this by choosing the symbol that appears most frequently, i.e. by applying ‘majority voting’ to the 0s and 1s that they receive in a column. For $\kappa > \frac{1}{2}$, the factor $\frac{1}{2}-\kappa$ has the opposite sign and the extremal strategy ρ_2^* is minority voting.

Note that ρ_2^* is not necessarily the strategy that the coalition actually applies. However, the distributor has to take into account that the colluders *could* be using ρ_2^*, and he has to choose his code length m accordingly. We are interested in this ‘extremal’ strategy because our aim is to derive a sharp lower bound on m.

Fig. 1 shows $\tilde{\mu}$ as a function of κ for the strategy ρ_2^*. The dashed line corresponds to the value $2/\pi$ obtained in the previous section. It is clear that $\kappa = \frac{1}{2}$ is the optimum. At the optimum we have $\tilde{\mu} = 2/\pi$, independent of c. The part of the curve with $\kappa < \frac{1}{2}$ hardly depends on c. The part with $\kappa > \frac{1}{2}$ becomes steeper with increasing c.

5.4 Non-binary alphabet

We now return to the general expression for $\tilde{\mu}$ given in (56). We work in the *restricted digit model*, where, at each position, the colluders can output only the symbols they have available. (In Appendices A and B we discuss the unreadable digit and arbitrary digit model).

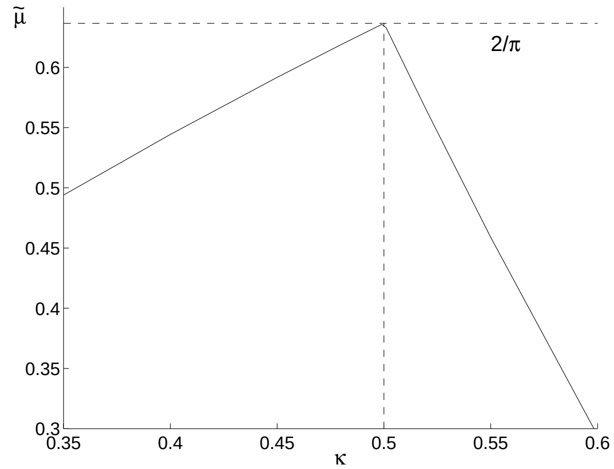

Fig. 1: $\tilde\mu$ as a function of κ for $q=2$, $c=80$, given the 'extremal' strategy ρ_2^*.

Note that the sum $\sum_\alpha P_{\mathbf b}(\alpha)(\cdots)$ in (56) represents an average over α. We obtain a lower bound for the sum from the fact that an average is at least as big as the smallest element in the summation. Thus we have

$$\begin{aligned}\tilde\mu \;\geq\; & \frac{\Gamma(\kappa q)}{[\Gamma(\kappa)]^q}\frac{c\cdot c!}{\Gamma(c+\kappa q)}\sum_{\mathbf b}[\prod_{\gamma=0}^{q-1}\frac{\Gamma(\kappa+b_\gamma)}{\Gamma(1+b_\gamma)}] \\ & \min_{\alpha|b_\alpha\neq 0}\frac{\Gamma(b_\alpha-\frac12+\kappa)}{\Gamma(b_\alpha+\kappa)}\frac{\Gamma(c-b_\alpha-\frac12+\kappa[q-1])}{\Gamma(c-b_\alpha+\kappa[q-1])}\left\{\tfrac12-\kappa-\frac{b_\alpha}{c}(1-\kappa q)\right\}.\end{aligned} \tag{61}$$

As we have assumed the restricted digit model, the minimum is taken only over those symbols that the colluders have received.

Eq.(61) allows us to identify the 'extremal' colluder strategy ρ_q^*, which forces the distributor to use the largest code length m. For each $\mathbf b$ separately, the colluders choose α such that the expression following '$\min_\alpha$' is minimized.

For $q\leq 10$ and a fixed coalition size $c=20$ we have numerically computed $\tilde\mu$ as a function of κ for the ρ_q^* strategy, i.e. taking the equality in (61). For large q and c the numerics are computationally expensive, since the number of terms in the $\mathbf b$-summation is of order c^{q-1}. Fig. 2 shows $\tilde\mu$ as a function of the steepness parameter κ. For $q\leq 7$ the maximum of the curve lies slightly to the right of $\kappa=1/q$. For $q\geq 8$ an extra hump is visible. The hump is a 'finite c effect'; it does not exist when the ratio q/c is small. Fig. 3 shows how $\tilde\mu$ varies when c is increased: The part of the curve at $\kappa<1/q$ is unaffected, while for $\kappa>1/q$ the curve goes downward and converges to a finite value.

We use the numerical results for $\tilde\mu$ to estimate the required code length ($m\propto\tilde\mu^{-2}$). We give estimates for the advantage that a q-ary code gives over the symmetric binary code with $\kappa=1/2$. The comparison with the binary case can be done in several ways, depending on the details of the watermark embedding. We give the two extreme comparison methods:

1. *Counting the number of symbols.* A q-ary symbol occupies as much space in the content as a binary symbol, regardless of q. Fig. 4 shows the $\frac{q\text{-ary case}}{\text{binary case}}$ ratio for the number of symbols. This ratio is given by $4/(\pi^2\tilde\mu^2)$.

2. *Counting the number of bits.* A q-ary symbol occupies $\log_2 q$ times more space in the content than a binary symbol. In this case it is not fair to compare code length expressed in symbols. One has to count bits. Fig. 5 shows the $\frac{q\text{-ary case}}{\text{binary case}}$ ratio for the number of bits. This ratio is given by $\log_2 q\cdot 4/(\pi^2\tilde\mu^2)$.

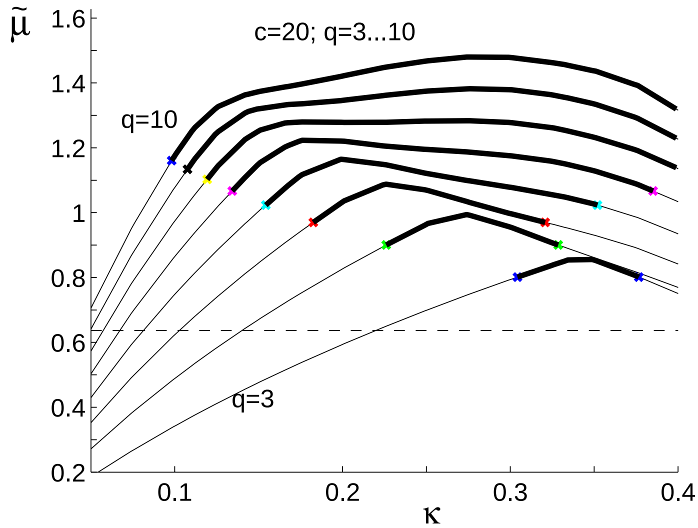

Fig. 2: $\tilde{\mu}$ as a function of κ for several alphabet sizes q. The coalition size is $c = 20$. The colluders employ the 'extremal' strategy.

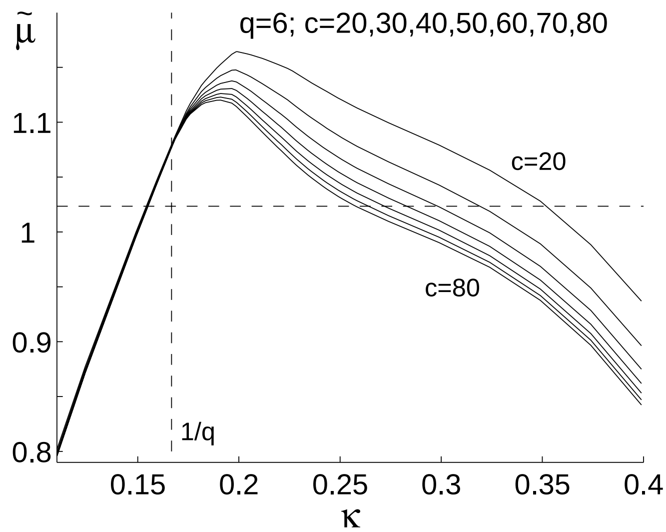

Fig. 3: $\tilde{\mu}$ as a function of κ for $q = 6$ at several coalition sizes. The colluders employ the 'extremal' strategy. The dashed horizontal line lies at $(2/\pi)\sqrt{\log_2 6}$. When $\tilde{\mu}$ lies above this line, the space (in bits) occupied in the $q = 6$ scheme is smaller than in the binary scheme.

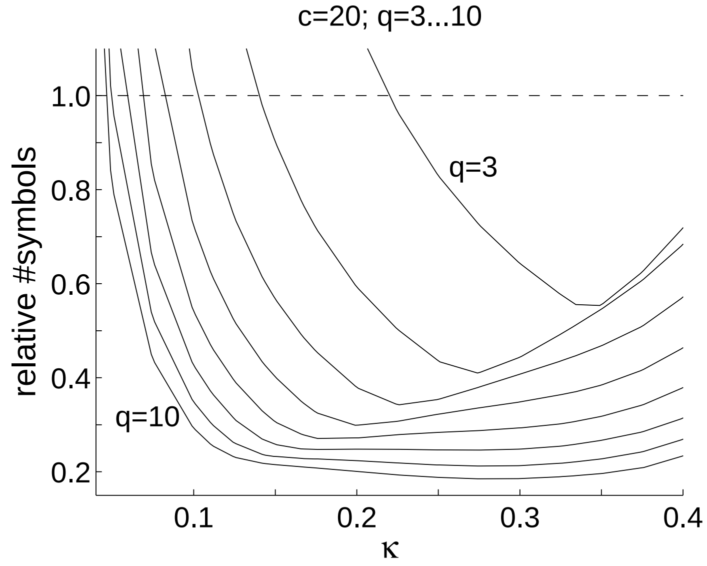

Fig. 4: Number of symbols in the codewords, relative to the binary case, for several alphabet sizes q. The coalition size is $c = 20$. The colluders employ the 'extremal' strategy.

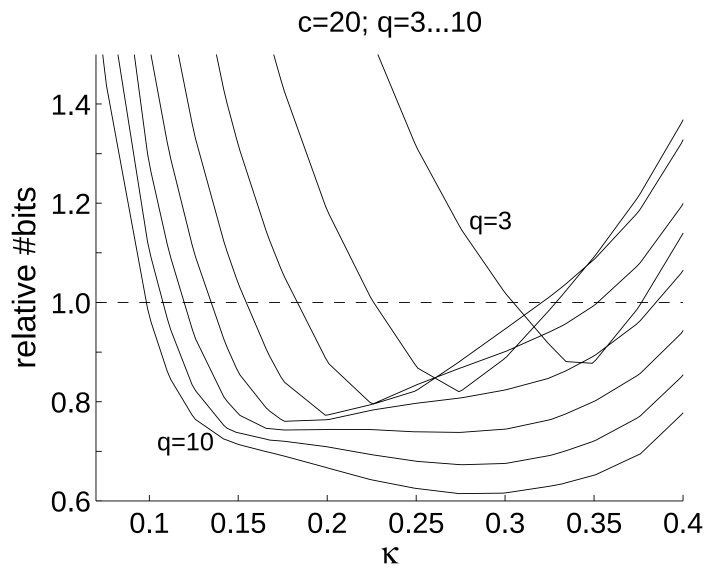

Fig. 5: Number of bits in the codewords, relative to the binary case, for several alphabet sizes q. The coalition size is $c = 20$. The colluders employ the 'extremal' strategy.

Type 1 is the most optimistic comparison possible, in the sense that it allows for the largest improvements w.r.t. the binary scheme. Type 2 comparison is the most pessimistic possible. Without giving a full argument, we state that in the case of video watermarking type 1 is more appropriate, even for large alphabets. When, for instance, symbols are embedded using a spread-spectrum watermark, where each spreading sequence corresponds to a different symbol in the alphabet, then the segment length can be kept almost independent of q without decreasing detection performance.

For completeness we give the results for both comparisons. The horizontal dotted line in Fig. 2 indicates the threshold for comparison of type 1. When $\tilde{\mu}$ rises above this threshold, the q-ary scheme needs fewer symbols than the binary scheme. The thick piece of each curve indicates the region where the q-ary scheme is better than the binary, using comparison type 2. Fig. 4 shows the code length m (the number of symbols) as a function of κ, for a number of q values, and Fig. 5 similarly shows $m \log_2 q$, the number of bits. Both graphs have their vertical axis normalised such that lengths are divided by corresponding lengths in the binary scheme. In both graphs the finite-c humps are visible. Not taking the humps into account, we see that for $3 \leq q \leq 10$ the number of symbols is reduced by 40%–80% w.r.t. the binary case, while the reduction in the number of bits is 11%–30%. Finite-c effects further improve these results. We conclude that in our symmetric scheme it is advantageous to use the largest possible alphabet allowed by the watermarking method employed.

6 The Gaussian approximation

6.1 Motivation

In this section we analyse the performance of the symmetric scheme using what we call the 'Gaussian approximation'. By this we mean the assumption that the accusation $\mathcal{A}_j$ for innocent j has a Gaussian probability density function. The assumption is motivated by the Central Limit Theorem (CLT): when a large number of i.i.d. variables are summed, the distribution of the sum converges to the normal distribution. The CLT applies when the moments of the summands' distribution meet certain conditions. The moments also determine the rate of convergence to the normal form.

The accusation $\mathcal{A}_j$ is computed by taking the sum over m independent accusations, each of which is based on a single symbol y_i in the unauthorized copy. All the separate accusations have the same probability distribution. The number of symbols, m, is large enough to guarantee 'sufficiently fast' convergence to the normal form. This informal statement is made more precise in Appendix C, where we derive a lower bound on c_0 as a function of q. When c_0 is above this bound, the deviations from the normal form become 'small enough' in the central region of the $\mathcal{A}_j$-distribution function. It turns out that the bound approximately lies between $c_0 = 10$ and $c_0 = 20$. Hence convergence is fast enough in many practical situations.

In Section 6.2 we analyse the symmetric scheme under the assumption that $\mathcal{A}_j$ has a Gaussian distribution. We obtain a lower bound on m that is a factor 2 smaller than Theorem 1.

In the discussion of the CLT in Appendix C it turns out that for $q \geq 3$ the cutoff parameter t can be sent to zero without causing any divergences. The cutoff parameter is discussed in Section 7.1.

6.2 Lower bound on the code length

Theorem 2: *Let $\mathcal{A}_j$ have a Gaussian probability density. For all $\delta > 0$ there exists a sufficiently large c_0 such that Property 1 is satisfied for parameter ε_1 and Property 2 is satisfied for parameters c_0, ε_2 when the code length is*

$$m > 2\tilde{\mu}^{-2}(1+\delta)c_0^2 \ln \varepsilon_1^{-1}. \tag{62}$$

Note that this code length is a factor 2 lower than the one in Theorem 1. We first give an informal argument why a bound of the form (62) follows from the Gaussian probability density. Then we give a formal proof.

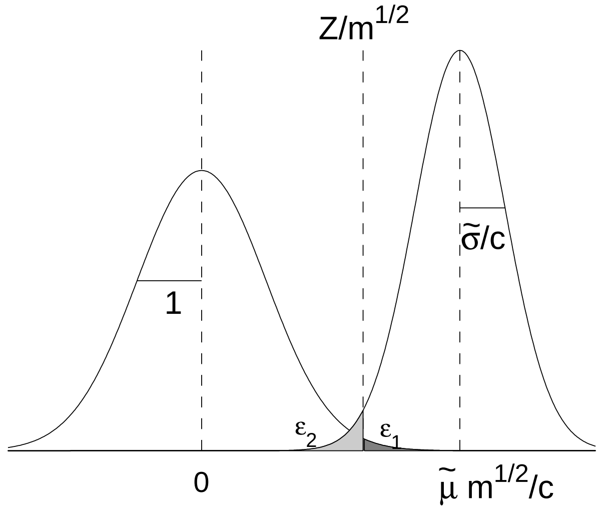

Fig. 6: Sketch of the probability density of $\mathcal{A}_j/\sqrt{m}$ (left) and $\frac{1}{c}\mathcal{A}_C/\sqrt{m}$ (right). The accusation threshold Z and the error rates ε_1 and ε_2 are also shown.

Informal argument: If the probability density of $\mathcal{A}_j$ is known, then that knowledge allows us to compute the FP and FN error rates as a function of $\tilde{\mu}_j$, $\tilde{\sigma}_j$, $\tilde{\mu}$, $\tilde{\sigma}$, m and Z. This is sketched in Fig. 6. The left curve is the probability density of the quantity $\mathcal{A}_j/\sqrt{m}$. It has mean $\tilde{\mu}_j = 0$ and variance $\tilde{\sigma}_j = 1$ (see Lemmas 2 and 3). The error rate ε_1 is given by the area to the right of the (rescaled) threshold $Z/\sqrt{m}$. The right curve is the probability density of the quantity $\frac{1}{c}\mathcal{A}_C/\sqrt{m}$. It has average $\frac{1}{c}\tilde{\mu}\sqrt{m}$ and variance $\tilde{\sigma}/c$. The error rate ε_2 is given by the area to the left of $Z/\sqrt{m}$. The horizontal axis is scaled such that the $\mathcal{A}_j$-curve does not depend on c and m. If we set ourselves the goal of having fixed error rates for arbitrary c, two observations can be made from Fig. 6:

- In order to have a fixed ε_1 for all c, the threshold line $Z/\sqrt{m}$ must not shift. Hence Z must be chosen as $Z \propto \sqrt{m}$ as far as the dependence on c is concerned.
- When c increases, the rightmost curve becomes narrower and shifts to the left. In order to prevent ε_2 from vanishing, m must be chosen proportional to c^2.

From this informal argument we obtain the proportionality $m \propto c_0^2$ in (62), but not the constants and the logarithmic dependence on ε_1.

Proof of Theorem 2: Let ρ_1 and ρ_2 be the density functions of $\mathcal{A}_j$ and $\mathcal{A}_C$, respectively, rescaled such that they both have zero mean and unit variance. We define cumulative distributions in the tails,

$$G_1(x) = \int_x^\infty \mathrm{d}x' \, \rho_1(x') \quad ; \quad G_2(x) = \int_{-\infty}^x \mathrm{d}x' \, \rho_2(x'). \tag{63}$$

Lemma 6: *In order to achieve a False Positive error rate $\leq \varepsilon_1$ and a False Negative error rate $\leq \varepsilon_2$ against any coalition of size $c \leq c_0$, it is sufficient to set the code length m according to*

$$m \geq c_0^2 \left[\frac{\tilde{\sigma}_j}{\tilde{\mu}} G_1^{\mathrm{inv}}(\varepsilon_1)\right]^2 \left\{1 - \frac{\tilde{\sigma}}{c_0\tilde{\sigma}_j} \frac{G_2^{\mathrm{inv}}(\varepsilon_2)}{G_1^{\mathrm{inv}}(\varepsilon_1)}\right\}^2. \tag{64}$$

Here the superscript 'inv' denotes the inverse function.

Proof: The proof is completely analogous to the derivation in Section 3.4 of [7]. □

Note that $G_2^{\rm inv}(\varepsilon_2) < 0$. Note further that the dependence of (64) on ε_2 vanishes in the limit $c_0 \to \infty$. (Remember that $\tilde\sigma_j = 1$ according to Lemma 3 and that $\tilde\sigma = \mathcal{O}(\sqrt{c_0})$ as a consequence of Lemma 4). If $\mathcal{A}_j$ and $\mathcal{A}_C$ have Gaussian distributions, then G_1 and G_2 are error functions[2] and we have $G_1^{\rm inv}(\varepsilon_1) = \sqrt{2}{\rm Erfc}^{\rm inv}(2\varepsilon_1)$, $G_2^{\rm inv}(\varepsilon_2) = -\sqrt{2}{\rm Erfc}^{\rm inv}(2\varepsilon_2)$. Substitution into (64), using $\tilde\sigma_j = 1$, gives

$$m \geq \frac{2}{\tilde\mu^2} c_0^2 \left[{\rm Erfc}^{\rm inv}(2\varepsilon_1)\right]^2 \left\{1 + \frac{\tilde\sigma}{c_0}\frac{{\rm Erfc}^{\rm inv}(2\varepsilon_2)}{{\rm Erfc}^{\rm inv}(2\varepsilon_1)}\right\}^2 . \tag{65}$$

In the regime $\varepsilon_1 \ll \varepsilon_2$, which is the relevant regime for e.g. movie distribution, the dependence of (65) on ε_2 is rather weak even for finite c_0, since ${\rm Erfc}^{\rm inv}(\varepsilon_2) < {\rm Erfc}^{\rm inv}(\varepsilon_1)$. We use the asymptotic form of the inverse error function for small arguments, ${\rm Erfc}^{\rm inv}(\varepsilon) = \sqrt{\ln\varepsilon^{-1}}[1 - \mathcal{O}(\frac{\ln\ln\varepsilon^{-1}}{\ln\varepsilon^{-1}})]$, to write

$$m \geq \frac{2}{\tilde\mu^2} c_0^2 \ln\frac{1}{\varepsilon_1}\left[1 - \mathcal{O}(\frac{\ln\ln\varepsilon_1^{-1}}{\ln\varepsilon_1^{-1}})\right]\left\{1 + \mathcal{O}\left(\frac{1}{\sqrt{c_0}}\sqrt{\frac{\ln\varepsilon_2^{-1}}{\ln\varepsilon_1^{-1}}}\right)\right\}. \tag{66}$$

For large c_0 the result (62) follows[3]. □

Hence for large enough c_0 a code length $m = 2\tilde\mu^{-2}c_0^2\ln\varepsilon_1^{-1}$ suffices. This is shorter by a factor 2 than the result obtained in Section 4.

7 Discussion

7.1 The cutoff parameter t

In this section we discuss the the effects of the cutoff $t = Tc_0^{-a}$ introduced in Section 2.2. The probabilities p_α lie in the restricted interval $[t/(q-1), 1-t]$. It is clear from Section 4 that the presented proof of Theorem 1 does not work for $t = 0$. In the limit $T \downarrow 0$, the allowed intervals for the auxiliary variables α_1 and α_2 (34,42) vanish, while both intervals need to be finite for the proof that Properties 1 and 2 are satisfied.

The speed of the convergence to the asymptotic result $A = 4/\tilde\mu^2$, $B = 4/\tilde\mu$ depends on the way in which the parameters $a \in (0,2)$ and T are chosen. The small parameters ψ_1 and ψ_2 (45,47) asymptotically behave as

$$\psi_1 \approx 1.7\frac{q}{\tilde\mu}\frac{\sqrt{T}}{c_0^{a/2}} \quad ; \quad \psi_2 \approx \frac{\ln\varepsilon_2}{\ln\varepsilon_1}\frac{1}{1.7\sqrt{T}c_0^{1-a/2}}. \tag{67}$$

Furthermore, condition (36), necessary for Property 1 to hold, can be written as

$$c_0 \gtrapprox \left[\frac{q-1}{T}(\frac{\tilde\mu}{3.4})^2\right]^{1/(2-a)} . \tag{68}$$

For practical reasons, we wish both ψ_1 and ψ_2 to become small at a reasonably low value of c_0, while the bound (68) also should not be too high. However, in the limit $T \downarrow 0$, both the c_0-bound (68) and the expression for ψ_2 in (67) diverge. Hence, when t tends to zero, the approach of Sections 4.1–4.3, based on the Markov inequality, can prove Properties 1 and 2 only for extremely large c_0.

The role of the cutoff t is completely different in the analysis using the Gaussian approximation.

[2] To avoid ambiguities due to conflicting definitions in the literature, we mention that we use the definition ${\rm Erfc}(x) = 1 - (2/\sqrt{\pi})\int_0^x e^{-u^2}{\rm d}u$.

[3] For proving Theorem 2 we do not have to assume that $\mathcal{A}_C$ has a Gaussian form. The ε_2-term in (64) vanishes for all functions $G_2^{\rm inv}$ because of the factor $\tilde\sigma/c_0 = \mathcal{O}(1/\sqrt{c_0})$. However, the computation of $\mathcal{A}_C$ involves even more summed contributions than $\mathcal{A}_j$, so it is safe to assume that when $\mathcal{A}_j$ is Gaussian, then $\mathcal{A}_C$ is Gaussian as well.

- The case $q = 2$. It was shown in [7] for the original Tardos scheme that the CLT can only be applied if $t > 0$. The probability distribution of the accusation U_i (for innocent users) due to the symbol y_i is proportional to $1/(1+U_i^2)^2$. The 3rd moment is zero. For distributions with vanishing 3rd moment, the CLT only holds when the 4th moment does not diverge. However, for $t = 0$ the 4th moment *does* diverge. Hence we need $t > 0$. Exactly the same reasoning applies to the symmetric scheme with $q = 2$.

- The case $q \geq 3$. For $q \geq 3$, the 3rd moment of the probability distribution of $\mathcal{A}_j^{(i)}$ (for innocent j) is always nonzero, no matter what the value of t is. This is shown in Appendix C, Eq.(74). Hence the CLT applies even if we set $t = 0$. In the Gaussian approximation, there is no reason to have a cutoff t for $q \geq 3$.

7.2 Different attack models

Up to this point we have only considered the restricted digit model. However, it is easy to obtain results for the other attack models listed in Section 1.2. As can be seen from (64), the bound on the code length is proportional to $\tilde{\sigma}_j^2/\tilde{\mu}^2$.[4] The differences between the various attack models give rise to different values $\tilde{\sigma}_j$, $\tilde{\mu}$, but the form (64) is independent of the attack model. Hence, in order to see the differences between the attack models, it is sufficient to compare the ratio $\tilde{\sigma}_j/\tilde{\mu}$.

The unreadable digit model is discussed in Appendix A. It is assumed that the colluders output an erasure symbol '?' whenever they can, and that the distributor gives zero accusation to locations with an erasure. It turns out that for large alphabets ($q \gtrapprox 7$) the colluder strategy of outputting erasures is good, and the distributor has to use longer codes than in the restricted digit model. However, for small alphabets it is better for the colluders not to use an erasure at each detectable position, as a '?' informs the distributor that the position is detectable.

Results for the arbitrary digit model are derived in Appendix B. Unsurprisingly, with this attack model a nonbinary scheme always performs worse than the symmetric binary scheme; the colluders have ample opportunity to incriminate innocent users while avoiding accusation themselves.

8 Summary

In this paper we have proposed a new construction for a randomized digital fingerprinting code, which is similar to a recent construction by Tardos but can be used with arbitrary size alphabets. We have analyzed the performance of our scheme, in the restricted digit model, in two ways.

First, we have proved a lower bound on the code length m such that the desired False Positive and False Negative error probabilities are achieved against any coalition of size $c \leq c_0$. Due to a different way of computing accusations, the proposed code allows for 10 times shorter codes (with respect to [9]) in the case of a binary alphabet. Moving to a code over a q-ary alphabet allows a further reduction of the code length of 35% at $q = 3$ and 80% at $q = 10$.

Second, we have analyzed our scheme under the assumption that the accusation sum $\mathcal{A}_j$ follows a Gaussian distribution. This 'Gaussian approximation' is valid at coalition sizes c_0 of approximately 10–20 and larger. We have shown that, in this approximation, the collusion resistance of the scheme is retained for a code length m that is twice as short as the bound obtained using no assumptions.

Acknowledgements

We kindly thank Joop Talstra and Henk Hollmann for valuable comments, and Guido Janssen for providing us with literature references.

[4] Theorem 1 is formulated after the substitution $\tilde{\sigma} \to 1$ has been done. If application of Lemma 3 is postponed in the analysis in Section 4, we get the proportionality to $\tilde{\sigma}_j^2$ in Theorem 1 as well.

References

[1] G.E. Andrews, R. Askey, and R. Roy. *Special Functions.* Encyclopedia of Mathematics and its Applications. Cambridge University Press, 1999.

[2] M.Z. Bazant. Random walks and diffusion. Technical report, MIT Lecture notes, `http://www-math.mit.edu/18.366/lec/lec04.pdf`, 2005.

[3] D. Boneh and J. Shaw. Collusion-secure fingerprinting for digital data. *IEEE Transactions on Information Theory*, 44(5):1897–1905, 1998.

[4] L. Devroye. *Non-Uniform Random Variate Generation.* Springer-Verlag, 1986. Available online at http://cg.scs.carleton.ca/~luc/rnbookindex.html.

[5] H.D.L. Hollmann, J.H. van Lint, J-P. Linnartz, and L.M.G.M. Tolhuizen. On codes with the identifiable parent property. *Journal of Combinatorial Theory*, 82:472–479, 1998.

[6] C. Peikert, A. Shelat, and A. Smith. Lower bounds for collusion-secure fingerprinting. In *Proceedings of the 14th Annual ACM-SIAM Symposium on Discrete Algorithms (SODA)*, pages 472–478, 2003.

[7] B. Škorić, T.U. Vladimirova, M. Celik, and J.C. Talstra. Tardos fingerprinting is better than we thought. Technical report, Submitted to IEEE Transactions on Information Theory. Preprint at arXiv repository, `http://www.arxiv.org/abs/cs.CR/0607131`, 2006.

[8] J.N. Staddon, D.R. Stinson, and R. Wei. Combinatorial properties of frameproof and traceability codes. *IEEE Transactions on Information Theory*, 47(3):1042–1049, 2001.

[9] G. Tardos. Optimal probabilistic fingerprint codes. In *Proceedings of the 35th Annual ACM Symposium on Theory of Computing (STOC)*, pages 116–125, 2003.

A Unreadable digit model

In this appendix we consider the case of the unreadable digit model. In this attack model, the colluders are allowed to output the erasure symbol '?' in detectable positions. For simplicity we make two assumptions: (i) The colluders generate an erasure whenever they can, and (ii) The distributor gives zero accusation in case of an erasure symbol.

The quantities $\tilde{\sigma}_j$ and $\tilde{\mu}$ are both affected by these assumptions. They are easily computed, since all detectable positions (leading to '?') are discarded by the distributor. This leaves only the undetectable positions, characterized by vectors $\mathbf{b}$ that consist of $q-1$ zero components and one component equal to c. We have

$$\tilde{\sigma}_j^2 = \sum_{\alpha=0}^{q-1} \int_{J(0,q)} \mathrm{d}^q p\, F(\mathbf{p}) p_\alpha^c = q \frac{\Gamma(\kappa q)\Gamma(c+\kappa)}{\Gamma(\kappa)\Gamma(c+\kappa q)} \tag{69}$$

and

$$\tilde{\mu} = \sum_{\alpha=0}^{q-1} \int_{J(0,q)} \mathrm{d}^q p\, F(\mathbf{p}) p_\alpha^c \cdot c g_1(p_\alpha) = cq \frac{\Gamma(\kappa q)\Gamma(c-\frac{1}{2}+\kappa)\Gamma(\frac{1}{2}+\kappa[q-1])}{\Gamma(\kappa)\Gamma(\kappa[q-1])\Gamma(c+\kappa q)}. \tag{70}$$

Recall from (64) that the required code length is proportional to $\tilde{\sigma}_j^2/\tilde{\mu}^2$. Using (69) and (70) we obtain

$$\begin{aligned} \frac{\tilde{\sigma}_j^2}{\tilde{\mu}^2} &= \frac{1}{qc^2} \frac{\Gamma(c+\kappa)\Gamma(c+\kappa q)}{[\Gamma(c-\frac{1}{2}+\kappa)]^2} \frac{\Gamma(\kappa)}{\Gamma(\kappa q)} \left[\frac{\Gamma(\kappa[q-1])}{\Gamma(\frac{1}{2}+\kappa[q-1])}\right]^2 \\ &\approx \frac{c^{-1+\kappa[q-1]}}{q} \frac{\Gamma(\kappa)}{\Gamma(\kappa q)} \left[\frac{\Gamma(\kappa[q-1])}{\Gamma(\frac{1}{2}+\kappa[q-1])}\right]^2. \end{aligned} \tag{71}$$

The last expression is obtained using the Stirling approximation of the Gamma function for large c. For $\kappa = 1/q$ the large q asymptotic behaviour is given by

$$\lim_{q\to\infty} \tilde{\sigma}_j^2/\tilde{\mu}^2 = 4/\pi. \tag{72}$$

Note that the result does not depend on c. Consequently the asymptotic relation $m \propto c_0^2$ holds not only in the restricted digit model, but also in the unreadable digit model. Eq. (72) demonstrates that it is unfavorable for the distributor to use a very large alphabet in the unreadable digit model, since the code length in bits ($m \log_2 q$) then grows as $\log_2 q$.

A graph of the (normalized) code length in bits $\propto \log_2(q)\tilde{\sigma}_j^2/\tilde{\mu}^2$, similar to the graphs in Section 5.4, is shown in Fig. 7 for $q = 3$ and $q = 7$. The number of bits increases as a function of q for the unreadable digit model, but it decreases in the restricted digit model. Apparently, the colluder strategy of outputting erasures whenever possible makes sense for large alphabets (the distributor has to use a longer code than in the restricted digit case), but not for small alphabets. Depending on the employed value of κ, the crossover value of q lies between approximately 5 and 8.

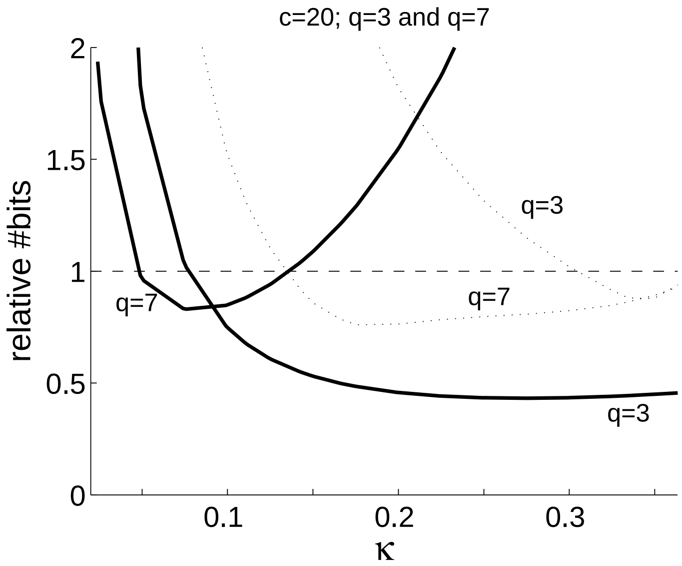

Fig. 7: Code length in bits as a function of κ in the unreadable digit model (solid lines), relative to the $q = 2$ restricted digit model. The colluders output an erasure whenever allowed by the marking condition. The dotted lines are the results for the restricted digit model (see Fig. 5).

B Arbitrary digit model

In this appendix we consider the case of the arbitrary digit model. In this attack model, the colluders are allowed to output any symbol $y \in \{0, \ldots, q-1\}$ (but not '?') in detectable positions.

This choice of attack model influences only $\tilde{\mu}$. The quantity $\tilde{\sigma}_j$ is unaffected by going from the restricted to the arbitrary digit model. We compute $\tilde{\mu}$ from expression (61) with one modification: The minimisation '$\min_\alpha$' now also includes symbols α for which $b_\alpha = 0$ (provided, of course, that none of the other symbols occurs c times).

Numerical results are shown in Fig. 8. For each q, the $\tilde{\mu}$ curve of the arbitrary digit model (solid curves) always lies below the curve of the restricted digit model (dotted curves). Note further that the nonbinary scheme is always worse than the binary in the arbitrary digit model. (The curves lie below $2/\pi$). Hence, if the arbitrary digit model applies, the distributor's best option is to use the binary scheme of Section 2.2.

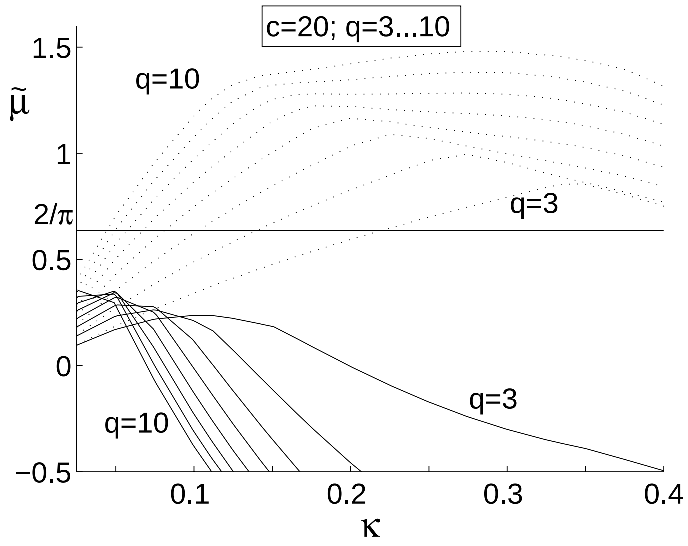

Fig. 8: $\tilde{\mu}$ as a function of κ in the arbitrary digit model (solid lines). The dotted lines are the results for the restricted digit model.

C Convergence to the normal distribution

In this appendix we study how fast (as a function of m) the distribution of $\mathcal{A}_j$ converges to the normal distribution. We primarily study the case $q \geq 3$, since for $q = 2$ the analysis of [7] suffices. We set $t = 0$. We use a theorem from [2] that gives the width of the central region where the normal form is a good approximation. This central region contains a fraction $1 - 2\varepsilon_1$ of the probability mass. By 'good approximation' it is meant that the deviation from the normal form, everywhere in the central region, is smaller than the value of the Gaussian at the edge of the central region. Applied to our accusation sum $\mathcal{A}_j$, the theorem gives the following width, expressed in standard deviations,

$$\#\text{sigmas} = \left(\frac{6\tilde{\sigma}_j^3}{|\lambda_3|}\right)^{1/3} m^{1/6}, \qquad \text{where } \lambda_3 := \mathbb{E}[\{\mathcal{A}_j^{(i)}\}^3]. \tag{73}$$

Here $\mathbb{E}$ stands for averaging first over X_{ji}, then y, then $\mathbf{X}_C$ and finally $\mathbf{p}$. The third moment is given by

$$\begin{aligned}\lambda_3 &= \frac{\Gamma(\kappa q)}{[\Gamma(\kappa)]^q} \sum_{\alpha=0}^{q-1} \sum_{\mathbf{b}} P_{\mathbf{b}}(\alpha) \binom{c}{\mathbf{b}} \frac{\prod_{\beta \neq \alpha} \Gamma(\kappa + b_\beta)}{\Gamma(c - b_\alpha + \kappa[q-1])} \\ &\quad \times \int_0^1 \mathrm{d}p_\alpha\, p_\alpha^{b_\alpha - 1 + \kappa} (1-p_\alpha)^{c - b_\alpha - 1 + \kappa[q-1]} \left[\frac{(1-p_\alpha)^{3/2}}{\sqrt{p_\alpha}} - \frac{p_\alpha^{3/2}}{\sqrt{1-p_\alpha}}\right].\end{aligned} \tag{74}$$

The integrals are all convergent[5] if the inequality $\kappa > 1/[2(q-1)]$ holds. (We remind the reader that $b_\alpha \geq 1$ due to the marking condition. Hence, the integrals always converge at $p_\alpha = 0$). From Fig. 2 we see that our region of interest lies at $\kappa > 1/q$, which means that the inequality indeed holds. Notice that for $q = 2$ the integral is antisymmetric under the mapping $(p_\alpha \to 1 - p_\alpha, b_\alpha \to c - b_\alpha)$, yielding $\lambda_3 = 0$. Notice too that we have set $t = 0$ without running into any divergences. In the proof of Theorem 1 it is impossible to set $t = 0$.

[5] We also have $\mathbb{E}[|\mathcal{A}_j^{(i)}|^3] < \infty$, and hence the Berry-Esséen theorem holds, stating that there is uniform convergence to a Gaussian distribution, with errors of order $1/\sqrt{m} = \mathcal{O}(1/c_0)$. Eq.(73) gives a sharper bound on the width of the central region than the Berry-Esséen theorem.

If the 'extremal' strategy of Section 5.4 is employed by the colluders, then (74) can be written as

$$\lambda_3 = \frac{\Gamma(\kappa q)}{[\Gamma(\kappa)]^q} \frac{c \cdot c!}{\Gamma(c+\kappa q)} \sum_{\mathbf{b}} [\prod_{\gamma=0}^{q-1} \frac{\Gamma(\kappa + b_\gamma)}{\Gamma(1+b_\gamma)}] \frac{\Gamma(b_y - \frac{1}{2} + \kappa)}{\Gamma(b_y + \kappa)} \frac{\Gamma(c - b_y - \frac{1}{2} + \kappa[q-1])}{\Gamma(c - b_y + \kappa[q-1])} \left\{ 1 - \frac{2b_y}{c} + \frac{\kappa[q-2]}{c} \right\}. \quad (75)$$

Here y is a function of $\mathbf{b}$, namely the symbol chosen by the colluders after they have observed $\mathbf{b}$, such that $\tilde{\mu}$ is minimized. Notice that (75) has the same form as (61); the only difference lies in the factor between the curly brackets. Numerical results for (75) are shown in Figs. 9 and 10. It is clear from Fig. 9 that λ_3 hardly depends on c.

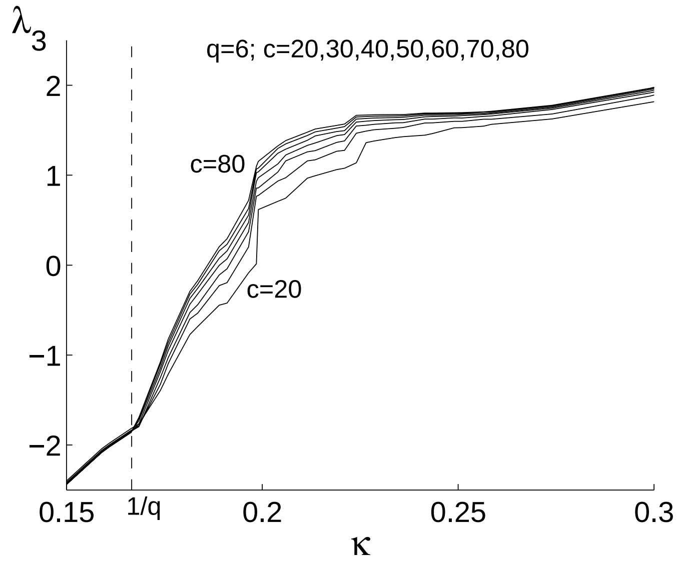

Fig. 9: Third moment λ_3 as a function of κ for various coalition sizes c, for $q = 6$. The colluders employ the 'extremal' strategy.

Finally we substitute some numerical values into (73). From Lemma 3 we have $\tilde{\sigma}_j = 1$. We use the result (62), $m = (2/\tilde{\mu}^2)c_0^2 \ln \varepsilon_1^{-1}$. We set $\varepsilon_1 = 10^{-15}$, corresponding to the probability of an 8-sigma event. We wish the CLT to apply in a central region with #sigmas≥ 8. According to (73), this requirement is satisfied for $c_0 \gtrapprox 10 \cdot \lambda_3 \tilde{\mu}$.

We use Fig. 10 to read off the value of λ_3 at the κ-value where $\tilde{\mu}$ (61) is in the optimal range (as shown in Fig. 2). Setting κ slightly larger than $1/q$, we see that $|\lambda_3| < 1$. Hence, for $q \leq 10$, given the $\tilde{\mu}$-values plotted in Fig. 2, we conclude that the Gaussian approximation applies when the code is built to resist coalitions of size c_0 larger than some threshold lying between approximately 10 and 20. The larger c_0, the better the Gaussian approximation.

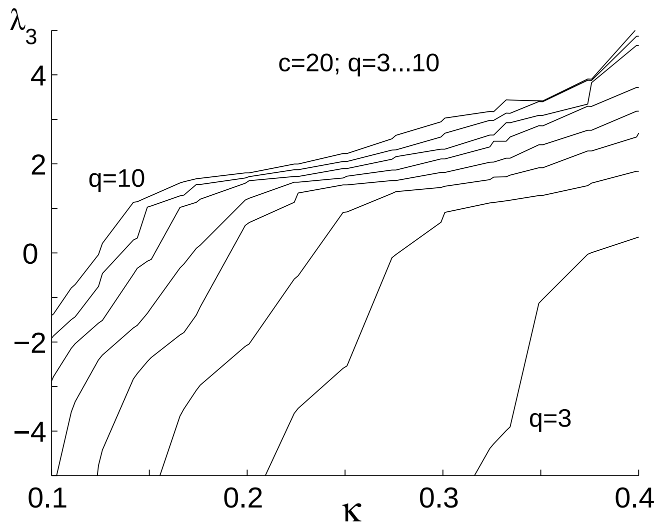

Fig. 10: Third moment λ_3 as a function of κ for various alphabet sizes q, for $c = 20$. The colluders employ the 'extremal' strategy.