False Positive probabilities in q-ary Tardos codes: comparison of attacks

Antonino Simone Boris Škorić

Abstract

We investigate False Positive (FP) accusation probabilities for q-ary Tardos codes in the Restricted Digit Model. We employ a computation method recently introduced by us, to which we refer as Convolution and Series Expansion (CSE). We present a comparison of several collusion attacks on q-ary codes: majority voting, minority voting, Interleaving, $\tilde{\mu}$-minimizing and Random Symbol (the q-ary equivalent of the Coin Flip strategy). The comparison is made by looking at the FP rate at approximately fixed False Negative rate. In nearly all cases we find that the strongest attack is either minority voting or $\tilde{\mu}$-minimizing, depending on the exact setting of parameters such as alphabet size, code length, and coalition size.

Furthermore, we present results on the convergence speed of the CSE method, and we show how FP rate computations for the Random Symbol strategy can be sped up by a pre-computation step.

1 Introduction

1.1 Collusion attacks against forensic watermarking

Fingerprinting provides a means for tracing the origin and distribution of digital data. Before distribution, the content is modified by applying an imperceptible fingerprint, which plays the role of a personalized serial number. The fingerprint is usually embedded through a watermarking algorithm. Once an unauthorized copy of the content is found, the identity can be determined of those users who participated in the creation of the unauthorized copy. This can be done using a tracing algorithm, which outputs a list of allegedly guilty users. This process is also known as 'forensic watermarking'.

Reliable tracing of content requires security against attacks that aim to remove the embedded information. Collusion attacks, where a group of pirates collude to compare their copies, are a particular threat. As any differences between the copies have to arise from the watermarks and not the content, such a comparison gives information which can be used to remove the watermark. To counter this threat, coding theory has produced a number of collusion-resistant codes. In any practical implementation, they must be combined with an embedding scheme. The resulting system has two layers [6, 12]: The coding layer determines which message to embed and protects against collusion attacks. The underlying watermarking layer hides symbols in segments of the content. The symbols are either binary or q-ary. The interface between the layers is usually specified in terms of the *marking assumption* plus additional assumptions that are referred to as a 'model'. The marking assumption states that the colluders are able to perform modifications only in those content segments where the colluders received differently marked content. These segments are called detectable positions. The 'model' specifies the kind of symbol manipulations that the attackers are able to perform in detectable positions. The commonly used *restricted digit model* only allows them to choose pieces from their copies of the content, i.e. each segment of the unauthorized copy carries exactly one symbol that the attackers have available. The *unreadable digit model* allows for slightly stronger attacks. The attackers are also able to erase the fingerprint at detectable positions. Under the *arbitrary digit model* they can put arbitrary symbols in detectable positions, while the *general digit model* additionally allows erasures at detectable positions.

1.2 Tardos codes

Many collusion resistant codes have been proposed in the literature. Most notable are the Boneh-Shaw construction [4] and the by now famous Tardos code [16]. The former construction uses a concatenation of an inner code with a random outer code, while the latter one is a fully randomized binary code. We briefly summarize some of the most important developments regarding Tardos codes.

The number of users is n. In Tardos' original paper [16] a binary code was given achieving length $m = 100c_0^2\lceil \ln \frac{1}{\varepsilon_1} \rceil$, along with a proof that $m \propto c_0^2$ is asympotically optimal[1] for large coalitions, for all alphabet sizes. Here c_0 denotes the number of colluders that can be resisted, and ε_1 is the maximum allowed probability of accusing a fixed[2] innocent user.

The original Tardos code construction contained two unfortunate design choices which caused the proportionality constant '100' to be so high. First, the false negative probability ε_2 (not accusing any of the guilty users) was coupled to ε_1 according to $\varepsilon_2 = \varepsilon_1^{c_0/4}$. This gives $\varepsilon_2 \ll \varepsilon_1$ which is highly unusual in the context of content distribution; a deterring effect is achieved already at $\varepsilon_2 \approx \frac{1}{2}$, while the false positive probability ($\approx n\varepsilon_1$) needs to be very small. In the subsequent literature (e.g. [18, 2]) the ε_2 was decoupled from ε_1, leading to a substantial improvement of the code length.

Second, the symbols 0 and 1 were not treated on an equal footing. Only segments where the attackers produce a 1 were taken into account. This procedure ignores 50% of all the available information. A fully symbol-symmetric version of the Tardos code was given in [17], leading to a further improvement of the code length by a factor 4.

A further improvement was achieved in [11]. The Tardos code construction consists of two probabilistic steps. In the first step, a bias parameter is generated for each segment. In Tardos' original construction the probability density function (pdf) for the bias is a continuous function, suitable for arbitrary coalition size. In [11] a class of discrete distributions was given that performs better against finite coalition sizes than the original pdf.

All the above mentioned work followed the so-called 'simple decoder' approach, i.e. an accusation score is computed for each user independently, and if it exceeds a certain threshold, the user is considered suspicious. In contrast, one can also use a 'joint decoder' which considers sets of users. Amiri and Tardos [1] have given a capacity-achieving joint decoder construction for the binary code. (Capacity refers to the information-theoretic treatment [15, 10, 7, 3] of the colluder attack as a communication channel.) However, the construction is rather impractical, requiring computations for many candidate coalitions. Even if more practical joint decoders are found, the simple decoder will serve as a stepping stone in their operation. Thus, interest in the simple decoder remains high.

In [17] the binary construction was generalized to alphabets of arbitrary size q, in the simple decoder approach. It was shown that, in the restricted digit model, the transition to a larger alphabet size has benefits beyond the mere fact that a q-ary symbol carries $\log_2 q$ bits of information.

1.3 Exact computation of the false positive error probability

The so-called 'Gaussian approximation' or 'Gaussian assumption', introduced in [18], has been a useful tool in the analysis of Tardos codes. The assumption is that the pdf of a user's accusation score has a normal distribution. When this is the case, the statistical analysis of the code's performance can be drastically simplified; the performance is almost completely determined by a single parameter, namely the average score $\tilde{\mu}$ of the coalition.

The Gaussian assumption is motivated by the Central Limit Theorem (CLT): A user accusation consists of a sum of per-segment contributions, which are independent and identically distributed

[1] The proportionality $m \propto c_0^2$ was already known in the context of spread-spectrum watermarking. Kilian et al. [9] showed that, if the watermarks have a component-wise normal distribution, then $\Omega(\sqrt{m/ln\, n})$ differently marked copies are required to successfully erase any mark with non-negligible probability.

[2] Not to be confused with the total false positive probability (which we denote as η). The relation is $\eta = 1 - (1-\varepsilon_1)^{n-c_0} \approx n\varepsilon_1$.

(i.i.d.). When many of these get added together, the result is close to normal-distributed, i.e. the pdf is very close to a Gaussian in a certain region around the average, and deviates in the tails. The longer the code becomes (i.e. the larger the coalition size c_0), the wider this central region. In [18] and [17] theoretical results were provided arguing that the central region is sufficiently wide to allow for application of the Gaussian approximation for realistic parameter choices. However, these arguments are not very precise.
In [14] and [13] an in-depth analytical and numerical investigation of the Gaussian approximation was given in the RDM case. The approach is based on the convolution rule for characteristic functions, and a way to express the false accusation probability as a power series expansion in the small parameter $1/m$. We will refer to this method as 'the CSE method' (Convolution and Series Expansion). The advantage of the CSE method over simulations is that it yields reliable results also when the required false positive probability is very small. For instance, if the false positive rate is around 10^{-10}, then a number of simulations of order at least 10^{10} is required to measure this rate; in contrast, the computational effort in the CSE method does not depend on the error rate.
The work of [14, 13] showed, for various parameter settings and attack strategies, how the false positive probability has a transition from Gaussian behaviour in the central region to worse-than-Gaussian behaviour outside the center. However, a number of questions were left unanswered. For instance, the method only performs well if $1/m$ is small enough. This boundary condition was not quantified in detail. The shape of the probability tails was also not investigated. Furthermore, only a small number of strategies was considered, and a small part of parameter space.

1.4 Contributions and outline

In this paper we study the simple decoder setting in the Restricted Digit Model using the CSE method. We extend the list of strategies that can be handled effectively, quantify the properties of the CSE method more precisely, and present numerics for a larger part of parameter space than was done previously.

- The CSE method can compute error probabilities fast only if the attack strategy allows for a certain pre-computation step. The pre-computation is a weighted sum over the attack parameters, and the result is referred to as 'K_b' in [14, 13]. The K_b was pre-computed for the following attacks: majority voting, minority voting, interleaving, and the attack that minimizes the coalition score.

  In this paper we show how the K_b is pre-computed for the 'Random Symbol' attack, i.e. the q-ary equivalent of the 'coin flip' attack, in which all symbols observed by the coalition have equal probability regardless of their occurrence frequency.

- We show how mixed strategies can be accommodated in the CSE method. By 'mixed' we mean that the strategy is allowed to differ for different content segments. The convolution property of characteristic functions applies equally for mixed strategies and ordinary strategies.

- We show graphs of the probability density function (pdf) of the single-segment score for an innocent user. Full equations for this pdf were already given in [14], but the graphs better illustrate the differences between the various strategies.

- We study the shape of the probability distribution of the total innocent score, by plotting the false positive probability as a function of the threshold. In the tail we find, as expected, power-law behaviour that precisely matches the single-segment tail probability.

- The expansion in the CSE needs a certain number of terms in order to achieve sufficient convergence. This is indicated by a power '$\nu_{\max}$' where the expansion is cut off. Computation times heavily depend on this power. For various attack strategies and parameter settings we tabulate required $\nu_{\max}$ values.

- We present a comparison of all the attack strategies for which we can pre-compute the K_b. Ideally this should be done via ROC curves. However, we cannot compute exact false *negative* probabilities.[3] Thus, our comparison is based on the false positive probability for a threshold chosen such that the false negative is 'under control' (some unknown number close to 2^{-c}).

  We find that in nearly all cases the strongest attack is either minority voting or $\tilde{\mu}$-minimizing.

The paper is structured as follows. In Section 2 we briefly review the q-ary Tardos code and the CSE method. In Section 3 we show the K_b pre-computation for the Random Symbol strategy. In Section 4 we analyze the FP tail power law for several strategies, showing that it is equal to the single-segment tail behaviour. In Section 5 we study the convergence properties of the series expansion in the CSE method. In Section 6 we present the comparison between five strategies over a large part of parameter space. A summary is given in Section 7.

2 Preliminaries

We briefly describe the q-ary version of the Tardos code as introduced in [17] and the CSE method for computing innocent accusation probabilities.

2.1 The q-ary Tardos code

The number of symbols in a codeword is m. The number of users is n. The alphabet is $\mathcal{Q}$, with size q. $X_{ji} \in \mathcal{Q}$ stands for the i'th symbol in the codeword of user j. The whole matrix of codewords is denoted as X.

Two-step code generation.

m vectors $\boldsymbol{p}^{(i)} \in [0,1]^q$ are independently drawn according to a Dirichlet distribution F, with

$$F(\boldsymbol{p}) = \delta(1 - \sum_{\beta \in \mathcal{Q}} p_\beta) \cdot \frac{1}{B(\kappa \mathbf{1}_q)} \prod_{\alpha \in \mathcal{Q}} p_\alpha^{-1+\kappa}. \tag{1}$$

Here $\mathbf{1}_q$ stands for the vector $(1, \cdots, 1)$ of length q, $\delta(\cdot)$ is the Dirac delta function, and B is the generalized Beta function. κ is a positive constant called the 'concentration parameter' of the Dirichlet distribution. For $v_1, \cdots, v_n > 0$ the Beta function is defined as[4]

$$B(\boldsymbol{v}) = \int_0^1 \mathrm{d}x^n \; \delta(1 - \sum_{a=1}^n x_a) \prod_{b=1}^n x_b^{-1+v_b} = \frac{\prod_{a=1}^n \Gamma(v_a)}{\Gamma(\sum_{b=1}^n v_b)}. \tag{2}$$

All elements X_{ji} are drawn independently according to $\Pr[X_{ji} = \alpha | \boldsymbol{p}^{(i)}] = p_\alpha^{(i)}$.

Attack.

The coalition is a subset of the set of all users. We denote the coalition as $\mathcal{C}$, with size c. The i'th segment of the attacked content contains a symbol $y_i \in \mathcal{Q}$. We define vectors $\boldsymbol{\sigma}^{(i)} \in \mathbb{N}^q$ as

$$\sigma_\alpha^{(i)} \;=\; |\{j \in \mathcal{C} : X_{ji} = \alpha\}| \tag{3}$$

satisfying $\sum_{\alpha \in \mathcal{Q}} \sigma_\alpha^{(i)} = c$. In words: $\sigma_\alpha^{(i)}$ counts how many colluders have received symbol α in segment i. The attack strategy may be probabilistic. As usual, it is assumed that this strategy is segment-symmetric (the same in all segments), symbol-symmetric (invariant under permutation of the alphabet) and attacker-symmetric (invariant under permutation of the attackers). The strategy is expressed as probabilities $\theta_{y|\boldsymbol{\sigma}}$ that apply independently for each segment. Omitting the column index,

$$\Pr[y|\boldsymbol{\sigma}] = \theta_{y|\boldsymbol{\sigma}}. \tag{4}$$

Some often studied strategies are listed below.

[3] This is work in progress.

[4] This is also known as a Dirichlet integral. The ordinary Beta function ($n = 2$) is $B(x,y) = \Gamma(x)\Gamma(y)/\Gamma(x+y)$.

Strategy	Abbrev.	Description	$\theta_{y\|\boldsymbol{\sigma}}$
Minority Voting	MinV	Select symbol that occurs least often	
Majority Voting	MajV	Select symbol that occurs most often	
Interleaving	Int	Select random attacker's symbol	σ_y/c
$\tilde{\mu}$-minimizing	$\tilde{\mu}$-min	Select $\sigma_y > 0$ that minimizes $\tilde{\mu}$ (see below)	
Random Symbol	RS	Choose uniformly from received symbols	$\frac{[\sigma_y>0]}{\|\{\alpha\in\mathcal{Q}:\sigma_\alpha>0\}\|}$

The Int attack has been proved [8] to minimize the achievable code rate for $c \to \infty$, i.e. it is the best attack in the so-called 'joint decoder' setting. This does not imply that it is also the strongest attack when the tracing party is limited to 'simple' decoders, i.e. accusations computed per user. Indeed, it is known that asymptotically $\tilde{\mu}$-min is the strongest attack against the symmetrized Tardos score function [17].

Accusation.

The watermark detector sees the symbols y_i. For each user j, the *accusation sum* S_j is computed,

$$S_j = \sum_{i=1}^{m} S_j^{(i)} \qquad \text{where} \qquad S_j^{(i)} = \; g_{[X_{ji}==y_i]}(p_{y_i}^{(i)}), \tag{5}$$

where the expression $[X_{ji} == y_i]$ evaluates to 1 if $X_{ji} = y_i$ and to 0 otherwise, and the functions g_0 and g_1 are defined as

$$g_1(p) = \sqrt{\frac{1-p}{p}} \qquad ; \qquad g_0(p) = -\sqrt{\frac{p}{1-p}}. \tag{6}$$

The total accusation of the coalition is $S = \sum_{j\in\mathcal{C}} S_j$. The choice (6) is the unique choice that, for innocent users, yields zero average accusation and variance equal to 1 independent of $\boldsymbol{p}$,

$$pg_1(p) + (1-p)g_0(p) = 0 \qquad ; \qquad p[g_1(p)]^2 + (1-p)[g_0(p)]^2 = 1. \tag{7}$$

This has been shown to have optimal properties for $q = 2$ [5, 18]. Its unique properties (7) also hold for $q \geq 3$; that is the main motivation for using (6). A user j is 'accused' if his accusation sum S_j exceeds a threshold Z, i.e. $S_j > Z$. The parameter $\tilde{\mu}$ is defined as $\frac{1}{m}\mathbb{E}[S]$, where $\mathbb{E}$ stands for the expectation value over all random variables. The $\tilde{\mu}$ depends on q, κ, the collusion strategy, and weakly on c. In the limit of large c it converges to a finite value, and the code length scales as $m \propto c^2/\tilde{\mu}^2$.

2.2 Marginal distributions and strategy parametrization

Because of the independence between segments, the segment index will be dropped from this point onward. For given $\boldsymbol{p}$, the vector $\boldsymbol{\sigma}$ is multinomial-distributed, $\mathbb{P}(\boldsymbol{\sigma}|\boldsymbol{p}) = \binom{c}{\boldsymbol{\sigma}} \prod_\alpha p_\alpha^{\sigma_\alpha}$. Averaged over $\boldsymbol{p}$, the $\boldsymbol{\sigma}$ has distribution $\mathbb{P}(\boldsymbol{\sigma}) = \binom{c}{\boldsymbol{\sigma}} \frac{B(\kappa\mathbf{1}_q+\boldsymbol{\sigma})}{B(\kappa\mathbf{1}_q)}$. Two important marginals were given in [14]. First, the marginal probability $\mathbb{P}_1(b) = \Pr[\sigma_\alpha = b]$ for one arbitrary component α,

$$\mathbb{P}_1(b) = \binom{c}{b} \frac{B(\kappa + b, \kappa[q-1] + c - b)}{B(\kappa, \kappa[q-1])}. \tag{8}$$

Second, given that $\sigma_\alpha = b$, the probability that the remaining $q-1$ components of the vector $\boldsymbol{\sigma}$ are given by $\boldsymbol{x}$,

$$\mathbb{P}_{q-1}(\boldsymbol{x}|b) = \binom{c-b}{\boldsymbol{x}} \frac{B(\kappa\mathbf{1}_{q-1} + \boldsymbol{x})}{B(\kappa\mathbf{1}_{q-1})}. \tag{9}$$

It is always implicit that $\sum_{\beta\in\mathcal{Q}\setminus\{\alpha\}} x_\beta = c - b$. An alternative parametrization was introduced for the collusion strategy, which exploits the fact that (i) $\theta_{\alpha|\boldsymbol{\sigma}}$ is invariant under permutation of the symbols$\neq \alpha$; (ii) $\theta_{\alpha|\boldsymbol{\sigma}}$ depends on α only through the value of σ_α.

$$\Psi_b(\boldsymbol{x}) = \theta_{\alpha|\boldsymbol{\sigma}} \text{ given that } \sigma_\alpha = b \text{ and } \boldsymbol{x} = \text{the other components of } \boldsymbol{\sigma}. \tag{10}$$

$\mathcal{Q}$	the alphabet
q	alphabet size $\lvert\mathcal{Q}\rvert$
n	number of users
$\mathcal{C}$	set of colluding users
c	number of colluders $\lvert\mathcal{C}\rvert$
c_0	coalition size that the code can resist
m	code length (number of segments)
X_{ji}	symbol in segment i for user j
$\boldsymbol{p}^{(i)}$	bias vector for segment i
F	distribution of the bias vector, $\boldsymbol{p}^{(i)} \sim F$
$f(p_\alpha)$	marginal distribution of F for one component of $\boldsymbol{p}$
κ	shape parameter contained in F
$\sigma_\alpha^{(i)}$	# occurrences of symbol α in attackers' segment i
$\mathbb{P}$	probability distribution for $\boldsymbol{\sigma}$
$\mathbb{P}_1$	marginal distribution for one component of $\boldsymbol{\sigma}$

$\mathbb{P}_{q-1}$	marginal distribution for $q-1$ components of $\boldsymbol{\sigma}$
y_i	symbol in segment i of attacked content
$\theta_{y\lvert\boldsymbol{\sigma}}$	prob. that attackers output symbol y, given $\boldsymbol{\sigma}$
S_j	accusation sum of user j
S	coalition accusation sum, $S = \sum_{j\in\mathcal{C}} S_j$
Z	accusation threshold
$\tilde{Z}$	$Z/\sqrt{m}$
ε_1	max. tolerable prob. of fixed innocent getting accused
ε_2	max. tolerable prob. of not catching any attacker
$\tilde{\mu}$	$\mathbb{E}[S]/m$; does not depend on m
φ	prob. distribution of innocent's one-segment score
$\Psi_b(\boldsymbol{x})$	$\theta_{y\lvert\boldsymbol{\sigma}}$ when $\sigma_y = b$ and $\boldsymbol{\sigma}\backslash\sigma_y = \boldsymbol{x}$
K_b	quantity derived from $\Psi_b(\boldsymbol{x})$ by averaging over $\boldsymbol{x}$

Thus, $\Psi_b(\boldsymbol{x})$ is the probability that the pirates choose a symbol that they have seen b times, given that the other symbols' occurrences are $\boldsymbol{x}$. Strategy-dependent parameters K_b were introduced as follows,

$$K_b = \mathbb{E}_{\boldsymbol{x}|b}\Psi_b(\boldsymbol{x}) = \sum_{\boldsymbol{x}} \mathbb{P}_{q-1}(\boldsymbol{x}|b)\Psi_b(\boldsymbol{x}). \tag{11}$$

Due to the marking assumption it holds that $K_0 = 0$ and $K_c = 1$. Furthermore, the K_b obey the sum rule $q\sum_{b=0}^{c} K_b\mathbb{P}_1(b) = 1$. Efficient pre-computation of the K_b parameters can significantly speed up the computation of a number of quantities of interest, among which the $\tilde{\mu}$ parameter. It was shown that $\tilde{\mu}$ can be expressed as

$$\tilde{\mu} = \sum_{\boldsymbol{\sigma}} \mathbb{P}(\boldsymbol{\sigma}) \sum_{\alpha\in\mathcal{Q}} \theta_{\alpha|\boldsymbol{\sigma}} T(\sigma_\alpha) = q\sum_{b=0}^{c} K_b\mathbb{P}_1(b)T(b), \tag{12}$$

where

$$T(b) = \left\{\tfrac{1}{2} - \kappa + \frac{b}{c}(\kappa q - 1)\right\} c\frac{\Gamma(b+\kappa-\frac{1}{2})}{\Gamma(b+\kappa)}\frac{\Gamma(c-b+\kappa[q-1]-\frac{1}{2})}{\Gamma(c-b+\kappa[q-1])}. \tag{13}$$

2.3 The CSE method for computing false accusation probabilities

We briefly review the method introduced in [14]. It is based on the convolution rule for generating functions (Fourier transforms): Let $A_1 \sim f_1$ and $A_2 \sim f_2$ be continuous random variables, and let $\tilde{f}_1$, $\tilde{f}_2$ be the Fourier transforms of the respective pdfs. Let $A = A_1 + A_2$, and $A \sim \Phi$. Then the easiest way to find Φ is to use the fact that $\tilde{\Phi}(k) = \tilde{f}_1(k)\tilde{f}_2(k)$. If i.i.d. variables $A_i \sim \varphi$ are summed, $A = \sum_{i=1}^{m} A_i$, then the pdf of A is found using $\tilde{\Phi}(k) = [\tilde{\varphi}(k)]^m$.

The pdf of an innocent user's one-segment accusation $S_j^{(i)}$ will be denoted as φ. It was found that φ has the following form,

$$u > 0 \qquad : \qquad \varphi(u) = \frac{2q}{B(\kappa, \kappa[q-1])} \sum_{b=1}^{c} \binom{c}{b} \frac{(u^2)^{\kappa[q-1]+c-b-\frac{1}{2}}}{(1+u^2)^{c+1+\kappa q}} K_b$$

$$u < 0 \qquad : \qquad \varphi(u) = \frac{2q}{B(\kappa, \kappa[q-1])} \sum_{b=1}^{c} \binom{c}{b} \frac{(u^2)^{\kappa+b-\frac{1}{2}}}{(1+u^2)^{c+1+\kappa q}} K_b. \tag{14}$$

Notice that the formulas contain the strategy-dependent parameters K_b. In Figure 1 we show φ for different strategies. The strategy has a minor influence on the left tail, but strongly affects the shape of the right tail. A long positive tail is favorable to the attackers, as it leads to (1) an increased probability of FP errors and (2) slower convergence of the total score pdf to the Gaussian form. We see that MinV causes the biggest tail, followed by RS and Int. MajV has the shortest tail. (The $\tilde{\mu}$-min attack is equivalent to MajV for the given parameter values.) This behaviour is easily understood from the powers of u occurring in (14) for $u > 0$. For $u \gg 1$, the summand is proportional to $K_b(1/u)^{3+2\kappa+2b}$. The dominant contribution to the tail occurs at $b = 1$. MinV has a very large K_1 due to its preference for symbols that occur infrequently. In contrast, MajV has $K_b = 0$ for $b < c/q$.

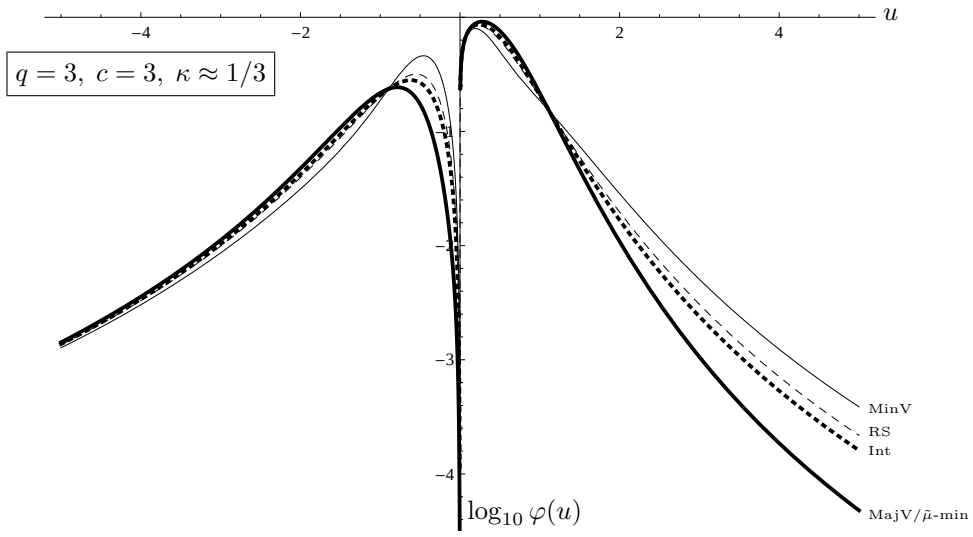

Figure 1: *The pdf φ of the single-segment score, shown for several strategies. The right tail strongly depends on the strategy, while the left tail is hardly affected.*

The Fourier transform $\tilde{\varphi}$ was computed in [14], and an expression for $[\tilde{\varphi}(k/\sqrt{m})]^m$ was derived as a power series in $k/\sqrt{m}$. The pdf of S_j for innocent j then follows from the inverse Fourier transform; finally the FP probability is the area under the tail at $S_j > Z$.
The result was formulated as follows. Let R_m be a function defined as $R_m(\tilde{Z}) := \Pr[S_j > \tilde{Z}\sqrt{m}]$ (for innocent j). Let Ω be the corresponding function in case the pdf of S_j is Gaussian, $\Omega(\tilde{Z}) = \frac{1}{2}\mathrm{Erfc}(\tilde{Z}/\sqrt{2})$. Then

$$R_m(\tilde{Z}) = \Omega(\tilde{Z}) + \frac{1}{\pi} \sum_{t=0}^{\infty} \omega_t(m) \Gamma(\nu_t) 2^{\nu_t/2} \mathrm{Im}\left[i^{-\alpha_t} H_{-\nu_t}(i\tilde{Z}/\sqrt{2}) \right]. \tag{15}$$

Here H is the Hermite function. The powers[5] ν_t satisfy $\nu_0 > 2$, $\nu_{t+1} > \nu_t$. In general the ν_t are not all integer. The $\omega_t(m)$ are real-valued coefficients that decrease as $m^{-\nu_t/6}$ or faster; The α_t are real-valued coefficients.

[5]The coefficients ν_t appear as powers of $k/\sqrt{m}$ in the series expansion of $\tilde{\varphi}^m$.

Computation of all the α_t, ω_t, ν_t up to a certain cutoff $t = t_{\max}$ is straightforward but very laborious, and leads to huge expressions if done analytically. It is best done semi-numerically using a software package such as Mathematica.
It holds that $\lim_{m\to\infty} R_m(\tilde{Z}) = \Omega(\tilde{Z})$, i.e. the pdf converges to a Gaussian. For a good numerical approximation it suffices to take terms up to some cutoff $t_{\max}$ or, equivalently, a power $\nu_{\max}$. The required $\nu_{\max}$ is a decreasing function of m. This will be discussed in Section 5.
It is worth remarking that the CSE method can be applied even when the colluders have the option of choosing a strategy for each content segment separately. Let φ_s denote the φ-function for some strategy s, and let m_s be the number of segments in which this strategy is applied. The only thing we have to do is replace

$$[\tilde{\varphi}(k/\sqrt{m})]^m \to \prod_{s\in\text{strategies}} [\tilde{\varphi}_s(k/\sqrt{m})]^{m_s} \tag{16}$$

and then follow all the derivation steps as before. The study of such situations is left for future work.

3 Pre-computation of K_b for the Random Symbol strategy

Our first contribution is the computation of the parameter K_b (11) for the RS strategy.

Definition 1 (Random Symbol strategy) *In the Random Symbol (RS) strategy, the coalition detects w distinct symbols in a content segment and chooses one of them uniformly with probability $1/w$.*

In the binary case this is known as the 'coin flip' strategy. The analysis of the RS strategy is different from other strategies because the RS strategy does not depend on the actual σ_α values (other than their being zero or nonzero). We obtain a result that looks similar to the formula for MajV in [14] and the more general 'class 2' strategies in [13], even though RS does not fit in the classification scheme introduced in [13].

Theorem 1 *Let $q > 2$ and $b \in \{1,\dots,c-1\}$. Let $N_b \in \mathbb{N}$ satisfy $N_b > (c-b)(q-2)$. Let $\tau_b = e^{i2\pi/N_b}$, and let G_{ba} be defined as*

$$G_{ba} = \sum_{z=1}^{c-b} \frac{\Gamma(\kappa+z)}{\tau_b^{az} z!}. \tag{17}$$

The K_b parameter for the RS strategy can then be expressed as

$$K_b^{\mathrm{RS}} = \frac{(c-b)!\Gamma(\kappa[q-1])\Gamma(\kappa)}{qN_b\Gamma(c-b+\kappa[q-1])} \sum_{a=0}^{N_b-1} \tau_b^{a(c-b)} \frac{(G_{ba}/\Gamma(\kappa)+1)^q - 1}{G_{ba}}. \tag{18}$$

Proof: See Appendix A.
For fixed b, naive evaluation of the $\boldsymbol{x}$-sum in (11) would involve $\mathcal{O}(c^{q-2}/(q-1)!)$ terms. Theorem 1 reduces the number of terms to $\mathcal{O}(qc^2)$: a factor $c-b$ from the z-sum and a factor $N_b = \mathcal{O}(qc)$ from the a-sum. We will use (18) for the numerics in Section 6.
Theorem 1 holds for $q > 2$. For the binary alphabet the result is much simpler.

Lemma 1 *Let $q = 2$ and $b \in \{1,\dots,c-1\}$. Then the K_b parameter for the RS strategy is*

$$K_b^{\mathrm{RS}} = \tfrac{1}{2}. \tag{19}$$

Proof:
With $b \in \{1,\dots,c-1\}$ it is guaranteed that both symbols in the alphabet are detected by the attackers. Then, by definition of the RS strategy, one of the two symbols is chosen uniformly at random. □

Lemma 2 *Let $q > 2$. Then*

$$K_1^{\mathrm{RS}} < K_2^{\mathrm{RS}} < \cdots < K_{c-1}^{\mathrm{RS}} = \tfrac{1}{2}. \tag{20}$$

Proof:
When b increases, the average number of symbols $\alpha \in \mathcal{Q}$ with $\sigma_\alpha > 0$ decreases. At $b = c - 1$ it is guaranteed that the number of detected symbols is exactly two. □

4 Power-law behaviour of the FP tail

Our second contribution is an analysis of the FP tails found by the CSE method. It was mentioned in [14] that the large number of terms in the segment-summation causes the pdf of S_j to converge to the Gaussian form (Central Limit Theorem) for small $|S_j|$, while for large $|S_j|$ the original power-laws from the single-segment pdf $\varphi(u)$ (14) prevail. However, the statement about the power laws was not explicitly demonstrated.
We present numerics showing that the tail of $R_m(\tilde{Z})$ (15) indeed has power law behaviour that follows directly from the dominant contribution in $\varphi(u)$ (14). In Section 2.3 we saw that all the investigated strategies, except MajV and sometimes $\tilde{\mu}$-min, have $K_1 > 0$. This leads to a dominant term $\propto (1/u)^{5+2\kappa}$ at $u \gg 1$. Hence, far into the right tail we have $\varphi(u) \propto (1/u)^{5+2\kappa}$. Integrating the tail beyond a threshold z we then get $\int_z^\infty \mathrm{d}u\, \varphi(u) \propto (1/z)^{4+2\kappa}$. Thus, if the statement about the tails is correct, we expect $\log R_m(\tilde{Z}) = -(4+2\kappa)\log \tilde{Z} + \text{constant}$ at $\tilde{Z} \gg 1$ for the MinV, RS, Int strategies (and $\tilde{\mu}$-min whenever it is not equivalent to majV). In Fig. 4 we show a log-log plot of $R_m(\tilde{Z})$ for several strategies, for one combination of q, c, m, κ. (Without providing evidence we mention that the behaviour is the same for other parameter choices.) We have also plotted the single-segment FP probability $R_1(\tilde{Z}) = \int_{\tilde{Z}}^\infty \mathrm{d}u\, \varphi(u)$ for Int. We notice the following

- The tails of MinV, RS, Int and $\tilde{\mu}$-min indeed follow the expected power law, as can be seen from the straight lines that are parallel to each other and to the single-segment curve.
- The curves for the different strategies lie in the same order as in Fig. 1.

The fact that the tail of the MinV curve lies higher than the rest was explained in Section 2.3: the K_1 parameter determines how strongly the dominant power $-(5+2\kappa)$ is present in $\varphi(u)$, and MinV has the highest K_1 of all strategies. The order of RS and Int can also be understood from the value of K_1.

Lemma 3 *It holds that $K_1^{\mathrm{RS}} \geq K_1^{\mathrm{Int}}$.*

Proof:
We have $\Psi_1^{\mathrm{Int}}(\boldsymbol{x}) = \frac{1}{c}$ and, $\Psi_1^{\mathrm{RS}}(\boldsymbol{x}) = \frac{1}{s(\boldsymbol{x})+1}$, where $s(\boldsymbol{x})$ is the number of non-zero elements in $\boldsymbol{x}$. We can bound $s(\boldsymbol{x})$ as $s(\boldsymbol{x}) + 1 \leq \min\{c, q\}$ since the number of distinct received symbols cannot exceed the alphabet size or the coalition size. This yields

$$\Psi_1^{\mathrm{RS}}(\boldsymbol{x}) = \frac{1}{s(\boldsymbol{x})+1} \geq \frac{1}{\min\{c,q\}} = \max\left\{\frac{1}{c}, \frac{1}{q}\right\} \geq \frac{1}{c} = \Psi_1^{\mathrm{Int}}(\boldsymbol{x}). \tag{21}$$

Finally, from the definition of K_b (11) we know that $\Psi_1^{\mathrm{RS}}(\boldsymbol{x}) \geq \Psi_1^{\mathrm{Int}}(\boldsymbol{x})$ implies $K_1^{\mathrm{RS}} \geq K_1^{\mathrm{Int}}$. □
The $\tilde{\mu}$-min strategy is more difficult to analyze. As shown in [13], it behaves in a rather complicated way, sometimes being equivalent to MinV, sometimes MajV, or something inbetween, depending on many parameters, mostly κ and q.

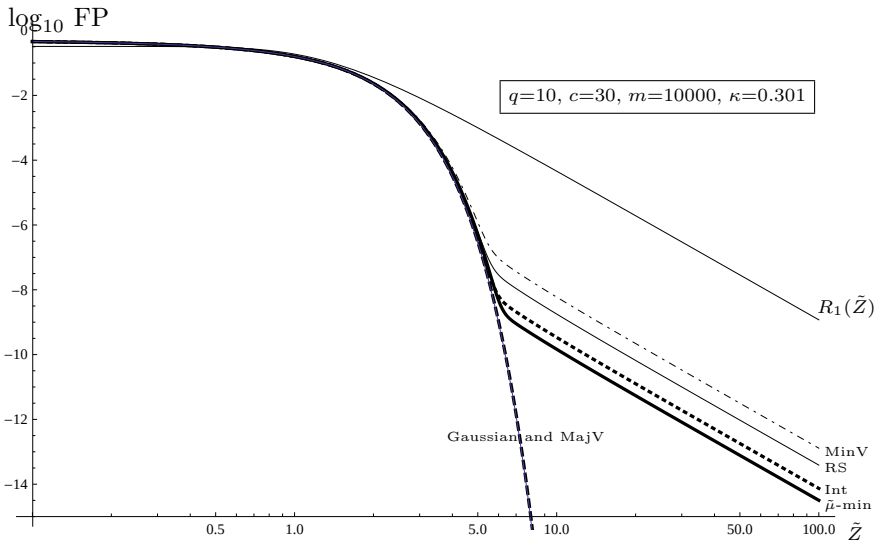

Figure 2: *Log-log plot of $R_m(\tilde{Z})$ for several strategies. The single-segment FP probability $R_1(\tilde{Z})$ for the Int is also plotted.*

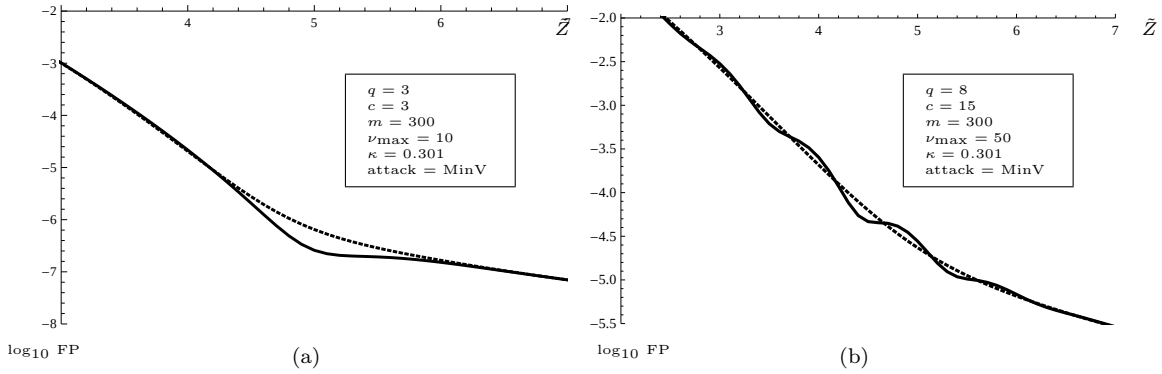

Figure 3: *Examples of incorrect $R_m(\tilde{Z})$ curves (solid line) when the cutoff $\nu_{\max}$ is chosen too small. Oscillations occur in the region where the curve departs from Gaussian behaviour. The dotted curve is the correct result.*

5 Numerical study of the power series cutoff

Our third contribution is an investigation how the cutoff power $\nu_{\max}$ should be chosen in order to achieve sufficient accuracy in the numerical computation of $R_m(\tilde{Z})$ while keeping the computation time of the series expansion (15) under control.

While it is obvious that increasing m should improve the convergence of the expansion (remember that the 'small parameter' in the expansion is $k/\sqrt{m}$), we have not been able to find an expression, or even a rule of thumb, that a priori predicts good values for $\nu_{\max}$. Several parameters have a large impact on the speed of convergence[6], in particular the attack strategy.

When $\nu_{\max}$ is chosen too small, we observe one of the following problems:

- $R_m(\tilde{Z})$ is not in the range $[0, 1]$ for all $\tilde{Z}$.

[6]By 'convergence' we mean convergence of the series to the correct value $R_m(\tilde{Z})$, not to be confused with the CLT effect that the pdf tends to the Gaussian form.

- $R_m(\tilde{Z})$ is not a smooth function of $\tilde{Z}$. The most pronounced effect is around the point where the curve leaves the Gaussian curve. Examples are shown in Fig. 3.

Table 1 shows $\nu_{\max}$ values which lead to a correct $R_m(\tilde{Z})$ curve, as a function of c, q, m and the attack strategy. The numbers listed in the last four columns are $\nu_{\max}$ values. We investigated $\nu_{\max} \in \{10, 20, 30, 40, 50\}$. The parameter κ is set to approximately $1/q$. The $\tilde{\mu}$-min strategy is then equivalent to MajV [13], so they are shown together in one column.
From the table we can see that $\nu_{\max} = 30$ is in general a safe choice. There are some rare cases where problems occur when $\nu_{\max}$ is *too large*. This happens just for MajV and MinV at small m. We suspect that the expansion parameter $\propto m^{-1/2}$ is not small enough in these cases, leading to an ill-defined series expansion in the CSE method (Section 1.3).
For the MajV strategy there are several situations in which the CSE method fails for the entire attempted $\nu_{\max}$ range. It happens at large c combined with small q.
Apparently, $\nu_{\max}$ has to be increased to get convergence. This effect is not easily explained. For $\kappa \approx 1/q$, the dominant power in the left tail of $\varphi(u)$ (14) is $(1/u)^{5-2/q}$ and in the right tail $(1/u)^{3+2(c+1)/q}$. (Remember, $K_b^{\mathrm{MajV}} = 0$ for $b < c/q$.) For large c/q, we see that the right tail vanishes much more quickly than the left tail. Perhaps this imbalance necessitates a larger number of terms in the expansion to get sufficient accuracy.

c	q	m	MajV / $\tilde{\mu}$-min	MinV	Int	RS
3	3	300	30	$\geq$20	$\geq$30	$\geq$30
	3	1000	$\geq$20	$\geq$20	$\geq$20	$\geq$20
	3	2000	$\geq$20	$\geq$20	$\geq$20	$\geq$20
	5	300	$\geq$20	$\geq$10	$\geq$10	$\geq$10
	5	1000	$\geq$10	$\geq$10	$\geq$10	$\geq$10
	5	2000	$\geq$10	$\geq$10	$\geq$10	$\geq$10
	8,15	300,1000,2000	$\geq$10	$\geq$10	$\geq$10	$\geq$10
5	3	300	-	$\geq$20	$\geq$30	$\geq$20
	3	1000	$\geq$40	$\geq$20	$\geq$20	$\geq$20
	3	2000	$\geq$30	$\geq$10	$\geq$20	$\geq$20
	5	300,1000,2000	$\geq$20	$\geq$10	$\geq$10	$\geq$10
	8	300	$\geq$20	$\geq$10	$\geq$10	$\geq$10
	8	1000	$\geq$20	$\geq$10	$\geq$10	$\geq$10
	8	2000	$\geq$10	$\geq$10	$\geq$10	$\geq$10
	15	300,1000,2000	$\geq$10	$\geq$10	$\geq$10	$\geq$10
8	3	300	-	$\geq$10	$\geq$30	$\geq$20
	3	1000	-	$\geq$10	$\geq$20	$\geq$20
	3	2000	-	$\geq$10	$\geq$20	$\geq$20
	5	300	-	$\geq$10	$\geq$10	$\geq$10
	5	1000	$\geq$20	$\geq$10	$\geq$10	$\geq$10
	5	2000	$\geq$20	$\geq$10	$\geq$10	$\geq$10
	8	300,1000,2000	$\geq$20	$\geq$10	$\geq$10	$\geq$10
	15	300	$\geq$20	10 - 40	$\geq$10	$\geq$10
	15	1000	$\geq$20	$\geq$10	$\geq$10	$\geq$10
	15	2000	$\geq$10	$\geq$10	$\geq$10	$\geq$10
15	3	300	-	$\geq$10	$\geq$30	$\geq$20
	3	1000	-	$\geq$10	$\geq$20	$\geq$20
	3	2000	-	$\geq$10	$\geq$20	$\geq$10
	5	300,1000,2000	-	$\geq$10	$\geq$10	$\geq$10
	8	300	-	10 - 30	$\geq$10	$\geq$10
	8	1000	$\geq$40	$\geq$10	$\geq$10	$\geq$10
	8	2000	$\geq$30	$\geq$10	$\geq$10	$\geq$10
	15	300	30	10	$\geq$10	$\geq$10
	15	1000	$\geq$20	$\geq$10	$\geq$10	$\geq$10
	15	2000	$\geq$20	$\geq$10	$\geq$10	$\geq$10

Table 1: *Cutoff values $\nu_{\max}$ giving proper convergence of the CSE method to $R_m(\tilde{Z})$, listed for various combinations of coalition size, alphabet size, code length and attack strategy. $\kappa \approx 1/q$.*

6 Comparison of attack strategies

Our fourth contribution is a comparison of attack strategies. In [14, 13] various FP plots were shown, but they did not span a large part of the parameter space, since their main purpose was to illustrate the CSE method and the $\tilde{\mu}$-min attack, respectively. In this chapter we aim to present a comprehensive overview of FP probabilities for all the attack strategies mentioned in Section 2.1, for a broad parameter range.

6.1 Comparison method: comparing FP at (approximately) equal FN

Ideally we would like to show ROC curves, but unfortunately the CSE method has not yet[7] been applied to the FN probability. Not having access to accurate FN numbers, we have chosen the following way to compare different attack strategies to each other: we approximately fix the FN probability and then compare the FP probabilities. Here the word 'approximately' needs some explanation. For each strategy we set the threshold Z to a different value. We set $Z = m\tilde{\mu}/c$, where $\tilde{\mu}$ depends on the strategy. We refer to this specific value as Z_{half}. Each colluder separately has a probability of approximately $\frac{1}{2}$ that his score stays below Z_{half} [18]; hence the FN probability is approximately $(\frac{1}{2})^c$. Other than this, very little information is available about the scores of the colluders. Fortunately, the pdf of the collective score S is known to be narrow. Consequently, a broad range of FN values is represented in a narrow interval around Z_{half}, and thus we do not lose much generality by setting $Z = Z_{\text{half}}$.

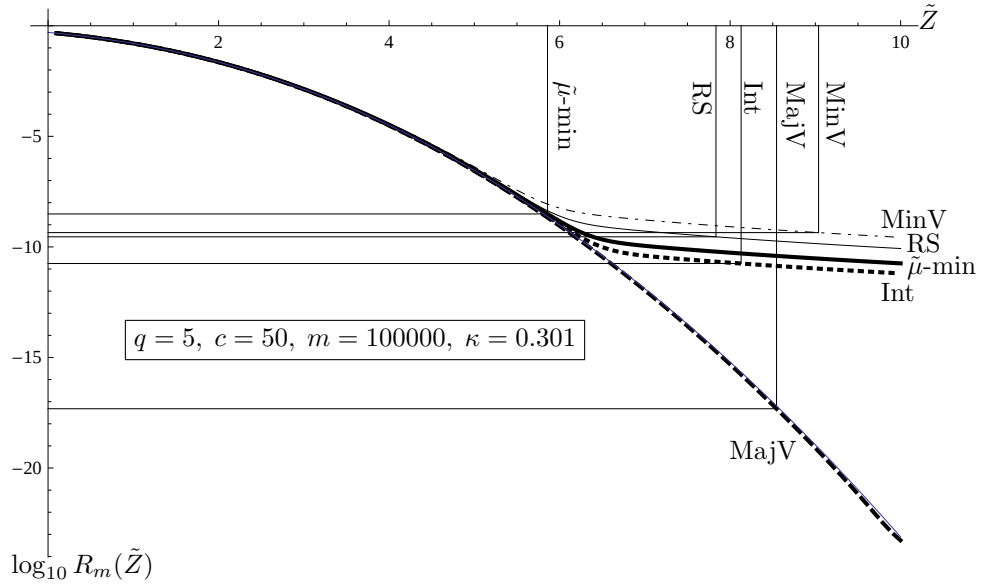

Figure 4: *FP probability as a function of the accusation threshold, for different strategies. The auxiliary lines connect each curve to its $\tilde{Z}_{\text{half}}$, allowing us to read off FP values for a fair comparison of strategies. Note that for the chosen parameter values, the $\tilde{\mu}$-min attack the $\tilde{Z}_{\text{half}}$ lies in the Gaussian part of the curve, making $\tilde{\mu}$-min the strongest attack.*

Our comparison method is illustrated in Figs. 4 and 5. At first sight, it looks as if MinV is always the strongest attack, since it causes the largest FP probability $R_m(\tilde{Z})$. However, we must not evaluate the curves at the same $\tilde{Z}$, but each at its own $\tilde{Z}_{\text{half}}$. The vertical lines connect each curve to its $\tilde{Z}_{\text{half}}$ point. The horizontal lines point to the corresponding FP probability. Comparing the FP values, we see that in Fig. 4 the $\tilde{\mu}$-min attack wins[8], while in Fig. 5 MinV wins. The c and the strategy-dependent behaviour of $\tilde{\mu}$ play a crucial role here. When the $\tilde{Z}_{\text{half}}^{\tilde{\mu}-\min}$ lies in the Gaussian

[7] This is work in progress.

[8] In fact, this is the first numerical corroboration of the statement made in [13] that the $\tilde{\mu}$-min attack is asymptotically optimal.

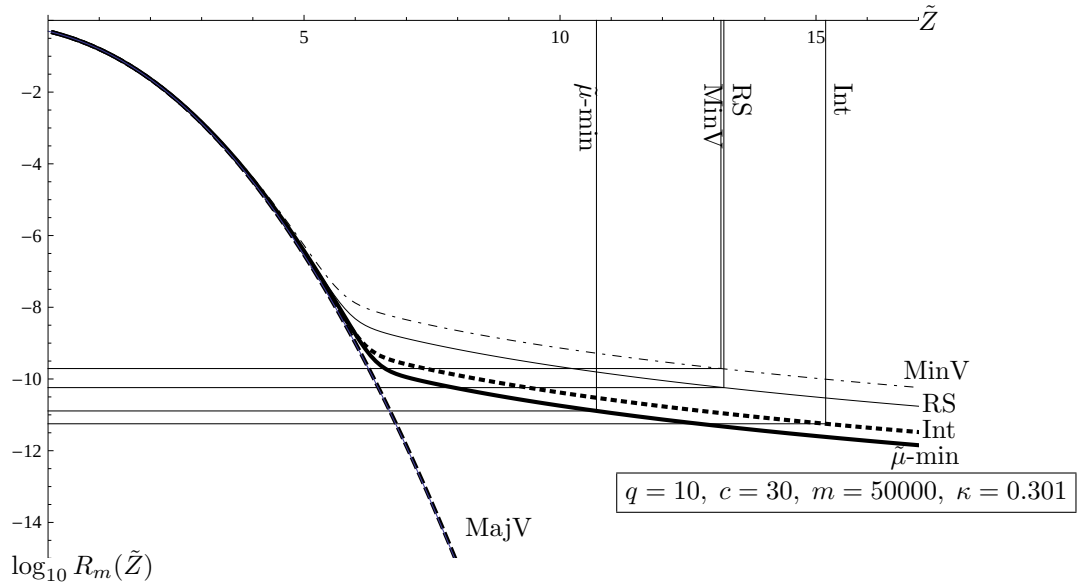

Figure 5: *Same type of plot as Fig. 4, but with different q, c and m. In this case the $\tilde{\mu}$-min attack has its $\tilde{Z}_{\text{half}}$ far outside the Gaussian part of the curve.*

part of the $\tilde{\mu}$-min curve, there can be no stronger attack than $\tilde{\mu}$-min. On the other hand, when it lies in the non-Gaussian part (which is often the case for small c) then the curves that lie higher than the $\tilde{\mu}$-min curve get a chance to yield a higher FP.

6.2 Study of the effect of c, q and m

We present plots for the dependence of the attacks on the three parameters c, q and m separately.

Varying the coalition size c

Fig. 6 shows four plots where $R_m(\tilde{Z}_{\text{half}})$ is computed as a function of c while q and m are kept fixed. Obviously, increasing c makes every attack type more powerful. (FP increases.) The $\tilde{\mu}$ strongly depends on the strategy, moderately depends on q, and weakly decreases with c. The $\tilde{Z}_{\text{half}} = \sqrt{m}\tilde{\mu}/c$ is a decreasing function of c, which means that the "read-off" point in a figure like Fig. 4 moves to the left, causing a higher FP probability. In several of the plots we see crossovers occurring, most notably between $\tilde{\mu}$-min and MinV.

Varying the alphabet size q

Fig. 7 analogously shows the dependance on q. All attacks weaken with increasing q. This is mainly caused by the fact that $\tilde{\mu}$ is an increasing function of q [17], forcing the "read-off" point in Fig. 4 to the right. We see crossovers occurring as a function of q too.

Varying the code length m

Fig. 8 shows the dependance on m. All attacks weaken with increasing m. This is due to two effects: the R_m curve becomes more Gaussian (Central Limit Theorem), and $\tilde{Z}_{\text{half}} \propto \sqrt{m}$ shifts to the right. The CLT effect differs per strategy, causing the observed crossovers.

Varying the parameter κ

Fig. 9 shows the dependance on κ. Apart from $\tilde{\mu}$-min, all the strategies have a smooth behaviour. As was explained in [13], the $\tilde{\mu}$-min strategy coincides with majV for small κ and with MinV for large κ. At intermediate κ there are jumps in the $\tilde{\mu}$-min curve, indicating a re-definition of the $\tilde{\mu}$-minimizing strategy.

For all the curves, the impact of κ on the FP rate is mostly due to the fact that $\tilde{\mu}$ depends on κ; the $\tilde{Z}_{\text{half}}$ in turn is linear in $\tilde{\mu}$.

From Fig. 9 we see that $\kappa \approx 0.3$ minimizes the coalition's effectiveness at $q = 3$ (given, of course, that they are restricted to the arsenal of strategies presented here). In general, the optimal κ choice lies close to $1/q$.

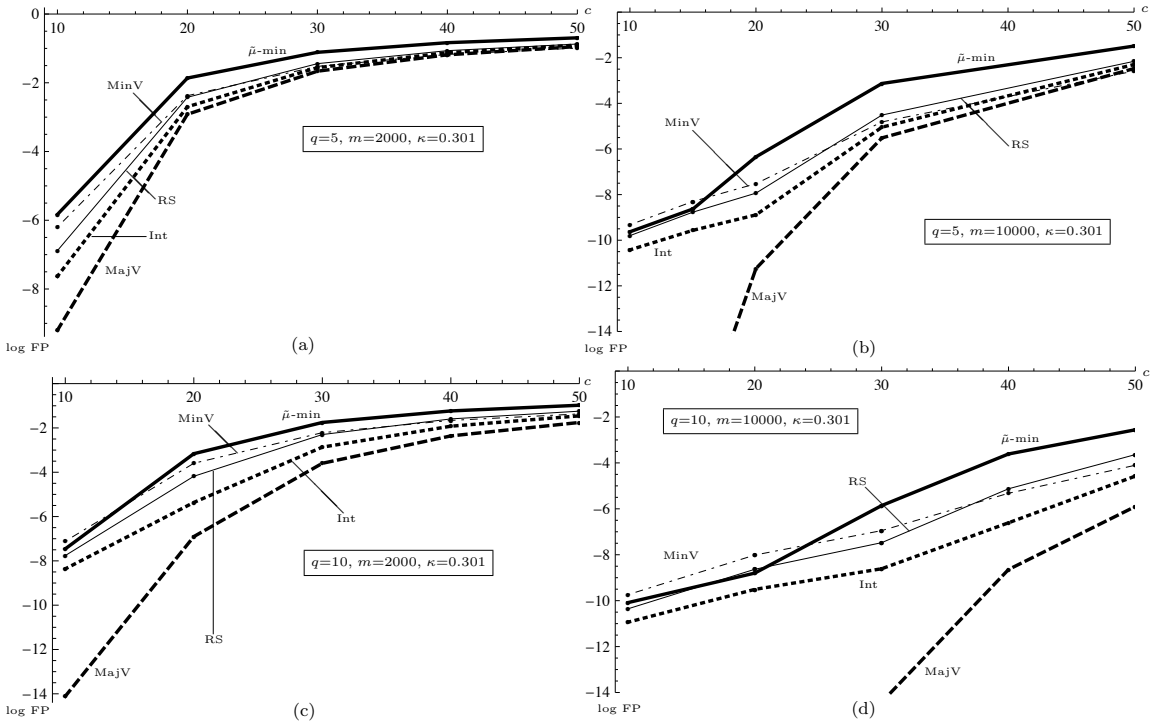

Figure 6: *FP probability $R_m(\tilde{Z}_{\text{half}})$ as a function of c for all the attack strategies. Four combinations of q and m are shown.*

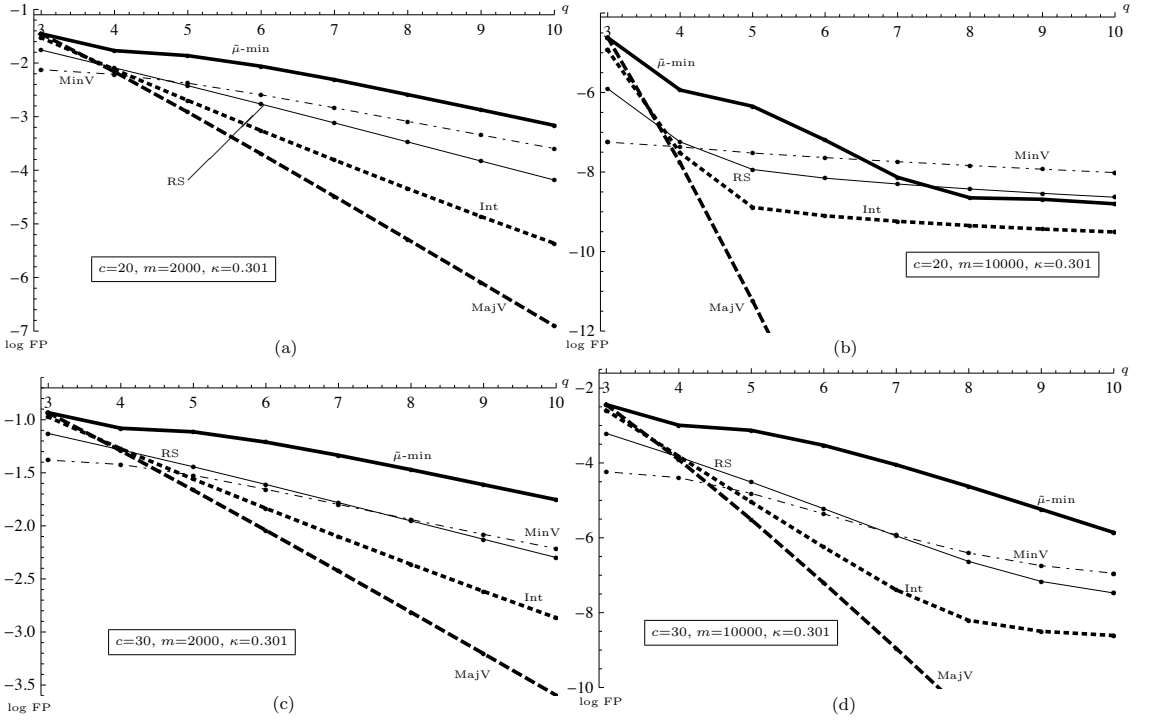

Figure 7: *FP probability $R_m(\tilde{Z}_{\text{half}})$ as a function of q for all the attack strategies. Four combinations of c and m are shown.*

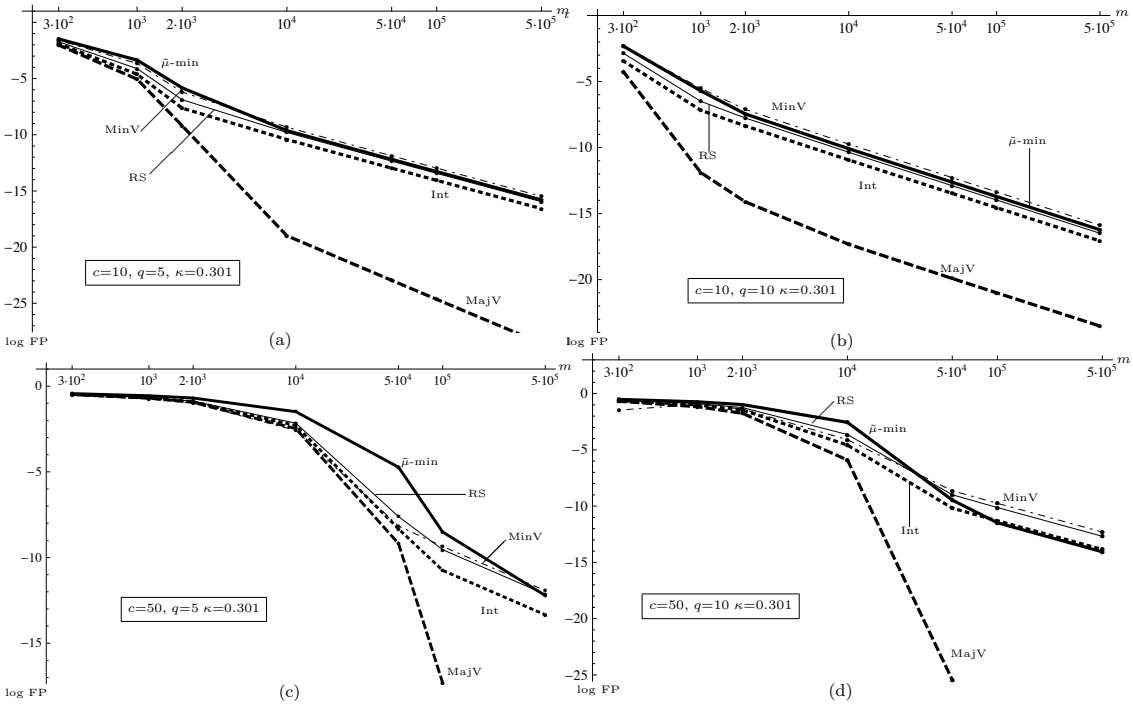

Figure 8: *FP probability $R_m(\tilde{Z}_{\text{half}})$ as a function of m for all the attack strategies. Four combinations of c and q are shown.*

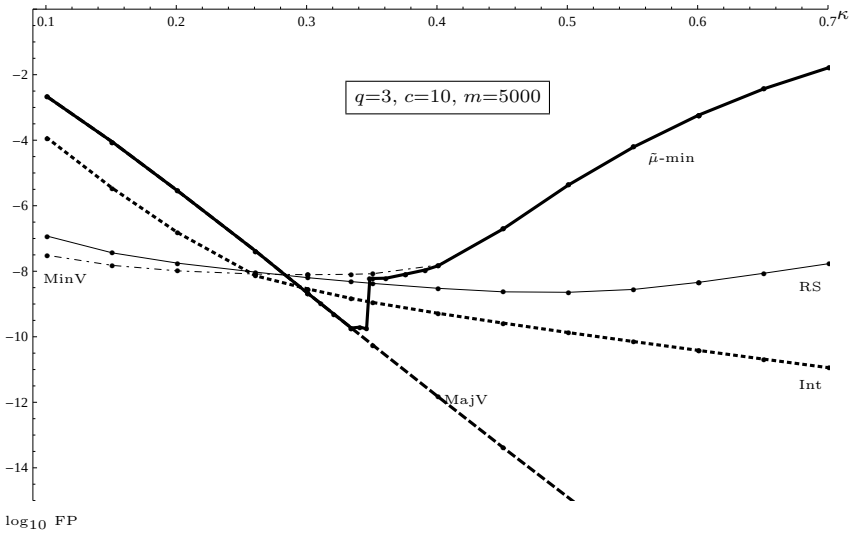

Figure 9: *FP probability* $R_m(\tilde{Z}_{\text{half}})$ *as a function of* κ.

7 Summary and future work

We have improved our understanding of the CSE method in various respects.
(i) We have empirically charted the power series cutoff values $\nu_{\max}$ that yield sufficiently accurate $R_m(\tilde{Z})$ results. For small m, the CSE method sometimes fails, especially in the case of the MajV strategy. (However, of all strategies MajV is closest to Gaussian, and we have seen that it is rarely a strong attack.) Unfortunately, we do not yet have a theoretical way to estimate $\nu_{\max}$. Perhaps this could be done if we had a better understanding of the ω parameters in (15).
(ii) We have demonstrated that the tail of the $R_m(\tilde{Z})$ curve follows the same power law as the single-segment pdf $\varphi(u)$. This is of course exactly what one expects, but it had not been explicitly verified before, and there have been doubts in the past about this issue.
(iii) We have plotted the $\varphi(u)$ curve for different strategies. While hardly a contribution, it does graphically indicate the relative strength of the attack strategies within one segment; and we have seen that these properties carry over to the FP probability $R_m(\tilde{Z})$ for the summed score.
Furthermore, we have broadened the 'scope' of the CSE method by increasing the list of strategies for which it is known how to do the K_b pre-computation. Theorem 1 shows how the K_b for RS can be computed with only $\mathcal{O}(qc^2)$ summation terms. RS does not fit in any of the classes defined in [13]. It has been necessary to introduce a new class that focuses on the number of zeros present in $\boldsymbol{\sigma}$ instead of the occurrences of the symbols.
Finally, we have compared the different attack strategies by looking at the $R_m(\tilde{Z}_{\text{half}})$ values, i.e. FP at an FN rate that is 'under control', by which we mean $P_{\text{FN}} \approx 2^{-c}$. Lacking the means to make ROC curves, this is currently the best we can do to come to a fair comparison. We are helped by the fact that a small change in $\tilde{Z}$ has a much stronger effect on FN than on FP. Application of the CSE method to FN is left for future work.
Our graphs show that the 'winner' in the strongest attack competition is nearly always MinV or $\tilde{\mu}$-min. This is heuristically understood from the fact that MinV has the longest $\varphi_+(u)$ tail and that $\tilde{\mu}$-min yields the smallest Z_{half}. The exact parameter settings determine which of these effects wins.

References

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

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

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

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

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

[6] S. He and M. Wu. Joint coding and embedding techniques for multimedia fingerprinting. *TIFS*, 1:231–248, 2006.

[7] Y.W. Huang and P. Moulin. Saddle-point solution of the fingerprinting capacity game under the marking assumption. In *ISIT 2009*.

[8] Y.W. Huang and P. Moulin. On fingerprinting capacity games for arbitrary alphabets and their asymptotics. In *IEEE International Symposium on Information Theory (ISIT 2012), Cambridge, MA, USA*, 2012.

[9] J. Kilian, F.T. Leighton, L.R. Matheson, T.G. Shamoon, R.E. Tarjan, and F. Zane. Resistance of digital watermarks to collusive attacks. In *ISIT 1998*, page 271.

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

[11] K. Nuida, M. Hagiwara, H. Watanabe, and H. Imai. Optimal probabilistic fingerprinting codes using optimal finite random variables related to numerical quadrature. *CoRR*, abs/cs/0610036, 2006.

[12] H.G. Schaathun. On error-correcting fingerprinting codes for use with watermarking. *Multimedia Systems*, 13(5-6):331–344, 2008.

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

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

[15] A. Somekh-Baruch and N. Merhav. On the capacity game of private fingerprinting systems under collusion attacks. *IEEE Trans. Inform. Theory*, 51:884–899, 2005.

[16] G. Tardos. Optimal probabilistic fingerprint codes. In *STOC 2003*, pages 116–125.

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

[18] B. Škorić, T.U. Vladimirova, M.U. Celik, and J.C. Talstra. Tardos fingerprinting is better than we thought. *IEEE Trans. on Inf. Theory*, 54(8):3663–3676, 2008.

Appendix

A Proof of Theorem 1

We first need the following Lemma:

Lemma 4

$$\sum_{w=1}^{q-1}\binom{q-1}{w}\frac{1}{w+1}\alpha^w\beta^{q-1-w}=\frac{(\alpha+\beta)^q-\beta^q}{\alpha q}-\beta^{q-1}. \tag{22}$$

Proof of Lemma 4
We define

$$A(\alpha):=\sum_{w=0}^{q-1}\binom{q-1}{w}\alpha^w\beta^{q-1-w}=(\alpha+\beta)^{q-1}. \tag{23}$$

Integrating A we have:

$$\int_0^\alpha A(\alpha')\mathrm{d}\alpha'=\sum_{w=0}^{q-1}\binom{q-1}{w}\frac{1}{w+1}\alpha^{w+1}\beta^{q-1-w}=\frac{(\alpha+\beta)^q-\beta^q}{q}. \tag{24}$$

Dividing both expressions by α and then subtracting the $w=0$ term β^{q-1}, the result (22) follows. □

Starting from the general definition of K_b (11) we have

$$K_b^{\mathrm{RS}}=\mathbb{E}_{\boldsymbol{x}|b}\Psi_b^{\mathrm{RS}}(\boldsymbol{x})=\sum_{\boldsymbol{x}}\binom{c-b}{\boldsymbol{x}}\frac{B(\kappa\mathbf{1}_{q-1}+\boldsymbol{x})}{B(\kappa\mathbf{1}_{q-1})}\Psi_b^{\mathrm{RS}}(\boldsymbol{x}). \tag{25}$$

Given that the strategy can be defined as

$$\Psi_b^{\mathrm{RS}}(\boldsymbol{x})=\frac{1}{w+1},\qquad w=|\{i:x_i>0\}| \tag{26}$$

we need to rewrite the $\boldsymbol{x}$-sum in (25) to take the w non-zero elements in $\boldsymbol{x}$ into account. We write $\boldsymbol{x}$ as a vector containing $q-1-w$ zeroes and w nonzero integers $z_1,\cdots,z_w$.

$$\sum_{\boldsymbol{x}}\{\cdots\}\to\sum_{w=1}^{q-1}\binom{q-1}{w}\sum_{\boldsymbol{z}\in\{1,\ldots,c-b\}^w}\delta_{0,c-b-\sum_{i=1}^w z_i}\{\cdots\} \tag{27}$$

$$=\sum_{w=1}^{q-1}\binom{q-1}{w}\sum_{z_1\in\{1,\ldots,c-b\}}\cdots\sum_{z_w\in\{1,\ldots,c-b\}}\delta_{0,c-b-\sum_{i=1}^w z_i}\{\cdots\} \tag{28}$$

where δ is the Kronecker delta. Next we use a sum representation of the Kronecker δ as follows:

$$\delta_{0,s}=\frac{1}{N_b}\sum_{a=0}^{N_b-1}\left(e^{i2\pi/N_b}\right)^{as} \tag{29}$$

with $s=c-b-\sum_{i=1}^w z_i$. This is a correct representation only if N_b is larger than the maximum $|s|$ that can occur. The most positive value of s is attained at $\boldsymbol{z}=0$, namely $s=c-b$. The most negative value is attained when $w=q-1$ and $z_k=c-b$ for all k, namely $s=-(c-b)(q-2)$. Hence N_b has to be larger than $(c-b)(q-2)$. Our expression for K_b now contains sums over z_k and a. We shift the a-sum completely to the left. Next we write

$$B(\kappa\mathbf{1}_{q-1}+\boldsymbol{x})=\frac{[\Gamma(\kappa)]^{q-1-w}\prod_{i=1}^w\Gamma(\kappa+z_i)}{\Gamma(c-b+\kappa[q-1])} \tag{30}$$

$$\binom{c-b}{\boldsymbol{x}}=\frac{(c-b)!}{\prod_{k=1}^w z_k!}. \tag{31}$$

All the expressions depending on the z_k variables are fully factorized; the part of the summand that contains the z_k is given by

$$\prod_{k=1}^{w}\left[\sum_{z_k=1}^{c-b}\frac{\Gamma(\kappa+z_k)}{z_k!\tau_b^{az_k}}\right]=(G_{ba})^w. \tag{32}$$

After some elementary rewriting we have

$$K_b^{\mathrm{RS}}=\frac{(c-b)!\Gamma(\kappa(q-1))}{N_b\Gamma(c-b+\kappa(q-1))}\sum_{a=0}^{N_b-1}\tau_b^{a(c-b)}\sum_{w=1}^{q-1}\frac{\binom{q-1}{w}}{w+1}\left[\frac{G_{ba}}{\Gamma(\kappa)}\right]^w. \tag{33}$$

We can go further applying Lemma 4 on the w-sum with $\alpha=\frac{G_{ba}}{\Gamma(\kappa)}$ and $\beta=1$, obtaining

$$\sum_{w=1}^{q-1}\frac{\binom{q-1}{w}}{w+1}\left[\frac{G_{ba}}{\Gamma(\kappa)}\right]^w=\frac{(G_{ba}/\Gamma(\kappa)+1)^q-1}{qG_{ba}/\Gamma(\kappa)}-1. \tag{34}$$

Substituting (34) into (33) we obtain

$$K_b^{\mathrm{RS}}=\frac{(c-b)!\Gamma(\kappa(q-1))}{N_b\Gamma(c-b+\kappa(q-1))}\left[\sum_{a=0}^{N_b-1}\tau_b^{a(c-b)}\frac{(G_{ba}/\Gamma(\kappa)+1)^q-1}{qG_{ba}/\Gamma(\kappa)}-\sum_{a=0}^{N_b-1}\tau_b^{a(c-b)}\right] \tag{35}$$

The second summation yields $\delta_{0,c-b}$ which is zero because we are looking at $b<c$. The result (18) follows. □