Error probabilities in Tardos codes

Error probabilities in Tardos codes

Antonino Simone

A catalogue record is available from the Eindhoven University of Technology Library.

ISBN: 978-90-386-3650-4

Error probabilities in Tardos codes

PROEFSCHRIFT

ter verkrijging van de graad van doctor aan de
Technische Universiteit Eindhoven, op gezag van de
rector magnificus, Prof. Dr. Ir. C.J. van Duijn, voor een
commissie aangewezen door het College voor
Promoties, in het openbaar te verdedigen
op woensdag 18 juni 2014 om 14:00 uur

door

Antonino Simone

geboren te Venetië, Italië

Dit proefschrift is goedgekeurd door de promotor:

Prof. Dr. S. Etalle

Copromotor:
Dr. B. Škorić

Acknowledgements

I feel bound to thank the many people that helped me during these years. Foremost, I want to thank my promotor, Sandro Etalle, that "back in the days" accepted me as a Ph.D. student in the SEC group. In these years he has always kept his door open for me offering his help whenever needed. I would also like to express my sincere gratitude to my supervisor Boris Škorić for the continuous support of my Ph.D. study and research, for his patience, guidance, motivation, enthusiasm, and immense knowledge. I could not have imagined having a better advisor and mentor.

I would like to thank the rest of my thesis committee: Emile Aarts, Gábor Tardos, Teddy Furon, Johan van Leeuwaarden and Berry Schoenmakers for reviewing my thesis and providing helpful comments.

I spent 5 great years in Eindhoven thanks to the many people I met during that period, in particular: Alessandro, Antonio A., Antonio C., Alberto, Bruno, Christiane, Çiçek, Costas, Daniel, Daniela, Dave, Dion, Elisa, Erik, Fred, Gaetan, Giovanni, Giulia, Henk, Irene, Jan-Jaap, Jerry, Jing, Jolande, Juan Carlos, Lida, Luca, Mark, Nicola, Patrizia, Richard, Sabine, Shona, Tanir, Vinicius. Thank you all for the great time! Also, I want to thank my poker, basket and LAN buddies: Agnieszka, Anders, Ali, Bas, Berkan, Bogdan, Erwin, Eugene, Hans, Iason, Jan-Willem, John, Kakuba, Kundan, Laurent, Lena, Maxim, Meilof, Mirella, Neda, Niels, Oleh, Oleg, Patricio, Patrick, Peter, Rob, Rostyslav, Sinatra, Sudhir, Sulva, Thomas, Upanshu, Valeriu, Volha, Yael, Yves, Zoran. Thanks to you all my playful side could stay alive. Also a big thank goes to the colleagues from SEC, CC and other groups: Anita, Benne, Dan, Fatih, José, Maxim, Milan, Peter B, Peter S, Peter vL, Relinde, Ruben, Ruud, Sebastiaan, Sokratis, Tanja, Tanya, Thijs, Wil. It was nice to share with you coffee breaks, movie nights and fruitful chats.

Un dovuto, immenso grazie va alla mia famiglia, che nonostante la lontananza ho sempre sentito vicina in questi anni. Mamma, Raffaele, lasciare casa é stata una dura decisione e non passa giorno in cui non senta la vostra mancanza. Grazie per il vostro continuo sostegno. Purtroppo questi ringraziamenti non potranno essere letti da due tra le piú care persone che ho avuto l'onore di avere al mio fianco fino a poco tempo fa. Nonna, il tuo affetto e le tue premure non saranno mai dimenticate. Papá, a te dedico questa tesi. Sei la persona che piú mi ha fatto appassionare alla scienza e che mi ha indirizzato verso questo percorso. Grazie per i sacrifici che hai fatto assieme alla mamma, per tutto quello

che mi hai insegnato e per essere stato un esempio.

Infine, il grazie piú grande va a Mayla, che con pazienza e amore continua da anni a prendersi cura di me. Il suo sostegno é stato fondamentale per raggiungere questo traguardo e mai le saró grato abbastanza per quanto ha fatto e continua a fare.

Contents

1

Introduction

1.1 Content protection

In the last decades we witnessed an unstoppable conversion from analog to digital: songs, movies, mail, money, books, TV programs and official documents are only a small set of products that we can obtain and manipulate just by using a home computer. Rather than buying music and other content types on physical carriers like CDs, nowadays it is increasingly common to buy them on-line. However, it is also extremely easy to find unauthorised free versions of these contents, a phenomenon known as *piracy*.

Since Internet is of common use, piracy has become frequent, particularly thanks to P2P (peer-to-peer) file sharing programs that allow people to easily share and find almost any kind of digital content while staying nearly anonymous. To contain this behaviour, music and video vendors have the following main options:

- *Copy prevention*, i.e. putting in place mechanisms that make it hard to produce copies. This technique, while effective against most users, is restrictive and unpractical since it often requires the usage of specific software or hardware.

- *Content tracing*, i.e. the vendor links the distributed content to the receivers by hiding a unique code, called *watermark*, inside the content. In this way, when an unauthorised copy is found, it is possible to recognise which users can be held responsible for starting the unauthorised sharing. It does not prevent a user from copying, nor does it hinder him when using the content. Instead,

it is a deterrent against piracy. Content tracing is also known as *traitor tracing*.

With the decreasing role of physical data carriers in content sales, one expects that tracing will become the predominant content protection technique for audio and video.

Online sales channels are particularly suitable for the implementation of content tracing. The one-to-one nature of the transaction and the lack of a physical data carrier make it easy for the vendor to customize the purchased data, i.e. to hide an *individual* watermark. Furthermore, vendor websites as a rule oblige buyers to provide personal data, which makes it possible to physically locate them.

Content tracing finds a natural place in *pay-TV* too. A pay-TV subscriber receives a device that decrypts only those TV channels that he has paid for. Such a device carries keys that are unique for each subscriber. By cleverly organizing the encrypted broadcast, the pay-TV operator can, to a certain extent, hand out different versions of the video stream to different devices, and thus make sure that different subscribers receive differently watermarked versions of the video.

1.2 Hidden watermarks

For the purpose of this thesis, a watermark is a sequence of small modifications that the vendor applies to a multimedia file in a secret way. The idea is that the vendor should be able to efficiently identify the watermark he inserted into the file, while the user should not notice anything out of the ordinary about the file he receives. The action of inserting a watermark into the original file is called *embedding*. A version of the content carrying a watermark is called a *watermarked copy*.

The embedding must satisfy two important requirements:

1. The watermark must be well hidden. In particular, the user should not be able to find the locations where changes were made. If the user is able to locate enough of the watermark, he may corrupt it, making it impossible to link the content to the user.

2. The presence of the watermark should not compromise the quality of the content. Defects in the multimedia may discourage users from purchasing the product.

Obviously, it is not efficient for the vendor to keep a full record of all the differently watermarked versions of the same content. In practice the data he stores consists merely of a compact description of the modifications that were made in the distributed versions. In audio/video, the number of different ways in which a file *position*[1] can be (robustly) watermarked is limited. In mathematical terms, these limited options can be represented as symbols from a small alphabet, and each user's watermark as a unique sequence of symbols.

Then, the vendor needs to store:

- For each user, the sequence of symbols present in his version.
- For each position, the differently watermarked versions of this position; or, alternatively, a short description how the watermark symbols were embedded. The details of how each symbol is embedded differ per position.

Obviously all this data is kept secret. When the vendor finds an unauthorized copy, he investigates it using a *watermark detector*. The detector takes as input the stored information as detailed above, the original (non-watermarked) file, and the unauthorized copy. For each position, the detector tries to determine which watermark symbol, if any, is present. This results in a sequence of detected symbols. The vendor compares this sequence to the stored user sequences, and in this way identifies which user illegitimately redistributed his copy.

1.3 Attacks on watermarks

Malicious users are interested in locating the watermark in their files, in order to alter it and thus destroy the link between the file and the user. Such an attempt to make the watermark unusable is called an *attack*.

Attacks on hidden watermarks can be grouped in two categories: attacks based on a single copy, and attacks based on multiple copies, also known as *collusion attacks* (or coalition attacks).

Today's watermarking techniques are particularly robust against single-copy attacks. One approach for the attacker is to insert noise, in the hope

[1] The description of an audio/video file as a sequence of positions is somewhat abstract. A 'position' in video typically refers to a complex combination of many screen pixels, spread out in time.

that the watermark gets damaged. However, the effectiveness of this approach is very limited. The noise must not degrade the perceptual quality of the audio/video. Furthermore, not knowing the secret embedding information as described in Section 1.2 the attacker cannot 'shape' the noise so as to target the watermark.

Another approach is the use of statistical techniques that lead to partial disclosure of a watermark. Such techniques, however, have a limited impact. If the requirements listed in Section 1.2 are satisfied, it is difficult for a single attacker to locate the watermark.

The situation changes when a collusion attack takes place. The colluders can compare each other's files, which immediately reveals differences in those locations where they did not all receive the same symbol. This additional information allows for a much stronger attack on the watermark.

1.4 Collusion resistant codes

The only defense against collusion attacks is to make sure that there are enough positions in the content where the colluders receive insufficient information to damage the watermark. In other words, enough redundancy must be built into the user sequences. This is achieved by using a *collusion resistant code* (*traitor tracing* code).

In the sequel, a sequence of watermark symbols can be considered as a *codeword* that identifies a recipient, while the set of all the codewords is called the *code*. The aim of collusion resistant codes consists in creating codewords such that, no matter how strong the collusion attack is, the resulting media still contains enough information to identify at least one of the colluders.

Figure 1.1 illustrates the elements in a traitor tracing scenario, from the content distribution to the tracing. After the watermark detection step, described in Section 1.2, the vendor uses a *decoder* to identify the attackers. In analogy with error-correcting codes, the decoder is an algorithm that tries to determine which codewords are closest to the sequence of detected symbols. The output of the decoder is a list of suspicious users. In making this list, there exist two important type of errors:

- *False positive* (FP) error: the decoder outputs one or more innocent users. Accusing an innocent user can be particularly damaging for the vendor because of potential (legal) repercussions. Furthermore,

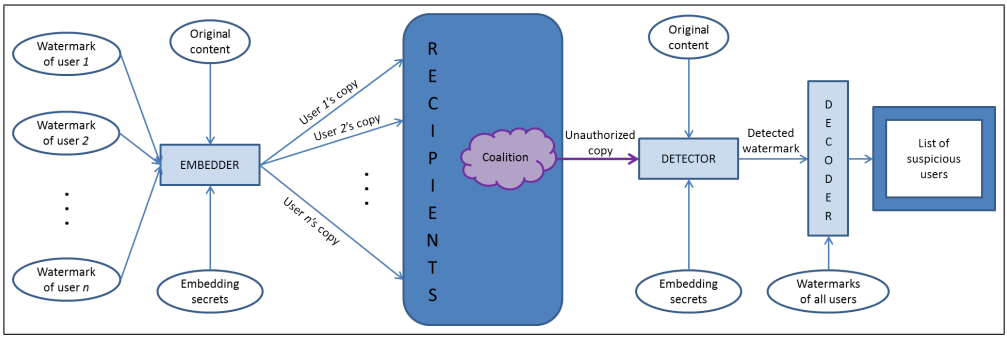


FIGURE 1.1: *Traitor tracing problem scenario. The watermarks are embedded into the original content. The personalised copies are distributed to the recipients. Some of them (a coalition) develop an unauthorised copy and distribute it. Once the vendor obtains this copy, the detector tries to extract the watermark symbols. The decoder makes a list of suspect users.*

having frequent FP errors may cause: (i) loss of credibility of the tracing scheme and (ii) waste of the vendor's time and resources in the follow-up accusation steps.

- *False negative* (FN) error: the list does not contain any colluder.[2] This type of error is considered less damaging than the FP (not finding any attacker causes less trouble than accusing an innocent). However, the tracing scheme obviously becomes useless if the attackers can evade capture too easily.

There is a rich literature on traitor tracing codes. The most studied class of traitor tracing codes in the past decade are the *Tardos codes* (or Tardos scheme, or bias-based codes), introduced in [38] and extended in [39]. These have the special and highly desirable property that they achieve optimal performance against large coalitions: the code length required to resist a collusion attack scales as the square of the coalition size. It has been proven that a better scaling is not possible [38]. A detailed description of the Tardos code is presented in Chapter 3.

[2]Usually the vendor's aim is to catch at least one attacker. An alternative aim could be to catch all colluders. This aim is usually not considered because it is too difficult to achieve when the colluders do not participate equally in the attack.

1.5 Problem description

Despite the popularity of the Tardos code, its performance was not fully understood prior to the work in this thesis. On the one hand, one can use simulations to determine the error probabilities. However, simulations for small FP probability take an infeasible amount of time [17, 7]: the required number of simulations scales as the inverse of the FP probability[3], which can be as small as 10^{-6}, or even smaller. On the other hand, there exist proven bounds on the error probabilities [38, 43, 42, 40]. Unfortunately, these bounds are not tight and generally are orders of magnitude too high.[4]

The knowledge of FP and FN probabilities is critical because the amount of 'space' available for embedding watermarks is very limited and has to be optimally exploited. According to some estimates, less than 1 byte of hidden watermark can be robustly embedded per minute of video (i.e. less than 960 bits in a two-hour movie). When the error probabilities are not known, the following problems can occur:

- The estimated errors are *higher* than the real ones.
  The vendor adopts a code that has too much redundancy, resulting in a waste of already scarce resources. The amount of redundancy that must be built in grows relatively weakly as a function of the FP rate (see Section 1.4), but even so an FP overestimate of orders of magnitude, which is typical when there is no simulation data, has a large impact on the coalition size that can be resisted.

- The estimated errors are *lower* than the real ones.
  The strength of the coalition has been underestimated and this gives more untraced colluders (FN) and/or more accusations of innocent users (FP) than anticipated.

Hence, the lack of knowledge about the actual performance of the code leads to a situation where the vendor either does not have a tracing capability or is faced with too many FP events.

[3] Typically the FP probability is fixed to be orders of magnitude lower than the FN. For this reason the FP determines how many simulations are needed.

[4] For example, simulations in [7] show that a codeword of 600 bits is enough for the Tardos scheme to resist 4 attackers and to provide FP and FN probabilities both of $2.5 \cdot 10^{-3}$. For the same setting, the upper bound in [2] guarantees that the error probabilities are below 0.47, far from the real value.

1.6 Research question

The above problems motivate the main research question of this thesis:

How to determine the real error rates of the Tardos code?

The aim of our research is to develop a method, preferably analytical, to efficiently compute the error probabilities at any parameter setting (codelength, alphabet size, coalition size, ...). The method has to be faster than the simulations and more accurate than the provable bounds; both by whole orders of magnitude. Knowing the precise error probabilities makes it possible to optimally exploit the available space for the watermark. The ability to precisely control the FP rate makes the crucial difference between a scheme that is unfit for use and an effective tracing system that can be used in practice.

A second research aim is to investigate the various attacks that a coalition can perform, in order to understand which attack works best against the Tardos code. In each content position, the colluders have a set of received symbols, and based on this set they have to decide what kind of content version to create in that position. The way in which they make their decision is called a *strategy*.[5] The aim is to compute error rates for a wide range of parameter settings and attack strategies; then (i) find the 'worst case' attack, i.e. which strategy maximises the error rates at given system parameters and (ii) conversely, which parameters settings provide resistance against all the tested attacks. This investigation should make clear what both the vendor and the colluders can hope to achieve.

1.7 Contributions

In this thesis we investigate the Tardos code, focusing on the FP and FN error probabilities. Our main results are the following:

- We develop a new procedure, which we call the 'CSE method' (Convolution and Series Expansion), to semi-analytically compute the FP and FN probabilities for the Tardos code. The method is based

[5] Some examples of strategies are majority voting (selecting the most frequent symbol), minority voting (selecting the least frequent symbol) and interleaving (taking the symbol of a random attacker).

on the **convolution** property of characteristic functions (see Chapter 4). It gives a recipe for calculating the error rates as a **series expansion** in a small parameter. The small parameter decreases as a function of the code length; hence for long codes the expansion converges more quickly than for short codes.

- The CSE method passes a number of consistency checks. First of all, it is consistent with simulations when simulations are doable. Furthermore, the tails of certain important probability distributions behave exactly according to theory.

- The expansion in the CSE needs a certain number of terms in order to achieve sufficient precision. For various attack strategies and parameter settings we tabulate the required number of terms.

- We introduce a new parametrization of attack strategies. This leads to two benefits: (i) more compact formulas and a better understanding, (ii) the possibility to do pre-computations, which speeds up the CSE method. Computing the error probability for a specific setting then takes from few seconds to several minutes on an average laptop.

- Using the CSE method we study the performance of the Tardos code against a variety of attack strategies, for a wide range of system parameters. This allows us to identify which of the strategies is the strongest given the system parameters. The data are combined in ROC (receiver operating characteristic) curves, to show the impact of the studied strategies on both the error probabilities at the same time. Our study shows that there are two distinct regions in the parameter space. In one region (short code and/or small coalition), the most powerful attack strategy is the well known minority voting attack. In the other region (long code and/or large coalition), it is a more complicated strategy that is tailored specifically against the decoder algorithm of the Tardos code.

The thesis is based on the publications [33], [32], [34] and [35].

2

Background on traitor tracing codes

In Chapter 1 we illustrated the traitor tracing problem showing the use of watermarks and their practical limitations. In this chapter we provide a more technical background. We focus on the coding, in line with the topic of this thesis. We will assume that the watermark embedding is properly done, so that the space in the content available for watermarking is fully exploited, and there is no effective single copy attack.

2.1 Concept and notation

In this section we introduce the notation we will use in this thesis and the basis concepts of the collusion resistant codes.

2.1.1 The marking assumption

The collusion attack introduced in Section 1.3 is a category of attacks in which several versions of the same content are used. The colluders, through the comparison of their copies, locate the segments that contain different symbols. These segments are called *detectable positions*.

The *marking assumption* (or marking condition) [4] states that an attack can take place only in detectable position. See Figure 2.1. Of course, as was discussed in Section 1.3, the marking assumption does not strictly hold. However, it is a good starting point for the analysis of traitor tracing codes and as such it is often adopted as a working

hypothesis. In the detectable positions, the attackers can change the watermark.

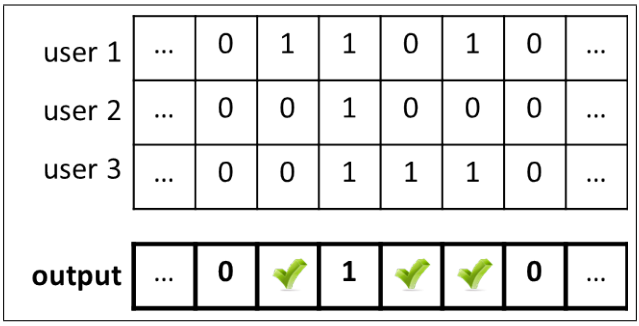


FIGURE 2.1: *Illustration of the marking assumption in the case of a binary alphabet. Three attackers compare their copies. The check-mark sign indicates the detectable positions. The other positions are kept unchanged because of the marking assumption.*

In the next section we introduce the models that are most commonly used to describe the coalition capabilities.

2.1.2 Attack models

Below we list often considered attack models. All of them except the CDM obey the Marking Assumption.

Restricted Digit Model: the output symbol for a position can just be one owned by at least one attacker in that position.[1]

Unreadable Digit Model: this model allows slightly stronger attacks than the Restricted Digit Model. The output for each detectable position can be either a symbol owned by at least one attacker in that position or an erasure (removal of the watermark).

Arbitrary Digit Model: the output symbol for a detectable position can be any of the symbols in the alphabet (but not the erasure).

General Digit Model: the output in a detectable position can be either a symbol in the alphabet or the erasure.

[1]Notice that this model does not really need to distinguish between detectable and undetectable positions. It implicitly follows the marking assumption.

Combined Digit Model: the output in a detectable position can be the fusion of any subset of the received symbols [41, 44]. The scenario allows signal processing and averaging attacks, typical in the spread-spectrum watermarking context. This symbol merging complicates the decoder step because the result of the fusion does not necessarily point to a specific alphabet symbol. The effect of merging many symbols can cause trouble to the detector yielding a probability that an erasure occurs. Furthermore the colluders are allowed to add noise, even in undetectable positions, which causes a small probability that a symbol is detected which is not part of the set received by the colluders. A variant of the binary Combined Digit Model is given by Kuribayashi in [16].

Example 1. Suppose that a 4-user coalition receives the symbols 0, 0, 1 and 2 and the alphabet is $\{0, 1, 2, 3\}$. Depending on the model, the generated output is:

- Restricted Digit Model: Any element of the set $\{0, 1, 2\}$ (Figure 2.2, second column).
- Unreadable Digit Model: Any element of the set $\{0, 1, 2, ?\}$, where ? denotes an erasure.
- Arbitrary Digit Model: Any element of the alphabet $\{0, 1, 2, 3\}$.
- General Digit Model: Any element of the set $\{0, 1, 2, 3, ?\}$.
- Combined Digit Model: The output is obtained from the symbol fusion of any non-empty subset of $\{0, 1, 2\}$.

Among all the models, the Combined Digit Model is the most realistic but also the hardest to analyze. The Arbitrary Digit Model and General Digit Model are not realistic due to the possibility of outputting any symbol, which requires information that is not available to the colluders. The Unreadable Digit Model, with the erasure possibility, gives to the attackers too much power. The model we are going to consider in the thesis will be the Restricted Digit Model (RDM). Even if it is the simplest among the ones listed, it is already a very complicated model and its full study is still not complete yet. Furthermore, the study of the RDM is a useful starting point for the analysis of more complicated models.

user 1	...	0	1	1	3	3	0	...
user 2	...	0	1	1	0	0	0	...
user 3	...	1	1	1	1	1	0	...
user 4	...	2	3	1	2	1	0	...
possible output	**...**	**0 1 2**	**1 3**	**1**	**0 1 2 3**	**0 1 3**	**0**	**...**

FIGURE 2.2: *Example of allowed output symbols in the Restricted Digit Model.*

2.1.3 Accusation scenarios

Once the distributor finds an unauthorised copy, the detecting and tracing phase begin. In the literature one finds two scenarios regarding the content vendor's aim:

- *catch all*: the distributor wants to trace all the attackers in the coalition. This is impossible to achieve if some colluders do not participate in the attack (or participate very little)

- *catch one*: the distributor is satisfied tracing at least one attacker. This is in many cases already a good deterrent.

We use the notation $\mathcal{C}$ to indicate the set of colluders and $\mathcal{L}$ the set of suspicious users. As mentioned before, in general the FP error is more grave than the FN, so it is critical to avoid innocent accusations. However, there are some contexts in which the FP error is not too damaging and its sporadic occurrence is acceptable. For example, in the pay TV scenario one can apply *dynamic traitor tracing*, where the distributor has real-time information about the attack and the watermarks can be generated dynamically. In this case, the consequence of a FP can consist of a *temporary* deactivation of a innocent user. An example of binary dynamic traitor tracing scheme is the one developed by Laarhoven et

al. [20]. In this thesis only the static scenario is considered, with the "catch one" aim.

We can start now to define some parameters.

m	Codelength.
n	Number of users.
$\mathcal{Q}$	Alphabet.
q	Alphabet size $\lvert\mathcal{Q}\rvert$.
$\mathcal{C}$	Set of colluding users.
c	Number of colluders $\lvert\mathcal{C}\rvert$.
c_0	Maximum number of colluders the scheme can resist.
$\boldsymbol{X}$	$n \times m$ matrix. The element X_{ji} indicates the symbol received by the user j in position i, while with X_j we indicate the entire codeword received by user j.
$\boldsymbol{X}_{\mathcal{C}}$	$c \times m$ matrix containing the codewords received by the c colluders.
$\boldsymbol{y}$	Sequence generated by the coalition $\mathcal{C}$ and detected by the distributor. With y_i we indicate the symbol in position i.
$\mathcal{L}$	List of accused users.

To develop an efficient scheme for the RDM is not trivial. There are many requirements to satisfy to obtain an usable scheme:

- short codelength
- resist c_0 colluders
- very low probability of FP error
- low probability of FN error
- small alphabet. Typically $q \leq 16$ for audio/video.

We give an short overview about the literature on the traitor tracing problem.

2.2 Traitor tracing evolution

The traitor tracing field has evolved in many directions. In this section we want to briefly describe the most famous results.

2.2.1 Deterministic and probabilistic approaches

A deterministic approach produces a list of accused users that is never empty and never contains innocent users, avoiding both the FP and FN errors. Hollmann et al. in 1998 [10] introduced Identifiable Parent Property (IPP) codes. These codes have the drawback of failing when the number of colluders is more then two. In 2001 Staddon et al. [36] proved the existence and provided the construction of a deterministic code with a codelength $m = c_0^2 \log_q n$, but the alphabet size needs to be $q \geq m - 1$.

It is often the case that an algorithm can be made more efficient by allowing a small probability of failure. In 1995 Boneh and Shaw [4] introduced a binary scheme ($q = 2$) using a partly randomized inner code with a deterministic outer code. The scheme needs a codelength $m = \mathcal{O}(c_0^4 \log \frac{n}{\eta} \log \frac{1}{\eta})$, where η indicates the probability to have a FP error. In the same work, they also provided a lower bound on the codelength, $m = \Omega(c_0 \log \frac{1}{c_0 \eta})$. In 2003 the lower bound became more precise thanks to Peikert et al. [30]: $m = \Omega(c_0^2 \log \frac{1}{c_0 \eta})$.

In the same year, Gábor Tardos [38] proved an even tighter bound of $m = \Omega(c_0^2 \log \frac{1}{\varepsilon_1})$, where ε_1 denotes the probability that one specific innocent user gets accused[2] and he gave a fully randomized binary code that achieves that bound, $m = 100 c_0^2 \lceil \ln \frac{1}{\varepsilon_1} \rceil$. This result represents an important turning point in the traitor tracing field and, as expected, it has been studied a lot from many points of view, to provide extensions, generalizations and improvements. Tardos' approach first generates a bias for each segment and then in each segment randomly draw a symbol for each user according to the bias. In the accusation step a score is computed for each user that consists of summing one-segment scores. The users whose score is higher than a threshold (decided a priori) are considered suspicious. There are a lot of works on all the separate components of the scheme. Among them, in 2008 Škorić et al. [42] developed a q-ary version of Tardos' original code. This q-ary generalisation is the scheme we are going to study. It will be fully detailed in Chapter 3.

2.2.2 Channel capacity and code rate

Collusion-resistant coding has been analyzed also from an information-theoretical point of view. The whole procedure that goes from the wa-

[2] For Tardos codes, ε_1 and η are related in the following way [40]: $1-\eta \approx (1-\varepsilon_1)^{n-c}$. Using $c \ll n$ and $\varepsilon_1 \ll 1/n$ this yields $\eta \approx n\varepsilon_1$.

termark matrix to the watermark symbols detected in the unauthorised copy can be seen as a communication channel: the colluders' codewords $\boldsymbol{X}_{\mathcal{C}}$ are the inputs of such a channel, and the output is $\boldsymbol{y}$. The applied attack is considered as channel noise. The knowledge about the channel capacity is important because it gives a lower bound for the codelength. As defined by Moulin and O'Sullivan in [25], the fingerprinting rate R is

$$R = \frac{\log_q n}{m}. \tag{2.1}$$

Suppose we have to specify one out of n users using m-segment codewords created with q-ary symbols. The numerator $\log_q n$ represents the number of q-ary symbols needed to specify one out of n users. This is divided by the number of symbols that is actually used. The rate R is the useful fraction of the codeword, i.e. the fraction that conveys the message about the identities of the colluders.

Thanks to the Shannon's channel coding theorem [21] we know that R must not exceed the channel capacity C to have a reliable data transmission.

The channel coding theorem also gives the asymptotic relation between error probability and codelength.

$$P_{\text{Err}} \leq q^{-(C-R)m}, \tag{2.2}$$

showing that for longer codes the error probability decreases provided that $R < C$. From (2.1) and (2.2) we have

$$\begin{aligned} P_{\text{Err}} &\leq q^{-(C-\frac{\log_q n}{m})m} && (2.3)\\ &= nq^{-Cm}. && (2.4) \end{aligned}$$

Taking the equality and isolating m, we get the sufficient code length m_{suff}

$$m_{\text{suff}} = -\frac{\log_q \frac{P_{\text{Err}}}{n}}{C} = \frac{\ln \frac{n}{P_{\text{Err}}}}{C \ln q}. \tag{2.5}$$

In the channel coding theorem an error means that the message is wrongly decoded. In the traitor tracing context this corresponds to a false accusation. Then it is possible to write (2.5) as

$$m_{\text{suff}} = \frac{\ln \frac{1}{\varepsilon_1}}{C \ln q}, \tag{2.6}$$

In [11] Huang and Moulin conjectured that asymptotically in the limit of large c the binary channel capacity is $1/(2c^2 \ln 2)$. The conjecture was then proved in two works independently: in [12] by Huang and Moulin, and in [1], where Amiri and Tardos also gave a binary capacity-achieving scheme. A more general result has been found by Boesten and Škorić in [3] for general q: the asymptotic fingerprinting capacity in the RDM is $C = (q-1)/(2c^2 \ln q)$. It was shown in [40] that the channel capacity cannot be reached by the Tardos scheme due to the applied accusation process.

2.2.3 Simple and joint decoders

Tardos' seminal paper and most of the later work follow the so-called *simple decoder* approach, i.e. a 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. The aforementioned Amiri and Tardos paper [1] introduced a capacity-achieving joint decoder construction for the binary code. However, the construction is impractical, requiring computations for many candidate coalitions which takes an amount of time proportional to n^c. Charpentier et al. in 2009 [5] presented the EM (Expectation Maximization) algorithm that tries to estimate both the coalition size and the applied attack, and then adapts the score system to the estimate. Nuida in 2010 [26] proposed a joint decoder against $c \leq 3$ attackers and at most some hundreds of users. A binary joint decoder was proposed in 2011 by Meerwald and Furon [22]. Their algorithm, called Don Quixote, can potentially look for coalition sets of size t building them from the $(t-1)$-sets. It begins with a simple decoder approach (equivalent to $t = 1$ sets) that gives a first ranking of the users. After that, the bigger sets are built taking just the users highest in the ranking. This pruning phase depends on the set size t and the computational power available. In their simulations they show how their joint decoder can find more efficiently set of colluders if compared with the simple decoder.

Even if more practical joint decoders are found, a simple decoder typically serves as a stepping stone in their operation. Thus, interest in simple decoders remains high. Furthermore, it was shown in [13] that the rate achievable by simple decoders asymptotically ($c \to \infty$) equals that for joint decoders. Finally, Oosterwijk et al. [29] presented a simple decoder that achieves capacity in the large c limit. This important result

took place after the work done in this thesis.

2.2.4 Alternative bias distributions

The binary Tardos scheme [38] uses the so-called arcsine distribution (see Section 3.1) to draw random biases. The q-ary generalisation [42] uses a Dirichlet distribution which reduces to the arsine distribution at $q = 2$. The bias distribution is one of the scheme's parameters that can be tweaked, and as such, alternatives have of course been studied. We briefly list some alternatives present in the literature.

In 2007 Nuida et al. [27] introduced a discrete bias distribution that depends on c_0. The accusation step is almost identical to Tardos' one. The modified bias distribution improves the tracing if $c \leq c_0$, but it has worse properties at $c > c_0$. This discrete procedure has been improved by the same team in [28]. Huang and Moulin [11] presented a discrete distribution that maximizes an information-theoretic figure of merit at given c_0. Finally, Laarhoven and de Weger in 2013 [19] have proven that asymptotically Nuida's discrete distribution converges to the arcsin distribution.

In this thesis we will keep with the Dirichlet distribution to avoid problems at $c > c_0$.

2.3 Group Testing

An alternative usage of the binary traitor tracing problem has been introduced by Stinson et al. in [37], where they link it to the nonadaptive group testing problem. In few words, group testing consists of detecting a small set of infected people in a large population. Individual blood tests are supposed to be too expensive to be done over the entire population, so, instead, blood mixtures are taken: the blood samples of several users are mixed together and the test is applied on this mix. A positive result indicates that at least one user is infected. The term "nonadaptive" indicates that the choice of users to mix does not depend on the outcome of earlier tests. The blood test corresponds to the all-1 strategy being applied by a coalition (outputting a 1 whenever possible).

The study made in [37] approached the group testing problem with the Boneh-Shaw scheme, being at that moment the most recent result on traitor tracing and having also the important property of being *frame-*

proof, i.e. having zero FP probability below a fixed coalition size. Another important contribution was made by Meerwald and Furon [23] where a joint decoder variation is used that resembles the Don Quixote algorithm. Kitagawa et al. [14], unlike the previous two approaches, use a group testing algorithm to solve the traitor tracing problem. The approach presents a deterministic joint decoder that achieves very high performance, but the whole work focuses just on coalitions of size 3 and on three particular strategies. Already with so few attackers, the required codelength is quite big, as expected with a deterministic approach.

The contributions in this thesis may have some impact on nonadaptive group testing by providing a method to determine low error probabilities.

3

The q-ary Tardos code

In this chapter we describe the q-ary generalization [42] of the Tardos code. It is "symbol symmetric", i.e. invariant under permutation of the alphabet, a property that was missing in Tardos' original scheme [38].

3.1 Code generation and embedding

The distributor generates a $n \times m$ matrix $\boldsymbol{X}$ filled with the q symbols present in $\mathcal{Q}$ ($q \geq 2$). m vectors $\boldsymbol{p}^{(i)} \in (0,1)^q, \sum_{\alpha \in \mathcal{Q}} p_\alpha = 1$ are independently drawn ($i \in [m]$) according to the Dirichlet distribution F with

$$F(\boldsymbol{p}) = \frac{1}{B(\kappa \mathbf{1}_q)} \prod_{\alpha \in \mathcal{Q}} p_\alpha^{-1+\kappa}, \tag{3.1}$$

where

- $\mathbf{1}_q$ stands for the vector $(1, \cdots, 1)$ of length q,
- κ is a positive constant called "concentration parameter" that determines the steepness of F,
- B is the generalized Beta function defined as follows:

Definition 3.1 (Generalized Beta function). *Let $\boldsymbol{v}$ be a n-component vector. The Beta function is defined as*

$$B(\boldsymbol{v}) := \frac{\prod_{a=1}^{n} \Gamma(v_a)}{\Gamma(\sum_{b=1}^{n} v_b)}. \tag{3.2}$$

For parameters $v_1, \cdots, v_n > 0$ the Beta function has the following Dirichlet integral representation:

$$B(\boldsymbol{v}) = \int_0^1 \mathrm{d}^n \boldsymbol{x}\ \delta\left(1 - \sum_{a=1}^{n} x_a\right) \prod_{b=1}^{n} x_b^{-1+v_b}. \tag{3.3}$$

where $\delta(\cdot)$ is the Dirac delta function.

At $q = 2$ and $\kappa = 1/2$ the scheme has a problem due to the presence of outliers that generate huge scores[1]. For this specific parameter choice (3.1) coincides with the arcsine distribution used originally by Tardos in [38]. The solution found by Tardos was to introduce a small parameter t to modify the range of possible values for p from $[0, 1]$ to $[t, 1-t]$, giving the following function:

$$F_{\text{Tardos}}(p) = \frac{1}{\pi - 4\arcsin\sqrt{t}} \frac{1}{\sqrt{p(1-p)}} \tag{3.4}$$

Tardos set $t = 1/300c_0$ for proof-technical reasons. Laarhoven and de Weger in [18], studying how to achieve shorter codelengths[2], showed that $t \propto c_0^{-4/3}$ is a better choice.

Once the biases $\boldsymbol{p}^{(i)}$ have been generated, all matrix elements X_{ji} are drawn independently according to the following distribution,

$$\Pr[X_{ji} = \alpha | \boldsymbol{p}^{(i)}] = p_\alpha^{(i)}. \tag{3.5}$$

Notice that the probabilities do not depend on the row index j, i.e. $\boldsymbol{p}^{(i)}$ determines the probabilities for a whole column of $\boldsymbol{X}$. Finally the codeword X_j is embedded in user j's content. The distribution of $\boldsymbol{X}$, non conditioned on $\boldsymbol{p}$, is known as the Polya distribution.

3.2 Attack

We work with the RDM as specified in Section 2.1.2. It means that, for each segment, the coalition can output a symbol only if at least one of the attackers has received it in that segment. We define vectors $\boldsymbol{\sigma}^{(i)} \in \{0, 1, \ldots, c\}^q$ as

$$\sigma_\alpha^{(i)} := |\{j \in \mathcal{C} : X_{ji} = \alpha\}| \tag{3.6}$$

[1] The scores will be introduced in Section 3.3.

[2] The scheme studied in [18] is a symbol symmetric version of the original Tardos scheme.

i.e. the number of occurrences (or tally) of the symbol α that the attackers see in column i. Obviously $\sum_{\alpha\in\mathcal{Q}} \sigma_\alpha^{(i)} = c$. For given q and c, we define the set of possible $\boldsymbol{\sigma}$ values as $\mathcal{S}_{qc} = \left\{\boldsymbol{\sigma} \in \{0,\ldots,c\}^q | \sum_{\alpha\in\mathcal{Q}} \sigma_\alpha = c\right\}$. We also define $\boldsymbol{\sigma}_{\setminus\alpha}$ as the tally vector $\boldsymbol{\sigma}$ without the element σ_α.

The attackers have a (probabilistic) strategy for choosing their output symbols. It is assumed that this strategy is fully column-symmetric, symbol-symmetric and attacker-symmetric. In other words, as listed in [40], we consider valid the following assumptions:

1. The strategy is invariant under alphabet permutation.

2. The strategy is invariant under attackers identity, i.e. the attackers equally share the risk.

3. The strategy is applied independently for each segment.

The first assumption is justified by the Dirichlet generating function (3.1), where the biases are drawn independently from the symbol, making each of them equally important. Furthermore, in many embedding schemes the embedded symbols are some random sequence that have not a natural ordering sequence. Moreover, the tracer can permute the alphabet at will in every segment independently and it will have zero effect. The second assumption is needed to balance the role of the attackers: considering that each colluder is sharing his copy, there should not be any kind of hierarchy in the guilty set. Finally, the third assumption is due to the independency among the $\boldsymbol{p}^{(i)}$ biases, being these m vectors drawn independently from (3.1). In conclusion, the symmetry present in the Tardos scheme motivates to build one-segment attacks. Furthermore, in [24] Moulin shows that in the context of fingerprinting capacity the most powerful attack has segment-symmetry.

The strategy is usually expressed as a set of probabilities $\theta_{y|\boldsymbol{\sigma}}$ that apply independently for each segment. Omitting the column index i, we have for each i

$$\Pr[\text{output } y, \text{ given } \boldsymbol{\sigma}] = \theta_{y|\boldsymbol{\sigma}}. \tag{3.7}$$

Due to the marking condition some of these probabilities are fixed. Let $\boldsymbol{e}_\alpha$ denote the vector $(0,\cdots,0,1,0,\cdots,0)$ with the '1' in position α. Then

$$\theta_{y|c\boldsymbol{e}_\alpha} = \delta_{y\alpha}, \tag{3.8}$$

where δ is the Kronecker delta.

Because of the assumed symbol symmetry of the attack, the $\theta_{y|\boldsymbol{\sigma}}$ is invariant under permutation of $\boldsymbol{\sigma}_{\setminus y}$. Hence the attack can be described by a smaller number of parameters. In [33] we introduced the following parametrization:

DEFINITION 3.2. *Let $\alpha \in \mathcal{Q}$, $b \in \{0, \ldots, c\}$, $\boldsymbol{x} \in \{0, \ldots, c\}^{q-1}$ and $\boldsymbol{\sigma} \in \mathcal{S}_{qc}$ such that $\sigma_\alpha = b$ and $\boldsymbol{\sigma}_{\setminus\alpha} = \boldsymbol{x}$. We define*

$$\Psi_b(\boldsymbol{x}) = \theta_{\alpha|\boldsymbol{\sigma}} \tag{3.9}$$

Due to the Marking Assumption we have that $\Psi_c(\mathbf{0}) = 1$ and $\Psi_0(\boldsymbol{x}) = 0$. Notice that $\boldsymbol{x}$ can be considered equivalently as a multi-set of the partition of $c - b$ into $q - 1$ parts. Similarly, $\boldsymbol{\sigma}$ is a multi-set of the partition of c into q parts. For the sake of simplicity we consider and treat $\boldsymbol{x}$ and $\boldsymbol{\sigma}$ as a non-negative vector.

The probability for outputting α given such a $\boldsymbol{\sigma}$ does not depend on the actual value of α, but only on b and $\boldsymbol{x}$. (In fact, it is even insensitive to permutations of $\boldsymbol{x}$.) In words: $\Psi_b(\boldsymbol{x})$ is the coalition's probability of outputting a symbol which for them occurs b times, with the other symbol tallies being $\boldsymbol{x}$. In the case of the binary alphabet, $\boldsymbol{x}$ has only one component equal to $c - b$.

We will investigate some of the most common strategies and their definitions and properties will be discussed in Chapter 5.

3.3 Accusation

The watermark detector sees the symbol y_i embedded in segment i of the attacked content. Users are classified as suspicious ('accused') or not suspicious according to the following algorithm. For each user j, the so-called *accusation sum* S_j is computed,

$$S_j = \sum_{i=1}^{m} S_j^{(i)} \quad \text{where} \quad S_j^{(i)} = g_{[X_{ji}==y_i]}(p_{y_i}^{(i)}), \tag{3.10}$$

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

$$g_1(p) = \sqrt{\frac{1-p}{p}} \quad ; \quad g_0(p) = -\sqrt{\frac{p}{1-p}}. \tag{3.11}$$

In words: Having the same symbol as the attacked content induces a positive contribution $g_1(p_{y_i})$ to the accusation sum, which becomes worse when y_i is unlikely to occur. Having a symbol different from y_i induces a negative amount $g_0(p_{y_i})$, which becomes more negative when y_i is likely to occur. The total accusation of the coalition is defined as $S_{\mathcal{C}} := \sum_{j\in\mathcal{C}} S_j$.

The choice (3.11) of g_0, g_1 is the unique combination of functions that satisfies

$$\mathbb{E}[S_j^{(i)}] = p_{y_i}^{(i)} g_1(p_{y_i}^{(i)}) + (1-p_{y_i}^{(i)}) g_0(p_{y_i}^{(i)}) = 0 \tag{3.12}$$

$$\mathrm{Var}[S_j^{(i)}] = p_{y_i}^{(i)} [g_1(p_{y_i}^{(i)})]^2 + (1-p_{y_i}^{(i)}) [g_0(p_{y_i}^{(i)})]^2 = 1. \tag{3.13}$$

where j is an innocent user. This choice has been shown to be optimal for the binary alphabet [8, 43], i.e. it minimizes the code length. Its unique properties (3.13) also hold for $q \geq 3$; that is the main motivation for using (3.11).

A user is 'accused' if his accusation sum exceeds a threshold Z. A list $\mathcal{L}$ is made of accused users,

$$\mathcal{L} = \{j : S_j > Z\}. \tag{3.14}$$

3.4 Performance

The 'performance' of the scheme involves four important parameters: the number of attackers that has to be resisted (c_0), the maximum tolerable false negative probability ε_2 (probability of not catching any of the attackers),

$$P_{\mathrm{FN}} = \Pr[\mathcal{L} \cap \mathcal{C} = \emptyset] \leq \varepsilon_2, \tag{3.15}$$

the maximum tolerable false positive probability ε_1

$$\text{for fixed innocent } j: \quad P_{\mathrm{FP}} = \Pr[j \in \mathcal{L}] \leq \varepsilon_1, \tag{3.16}$$

and the length m of the code. One way of measuring how well the scheme works is to look at how big m has to be as a function of c_0, ε_1 and ε_2. The smaller m, the better the scheme. It is important to note that in forensic watermarking of audio/video content, a small P_{FP} is the primary requirement. The P_{FN} is far less important, since the deterring effect of forensic watermarking is preserved even for large ε_2, of the order

of $\frac{1}{2}$. Hence m essentially becomes a function of c_0 and ε_1. In [42] an asymptotic result was obtained for large c_0,

$$m = \frac{2}{\tilde{\mu}^2} c_0^2 \ln \frac{1}{\varepsilon_1 \sqrt{2\pi}}. \tag{3.17}$$

Here $\tilde{\mu}$ is the expectation value of the collective accusation sum of the coalition, scaled in such a way that the dependence on m is removed: $\tilde{\mu} = \mathbb{E}[S_{\mathcal{C}}]/m$. In the case of the binary scheme (with $\kappa = 1/2$ and $t \to 0$ because $c_0 \to \infty$), $\tilde{\mu} = 2/\pi \approx 0.64$. For larger alphabets the $\tilde{\mu}$ depends on the parameter κ in a complicated way; for optimal κ, the $\tilde{\mu}$ takes values from approximately 0.8 to 1.4 as q goes from 3 to 10.

The *global* false positive probability is denoted as $P_{\mathrm{FP}}^{\mathrm{global}}$:

$$P_{\mathrm{FP}}^{\mathrm{global}} = \Pr[\mathcal{L} \setminus \mathcal{C} \neq \emptyset]. \tag{3.18}$$

In words: $P_{\mathrm{FP}}^{\mathrm{global}}$ is the probability that at least one innocent user is accused. For $n \gg c$ the following relation with P_{FP} was proven[3]:

$$P_{\mathrm{FP}}^{\mathrm{global}} = nP_{\mathrm{FP}} \left[1 - \frac{c}{n} - \mathcal{O}\left(nP_{\mathrm{FP}}\right)\right] \approx nP_{\mathrm{FP}}. \tag{3.19}$$

The intuition comes from the fact that the dependency between innocent user scores is quite weak. Besides, being usually $n \gg c$, the approximation trivially follows from (3.19).

3.5 The Gaussian approximation

We briefly review the analysis of error probabilities performed in [42], which leads to the result (3.17).

Consider, for some innocent user j, the probability distribution function (pdf) ρ_m of the quantity $S_j/\sqrt{m}$. (Note that the pdf itself depends on m.) From (3.13) it follows that ρ_m has zero mean and unit variance. For brevity we now introduce the notation $\tilde{Z} = Z/\sqrt{m}$. The probability of falsely accusing j is given by the cumulative distribution function (cdf)

$$R_m(\tilde{Z}) := \int_{\tilde{Z}}^{\infty} \mathrm{d}x \; \rho_m(x). \tag{3.20}$$

This is depicted as the shaded area 'FP' in Fig. 3.1. We require

$$R_m(\tilde{Z}) \leq \varepsilon_1. \tag{3.21}$$

[3]Result learnt via personal communication with Jan-Jaap Oosterwijk.

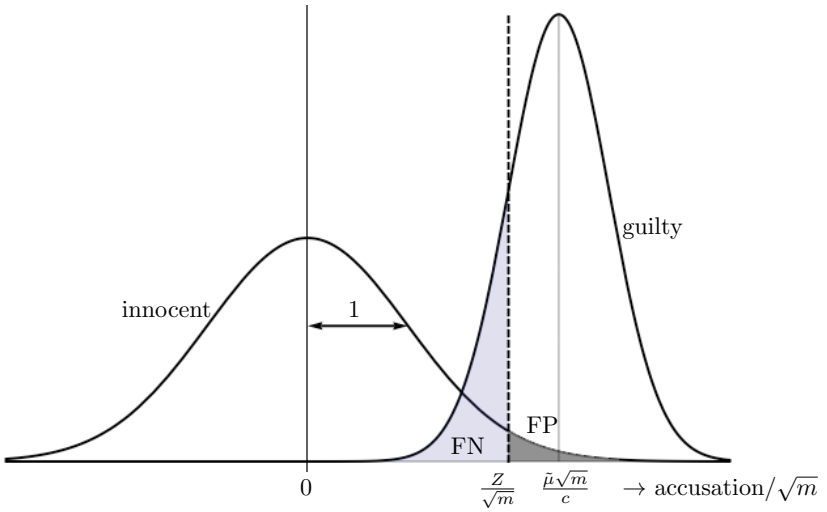


FIGURE 3.1: *Sketch of the probability distributions of $S_j/\sqrt{m}$ for some fixed innocent j, and of $S_{\mathcal{C}}/(c\sqrt{m})$. The horizontal axis is scaled by a factor $\sqrt{m}$ so that the variance of the innocent curve is exactly 1.*

Similarly, consider the probability distribution τ_m of the quantity $S_{\mathcal{C}}/(c\sqrt{m})$, but normalized in such a way that the mean is zero and the variance is 1. The cdf is

$$T_m(x) := \int_{-\infty}^{x} \mathrm{d}x' \; \tau_m(x'). \tag{3.22}$$

It was shown in [38] that $P_{\mathrm{FN}} \leq \Pr[S_{\mathcal{C}} < cZ]$. Hence if $\Pr[S_{\mathcal{C}} < cZ] \leq \varepsilon_2$ then automatically $P_{\mathrm{FN}} \leq \varepsilon_2$. The shaded area in Fig. 3.1 labeled as 'FN' is actually $\Pr[S < cZ]$, which acts as a convenient bound on the FN. This area is given by $T_m([\tilde{Z} - \frac{\tilde{\mu}\sqrt{m}}{c}]/\frac{\tilde{\sigma}}{c}) = T_m(\frac{c\tilde{Z} - \tilde{\mu}\sqrt{m}}{\tilde{\sigma}})$, where $\tilde{\sigma}$ is the (scaled) standard deviation of the collective accusation, $m\tilde{\sigma}^2 := \mathbb{E}[S_{\mathcal{C}}^2] - (\mathbb{E}[S_{\mathcal{C}}])^2$. The requirement on the FN probability in case of c_0 attackers is then formulated as

$$T_m\left(\frac{c_0\tilde{Z} - \tilde{\mu}\sqrt{m}}{\tilde{\sigma}}\right) \leq \varepsilon_2. \tag{3.23}$$

The two equations (3.21) and (3.23) for given c_0, ε_1, ε_2 can be thought of as constraints in the (Z, m)-plane. It was shown [42] that these constraints can be satisfied only if

$$m \geq \frac{1}{\tilde{\mu}^2} c_0^2 \left[R_m^{\mathrm{inv}}(\varepsilon_1) - \frac{\tilde{\sigma}}{c_0} T_m^{\mathrm{inv}}(\varepsilon_2)\right]^2 \tag{3.24}$$

where R_m^{inv} and T_m^{inv} are the inverse functions of R_m and T_m respectively. Note that $T_m^{\text{inv}}(\varepsilon_2) < 0$ for ε_2 smaller[4] than approximately $1/2$; decreasing ε_2 leads to a longer code. It was shown that the T_m^{inv} term is negligible with respect to the R_m^{inv} term if c_0 is large and/or $\varepsilon_2 \approx 1/2$. Hence, (3.24) in practice reduces to

$$m \geq m_{\min} \quad ; \quad m_{\min} \approx \frac{1}{\tilde{\mu}^2} c_0^2 \left[R_m^{\text{inv}}(\varepsilon_1) \right]^2 . \tag{3.25}$$

Eq. (3.25) in itself is not immediately useful, because R_m depends on m. In the limit of large m, however, ρ_m simply becomes a Gaussian independent of m, and R_m is the area under a Gaussian tail, which we denote as $\Omega(\tilde{Z}) = \frac{1}{2}\text{Erfc}\frac{\tilde{Z}}{\sqrt{2}}$. (Here Erfc is the complementary error function.) The result (3.17) follows by applying the bound $[\Omega^{\text{inv}}(\varepsilon_1)]^2 = [\sqrt{2}\,\text{Erfc}^{\text{inv}}(2\varepsilon_1)]^2 < 2\ln(\varepsilon_1\sqrt{2\pi})^{-1}$.

To the best of our knowledge, the above reasoning is the simplest argument available that yields the asymptotic relation $m \propto c_0^2$.

It was argued in [43] and [42] that m is so large that ρ_m is Gaussian even a sufficient number of standard deviations away from 0. ('Sufficient' here means that the area under the Gaussian part is at least $1-2\varepsilon_1$ so that the area under the right tail is estimated accurately.) The argument was based on the moments of the innocent accusation and the Berry-Esseen theorem [6]. From Figure 3.1 is possible to learn the following points:

- For fixed Z and m, the effect of increasing c is that the guilty curve shifts to the left. Later we will show that $\tilde{\mu}$ hardly changes as a function of c.
- In the Gaussian regime, the coalition has very little effect on P_{FP}, protecting the innocent users from big guilty sets.
- Z needs to be proportional to $\sqrt{m}$ and $m \sim c^2$ in order to obtain the desired ε_1. Hence $Z \propto c$, indeed as shown in [38].

A full analysis of the tails of ρ is important for the following reason: as (3.25) shows, it is advantageous for the attackers not only to decrease $\tilde{\mu}$, but also to modify the shape of R_m such that $R_m^{\text{inv}}(\varepsilon_1)$ increases, i.e. such that the right-hand tail of the innocent's accusation probability becomes longer. How much influence their strategy has on the shape of R_m will

[4] If one is willing to set $\varepsilon_2 > 1/2$, the contribution from $T_m^{\text{inv}}(\varepsilon_2)$ may even reduce the code length.

be studied later. If there is hardly any influence, then the value of $\tilde{\mu}$ uniquely determines $m_{\min}$, and the optimal strategy is to minimize $\tilde{\mu}$; if there is a significant influence, then the attackers' aim is to maximize the quotient $R_m^{\text{inv}}(\varepsilon_1)/\tilde{\mu}$.

3.6 Prior analysis on the Tardos code

As mentioned in Section 1.5, existing analytical approaches do not lead to a precise estimation of the error probabilities, while numerical approaches fail for very small error probabilities.

The first analysis was conducted by Tardos in [38], where he found upper bounds for both the FP and FN probabilities using the Markov inequality. Later, tighter bounds were obtained thanks to more precise studies by Blayer and Tassa [2] and by Škorić et al. [43]. In both cases, some of the parameters initially fixed by Tardos are substituted with variables that are later set to the optimal value. In [42] the scheme was modified, making it symbol-symmetric, which increases the ($q = 2$) code rate by a factor 4. However, the method of analysis was not improved. Later, in [18] Laarhoven and de Weger further improved the binary symmetric case improving the approach of Blayer and Tassa [2]. All these proofs were based on the Markov inequality. Only recently Škorić and Oosterwijk in [40] chose Bernstein and Bennett inequalities instead of the Markov inequality, obtaining simpler proofs and a tighter FN bound in the non-binary case. However, the price to pay is that a position-symmetric attack is assumed, thus losing generality.

The alternative to the analytic approach is the numerical approach that computes the outcomes of the Tardos scheme using simulations of the scheme. The drawback of this approach is that the computation is unfeasible for very small probabilities. Furon et al. [7] tried to overcome the problem of simulating small FP probabilities by using techniques from rare-event analysis. In the algorithm a low probability is re-formulated as a product of larger probabilities that can be estimated separately. The level of accuracy of this approach is not completely clear to us as it does not reproduce the power-law probability tails in the non-Gaussian regime (see Section 6.2).

4

The CSE method

The precise knowledge of the FP and FN error probabilities is crucial to determine the real strength of Tardos' codes. In the literature, the proven bounds on P_{FP} use Markov, Bennet and Bernstein inequalities providing a not so tight result. As consequence, the sufficient codelength is also unknown. Also simulations are infeasible when the FP error probability is too small because they would require a huge number of computations.

In this chapter we explain the Convolution and Series Expansion (CSE) method as developed in [33]. This approach makes possible to know the exact value of P_{FP} for almost any desired combination of parameters (code length, alphabet size, coalition size, strategy, ...). The whole procedure is based on the convolution property of the characteristic functions (or Fourier transform).

Definition 4.1 (Fourier transform). *Let $f : \mathbb{R} \to \mathbb{C}$ be a function. The Fourier transform of f is denoted as $\tilde{f}$ and defined as*

$$\tilde{f}(k) = \int_{-\infty}^{\infty} \mathrm{d}x \; e^{-ikx} f(x) \qquad \textit{with } k \in \mathbb{R}. \tag{4.1}$$

Property 4.1 (Convolution). *When random variables are added, the pdf of the sum is obtained by multiplying the Fourier transforms of their respective pdf's and then doing a Fourier back-transform. In other words, if $X \sim f_1$, $Y \sim f_2$ and $Z = X + Y \sim f_3$, then $\tilde{f}_3 = \tilde{f}_1 \tilde{f}_2$.*

Proof. For $z = x + y$ we have that

$$f_3(z) = \int_{-\infty}^{\infty} \mathrm{d}x \; f_1(x) f_2(z - x). \tag{4.2}$$

Then it follows that

$$\begin{aligned}
\tilde{f}_3(k) &= \int_{-\infty}^{\infty} \mathrm{d}z\; e^{-ikz} f_3(z) && (4.3)\\
&= \int_{-\infty}^{\infty} \mathrm{d}z\; e^{-ikz} \int_{-\infty}^{\infty} \mathrm{d}x\; f_1(x) f_2(z-x) && (4.4)\\
&= \int_{-\infty}^{\infty} \mathrm{d}x\; e^{-ikx} f_1(x) \int_{-\infty}^{\infty} \mathrm{d}(z-x)\; e^{-ik(z-x)} f_2(z-x) && (4.5)\\
&= \tilde{f}_1(k)\tilde{f}_2(k). && (4.6)
\end{aligned}$$

□

Therefore, knowing the single-segment-score pdf we can find the pdf for the entire score.

4.1 Preliminaries

4.1.1 Probabilities and expectation values

In this section we are going to introduce several new functions and variables in order to reorganise the probabilistic aspect of the Tardos' scheme. These ingredients will be necessary to find the one-segment pdf's for innocent and guilty users. Then applying the CSE method we will end with the m-segments pdf's.

Because of the column symmetry (Section 3.2), the references to specific segments will be omitted (unless strictly necessary). Especially in the following part, in which we will investigate the single segment properties, the column index i will be omitted for the sake of simplicity. For example, to indicate a bias vector we will use $\boldsymbol{p}$ instead of $\boldsymbol{p}^{(i)}$.

Note about the notation: for a scalar x and a vector $\boldsymbol{p}$, the notation $\boldsymbol{p}^x$ stands for $\prod_\alpha p_\alpha^x$. For vectors $\boldsymbol{p}, \boldsymbol{x}$, the notation $\boldsymbol{p}^{\boldsymbol{x}}$ means $\prod_\alpha p_\alpha^{x_\alpha}$.

Expectation values

We will need to compute expectation values over several random variables mentioned above. To this end we list a number of lemmas that will be useful later.

Expectation over $\boldsymbol{p}$: Let $r(\boldsymbol{p})$ be an arbitrary function. Then the expectation over $\boldsymbol{p}$ is defined as

$$\mathbb{E}_{\boldsymbol{p}}[r(\boldsymbol{p})] := \int_0^1 \mathrm{d}^q \boldsymbol{p}\; F(\boldsymbol{p}) r(\boldsymbol{p}). \tag{4.7}$$

with F as given in 3.1. The following lemma is helpful when one component of $\boldsymbol{p}$ has a special status, for instance p_y, with y the symbol chosen by the attackers. Similarly with $\boldsymbol{\sigma}_{\setminus\alpha}$, the rest of $\boldsymbol{p}$ is denoted as $\boldsymbol{p}_{\setminus y}$.

LEMMA 4.2 (Marginals of the Dirichlet distribution). *Let r be any function of $\boldsymbol{p}$. The expectation value $\mathbb{E}_{\boldsymbol{p}}$ can be split into two parts as*

$$\mathbb{E}_{\boldsymbol{p}}[r(\boldsymbol{p})] = \mathbb{E}_{p_y}\left[\mathbb{E}_{\boldsymbol{p}_{\setminus y}|p_y}[r(\boldsymbol{p})]\right], \tag{4.8}$$

with

$$\mathbb{E}_{p_y}[\cdots] = \frac{1}{B(\kappa, \kappa[q-1])} \int_0^1 \mathrm{d}p_y\; p_y^{-1+\kappa}(1-p_y)^{-1+\kappa[q-1]}[\cdots] \tag{4.9}$$

$$\mathbb{E}_{\boldsymbol{p}_{\setminus y}|p_y}[r(\boldsymbol{p})] = \frac{1}{B(\kappa \mathbf{1}_{q-1})} \int_0^1 \mathrm{d}^{q-1}\boldsymbol{s}\; \delta\left(1 - \sum_{\beta \in \mathcal{Q}\setminus\{y\}} s_\beta\right) \boldsymbol{s}^{-1+\kappa} r(\boldsymbol{p})\big|_{\boldsymbol{p}_{\setminus y} = (1-p_y)\boldsymbol{s}}. \tag{4.10}$$

PROOF. See Appendix A □

Expectation over $\boldsymbol{\sigma}|\boldsymbol{p}$: Let $r(\boldsymbol{\sigma})$ be an arbitrary function. Then

$$\mathbb{E}_{\boldsymbol{\sigma}|\boldsymbol{p}}[r(\boldsymbol{\sigma})] := \sum_{\boldsymbol{\sigma} \in \mathcal{S}_{qc}} \binom{c}{\boldsymbol{\sigma}} \boldsymbol{p}^{\boldsymbol{\sigma}} r(\boldsymbol{\sigma}). \tag{4.11}$$

Expectation over $y|\boldsymbol{\sigma}$: Let $r(y)$ be an arbitrary function. Then

$$\mathbb{E}_{y|\boldsymbol{\sigma}}[r(y)] := \textstyle\sum_{y \in \mathcal{Q}} \theta_{y|\boldsymbol{\sigma}} r(y). \tag{4.12}$$

Expectation over $y|\boldsymbol{p}$: Let $r(y)$ be an arbitrary function. We introduce the notation $T_{y|\boldsymbol{p}}$ to denote the following sum,

$$T_{y|\boldsymbol{p}} = \sum_{\boldsymbol{\sigma} \in \mathcal{S}_{qc}} \binom{c}{\boldsymbol{\sigma}} \boldsymbol{p}^{\boldsymbol{\sigma}} \theta_{y|\boldsymbol{\sigma}}, \tag{4.13}$$

where the condition $\sum_\alpha p_\alpha = 1$ is *not* enforced. This will allow us to write several important expressions compactly in terms of partial derivatives of T. The notation $\tau_{y|\boldsymbol{p}}$ is defined as $T_{y|\boldsymbol{p}}$ where we *do* enforce the 'on-shell' condition $\sum_\alpha p_\alpha = 1$. It represents the conditional probability that y occurs given $\boldsymbol{p}$. Then, the expectation value over $y|\boldsymbol{p}$ trivially follows:

$$\mathbb{E}_{y|\boldsymbol{p}}[r(y)] = \sum_{y\in\mathcal{Q}} \tau_{y|\boldsymbol{p}} r(y). \tag{4.14}$$

Probabilities

The information owned by the distributor is obviously much more wide then the colluders one and, moreover, the marking assumptions combined with the RDM definition increase his power. Indeed, they not only reduce the positions and the symbols that the attackers are allowed to use to apply their attack, also they allow to apply an 'a priori' study to restrict deterministically the coalition set. The idea is quite intuitive: if colluders output the symbol y_i in position i, then the set of users that own that symbol in position i contains certainly at least one of the attackers, and this concept can be extended on more segments. In the lucky event in which just a user has the symbol outputted[1], then we can accuse him with 0% chance to make a mistake. However, our study is based only on Tardos' pure approach and these analysis are not taken into consideration.

For given $\boldsymbol{p}$, the probability that the c colluders receive symbol tallies $\boldsymbol{\sigma}$ is the multinomial distribution. We use the following notation,

$$\mathbb{P}(\boldsymbol{\sigma}|\boldsymbol{p}) := \binom{c}{\boldsymbol{\sigma}} \prod_{\alpha\in\mathcal{Q}} p_\alpha^{\sigma_\alpha}, \tag{4.15}$$

where $\binom{c}{\boldsymbol{\sigma}} = c!/(\prod_\alpha \sigma_\alpha!)$. It is always implicitly understood that $\sum_\alpha \sigma_\alpha = c$.

Similarly, the marginal distribution for a single component σ_α is the binomial. We use the notation

$$\mathbb{P}_1(b|p) := \Pr[\sigma_\alpha = b|p_\alpha = p] = \binom{c}{b} p^b (1-p)^{c-b}. \tag{4.16}$$

[1]Such unexpected event can happen more easily if the strategy used is, for example, minority voting (see Chapter 5) combined with a big value of m and the Dirichlet distribution nature to emphasize one symbol to the detriment of the othes.

LEMMA 4.3. *The overall probability that the colluders receive symbol occurrences $\boldsymbol{\sigma}$ is given by*

$$\mathbb{P}(\boldsymbol{\sigma}) := \binom{c}{\boldsymbol{\sigma}} \frac{B(\kappa \mathbf{1}_q + \boldsymbol{\sigma})}{B(\kappa \mathbf{1}_q)}. \tag{4.17}$$

PROOF. We have $\Pr[\boldsymbol{\sigma}] = \mathbb{E}_{\boldsymbol{p}} \mathbb{P}(\boldsymbol{\sigma}|\boldsymbol{p})$, with $\mathbb{P}(\boldsymbol{\sigma}|\boldsymbol{p})$ given by (4.15). The lemma follows by applying the Dirichlet integration rule (3.3). □

LEMMA 4.4. *The marginal probability distribution $f(p_\alpha)$ for a single component of the vector $\boldsymbol{p}$ is*

$$f(p_\alpha) = \frac{1}{B(\kappa, \kappa[q-1])} p_\alpha^{-1+\kappa} (1-p_\alpha)^{-1+\kappa[q-1]}. \tag{4.18}$$

PROOF. We have $1 = \int_0^1 \mathrm{d}p_\alpha \; f(p_\alpha) = \int_0^1 \mathrm{d}^q \boldsymbol{p} \; F(\boldsymbol{p})$. From Lemma 4.2 we have

$$f(p_\alpha) = p_\alpha^{-1+\kappa} (1-p_\alpha)^{-1+\kappa[q-1]} \frac{1}{B(\kappa \mathbf{1}_q)} \int_0^1 \mathrm{d}^{q-1} \boldsymbol{s} \; \delta \left(1 - \sum_{\gamma \in \mathcal{Q} \setminus \{\alpha\}} s_\gamma \right) \boldsymbol{s}^{-1+\kappa}. \tag{4.19}$$

The lemma follows after evaluation of the $\int \mathrm{d}^{q-1} \boldsymbol{s}$ integral using (3.3). □

LEMMA 4.5. *The overall marginal probability distribution for one component of $\boldsymbol{\sigma}$ is*

$$\mathbb{P}_1(b) := \Pr[\sigma_\alpha = b] = \binom{c}{b} \frac{B(\kappa + b, \kappa[q-1] + c - b)}{B(\kappa, \kappa[q-1])}. \tag{4.20}$$

PROOF. We have

$$\Pr[\sigma_\alpha = b] = \int_0^1 \mathrm{d}p_\alpha f(p_\alpha) \mathbb{P}_1(b|p_\alpha) \tag{4.21}$$

with $\mathbb{P}_1(b|p_\alpha)$ and $f(p_\alpha)$ given by (4.16) and Lemma 4.4 respectively. The integral is evaluated using (3.3). □

COROLLARY 4.6. *Let $\boldsymbol{\sigma}_{\setminus\alpha}$ denote the vector $\boldsymbol{\sigma}$ without the component σ_α. The probability distribution of $\boldsymbol{\sigma}_{\setminus\alpha}$ conditioned on σ_α is given by*

$$\mathbb{P}_{q-1}(\boldsymbol{x}|b) := \Pr[\boldsymbol{\sigma}_{\setminus\alpha} = \boldsymbol{x} | \sigma_\alpha = b] = \binom{c-b}{\boldsymbol{x}} \frac{B(\kappa \mathbf{1}_{q-1} + \boldsymbol{x})}{B(\kappa \mathbf{1}_{q-1})}. \tag{4.22}$$

PROOF. Follows directly from Lemmas 4.3 and 4.5 by taking $\Pr[\boldsymbol{\sigma}_{\setminus\alpha} = \boldsymbol{x}|\sigma_\alpha = b] = \mathbb{P}(\boldsymbol{\sigma} = (\boldsymbol{x}, b))/\mathbb{P}_1(b)$ and simplifying the Beta functions. □

We now introduce a new parameter very important for our research and for which we will dedicate the whole Chapter 5. Its definition is

$$K_b := \mathbb{E}_{\boldsymbol{x}|b}\Psi_b(\boldsymbol{x}) = \sum_{\boldsymbol{x}} \mathbb{P}_{q-1}(\boldsymbol{x}|b)\Psi_b(\boldsymbol{x}). \tag{4.23}$$

It is implicit that $\sum_{\beta\in\mathcal{Q}\setminus\{\alpha\}} x_\beta = c - b$. For $q = 2$ we define $K_b = \Psi_b$. (In some of the literature the notation $\theta_x := \Pr[y = 1|\ \#\text{received 1s} = x]$ is used for the binary case. The relation with our notation is: $\theta_b = \Psi_b$.)

For any strategy we have

$$K_0 = 0 \qquad ; \qquad K_c = 1 \tag{4.24}$$

due to the marking assumption.

LEMMA 4.7. *The numbers K_b satisfy*

$$q\sum_{b=1}^{c} K_b\mathbb{P}_1(b) = 1. \tag{4.25}$$

PROOF. The factor q can be replaced by $\sum_{y\in\mathcal{Q}}$. Using the definition (4.23) we get

$$\sum_y\sum_b K_b\mathbb{P}_1(b) = \sum_b\sum_{\boldsymbol{x}}\mathbb{P}(\boldsymbol{x}, b)\cdot\sum_y\Psi_b(\boldsymbol{x}) \tag{4.26}$$

$$= \sum_b\sum_{\boldsymbol{x}}\mathbb{P}(\boldsymbol{x}, b)\cdot\sum_y\theta_{y|\sigma_y=b,\boldsymbol{\sigma}_{\setminus y}=\boldsymbol{x}} \tag{4.27}$$

$$= \sum_b\sum_{\boldsymbol{x}}\mathbb{P}(\boldsymbol{x}, b) = 1. \tag{4.28}$$

□

The meaning of K_b is not straightforward. It indicates the overall probability that a symbol with tally b (if such a symbol exists) gets chosen by the attackers. Unlike θ and ψ, K_b is averages over all the other random variables. Then the interpretation of 4.25 can be seen as follows: the probability that some b gets chosen is 100%.

4.1.2 Integrals and Gamma function equalities

Before going further, we need some more lemmas and definitions that will become useful in the future.

LEMMA 4.8. *For $d > 0$, $v > 0$, the following holds*

$$\int_0^\infty \mathrm{d}u \, \frac{u^{2d-1}}{(1+u^2)^{d+v}} = \tfrac{1}{2} B(d, v). \tag{4.29}$$

PROOF. Apply a change of variables $u = \sqrt{p/(1-p)}$, with $p \in [0, 1]$. This gives $1 + u^2 = 1/(1-p)$ and $\mathrm{d}u = \frac{1}{2} p^{-1/2}(1-p)^{-3/2}\mathrm{d}p$. The integral becomes $\frac{1}{2}\int_0^1 \mathrm{d}p \, p^{-1+d}(1-p)^{-1+v}$ which has the Dirichlet form (3.3). □

LEMMA 4.9. *For $x \gg 1$, and $a_1, a_2 \in \mathbb{R}$ such that $|a_1| \ll x$ and $|a_2| \ll x$ and a_1, a_2 independent of x, it holds that*

$$\frac{\Gamma(x+a_1)}{\Gamma(x+a_2)} = x^{a_1-a_2}\left[1 + \mathcal{O}\left(\frac{1}{x}\right)\right]. \tag{4.30}$$

PROOF. Follows directly from Stirling's approximation

$$\Gamma(z+1) \approx \sqrt{2\pi z}\left(\frac{z}{e}\right)^z \left(1 + \mathcal{O}\left(\frac{1}{z}\right)\right). \tag{4.31}$$

□

COROLLARY 4.10. *Let $c \gg 1$ and $1 \ll b \leq c$. Let $|\alpha_1|, |\alpha_2|, |\beta_1|, |\beta_2| \ll b$ and $\alpha_1, \alpha_2, \beta_1, \beta_2$ independent of b and c. Then*

$$\frac{B(b+\alpha_1, c-b+\beta_1)}{B(b+\alpha_2, c-b+\beta_2)} = \left(\frac{b}{c}\right)^{\alpha_1-\alpha_2}\left(1-\frac{b}{c}\right)^{\beta_1-\beta_2}\left[1 + \mathcal{O}\left(\frac{1}{b}\right)\right]. \tag{4.32}$$

PROOF. Follows directly from writing out the Beta functions in terms of Gamma functions and then applying Lemma 4.9. □

DEFINITION 4.2. *We define $\Omega(z)$ as the probability mass in the right tail of the normal distribution beyond point z,*

$$\Omega(z) = \frac{1}{\sqrt{2\pi}} \int_z^\infty \mathrm{d}x \, e^{-x^2/2}. \tag{4.33}$$

LEMMA 4.11 (See e.g. Eq. 9.254.1 in [9]). *For* $x \in \mathbb{R}$

$$\frac{1}{2\pi i}\int_{-\infty}^{\infty} \mathrm{d}k\, \frac{e^{ikx}}{k} e^{-k^2/2} = \tfrac{1}{2} - \Omega(x). \tag{4.34}$$

LEMMA 4.12 (See e.g. Eq. 3.462.1 in [9]). *For* $\nu > 0$ *and* $x \in \mathbb{R}$

$$\int_0^{\infty} \mathrm{d}k\, k^{\nu-1} e^{-\frac{1}{2}k^2} e^{ikx} = \Gamma(\nu) 2^{\nu/2} H_{-\nu}\left(\frac{-ix}{\sqrt{2}}\right). \tag{4.35}$$

Here H *is the Hermite function.*

COROLLARY 4.13. *For* $x \in \mathbb{R}$ *and* $\nu > 0$

$$\int_{-\infty}^{\infty} \frac{\mathrm{d}k}{2\pi} (i \operatorname{sgn} k)^{\alpha-1} |k|^{\nu-1} e^{-k^2/2} e^{ikx} = \frac{1}{\pi}\Gamma(\nu) 2^{\nu/2} \operatorname{Im}\left[i^{-\alpha} H_{-\nu}\left(\frac{ix}{\sqrt{2}}\right)\right] \tag{4.36}$$

PROOF. The first equality follows by applying Lemma 4.12 twice (once for the positive part of the integral, once for the negative). □

4.1.3 Fourier transforms

We now introduce few lemmas regarding the Fourier transform defined previously in Definition 4.1.

LEMMA 4.14. *If* f *is a real-valued function, then* $\tilde{f}(-k) = [\tilde{f}(k)]^*$.

PROOF. $[\int \mathrm{d}x\, e^{-ikx} f(x)]^* = \int \mathrm{d}x\, [e^{-ikx} f(x)]^* = \int \mathrm{d}x\, e^{ikx} f(x) = \tilde{f}(-k)$. □

COROLLARY 4.15. *If* f *is a real-valued function, then the even part of* $\tilde{f}(k)$ *is* $\operatorname{Re}\tilde{f}(k)$*, and the odd part is* $i \cdot \operatorname{Im}\tilde{f}(k)$.

PROOF. By Lemma 4.14, the even part is $\frac{1}{2}[\tilde{f}(k) + \tilde{f}(-k)] = \frac{1}{2}\tilde{f}(k) + \frac{1}{2}[\tilde{f}(k)]^* = \operatorname{Re}\tilde{f}(k)$. The odd part is $\frac{1}{2}[\tilde{f}(k) - \tilde{f}(-k)] = \frac{1}{2}\tilde{f}(k) - \frac{1}{2}[\tilde{f}(k)]^* = i\operatorname{Im}\tilde{f}(k)$. □

LEMMA 4.16. *Let* $f(x)$ *be a probability distribution function, and* X *a random variable with* $X \sim f$. *Then*

$$\left.\frac{\partial^n \tilde{f}(k)}{\partial k^n}\right|_{k=0} = (-i)^n \mathbb{E}[X^n]. \tag{4.37}$$

PROOF. $\frac{\partial^n \tilde{f}(k)}{\partial k^n} = \int \mathrm{d}x\, [\frac{\partial^n}{\partial k^n} e^{-ikx}] f(x) = (-i^n) \int \mathrm{d}x\, x^n e^{-ikx} f(x)$. Setting $k = 0$ gives the result. □

Now we have all the ingredients to compute P_{FP} using the CSE method.

4.2 CSE method for the innocent user score

Next step consists in deriving the probability for an innocent user to obtain a specific score in a single segment. We denote this score $S_j(i) = u$ and the pdf as $\varphi(u)$ We then calculate its Fourier transform necessary to apply Property 4.1 obtaining finally the m-segment pdf ρ_m.

Before going through φ and $\tilde{\varphi}$ mathematical definitions, we are going to talk about R_m instead. This choice, that could appear not much intuitive, has the advantage to show which "form" of $\tilde{\varphi}$ is required to write R_m, avoiding intermediate unnecessary steps. This will become more clear in the next section.

4.2.1 From one segment to the full score

Lemma 4.16 combined with (3.13) can already produce the next helpful corollary.

COROLLARY 4.17. *Let φ be the probability distribution function of the one-symbol accusation $S_j^{(i)}$ for an innocent user j. Then its Fourier transform $\tilde{\varphi}$ has the following power series expansion,*

$$\tilde{\varphi}(k) = 1 - \tfrac{1}{2}k^2 + \textit{higher powers of } k, \tag{4.38}$$

where the higher powers of k are allowed to be irrational.

PROOF. Trivially $\mathbb{E}[u^0] = 1$. From (3.13) we know that $\mathbb{E}[u] = 0$ and $\mathbb{E}[u^2] = 1$. Hence by Lemma 4.16 we have $\tilde{\varphi}(0) = 1$, $\tilde{\varphi}'(0) = 0$ and $\tilde{\varphi}''(0) = -1$. The expansion in the corollary is consistent with these values. Higher orders of k do not have to be integer. In fact, if $\mathbb{E}[u^3] \neq 0$, $\mathbb{E}[u^3] < \infty$ and $\mathbb{E}[u^4] = \infty$ (as we will see later) then there is a k^3 term in the expansion, and the lowest power of k higher than 3 lies somewhere between 3 and 4. □

We now apply Property 4.1 to obtain ρ_m from φ. Knowing ρ_m we get the probability mass in the right tail (R_m) by integration 3.20. When the convolution is applied to the m random variables in the accusation sum, it leads to the following result.

THEOREM 4.18. *Let j be an innocent user. Let φ denote the pdf of $S_j^{(i)}$, with $S_j^{(i)}$ as defined in (3.10). Let $\tilde{\varphi}$ be the Fourier transform of φ. Then*

the probability $P_{\text{FP}} = \Pr[S_j > Z]$ *is given by*

$$R_m(\tilde{Z}) = \frac{1}{2} + \frac{i}{2\pi} \int_{-\infty}^{\infty} \mathrm{d}k \, \frac{\exp ik\tilde{Z}}{k} \left[\tilde{\varphi}\left(\frac{k}{\sqrt{m}} \right) \right]^m . \tag{4.39}$$

PROOF. See Appendix B. □

This result gives us a closed-form expression for $R_m(\tilde{Z})$ that contains only a single integration and a limited number of sums. (The sums are contained in the evaluation of $\tilde{\varphi}$, as will become apparent in Section 4.2.3.) These will have to be evaluated numerically. Note that $\Pr[S_j > 0]$ is not necessarily equal to $\frac{1}{2}$.

It turns out that numerical evaluation of the integral in (4.39) is difficult, because of the fast oscillations of the integrand at large k. For this reason, we have chosen for a somewhat indirect method of evaluating (4.39). It is based on a series expansion in powers of k. It has the advantage that the accuracy of the numerical evaluation is well under control, and that the dependence of R_m on m is visible. The disadvantage is that many terms in the expansion have to be kept.

THEOREM 4.19. *Let j be an innocent user. Let φ have a finite third moment. Then it is possible to write*

$$\left[\tilde{\varphi}\left(\frac{k}{\sqrt{m}} \right) \right]^m = e^{-\frac{1}{2}k^2} \left[1 + \sum_{t=0}^{\infty} \omega_t(m) (i \operatorname{sgn} k)^{\alpha_t} |k|^{\nu_t} \right], \tag{4.40}$$

where α_t are real numbers; the coefficients $\omega_t(m)$ are real; the powers ν_t satisfy $\nu_0 = 3$ and $\nu_{t+1} > \nu_t$. The ν_t are not necessarily integer. All the coefficients $\omega_t(m)$ are decreasing functions of m, decreasing as $m^{-\nu_t/6}$ or faster.

The probability of accusing user j is given by

$$R_m(\tilde{Z}) = \Omega(\tilde{Z}) + \frac{1}{\pi} \sum_{t=0}^{\infty} \omega_t(m) \Gamma(\nu_t) 2^{\nu_t/2} \operatorname{Im} \left[i^{-\alpha_t} H_{-\nu_t}(i\tilde{Z}/\sqrt{2}) \right]. \tag{4.41}$$

Here H is the Hermite function.

PROOF. See Appendix C. □

The proof closely follows one of the standard proofs of the Central Limit Theorem. In the limit $m \to \infty$ all the coefficients ω_t vanish, leaving only the term $\Omega(\tilde{Z})$ which is the right-hand tail mass of the normal

distribution. For integer ν the function $H_{-\nu}$ reduces to the Hermite polynomial of order $\nu-1$, multiplied by a factor $\exp(-\frac{1}{2}\tilde{Z}^2)$. (See Corollary 4.13.) The Gaussian convergence of pdfs with finite third moment has been investigated also in the Berry-Esseen theorem [6]:

THEOREM 4.20 (Berry-Esseen). *Let be $X_1, X_2, \ldots$ i.i.d. random variables with $\mathbb{E}(X_1) = 0, \mathbb{E}(X_1^2) = \sigma^2 > 0$ and $\mathbb{E}(|X_1|^3) = M_3 < \infty$. Let be*

$$Y_n = \frac{X_1 + X_2 + \cdots + X_n}{n} \tag{4.42}$$

the simple mean, F_n the cdf of $\frac{Y_n\sqrt{n}}{\sigma}$ and $\Upsilon(x)$ the cdf of the standard normal distribution. Then it exists $C > 0$ s.t., for all x and n

$$|F_n(x) - \Upsilon(x)| \leq \frac{CM_3}{\sigma^3\sqrt{n}}. \tag{4.43}$$

In other words, the Berry-Esseen theorem proves that if a pdf ξ has a finite third moment then the convergence to the normal distribution will be reached adding together a large number of i.i.d. $X_i \sim \xi$.

In Section 4.2.2 we determine the distribution φ. In Section 4.2.3 the Fourier transform $\tilde{\varphi}$ is computed and the leading order parameters ν_t, ω_t, α_t are derived.

4.2.2 Distribution function of an innocent user's single-segment score

THEOREM 4.21. *For an innocent user j, the distribution function φ of $S_j^{(i)}$ is given by*

$$u > 0 \quad : \quad \varphi_+(u) = \frac{2q}{B(\kappa, \kappa[q-1])} \sum_{b=1}^{c} \binom{c}{b} \frac{(u^2)^{\kappa[q-1]+c-b-\frac{1}{2}}}{(1+u^2)^{c+1+\kappa q}} K_b, \tag{4.44}$$

$$u < 0 \quad : \quad \varphi_-(u) = \frac{2q}{B(\kappa, \kappa[q-1])} \sum_{b=1}^{c} \binom{c}{b} \frac{(u^2)^{\kappa+b-\frac{1}{2}}}{(1+u^2)^{c+1+\kappa q}} K_b. \tag{4.45}$$

PROOF. See Appendix D. □

Note that all dependence on the strategy is contained in the numbers $K_b \in [0, 1]$. Furthermore we see that the left tail and the right tail of $\varphi(u)$ have different power law behaviour. This is summarized in Table 4.1.

b	Left tail	Right tail	$u \uparrow 0$	$u \downarrow 0$
1	$(\frac{1}{\lvert u\rvert})^{2c+1+2\kappa[q-1]}$	$\boldsymbol{(\frac{1}{u})^{5+2\kappa}}$	$\boldsymbol{\lvert u\rvert^{1+2\kappa}}$	$u^{2c-3+2\kappa[q-1]}$
c	$\boldsymbol{(\frac{1}{\lvert u\rvert})^{3+2\kappa[q-1]}}$	$(\frac{1}{u})^{2c+3+2\kappa}$	$\lvert u\rvert^{2c-1+2\kappa}$	$\boldsymbol{u^{-1+2\kappa[q-1]}}$

TABLE 4.1: Powers in $\varphi(u)$ in the tails and close to $u = 0$. Dominant powers are shown in boldface.

Note also that for $2\kappa[q-1] > 1$ the absolute third moment exists: the integral $\int \mathrm{d}u\ |u|^3\varphi(u)$ is convergent in both tails. (As opposed to the binary case with $\kappa = 1/2$.) Consequently, there is a *guaranteed convergence to the normal distribution* when i.i.d. random variables $u_i \sim \varphi$ are added together in large numbers.

The right tail is dominated by the $b = 1$ term; it is proportional to $(1/u)^{5+2\kappa}$. The left tail is dominated by the $b = c$ term, and is proportional to $(1/|u|)^{3+2\kappa q-2\kappa}$. It was found numerically in [42] that the 'optimal' κ (in terms of maximizing $\tilde{\mu}$) lies close to $1/q$; for such a choice of κ the left tail is heavier than the right tail. Such a property is obviously beneficial for not accusing *innocent* users.

The behaviour of $\varphi(u)$ around $u = 0$ is also noteworthy. For $u \uparrow 0$ the function is dominated by the $b = 1$ contribution $|u|^{1+2\kappa}$, which has zero derivative at $u = 0$. For $u \downarrow 0$ the $b = c$ term $u^{-1+2\kappa[q-1]}$ dominates; this one, however, has infinite derivative for $\kappa < 1/(q-1)$ (which is the case when e.g. $\kappa \approx 1/q$).

In Chapter 5 we will see that Table 4.1 provides useful information that allows us to compare colluder strategies.

COROLLARY 4.22. *For an innocent user, the overall probability of positive and negative accusation are in general unequal, and are given by*

$$\Pr[u > 0] = q \sum_{b=1}^{c} K_b \mathbb{P}_1(b) \frac{b+\kappa}{c+\kappa q} \tag{4.46}$$

$$\Pr[u < 0] = q \sum_{b=1}^{c} K_b \mathbb{P}_1(b) \frac{c-b+\kappa[q-1]}{c+\kappa q}. \tag{4.47}$$

PROOF. Follows by evaluating the u-integrals with Lemma 4.8, then applying Lemma 4.5 and finally rewriting the Beta functions using $B(x, y+1) = B(x,y)\frac{y}{x+y}$. □

Note that the probabilities (4.46) properly add up to 1; this is readily seen from Lemma 4.7. Note also that it is also visible from Theorem 4.18 that $\Pr[u > 0] \neq \frac{1}{2}$ in general.

4.2.3 The Fourier transform of φ

We finally arrive to the last missing component needed to obtain R_m: the Fourier transform (characteristic function) of $\varphi(u)$. To compute $\tilde{\varphi}$ we are going to use the following lemma:

LEMMA 4.23 (From [31], section 2.5.9). *Let $k \in \mathbb{R}$, Re $v > -\frac{1}{2}$, and $d > 0$. Let the function Λ be defined as the following convergent integral,*

$$\Lambda(d, v; k) := \int_0^\infty \mathrm{d}u \, \frac{u^{2d-1}}{(u^2+1)^{v+d}} e^{iku}. \tag{4.48}$$

This integral is expressed in terms of hypergeometric ${}_1F_2$ functions as

$$\begin{aligned} \Lambda(d, v; k) &= (-ik)^{2v}\Gamma(-2v)\, {}_1F_2\left(v+d; v+\frac{1}{2}, v+1; \frac{k^2}{4}\right) \\ &\quad + \frac{1}{2}\sum_{j=0}^{\infty} \frac{(ik)^j}{j!} B\left(d+\frac{j}{2}, v-\frac{j}{2}\right) \qquad (4.49) \\ &= (-ik)^{2v}\Gamma(-2v)\, {}_1F_2\left(v+d; v+\frac{1}{2}, v+1; \frac{k^2}{4}\right) \\ &\quad + \frac{1}{2}B(d, v)\, {}_1F_2\left(d; \frac{1}{2}, 1-v; \frac{k^2}{4}\right) \\ &\quad + \frac{ik}{2} B\left(d+\frac{1}{2}, v-\frac{1}{2}\right)\, {}_1F_2\left(d+\frac{1}{2}; \frac{3}{2}, \frac{3}{2}-v; \frac{k^2}{4}\right). \qquad (4.50) \end{aligned}$$

Notice that in general $\Lambda(d, v; k)$ is not an entire function of k due to the appearance of the factor k^{2v} in the first term, which for general v is not an entire function.

The hypergeometric function ${}_1F_2$ has the sum representation ${}_1F_2(\alpha; \beta_1, \beta_2; z) = \sum_{j=0}^{\infty} \frac{(\alpha)_j}{j!(\beta_1)_j(\beta_2)_j} z^j$ where $(\alpha)_j = \alpha(\alpha+1)\cdots(\alpha+j-1)$ is the Pochhammer symbol. The radius of convergence is infinity. The ${}_1F_2$ function can be evaluated by using software packages such as Mathematica.

THEOREM 4.24. *The Fourier transform of φ is given by*

$$\tilde{\varphi}(k) = \frac{2q}{B(\kappa, \kappa[q-1])} \sum_{b=1}^{c} \binom{c}{b} K_b \cdot \Big[\Lambda(d_b, v_b; k) + \Lambda(D_b, V_b; -k) \Big], \tag{4.51}$$

with Λ as defined in Lemma 4.23, and

$$\begin{aligned} d_b &:= b + \kappa \quad ; \quad v_b := c - b + \kappa[q-1] + 1 \\ D_b &:= c - b + \kappa[q-1] \quad ; \quad V_b := b + \kappa + 1. \end{aligned} \tag{4.52}$$

PROOF. The Fourier transform is defined as $\tilde{\varphi}(k) = \int_{-\infty}^{\infty} \mathrm{d}u \; \varphi(u) e^{-iku}$. We use the expression for φ given in Theorem 4.21. The integral for the summands in φ_+ is immediately of the form appearing in Lemma 4.23 and yields $\Lambda(D_b, V_b; -k)$. The integral over the φ_- terms is of the form $\int_{-\infty}^{0} \mathrm{d}u \; f(u^2) e^{-iku}$, which can be rewritten as $\int_0^{\infty} \mathrm{d}u \; f(u^2) e^{iku}$; this has the form of the integral in Lemma 4.23 and yields $\Lambda(d_b, v_b; k)$. □

For $q \geq 3$ and realistic κ, none of the values d_b, v_b, D_b, V_b in (4.73) is integer or half-integer. Hence substitution into all the Gamma functions and Pochhammers contained in the ${}_1F_2$ functions of Lemma 4.23 is well defined. Note that, given the summation range $1 \leq b \leq c$, the smallest possible value of v_b or V_b is $v_c = 1 + \kappa[q-1] > 1$. Hence, in a power series expansion for small k, the k^{2v} term in (4.49) always comes 'after' the k^3 power. In fact, for $q \geq 3$ and $\kappa \approx 1/q$ we have $2v_c \in (3, 4)$.

COROLLARY 4.25. *For $q \geq 3$ the leading order terms in the expansion of $\tilde{\varphi}(k)$ are given by*

$$\begin{aligned} \tilde{\varphi}(k) \;=\; & 1 - \tfrac{1}{2}k^2 + \frac{2q}{B(\kappa, \kappa[q-1])} \cdot \\ & \Bigg\{ \frac{(ik)^3}{2 \cdot 3!} \sum_{b=1}^{c} K_b [B(d_b + \tfrac{3}{2}, v_b - \tfrac{3}{2}) - B(D_b + \tfrac{3}{2}, V_b - \tfrac{3}{2})] \\ & + (-ik)^{2+2\kappa[q-1]} \Gamma(-2 - 2\kappa[q-1]) \\ & + \frac{(ik)^4}{2 \cdot 4!} \sum_{b=1}^{c} K_b [B(d_b + 2, v_b - 2) - B(D_b + 2, V_b - 2)] \\ & + (ik)^{4+2\kappa} K_1 \Gamma(-4 - 2\kappa) \\ & + \cdots \Bigg\} \end{aligned} \tag{4.53}$$

Proof. Follows by substituting the first expression for Λ from Lemma 4.23 into Theorem 4.24, and then cutting off the small-argument power series of the ${}_1F_2$ function (which is preceded by a factor $(-ik)^{2v}$) after the k^0 term. □

We refer to the recipe detailed above as the Convolution and Series Expansion (CSE) method.

All the components are now in place to obtain the false positive probability P_{FP}. This topic will be discussed in Chapters 5 and 6 where we will show the complexity of the various parts of R_m and how we manage to efficiently obtain numerics.

4.3 Guilty user's probability distribution functions

To compute guilty user's pdf the approach will be exactly the same as for the innocent one. Our target now will be to calculate the $\Pr[S_j > Z]$, that as for the innocent case, it is all based on just one user j that, this time, he is going to be part of the coalition $\mathcal{C}$. This small but fundamental difference will make slightly more complicated the computations for two main reasons:

1. being in $\mathcal{C}$, j has directly contributed to the generation of y, fact that was (pleasantly) missing in the innocent case,

2. the useful properties of g_0 and g_1 do not hold anymore for the guilty users, making more complicated to obtain P_{FN}.

It is important to point immediately the fact that, being the following a one-user-only approach, the results we are going to obtain do not allow us to compute the quantity P_{FN} exactly.

4.3.1 Relation between $\Pr[S_j > Z]$ and P_{FN}

The quantity that we compute, $\Pr[S_j > Z]$ (for a guilty user j), is not equal to the quantity that we are most interested in, namely P_{FN}. However, we can use $\Pr[S_j > Z]$ to give an upper bound on P_{FN}, as we state in the following lemma.

Lemma 4.26. *Let $j \in \mathcal{C}$. It holds that $P_{\text{FN}} < 1 - \Pr[S_j > Z]$.*

PROOF. Let $\mathcal{L}$ be the set of accused users, and $\mathcal{A} = \mathcal{L} \cap \mathcal{C}$ the set of attackers that end up in $\mathcal{L}$. Then

$$1 = \Pr[|\mathcal{A}| = 0] + \Pr[|\mathcal{A}| > 0] \tag{4.54}$$
$$= P_{\text{FN}} + \Pr[|\mathcal{A}| > 0 \wedge j \in \mathcal{A}] + \Pr[|\mathcal{A}| > 0 \wedge j \notin \mathcal{A}] \tag{4.55}$$
$$= P_{\text{FN}} + \Pr[j \in \mathcal{A}] + \Pr[|\mathcal{A}| > 0 \wedge j \notin \mathcal{A}] \tag{4.56}$$

□

Unfortunately, the bound provided in Lemma 4.26 is not always tight. Indeed, the value of the last term in Eq. (4.56) can in some cases result in a quite high probability. However, we have not been able to prove a tight upper bound on P_{FN}.

4.3.2 The expected coalition score $\tilde{\mu}$

One very important parameter we are going to use later is $\tilde{\mu}$. This value has been introduced in Section 3.4 as the single-segment coalition expected score, or more formally $\tilde{\mu} = \mathbb{E}[S]/m$. A more precise formula of $\tilde{\mu}$ has been given in [42] with the following expression (for the case $q \geq 3$),

$$\tilde{\mu} = \sum_{\boldsymbol{\sigma}} \mathbb{P}(\boldsymbol{\sigma}) \sum_{y \in \mathcal{Q}} \theta_{y|\boldsymbol{\sigma}} W(\sigma_y) \left\{ \tfrac{1}{2} - \kappa + \frac{\sigma_y}{c}(\kappa q - 1) \right\} \tag{4.57}$$

$$W(b) := c \frac{\Gamma(b + \kappa - \frac{1}{2})}{\Gamma(b + \kappa)} \frac{\Gamma(c - b + \kappa[q-1] - \frac{1}{2})}{\Gamma(c - b + \kappa[q-1])}. \tag{4.58}$$

The colluders want to minimize $\tilde{\mu}$, while the content owner wants to maximize it.

This equation is clearly depending on the strategy chosen by the coalition, that in this case is defined as $\theta_{y|\boldsymbol{\sigma}}$. We want to rewrite (4.57) introducing instead the new variable K_b.

THEOREM 4.27. *The quantity $\tilde{\mu}$ as defined in (4.57) can be written as*

$$\tilde{\mu} = q \sum_{b=1}^{c} \mathbb{P}_1(b) K_b W(b) \left\{ \tfrac{1}{2} - \kappa + \frac{b}{c}(\kappa q - 1) \right\}. \tag{4.59}$$

PROOF. In (4.57) we shift the $\sum_y$ to the front and write $\mathbb{P}(\boldsymbol{\sigma}) = \Pr[\sigma_y = b] \Pr[\boldsymbol{\sigma}_{\setminus y} = \boldsymbol{x} | \sigma_y = b]$ and $\sum_{\boldsymbol{\sigma}} = \sum_b \sum_{\boldsymbol{x}}$. The $\sum_{\boldsymbol{x}}$ of $\theta_{y|\boldsymbol{\sigma}}$ yields K_b according to the definition (4.23). □

Corollary 4.28. *For $\kappa > \frac{1}{2(q-1)}$ the contribution of the $b = c$ term to $\tilde{\mu}$ vanishes in the limit of large c.*

Proof. In (4.59) we split off the $b = c$ term, which has $K_c = 1$ due to the marking conditon. After some rewriting of Gamma functions this yields

$$\tilde{\mu} = cq\frac{B(c+\kappa-\frac{1}{2},\kappa[q-1]+\frac{1}{2})}{B(\kappa,\kappa[q-1])}+q\sum_{b=1}^{c-1}\mathbb{P}_1(b)K_bW(b)\left\{\tfrac{1}{2}-\kappa+\frac{b}{c}(\kappa q-1)\right\}. \tag{4.60}$$

In the limit of large c, the first term scales as $(1/c)^{\kappa[q-1]-1/2}$. For $\kappa[q-1] > \frac{1}{2}$ this vanishes asymptotically. □

Corollary 4.28 tells us that in the relevant case $\kappa \approx 1/q$, the contributions to $\tilde{\mu}$ work completely different than in the usual binary scheme ($q = 2, \kappa = \frac{1}{2}$). There the $b = c$ term scales as c^0 and all the $b < c$ terms are zero.

4.3.3 Distribution function of a guilty-user's score

Throughout this section we will use the shorthand notation u for $S_j^{(i)}$. We define $\psi(u)$ the probability for a guilty user to obtain a score equal to u.We derive the distribution function $\psi(u)$ as follows. First we fix $\boldsymbol{p}$ and compute the conditional pdf $\psi(u|\boldsymbol{p})$. Then the end result follows by taking the expectation value over $\boldsymbol{p}$: $\psi(u) = \mathbb{E}_{\boldsymbol{p}}[\psi(u|\boldsymbol{p})]$. Because of the different behavior of positive and negative scores we introduce the notation ψ_+ for $u > 0$ and ψ_- for $u < 0$.

Theorem 4.29. *Let $T_{y|\boldsymbol{p}}$ and $\tau_{y|\boldsymbol{p}}$ be functions as defined in Section 4.1.1. For a guilty user, the probability distribution of the score conditioned on*

$\boldsymbol{p}$ is given by

$$u<0: \quad \psi_-(u|\boldsymbol{p}) = \sum_{y\in\mathcal{Q}} \delta\left(u-g_0(p_y)\right) \sum_{\boldsymbol{\sigma}} \binom{c}{\boldsymbol{\sigma}} \left(1-\frac{\sigma_y}{c}\right) \boldsymbol{p}^{\boldsymbol{\sigma}} \theta_{y|\boldsymbol{\sigma}} \tag{4.61}$$

$$= \sum_{y\in\mathcal{Q}} \delta\left(u-g_0(p_y)\right) \left[\tau_{y|\boldsymbol{p}} - \frac{p_y}{c}\frac{\partial T_{y|\boldsymbol{p}}}{\partial p_y}\right], \tag{4.62}$$

$$u>0: \quad \psi_+(u|\boldsymbol{p}) = \sum_{y\in\mathcal{Q}} \delta\left(u-g_1(p_y)\right) \sum_{\boldsymbol{\sigma}} \binom{c}{\boldsymbol{\sigma}} \frac{\sigma_y}{c} \boldsymbol{p}^{\boldsymbol{\sigma}} \theta_{y|\boldsymbol{\sigma}} \tag{4.63}$$

$$= \frac{1}{c} \sum_{y\in\mathcal{Q}} \delta\left(u-g_1(p_y)\right) p_y \frac{\partial T_{y|\boldsymbol{p}}}{\partial p_y}. \tag{4.64}$$

PROOF. See Appendix E. □

THEOREM 4.30. *For a guilty user, the distribution function ψ of the score in one segment is given by*

$$u<0: \quad \psi_-(u) = \frac{2q}{B(\kappa,\kappa[q-1])} \sum_{b=1}^{c-1} \left(1-\frac{b}{c}\right) \binom{c}{b} \frac{(u^2)^{b+\kappa-\frac{1}{2}}}{(1+u^2)^{c+\kappa q}} K_b, \tag{4.65}$$

$$u>0: \quad \psi_+(u) = \frac{2q}{B(\kappa,\kappa[q-1])} \sum_{b=1}^{c} \frac{b}{c} \binom{c}{b} \frac{(u^2)^{c-b+\kappa[q-1]-\frac{1}{2}}}{(1+u^2)^{c+\kappa q}} K_b. \tag{4.66}$$

PROOF. See Appendix F. □

The expressions (4.65, 4.66) are rather complicated. We have double-checked their correctness by verifying the normalization and the first moment.

CONSISTENCY CHECK 1. The function $\psi(u)$ given in Theorem 4.30 is correctly normalized, $\int_{-\infty}^{\infty} \mathrm{d}u\, \psi(u) = 1$.

PROOF. See Appendix G. □

CONSISTENCY CHECK 2. The function $\psi(u)$ has the correct first moment, $\int_{-\infty}^{\infty} \mathrm{d}u\, \psi(u) u = \tilde{\mu}/c$.

PROOF. See Appendix H. □

Left tail	Right tail	$u \uparrow 0$	$u \downarrow 0$
$K_{c-1}\lvert u\rvert^{-3-2\kappa[q-1]}$	$K_1 u^{-3-2\kappa}$	$K_1(c-1)\lvert u\rvert^{1+2\kappa}$	$u^{-1+2\kappa[q-1]}$

Table 4.2: *Dominant powers of $\psi(u)$ in the tails and near $u = 0$. All the values above are multiplied by $\frac{2q}{B(\kappa,\kappa[q-1])}$.*

The behavior in the tails and near $u = 0$ is summarized in Table 4.2. The right tail is dominated by the $b = 1$ term; it is proportional to $(1/u)^{3+2\kappa}$. The integral $\int_0^\infty \mathrm{d}u\ \psi_+(u)u^a$ converges for $a < 2 + 2\kappa$. The left tail is dominated by the $b = c - 1$ term, and is proportional to $(1/|u|)^{3+2\kappa[q-1]}$. The integral $\int_{-\infty}^0 \mathrm{d}u\ \psi_-(u)|u|^a$ converges for $a < 2 + 2\kappa[q-1]$. Hence, for $\kappa \in \left(0, \frac{1}{2}\right)$, the usual choice, the second moment always exists, but not the third absolute moment. We see that the right tail is heavier than the left tail, meaning that extreme positive scores are more likely than extreme negative scores. Such a property is obviously beneficial for accusing guilty users.

Definition 4.3. *We denote the second moment of the pdf ψ as M_2,*

$$M_2 := \int_{-\infty}^{\infty} \mathrm{d}u\ \psi(u)u^2. \tag{4.67}$$

Definition 4.4. *We denote the variance of the pdf ψ as V,*

$$V := M_2 - \tilde{\mu}^2/c^2. \tag{4.68}$$

Lemma 4.31. *The second moment M_2 as defined in Def. 4.3 is given by*

$$M_2 = q \sum_{b=1}^{c} K_b \mathbb{P}_1(b) \left[\left(1 - \frac{b}{c}\right) \frac{b+\kappa}{c-b+\kappa[q-1]-1} + \frac{b}{c} \frac{c-b+\kappa[q-1]}{b+\kappa-1} \right]. \tag{4.69}$$

Proof. See Appendix I. □

Remark 4.32. The scores of guilty users are not independent. As a consequence, the variance of the coalition score $S_{\mathcal{C}}^{(i)}$ is not a simple

multiple of V. Let the covariance between two guilty user scores be $K_{jj'} = \mathbb{E}[S_j^{(i)} S_{j'}^{(i)}] - \tilde{\mu}^2/c^2$. Then we have

$$\begin{aligned}\mathbb{E}\left[(S_{\mathcal{C}}^{(i)})^2\right] &= \mathbb{E}\left[\sum_{j,j'\in\mathcal{C}} S_j^{(i)} S_{j'}^{(i)}\right] \\ &= c\mathbb{E}\left[(S_j^{(i)})^2\right] + \sum_{j\neq j'}\left(\mathbb{E}\left[S_j^{(i)} S_{j'}^{(i)}\right] - \tilde{\mu}^2/c^2\right) + \tilde{\mu}^2\left(1-\frac{1}{c}\right).\end{aligned} \tag{4.70}$$

which yields

$$\tilde{\sigma}^2 := \mathrm{Var}(S_{\mathcal{C}}^{(i)}) = cV + \textstyle\sum_{j\neq j'} K_{jj'}. \tag{4.71}$$

In [42] the variance was bounded as $\tilde{\sigma}^2 < qc - \tilde{\mu}^2$. From this bound we learn that the sum $\sum_{j\neq j'} K_{jj'}$ scales at most linearly in c, even though it contains two sums over the coalition. A study of the covariances is left for future work.

4.3.4 Fourier transform of ψ

THEOREM 4.33. *The Fourier transform of ψ is given by*

$$\tilde{\psi}(k) = \frac{2q}{B(\kappa,\kappa[q-1])}\sum_{b=1}^{c}\binom{c}{b} K_b \cdot \left[\left(1-\frac{b}{c}\right)\Lambda(d_b', v_b'; k) + \frac{b}{c}\Lambda(D_b', V_b'; -k)\right], \tag{4.72}$$

with Λ as defined in Lemma 4.23, and

$$\begin{aligned} d_b' &= b+\kappa \quad ; \quad v_b' = c-b+\kappa[q-1] \\ D_b' &= c-b+\kappa[q-1] \quad ; \quad V_b' = b+\kappa. \end{aligned} \tag{4.73}$$

PROOF. We use the expression for ψ given in Theorem 4.30. The Fourier integral for the summands in ψ_+ is immediately of the form (4.49) and yields $\Lambda(D_b', V_b'; -k)$. The integral over the ψ_- terms is of the form $\int_{-\infty}^{0} \mathrm{d}u\ f(u^2)e^{-iku}$, which can be rewritten as $\int_0^{\infty} \mathrm{d}u\ f(u^2)e^{iku}$; this too has the form (4.49) and yields $\Lambda(d_b', v_b'; k)$. □

COROLLARY 4.34. *For $q \geq 3$ and $\frac{1}{2(q-1)} \leq \kappa < \frac{1}{2}$, the $\tilde{\psi}$ has the following power series expansion,*

$$\tilde{\psi}(k) = 1 - i\frac{\tilde{\mu}}{c}k - \tfrac{1}{2}M_2 k^2 + A(-ik)^{2+2\kappa} + O(k^3), \tag{4.74}$$

$$\text{where } A := \frac{2q}{B(\kappa,\kappa[q-1])} K_1 \Gamma(-2-2\kappa). \tag{4.75}$$

Proof. Trivially $\mathbb{E}[u^0] = 1$. From Consistency check 2 and Lemma 4.31 we know that $\mathbb{E}[u] = \frac{\tilde{\mu}}{c}$ and $\mathbb{E}[u^2] = M_2$. Hence by Lemma 4.16 we have $\tilde{\psi}(0) = 1$, $\tilde{\psi}'(0) = -i\frac{\tilde{\mu}}{c}$ and $\tilde{\psi}''(0) = -M_2$. The expansion in (4.74) is consistent with these values. After k^2 the powers can be non-integer. The next term in the series expansion is $k^{2+2\kappa}$. The exponent comes from the application of Lemma 4.23 in Theorem 4.33: in the first term of (4.49) the k^{2v} factor can build irrational powers of k. The minimum value generated is for $V_1' = 1+\kappa$, with V_b' as defined in (4.73). Note that the Λ term obtained from v_c' is not present because it is multiplied by $1 - b/c$. The next contribution is $v_{c-1}' = 1 + \kappa[q-1]$ which (for $q \geq 3$) is larger than V_1'. Finally, the coefficient A follows from the $\Lambda(D_1', V_1', -k)$ term in (4.72), taking only the leading term (=1) in the sum representation of the ${}_1F_2$ function. □

In order to apply the CSE method we will have to work with a zero-mean pdf. For this reason we introduce a 'centered' version of ψ.

Definition 4.5. *We define the pdf χ as a shifted version of ψ,*

$$\chi(r) := \psi\left(\frac{\tilde{\mu}}{c} + r\right). \tag{4.76}$$

We will use shorthand notation $r = u - \tilde{\mu}/c$. From the definition it trivially follows that $\mathbb{E}[r] = 0$ and $\mathbb{E}[r^2] = V$.

Lemma 4.35. *The Fourier transform of χ is given by*

$$\tilde{\chi}(k) = e^{ik\frac{\tilde{\mu}}{c}}\tilde{\psi}(k). \tag{4.77}$$

Proof. $\tilde{\chi}(k) = \int_{-\infty}^{\infty} \mathrm{d}r\ e^{-ikr}\chi(r) = \int_{-\infty}^{\infty} \mathrm{d}u\ e^{-ik\left(u-\frac{\tilde{\mu}}{c}\right)}\psi(u) = e^{ik\frac{\tilde{\mu}}{c}}\tilde{\psi}(k)$. □

Corollary 4.36. *Let $\frac{1}{2[q-1]} < \kappa < \frac{1}{2}$ and let χ be as given in Definition 4.5. Then $\tilde{\chi}$ has the following power series expansion,*

$$\tilde{\chi}(k) = 1 - \tfrac{1}{2}Vk^2 + A(-ik)^{2+2\kappa} + O(k^3) \tag{4.78}$$

with A as given in (P.2).

Proof. From (4.77) we can rewrite $\tilde{\chi}(k)$ as a product of the series expansions of $e^{ik\frac{\tilde{\mu}}{c}}$ and $\tilde{\psi}(k)$. Since $e^{ik\frac{\tilde{\mu}}{c}} = 1 + i\frac{\tilde{\mu}}{c}k - \frac{1}{2}\frac{\tilde{\mu}^2}{c^2}k^2 + O(k^3)$, and

the $\tilde{\psi}(k)$ expansion was given in (4.74), we have

$$e^{ik\frac{\tilde{\mu}}{c}}\tilde{\psi}(k) = \left[1 + i\frac{\tilde{\mu}}{c}k - \tfrac{1}{2}\frac{\tilde{\mu}^2}{c^2}k^2 + O(k^3)\right] \cdot \left[1 - i\frac{\tilde{\mu}}{c}k - \tfrac{1}{2}M_2k^2 + A(-ik)^{2+2\kappa} + O(k^3)\right] \tag{4.79}$$

$$= 1 + 0k + \left(-\frac{M_2}{2} - \frac{\tilde{\mu}^2}{2c^2} + \frac{\tilde{\mu}^2}{c^2}\right)k^2 + A(-ik)^{2+2\kappa} + O(k^3), \tag{4.80}$$

and (4.78) follows after some simplification. □

REMARK 4.37. The $1 - \frac{1}{2}Vk^2$ part of (4.78) can be also found using Lemma 4.16, since we know that $\mathbb{E}[r] = 0$ and $\mathbb{E}[r^2] = V$.

In the expression (4.78) there are no powers between k^0 and k^2. This makes it possible for us to use the CSE method.

4.3.5 C_m definition: application of the CSE method to the guilty user score

We are now finally in a position to compute accusation probabilities for guilty users. The Fourier transform $\tilde{\chi}$ serves as the basis; raising it to the power m yields the Fourier-transformed pdf of the total accusation S_j. The computational steps are almost identical to the case of the innocent score distribution [33], with two minor differences:

1. The variance of the single-segment pdf is V instead of 1;
2. The pdf has non-zero average.

Below we list the (slight) modifications in the CSE method, as compared to the innocent case, induced by the $V \neq 1$ variance and the nonzero mean. First, the tail of the Gaussian distribution changes.

LEMMA 4.38. *Let $V > 0$ be the variance defined in (4.68). Then, for $x \in \mathbb{R}$ it holds that*

$$\frac{1}{2\pi i}\int_{-\infty}^{\infty} \mathrm{d}k\, \frac{e^{ikx}}{k} e^{-\frac{V}{2}k^2} = \tfrac{1}{2} - \Omega(x/\sqrt{V}). \tag{4.81}$$

Proof. From Eq. 9.254.1 in [9] we have that $\frac{1}{2\pi i}\int_{-\infty}^{\infty}\mathrm{d}k\ \frac{e^{ikx}}{k}e^{-k^2/2} = \frac{1}{2} - \Omega(x)$. Changing the integration variable in (4.81) to $k' = k\sqrt{V}$ immediately yields the result. □

The modified Gaussian tail leads to modifications in all the integrals involving the tail.

Lemma 4.39. *Let $V > 0$ be the variance defined in (4.68). For $x \in \mathbb{R}$ and $\nu > 0$ it holds that*

$$\int_{-\infty}^{\infty}\frac{\mathrm{d}k}{2\pi}(i\,\mathrm{sgn}\,k)^{\alpha-1}|k|^{\nu-1}e^{-\frac{V}{2}k^2}e^{ikx} = \frac{1}{\pi V^{\frac{\nu}{2}}}\Gamma(\nu)2^{\nu/2}\ \mathrm{Im}\left[i^{-\alpha}H_{-\nu}\left(\frac{ix}{\sqrt{2V}}\right)\right]. \tag{4.82}$$

Proof. Corollary 2 in [33] states that for $x \in \mathbb{R}$ and $\nu > 0$:

$$\int_{-\infty}^{\infty}\frac{\mathrm{d}k}{2\pi}(i\,\mathrm{sgn}\,k)^{\alpha-1}|k|^{\nu-1}e^{-k^2/2}e^{ikx} = \frac{1}{\pi}\Gamma(\nu)2^{\nu/2}\ \mathrm{Im}\left[i^{-\alpha}H_{-\nu}\left(\frac{ix}{\sqrt{2}}\right)\right]. \tag{4.83}$$

A change of integration variable to $k\sqrt{V}$ in (4.82) directly leads to the end result. □

The nonzero expectation value $\mathbb{E}[S_j] = m\tilde{\mu}/c$ gives rise to a 'shifted' version of the formula for the accusation probability. We introduce a shifted accusation threshold Δ,

$$\Delta := Z - m\tilde{\mu}/c \qquad ; \qquad \tilde{\Delta} := \Delta/\sqrt{m}. \tag{4.84}$$

The accusation probability can be expressed as a function of $\tilde{\Delta}$, as shown in the following two theorems.

Theorem 4.40. *Let j be a guilty user. Let C_m denote the accusation probability* $\Pr[S_j > Z]$. *Then*

$$C_m(\tilde{\Delta}) = \frac{1}{2} + \frac{i}{2\pi}\int_{-\infty}^{\infty}\mathrm{d}k\ \frac{\exp(ik\tilde{\Delta})}{k}\left[\tilde{\chi}\left(\frac{k}{\sqrt{m}}\right)\right]^m. \tag{4.85}$$

Proof. Exactly the same as the proof of Theorem 4.18, but with $\tilde{\Delta}$ replacing $\tilde{Z}$. □

THEOREM 4.41. *Let j be a guilty user and $\frac{1}{2[q-1]} < \kappa < \frac{1}{2}$. Then it is possible to write*

$$\left[\tilde{\chi}\left(\frac{k}{\sqrt{m}}\right)\right]^m = \exp(-\tfrac{1}{2}Vk^2)\left[1+\sum_{t=0}^{\infty}\omega_t(m)(i\operatorname{sgn}k)^{\alpha_t}|k|^{\nu_t}\right] \quad (4.86)$$

where α_t are real numbers; the coefficients $\omega_t(m)$ are real; the powers ν_t satisfy $\nu_0 = 2+2\kappa$ and $\nu_{t+1} > \nu_t$. The ν_t are not necessarily integer. All the coefficients $\omega_t(m)$ are decreasing functions of m. The probability of accusing user j is given by

$$C_m(\tilde{\Delta}) = \Omega(\tilde{\Delta}/\sqrt{V}) + \tfrac{1}{\pi}\textstyle\sum_{t=0}^{\infty}\omega_t(m)\Gamma(\nu_t)(2/V)^{\nu_t/2}\operatorname{Im}\left[i^{-\alpha_t}H_{-\nu_t}(i\tilde{\Delta}/\sqrt{2V})\right]. \quad (4.87)$$

PROOF. See Appendix J. □

Numerics obtained from the application of the CSE method are present in Chapter 6.

4.4 Mixed strategies

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

$$[\tilde{\varphi}(k/\sqrt{m})]^m \to \prod_{s\in\text{strategies}}[\tilde{\varphi}_s(k/\sqrt{m})]^{m_s} \quad (4.88)$$

and then follow all the derivation steps as before. The same procedure can be applied on the ψ function.

4.5 Research question, revisited

With the technical background of Chapters 2 and 3, it is now possible to formulate our research question in more technical terms:

What are the error rates of the q-ary Tardos scheme [42] in the Restricted Digit Model?

We restrict ourselves to the RDM because it lends itself to analysis.

In addition to the questions raised in Section 1.6, another important aspect of Tardos codes studied in this work is the convergence to Gaussian distributions. We investigate under which circumstances the Gaussian assumption is justified.

5

Strategy classification and K_b computation

The attack chosen by the colluders has two different targets, both that aim to make worthless the tracing scheme: to not be accused (keeping the attacker's scores low) and to make accused many innocent users (trying to raise the innocent's score). The motivation in the first case is obvious. In the second, instead, if many innocent users get accused, the distributor can be blamed for being untrustworthy, with the consequence that his traitor tracing scheme is considered unreliable.

As specified in Section 3.2, the attack is assumed to have three symmetries:

1. Symbol symmetry,
2. Segment symmetry,
3. Attacker symmetry.

The possible strategies that attackers may use are infinite. However, just few of them have an interesting effect on the classic Tardos' scheme. The explanation is given by Figure 3.1 that shows the error probabilities. The two areas are the attackers' targets and they want to raise them as much as possible. To convert this task into a strategy is not trivial at all.

The P_{FP} and P_{FN} depend obviously on many parameters (q, m, κ, c and the strategy) and the applied strategy is just one of them. The combination of the parameters creates several different scenarios, making unattainable to proclaim a particular strategy "the best one".

Back to Figure 3.1, the P_{FP} corresponds to the innocent right tail area from the point $Z/\sqrt{m}$ (now known also as $\tilde{Z}$) on. Supposing that $\tilde{Z}$ has been fixed to a specific value , to raise P_{FP}, there are two ways: to shift the entire innocent curve to the right or to make its right tail heavier. Unluckily for the guilty coalitions, the first option is unfeasible for a simple reason: in (3.13) is shown that the one-segment average innocent score is zero independently from the strategy. Obviously, this value holds also when we consider the total innocent score. As consequence, the innocent curve cannot be shifted in any direction. That leaves the second option. For the Central Limit Theorem, when $m \to \infty$ the innocent curve shape converges exactly to a Gaussian and this shape cannot be altered. In reality, being $m < \infty$ there will be always a portion of the right innocent tail that will follow a non-Gaussian behaviour, and in this section is possible, using the right strategy, to tweak the tail heaviness. The lower the m, the longer the non-Gaussian part.

For the P_{FN} can be applied the same reasoning done for P_{FP}. To increase its value, the coalition can try to shift the guilty curve to the left or to raise the guilty left tail. This time the first option is totally feasible, as we can see in Theorem 4.27 where is shown how $\tilde{\mu}$ depends on K_b. A strategy that can reduce $\tilde{\mu}$ value will automatically shift the entire curve to the left. The second option, instead, is not a good target for the colluders as we are going to show in the following lines. When m increases, the guilty curve becomes more Gaussian. Hence, for large m, the strategy has less effect on the shape of the tails, and the best attack is to reduce $\tilde{\mu}$. At "small" m, there is a complicated tradeoff between stretching the innocent right tail and reducing $\tilde{\mu}$.

The structure of this chapter is the following: in Section 5.1 we define formally the strategies we have investigated; in Section 5.2 we introduce a classification for the attacks and we compute K_b for the investigates strategies; in Section 5.3 we present some analytic results.

5.1 Strategy definitions

In our research the study of the q-ary Tardos' scheme has been done focusing on the P_{FP} and P_{FN} behaviour under the changes of all the parameters involved. From this point of view, we were searching the strategies that could behave better to accomplish the task of increasing both these probabilities with a simple decoder approach. It is important

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 Section 5.1.5)	
Random Symbol	RS	Choose uniformly from received symbols	$\frac{[\sigma_y>0]}{\lvert\{\alpha\in\mathcal{Q}:\sigma_\alpha>0\}\rvert}$

Table 5.1: List of the strategies investigated with brief description.

to remark that this is not the only way to judge the efficiency of an attack. For example, from an information-theoretical point of view, the best strategy for the coalition is the one that minimizes the fingerprinting capacity, while the tracer sets up the scheme trying to maximise it. In [13] it has been found that the optimal choices are the interleaving attack for the attackers and the Dirichlet distribution with $\kappa = \frac{1}{2}$ for the tracer. Unfortunately, in [13] it was not present an optimal score system to use.

The strategies we are going to study are five: majority voting, minority voting, interleaving attack, random symbol attack, and $\tilde{\mu}$-minimizing attack. We start providing their definitions, arriving to their representation $\Psi_b(\boldsymbol{x})$. This will prepare the way to computing the K_b given in Section 5.2.2. For the sake of simplicity, from now on we are going to use the abbreviations defined in Table 5.1 to refer to the various strategies. There is also an extremely brief description of the strategies and, when simple to do, its $\theta_{y|\boldsymbol{\sigma}}$. These abbreviations will be also used as labels on $\theta_{y|\boldsymbol{\sigma}}$, $\Psi_b(\boldsymbol{x})$ and K_b.

5.1.1 Majority voting

Definition 5.1 (Majority Voting). *The MajV attack selects the symbol that highest tally in the segment. In case of two or more symbols occur the most, one of them will be randomly drawn. Formally, its $\theta_{y|\boldsymbol{\sigma}}$ definition*

is:

$$\theta^{\mathrm{MajV}}_{y|\boldsymbol{\sigma}} = \begin{cases} \frac{1}{a} & \text{if } \sigma_y = \max\limits_{\beta\in\mathcal{Q}} \sigma_\beta \\ 0 & \text{otherwise} \end{cases}, \tag{5.1}$$

where $a = |\{\alpha \in \mathcal{Q} : \sigma_\alpha = \max_{\beta\in\mathcal{Q}} \sigma_\beta\}|$.

LEMMA 5.1. *For MajV,* $\Psi_b(\boldsymbol{x})$ *parametrization is:*

$$\Psi^{\mathrm{MajV}}_b(\boldsymbol{x}) = \begin{cases} \frac{1}{\ell+1} & \text{if } b = \max\limits_{1\le i\le q-1} x_i \\ 0 & \text{otherwise} \end{cases}, \tag{5.2}$$

where $\ell = |\{i : x_i = b\}|$

Intuitively, to output the most frequent symbol has the side effect of giving a high coalition score because many attackers will have a positive match between the chosen symbol and their received symbol. As consequence, the score function g_1 will be applied more often. On the other hand, it is likely that also many innocent people have received the same symbol (especially when c is big), and then also their score will be higher. Furthermore, if the symbol in question is often present as consequence of a high probability in the generating $\boldsymbol{p}$, then the score gained $g_1(p_y)$ will not be very big and then not much damaging for the coalition.

5.1.2 Minority voting

DEFINITION 5.2 (Minority Voting). *The MinV attack selects the symbol with the smallest non-zero tally in the segment. In case of two or more symbols occur the least, one of them will be randomly drawn. Formally, its* $\theta_{y|\boldsymbol{\sigma}}$ *definition is:*

$$\theta^{\mathrm{MinV}}_{y|\boldsymbol{\sigma}} = \begin{cases} \frac{1}{a} & \text{if } \sigma_y = \min\limits_{\substack{\beta\in\mathcal{Q}:\\ \sigma_\beta>0}} \sigma_\beta \\ 0 & \text{otherwise} \end{cases}, \tag{5.3}$$

where $a = |\{\alpha \in \mathcal{Q} : \sigma_\alpha = \min_{\beta\in\mathcal{Q}} \sigma_\beta\}|$.

LEMMA 5.2. *For MajV,* $\Psi_b(\boldsymbol{x})$ *parametrization is:*

$$\Psi^{\mathrm{MinV}}_b(\boldsymbol{x}) = \begin{cases} \frac{1}{\ell+1} & \text{if } b = \min\limits_{\substack{1\le i\le q-1\\ x_i>0}} x_i \\ 0 & \text{otherwise} \end{cases}, \tag{5.4}$$

where $\ell = |\{i : x_i = b\}|$

Choosing the least frequent symbol, the colluders reduce their score by having as few matches with the chosen symbol as possible, collecting more negative scores. Many innocent users will also have a negative score. On the other hand, any match will cause a high positive score when the p_y is low.

5.1.3 Interleaving attack

Definition 5.3 (Interleaving attack). *The Int attack output the symbol of a randomly drawn attacker. Formally, its $\theta_{y|\boldsymbol{\sigma}}$ definition is:*

$$\theta^{\text{Int}}_{y|\boldsymbol{\sigma}} = \frac{\sigma_y}{c} \tag{5.5}$$

Lemma 5.3. *For Int, $\Psi_b(\boldsymbol{x})$ parametrization is:*

$$\Psi^{\text{Int}}_b(\boldsymbol{x}) = \frac{b}{c} \tag{5.6}$$

The 'interleaving' colluder strategy, which is known to be information-theoretically optimal [11, 13] for $c_0 \to \infty$, turns out to have special properties: the pdf and $\tilde{\mu}$ do not depend on the coalition size (see Section 5.3.2); the left and right tail are maximally heavy (see Tables 4.1 and 4.2).

This strategy has an important difference from MajV and MinV: its has an extra probabilistic step thank to which it is not possible to know in advance which symbol will be chosen even knowing $\boldsymbol{\sigma}$. This does not happen for MajV and MinV, where in many cases $\Psi_b(\boldsymbol{x})$ is populated just by 0s and 1s.

It is important to notice that, for large c, $\theta^{\text{Int}}_{\alpha|\boldsymbol{\sigma}} \approx p_\alpha$, meaning that big coalitions look like innocent users.

5.1.4 Random symbol attack

Definition 5.4 (Random symbol attack). *The RS attack selects uniformly one of the symbols detected by the coalition. Formally, its $\theta_{y|\boldsymbol{\sigma}}$ definition is:*

$$\theta^{\text{RS}}_{y|\boldsymbol{\sigma}} = \begin{cases} \frac{1}{a'} & \text{if } \sigma_y > 0 \\ 0 & \text{otherwise} \end{cases} \tag{5.7}$$

where $a' = |\{\alpha \in \mathcal{Q} : \sigma_\alpha > 0\}|$.

LEMMA 5.4. *For RS,* $\Psi_b(\boldsymbol{x})$ *parametrization is:*

$$\Psi_b^{\mathrm{RS}}(\boldsymbol{x}) = \begin{cases} \frac{1}{w+1} & \text{if } b > 0 \\ 0 & \text{otherwise} \end{cases}, \tag{5.8}$$

where $w = |\{i : x_i > 0\}|$

5.1.5 $\tilde{\mu}$-minimizing attack

DEFINITION 5.5 ($\tilde{\mu}$-minimizing attack). *The* $\tilde{\mu}$*-min attack chooses the symbol such that the expected single-segment coalition score* $\tilde{\mu}$ *is minimized.*

This is the strongest attack in the Gaussian regime. It is very easy to describe $\tilde{\mu}$-min in words, while it is difficult to do it as $\theta_{y|\boldsymbol{\sigma}}$ or $\Psi_b(\boldsymbol{x})$. The symbol chosen depends on all the parameters due to $\tilde{\mu}$ definition (see (4.59)). However, we will show later how is instead possible to give its definition in K_b form, that is the one we will really need to apply the CSE method.

Asymptotically for large code lengths the colluder strategy has negligible impact on the Gaussian shape of the innocent (and guilty) accusation pdf. For $q \geq 3$ the main impact of their strategy is on the value of the statistical parameter $\tilde{\mu}$. (For the binary symmetric scheme with $\kappa = \frac{1}{2}$, the $\tilde{\mu}$ is fixed at $\frac{2}{\pi}$; the attackers cannot change it. Then the strategy's impact on the pdf shape is *not* negligible.)

Hence for $q \geq 3$ the strategy that minimizes $\tilde{\mu}$ is asymptotically a 'worst-case' attack in the sense of maximizing the false positive probability. This was already argued in [42], and it was shown how the attackers can minimize $\tilde{\mu}$.

From (4.59) let be

$$T(b) = W(b)\left\{\tfrac{1}{2} - \kappa + \frac{b}{c}(\kappa q - 1)\right\}. \tag{5.9}$$

It is evident that, for a given $\boldsymbol{\sigma}$, the attackers must choose the symbol y such that $T(\sigma_y)$ is minimal[1], i.e. $y = \arg\min_\alpha T(\sigma_\alpha)$. In case of a tie it does not matter which of the best symbols is chosen, i.e. if the minimum $T(\sigma_\alpha)$ is shared by N different symbols, then each of these symbols will have probability $1/N$ of being elected.

[1]Notice that $\mathbb{P}_1(b)$ is unknown to the coalition

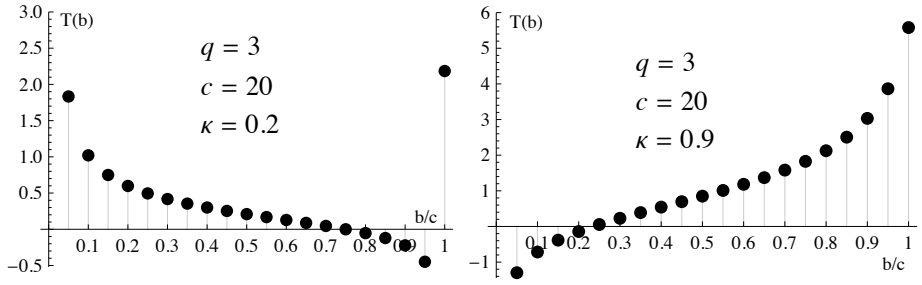


FIGURE 5.1: *The function $T(b)$ for $q = 3$, $c = 20$ and two values κ outside $(\frac{1}{2[q-1]}, \frac{1}{2})$.*

Let us introduce the notation $x = b/c$, $x \in (0, 1)$. Then for large c we have [33]

$$T(cx) \approx \frac{\frac{1}{2} - \kappa + x(\kappa q - 1)}{\sqrt{x(1-x)}}. \tag{5.10}$$

From (5.10) we deduce some elementary properties of the function T.

- If $\kappa < \frac{1}{2(q-1)}$ then T is monotonically decreasing, and $T(b)$ may become negative at large b.
- If $\kappa > \frac{1}{2}$, then T is monotonically increasing, and $T(b)$ may become negative at small b.
- For κ in between those values, $T(b)$ is nonnegative and has a minimum at $\frac{b}{c} \approx \frac{1}{q-2}(\frac{1}{2\kappa} - 1)$.

We expect that the existence of negative $T(b)$ values has a very bad impact on $\tilde{\mu}$ (from the accuser's point of view), and hence that κ is best chosen in the interval $(\frac{1}{2(q-1)}, \frac{1}{2})$.

Fig. 5.1 shows the function $T(b)$ for two values of κ outside this 'safe' interval. For $\kappa = 0.2$ it is indeed the case that $T(b) < 0$ at large b, and for $\kappa = 0.9$ at small b. Note that $T(c)$ is always positive due to the Marking Assumption. For small κ, the $T(b)$-ranking of the points is clearly such that majority voting is the best strategy; similarly, for large κ minority voting is best. For intermediate values of κ a more complicated ranking will occur.

These are the five strategies we are going to investigate. In the next section we are going to compute their K_bs. This is the last ingredient necessary to compute the error curves.

5.2 Strategy classifications and K_b precomputation

The K_b pure definition given in (4.23) can require an enormous amount of time to be computed. The full sum over $\boldsymbol{x}$ can make unpractical to compute K_b for some strategies. Luckily, many attacks can drastically simplify K_b formula.

One of our contribution consists in defining some $\Psi_b(\boldsymbol{x})$ strategy classes for which the computation of K_b becomes quicker. In the next section we are going to define these classes, describe the achieved speed-up and show in which classes the five strategies defined in Section 5.1 fit.

5.2.1 Strategy classes

The strategy classification we are going to provide is a prescription for efficiently computing the K_b parameters for more general colluder strategies than those studied in [33]. We consider the strategy parametrization $\Psi_b(\boldsymbol{x})$ with $b \neq 0$. The vector $\boldsymbol{x} \in \mathbb{N}^{q-1}$ can contain several entries equal to b. The number of such entries will be denoted as ℓ. (The dependence of ℓ on b and $\boldsymbol{x}$ is suppressed in the notation for the sake of brevity.) The number of remaining entries is $r \triangleq q - 1 - \ell$. These entries will be denoted as $\boldsymbol{z} = (z_1, \cdots, z_r)$, with $z_j \neq b$ by definition. Many symmetric strategies can be parameterized as a function $\Psi_b(\boldsymbol{x})$ which in turn can be expressed as a function of b, ℓ and $\boldsymbol{z}$; it is invariant under permutation of the entries in $\boldsymbol{z}$. We will concentrate on the following 'factorizable' classes of attack, each one a sub-class of the previous one.

Class 1: $\Psi_b(\boldsymbol{x})$ is of the form $w(b,\ell)\prod_{k=1}^{r} W(b,\ell,z_k)$

Class 2: $\Psi_b(\boldsymbol{x})$ is of the form $\frac{w(b)}{\ell+1}\prod_{k=1}^{r} W(b,z_k)$

Class 3: $\Psi_b(\boldsymbol{x})$ is of the form $\frac{1}{\ell+1}\prod_{k=1}^{r} W(b,z_k)$, with $W(b,z_k) \in \{0,1\}$ and $W(b,z_k) + W(z_k,b) = 1$. By definition $W(b,0) = 1$.

Class 1 merely restricts the dependence on $\boldsymbol{z}$ to a form factorizable in the components z_k. This is a very broad class, and contains e.g. the Int attack ($\theta_{\alpha|\boldsymbol{\sigma}} = \frac{\sigma_\alpha}{c}$, $\Psi_b(\boldsymbol{x}) = \frac{b}{c}$) which has no dependence on $\boldsymbol{z}$.

Class 2 puts a further restriction on the ℓ-dependence. The factor $1/(\ell+1)$ implies that symbols with equal occurrence have equal probability of being selected by the colluders. (There are $\ell+1$ symbols that occur b times.)

Class 3 restricts the function W to a binary 'comparison' of its two arguments: $\Psi_b(\boldsymbol{x})$ is nonzero only if for the attackers b is 'better' than z_k for all k, i.e. $W(b, z_k) = 1$. An example of such a strategy is MajV, where $\Psi_b(\boldsymbol{x}) = 0$ if there exists a k such that $z_k > b$, and $\Psi_b(\boldsymbol{x}) = \frac{1}{\ell+1}$ if $z_k < b$ for all k. Class 3 also contains MinV, and in fact any strategy which uses a strict ordering or 'ranking' of the tallies b, z_k. (Here a zero always counts as 'worse' than nonzero.)

Our motivation for introducing classes 1 and 2 is mainly technical, since they affect to which extent the K_b parameters can be computed analytically.

Theorem 5.5. *Let $N_b \in \mathbb{N}$ satisfy $N_b > \max\{c-b, bq-c, (c-b)(q-2)\}$. Let $\tau_b \triangleq e^{i2\pi/N_b}$, and let*

$$G_{ba\ell} \triangleq \sum_{z\in\{0,\ldots,c-b\}\setminus\{b\}} \frac{\Gamma(\kappa+z)W(b,\ell,z)}{\tau_b^{az} z!}, \qquad v_{ba} \triangleq \frac{\Gamma(\kappa+b)}{\tau_b^{ab} b!}. \quad (5.11)$$

Then for strategies in class 1 it holds that

$$K_b = \frac{(c-b)!}{N_b\Gamma(c-b+\kappa[q-1])B(\kappa\mathbf{1}_{q-1})} \cdot \sum_{a=0}^{N_b-1} \tau_b^{a(c-b)} \sum_{\ell=0}^{q-1} \binom{q-1}{\ell} G_{ba\ell}^{q-1-\ell} w(b,\ell) v_{ba}^{\ell}. \quad (5.12)$$

Proof. See Appendix K. □

Theorem 5.6. *For strategies in class 2 the quantity $G_{ba\ell}$ as defined in (5.11) does not depend on ℓ and can be denoted as G_{ba} (with $W(b,\ell,z)$ replaced by $W(b,z)$). It then holds that*

$$K_b = \frac{b!(c-b)!\; w(b)}{qN_b\Gamma(\kappa+b)\Gamma(c-b+\kappa[q-1])B(\kappa\mathbf{1}_{q-1})} \sum_{a=0}^{N_b-1} \tau_b^{ac} \left[(G_{ba}+v_{ba})^q - G_{ba}^q\right]. \quad (5.13)$$

Proof. See Appendix L. □

THEOREM 5.7. *For strategies in class 3, Theorem 5.6 holds, where* $w(b) = 1$ *and* G_{ba} *can be expressed as*

$$G_{ba} = \sum_{\substack{z\in\{0,\ldots,c-b\}\setminus\{b\} \\ W(b,z)=1}} \frac{\Gamma(\kappa+z)}{\tau_b^{az} z!}. \tag{5.14}$$

PROOF. See Appendix M. □

Note that also $\tilde{\mu}$-min fits in class 3. The function $W(b, z_k)$ evaluates to 1 if $T(b) < T(z_k)$ and to 0 otherwise.[2]

Without these theorems, straightforward computation of K_b following (4.23) would require a full sum over $\boldsymbol{x}$, which for large c comprises $\mathcal{O}(c^{q-2}/(q-1)!)$ different terms. ($q-1$ variables $\leq c-b$, with one constraint, and with permutation symmetry. We neglect the dependence on b.) Theorem 5.5 reduces the number of terms to $\mathcal{O}(q^2c^2)$ at worst; a factor c from computing G_{ba}, a factor q from $\sum_\ell$ and a factor N_b from $\sum_a$, with $N_b < qc$. In Theorem 5.6 the ℓ-sum is eliminated, resulting in $\mathcal{O}(qc^2)$ terms.

We conclude that, for $q \geq 5$ and large c, Theorems 5.5 and 5.6 can significantly reduce the time required to compute the K_b parameters.[3] A further reduction occurs in Class 3 if the $W(b, z)$ function is zero for many z.

5.2.2 K_b computation

We compute K_b for the five strategies and study the computational effort needed. As aforementioned, the naive approach would require $\mathcal{O}(c^{q-2}/(q-1)!)$ terms. The knowledge of K_b provides a better understanding of the attacks when combined with the study done before on φ and ψ.

[2] For $x, y \in \mathbb{N}$, with $x \neq y$, it does not occur in general that $T(x) = T(y)$. The only way to make this happen is to choose κ in a very special way as a function of q and c. W.l.o.g. we assume that κ is not such a pathological case.

[3] To get some feeling for the orders of magnitude: The crossover point where $qc^2 = c^{q-2}/(q-1)!$ lies at $c = 120$, 27, 18, 15, 13, for q =5, 6, 7, 8, 9 respectively.

Computing K_b^{MajV}

Lemma 5.8. *Let the colluder strategy be MajV. Let $N_b \in \mathbb{N}$ with $N_b > \max\{c-b, bq-c\}$, and let τ_b and G_{ba} be defined as*

$$\tau_b = e^{i2\pi/N_b} \quad ; \quad G_{ba} = \sum_{z=0}^{b-1} \frac{\Gamma(\kappa+z)}{\tau_b^{az} z!}. \tag{5.15}$$

Then K_b^{MajV} is given by

$$b < \frac{c}{q} : K_b^{\mathrm{MajV}} = 0 \tag{5.16}$$

$$\frac{c}{q} \le b < \frac{c}{2} : K_b^{\mathrm{MajV}} = \frac{b!(c-b)!}{qN_b\Gamma(\kappa+b)\Gamma(c-b+\kappa[q-1])B(\kappa\mathbf{1}_{q-1})} \cdot \sum_{a=0}^{N_b-1} \tau_b^{ac}\left[(G_{ba}+v_{ba})^q - G_{ba}^q\right] \tag{5.17}$$

$$b = \frac{c}{2} : K_{c/2}^{\mathrm{MajV}} = 1 - \frac{q-1}{2} \cdot \frac{B(\kappa\mathbf{1}_{q-1} + \frac{c}{2}\boldsymbol{e}_1)}{B(\kappa\mathbf{1}_{q-1})} \tag{5.18}$$

$$= 1 - \tfrac{1}{2}\frac{(1+\kappa)_{c/2-1}}{(1+\kappa[q-1])_{c/2-1}} \tag{5.19}$$

$$b > \frac{c}{2} : K_b^{\mathrm{MajV}} = 1. \tag{5.20}$$

Proof. See Appendix N. □

These expressions look very complicated. However, they are easier to evaluate numerically than (4.23). Evaluation of (5.17) requires only two summations: for every a, the computation of G_{ba} involves fewer than $c/2$ terms, and the a-sum has N_b terms, with $N_b = \mathcal{O}(cq/2)$. The total number of terms is $\mathcal{O}(c^2q/4)$.

Note that a large number N can be chosen that satisfies $N > \max\{c-b, bq-c\}$ for all $c/q \le b < c/2$. Then all the N_b values in (5.17) can be set to N. The price one pays for this small simplification is that the sums contain more terms.

Remark 5.9. Eq. (5.17) holds for all $b \in \{1, \ldots, c\}$. However, it is not evident to see how it reduces to (5.16), (5.18) and (5.20) without doing the derivation in Appendix N backwards.

Computing K_b^{MinV}

LEMMA 5.10. *Let the colluder strategy be MinV. Let $N_b \in \mathbb{N}$ with*

$$N_b > \begin{cases} c-2b & \text{if } q=2 \\ (c-b)(q-2) & \text{if } q>2 \end{cases}, \tag{5.21}$$

and let τ_b and G_{ba} be defined as

$$\tau_b = e^{i2\pi/N_b} \quad ; \quad G_{ba} = \sum_{z=b+1}^{c-b} \frac{\Gamma(\kappa+z)}{\tau_b^{az} z!}. \tag{5.22}$$

Then K_b^{MinV} is given by

$$b \le \frac{c}{q} \quad : \quad K_b^{\mathrm{MinV}} = \frac{b!(c-b)!}{qN_b\Gamma(\kappa+b)\Gamma(c-b+\kappa[q-1])B(\kappa\mathbf{1}_{q-1})} \cdot \sum_{a=0}^{N_b-1} \tau_b^{ac}\left[(G_{ba}+v_{ba})^q - G_{ba}^q\right] \tag{5.23}$$

$$b > \frac{c}{q} \quad : \quad K_b^{\mathrm{MinV}} = 0. \tag{5.24}$$

PROOF. See Appendix O. □

Similarly to MajV, the evaluation of (5.23) contains two summations: for every a, the computation of G_{ba} involves $c-2$ terms in the worst case ($b=1$), and the a-sum has N_b terms, with $N_b = \mathcal{O}(cq)$. The total number of terms is $\mathcal{O}(c^2q)$, more then the MajV case but still a huge improve compared to (4.23).

Computing K_b^{Int}

LEMMA 5.11. *If the colluder strategy is the Int attack, $\theta_{y|\boldsymbol{\sigma}} = \frac{\sigma_y}{c}$, then $K_b = b/c$.*

PROOF. This strategy implies $\Psi_b(\boldsymbol{x}) = b/c$ independent of $\boldsymbol{x}$. Substitute this into (4.23) and use the fact that the probabilities add up to 1. □

The computational complexity for Int is negligible considering the helpful simplification given by Lemma 5.11.

Computing $K_b^{\tilde{\mu}-\min}$

LEMMA 5.12. *Let the colluder strategy be $\tilde{\mu}$-min. Then the form of $K_b^{\tilde{\mu}-\min}$ corresponds to the one of Theorem 5.7 with*

$$W(b,z) = \begin{cases} 1 & \text{if } T(b) < T(z) \\ 0 & \text{otherwise} \end{cases} \tag{5.25}$$

for T as defined in (5.9).

PROOF. As discussed in Section 5.2.1, $\tilde{\mu}$-min fits in Class 3 description. Then $K_b^{\tilde{\mu}-\min}$ is attainable trivially using Theorem 5.7 and (5.25). □

The complicated nature of $\tilde{\mu}$-min does not allow to find ranges of b for which the computation can be quicker. However, it is possible to predict the tally ranking for some particular values of κ. Indeed, as shown in (5.10), the function T can have three different behaviours as a function of κ: monotonically decreasing, monotonically increasing and cup-shape. When T has a monotonic behaviour, it is trivial to understand which tally b will minimize T. From the properties obtained about T in (5.10), follows that:

- If $\kappa < \frac{1}{2(q-1)}$, then T is monotonically decreasing and it is minimized by high b. Then $K_b^{\tilde{\mu}-\min} \equiv K_b^{\text{MajV}}$
- If $\kappa > \frac{1}{2}$, then T is monotonically increasing, and $T(b)$ and it is minimized by low b. Then $K_b^{\tilde{\mu}-\min} \equiv K_b^{\text{MinV}}$

A very important scenario that we have already encountered appears when $\kappa \approx 1/q$. In this case, even if $T(b)$ has a cup-shape behaviour, $\tilde{\mu}$-min behaves as MajV, as shown in the following theorem.

THEOREM 5.13. *For $q \geq 3$ and $\kappa \approx 1/q$, the MajV strategy minimizes $\tilde{\mu}$.*

PROOF. The 'optimal' colluder strategy (in the sense of making $\tilde{\mu}$ as small as possible) is, for given $\boldsymbol{\sigma}$, to choose y such that the expression $W(\sigma_y)\{\frac{1}{2} - \kappa + \frac{\sigma_y}{c}(\kappa q - 1)\}$ is minimized. Putting $\kappa \approx 1/q$ in (4.57), we see that the optimal attack strategy is effectively to minimize W, i.e. the coalition chooses $y = \text{argmin}_{\alpha \in \mathcal{Q}: \sigma_\alpha > 0} W(\sigma_\alpha)$. Numerical inspection shows that the function $W(b)$ has a minimum at $b = \lceil c/2 \rceil$ (see Fig. 5.2).

For large c this is easily understood: application of Lemma 4.9 for large b and $c-b$ gives $W(b) \approx [\frac{b}{c}(1-\frac{b}{c})]^{-1/2}$, a function with its minimum

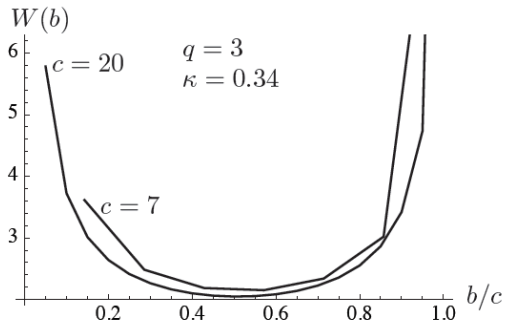


FIGURE 5.2: *Example of $W(b)$ for $q = 3$, $\kappa = 0.34$.*

at $b = c/2$ and symmetric around this minimum. Hence the optimal strategy consists of choosing the symbol α whose σ_α is closest to $c/2$. It turns out that this is precisely the same as majority voting. This can be seen as follows. First consider the case where the 'closest to $c/2$' strategy results in $\sigma_y > c/2$. Because of the sum rule $\sum_\alpha \sigma_\alpha = c$, there can be no $\alpha \neq y$ with $\sigma_\alpha > c/2$; hence the strategy has resulted in selecting the majority symbol. Second, consider the 'closest to $c/2$' strategy yielding $\sigma_y = c/2 - \delta$, with $\delta > 0$. If there is any $\alpha \neq y$ with $\sigma_\alpha > \sigma_y$, it will have to satisfy $\sigma_\alpha \geq c/2 + \delta = c - \sigma_y$. Only the equality is allowed ($\sigma_\alpha = c - \sigma_y$) by the sum rule; it gives rise to almost the same amount of accusation as σ_y, since $W(b)$ is very close to symmetric around $c/2$. □

Computing K_b^{RS}

The RS attack does not fit in any of the classes defined in Section 5.2.1. Indeed, the factor $1/w$ in Definition 5.4 does not fit with the ℓ Even so, K_b^{RS} can also be simplified obtaining a result that looks similar to the formula for Class 2.

THEOREM 5.14. *Let $q > 2$ and $b \in \{1, \ldots, c-1\}$. 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*

$$G_{ba} = \sum_{z=1}^{c-b} \frac{\Gamma(\kappa + z)}{\tau_b^{az} z!}. \tag{5.26}$$

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

$$K_b^{\mathrm{RS}} = \frac{(c-b)!\Gamma(\kappa[q-1])\Gamma(\kappa)}{qN_b\Gamma(c-b+\kappa[q-1])} \sum_{a=0}^{N_b-1} \tau_b^{a(c-b)} \frac{(G_{ba}/\Gamma(\kappa)+1)^q - 1}{G_{ba}}. \tag{5.27}$$

Proof. See Appendix P. □

Theorem 5.14 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.

Theorem 5.14 holds for $q > 2$. For the binary alphabet the result is much simpler.

Lemma 5.15. *Let $q = 2$ and $b \in \{1, \dots, c-1\}$. Then the K_b parameter for the RS strategy is*

$$K_b^{\mathrm{RS}} = \tfrac{1}{2}. \tag{5.28}$$

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

$$K_1^{\mathrm{RS}} < K_2^{\mathrm{RS}} < \cdots < K_{c-1}^{\mathrm{RS}} = \tfrac{1}{2}. \tag{5.29}$$

Proof sketch. 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. □

5.3 Analytic results

Some of the results given in Chapter 4 can be rewritten using the K_b formulas of Section 5.2.2.

5.3.1 Dominant power on tails

In φ definition (Theorem 4.2.2) K_b has an important role and the tail behaviours shown in Table 4.1 does not hold for MajV. Indeed, the discrepancy between the tails is even more pronounced if the attackers use MajV strategy (which for $q \geq 3$, $\kappa \approx 1/q$ minimizes $\tilde{\mu}$, as shown in Theorem 5.13) as shown in Figure 5.3.

Then the right tail is dominated by the $b = \lceil c/q \rceil$ term, which behaves as $(1/u)^{3+2\lceil c/q \rceil + 2\kappa}$, which for $c > q$ decreases even faster than $(1/u)^{5+2\kappa}$. From this perspective it may be better for the attackers not to use MajV; another strategy may yield a form of the ρ curve that is better for them. The best strategy strikes a balance between decreasing $\tilde{\mu}$ and lengthening the tail of $\varphi_+(u)$.

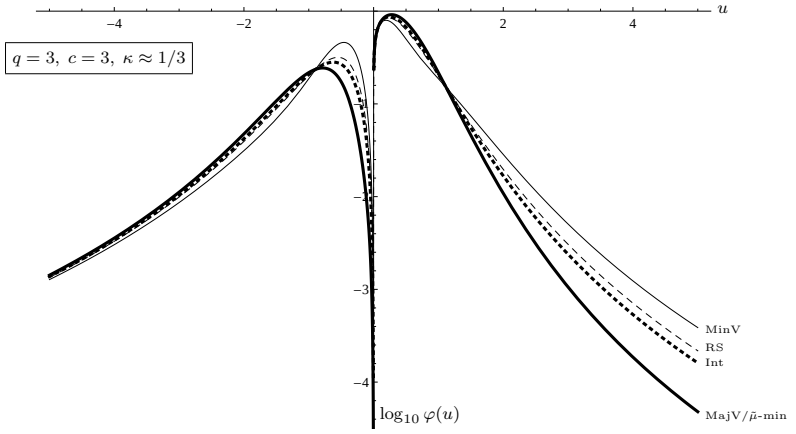


FIGURE 5.3: *The pdf φ of the single-segment score, shown for several strategies. The right tail strongly depends on the strategy, while the left tail is hardly affected.*

In the binary case, it is easy to identify where the balance lies: For $\kappa \approx \frac{1}{2}$, the strategy has practically no effect on $\tilde{\mu}$, so the attackers should concentrate on lengthening the $\varphi_+(u)$ tail. This is achieved by setting Ψ_b nonzero for small values of b, e.g. Int or MinV.

Note too what happens to the overall probability when the colluders choose a MajV strategy: then K_b tends to be small for small b and large for large b. The terms with large b then dominate the summations in Corollary 4.22, and consequently $\Pr[u > 0] > \Pr[u < 0]$. This is consistent with the fact that the left ($u < 0$) tail is heavier: the probability mass at $u < 0$ must be further removed from $u = 0$ in order to cause $\mathbb{E}[u] = 0$.

Let us now analyse the effects on $\psi(u)$ when the strategies defined are used. In case the chosen strategy is MajV, the right tail is dominated by the $b = \lceil c/q \rceil$ term, which behaves as $(1/u)^{2\lceil c/q \rceil + 2\kappa + 1}$, which for $c > q$ decreases faster than $(1/u)^{3+2\kappa}$. For MinV the left tail is dominated by $b = \lfloor c/2 \rfloor$, which behaves as $(1/|u|)^{2\lceil c/2 \rceil + 2\kappa[q-1]+1}$ and decreases faster than $(1/|u|)^{3+2\kappa[q-1]}$ for $c > 2$. Since K_1 is the coefficient associated with the dominant power in the right tail, we find that MinV yields the most pronounced right tail. On the left side it is MajV, the strategy that most emphasizes K_{c-1}. Fig. 5.4 illustrates these trends.

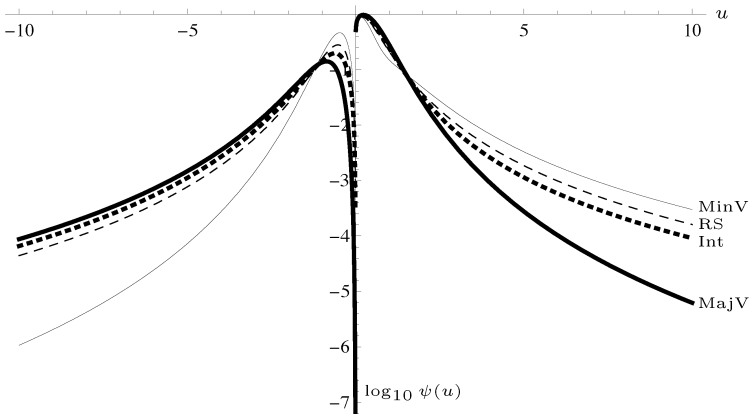


Figure 5.4: *The pdf ψ of the single-segment score of a guilty user, shown for several strategies. $c = 4$, $q = 3$, $\kappa \approx 1/3$.*

5.3.2 Simplifications for Int attack

The extremely effective simplification on K_b^{Int} let us simplified also other equations.

Corollary 5.17 (Of Theorem 4.27). *For the Int strategy, the $\tilde{\mu}$ parameter becomes*

$$\tilde{\mu}_{\text{Int}} = q\frac{B(\kappa + \frac{1}{2}, \kappa[q-1] + \frac{1}{2})}{B(\kappa, \kappa[q-1])}. \tag{5.30}$$

Proof. From the definition of $\tilde{\mu}$ it follows that it can be computed as an expectation value in a single content segment, $\tilde{\mu} = \mathbb{E}[\sigma_y g_1(p_y) + (c - \sigma_y)g_0(p_y)]$, with $\mathbb{E}$ the expectation over $\boldsymbol{p}$, $\boldsymbol{\sigma}$ and y, and g_1 and g_0 as defined in (3.11). The $\mathbb{E}_y(\cdots)$ expectation is given by $\sum_y \frac{\sigma_y}{c}(\cdots)$. We write

$$\frac{\sigma_y}{c}\left[\sigma_y g_1(p_y) + (c - \sigma_y)g_0(p_y)\right] = p_y\frac{\sigma_y - cp_y}{\sqrt{p_y(1-p_y)}} + \frac{1}{c}\frac{(\sigma_y - cp_y)^2}{\sqrt{p_y(1-p_y)}}. \tag{5.31}$$

From the properties of the multinomial distribution we get $\mathbb{E}_{\boldsymbol{\sigma}}[\sigma_y - cp_y] = 0$ and $\mathbb{E}_{\boldsymbol{\sigma}}[(\sigma_y - cp_y)^2] = cp_y(1-p_y)$. Next, the expectation $\mathbb{E}_{\boldsymbol{p}}$ over the full vector $\boldsymbol{p}$ reduces to the expectation over the component p_y, for which we use the marginal pdf $f(p)$ (Lemma 4.4). This gives

$$\tilde{\mu}_{\text{Int}} = \sum_y \frac{1}{B(\kappa, \kappa[q-1])}\int_0^1 \mathrm{d}p_y\, p_y^{-1+\kappa}(1-p_y)^{-1+\kappa[q-1]}\sqrt{p_y(1-p_y)}. \tag{5.32}$$

The result of the integration does not depend on y, so the $\sum_y$ yields a factor q. The integral yields $B(\kappa+\frac{1}{2},\kappa[q-1]+\frac{1}{2})$. □

COROLLARY 5.18 (Of Theorem 4.21). *If the colluder strategy is the Int, then*

$$\varphi_+^{\mathrm{Int}}(u) = \frac{2q}{B(\kappa,\kappa[q-1])}\frac{(u^2)^{\kappa[q-1]-\frac{1}{2}}}{(1+u^2)^{2+\kappa q}} \tag{5.33}$$

$$\varphi_-^{\mathrm{Int}}(u) = \frac{2q}{B(\kappa,\kappa[q-1])}\frac{(u^2)^{\kappa+\frac{1}{2}}}{(1+u^2)^{2+\kappa q}}, \tag{5.34}$$

and $\Pr[u>0]=\frac{\kappa+1}{\kappa q+1}$.

PROOF. The first part follows directly by applying Lemma 5.11 to (4.45) and using $\sum_{b=0}^{c}\binom{c}{b}bx^b = xc(1+x)^{c-1}$. The second part follows from computing the integral $\int_0^\infty \mathrm{d}u\ \varphi_+(u)$ using Lemma 4.8. □

It is interesting to note that the Int attack yields a $\varphi(u)$ distribution that has the heaviest possible tails for both positive and negative u (see Table 4.1): proportional to $(1/|u|)^{3+2\kappa[q-1]}$ for the left tail and $(1/u)^{5+2\kappa}$ for the right tail. It also has the lowest possible dominant powers around $u=0$. Furthermore, $\varphi(u)$ has the special property that it is completely independent of c.

The simplified $\varphi(u)$ given by Corollary 5.18 yields also a simplified Fourier transform:

COROLLARY 5.19 (Of Theorem 4.24). *If the colluders use the Int attack,*

then

$$\tilde{\varphi}_{\mathrm{Int}}(k) = 1 - \frac{1}{2}k^2 + \frac{2q}{B(\kappa, \kappa[q-1])} \cdot \Bigg[(ik)^{4+2\kappa}\Gamma(-4-2\kappa)\,{}_1F_2\left(\kappa q; \kappa + \frac{5}{2}, \kappa + 3; \frac{k^2}{4}\right) + (-ik)^{2+2\kappa[q-1]}\Gamma(-2-2\kappa[q-1]) \cdot {}_1F_2\left(\kappa q; \kappa[q-1] + \frac{3}{2}, \kappa[q-1] + 2; \frac{k^2}{4}\right) + \frac{1}{2}\sum_{j=3}^{\infty} \frac{(ik)^j}{j!} \left[B\left(\kappa + 1 + \frac{j}{2}, \kappa[q-1] + 1 - \frac{j}{2}\right) + (-1)^j B\left(\kappa[q-1] + \frac{j}{2}, \kappa + 2 - \frac{j}{2}\right) \right] \Bigg]. \tag{5.35}$$

Proof. The Fourier integrals of the φ_+ and φ_- given in Corollary 5.18 are precisely of the form handled in Lemma 4.23, with $(d = \kappa[q-1], v = \kappa + 2)$ and $(d = \kappa + 1, v = \kappa[q-1] + 1)$ respectively. □

Corollary 5.20 (Of Theorem 4.30). *If the colluder strategy is the Int, then*

$$\psi_-^{\mathrm{Int}}(u) = (1 - \tfrac{1}{c}) \frac{2q}{B(\kappa, \kappa[q-1])} \frac{(u^2)^{\kappa+1/2}}{(1+u^2)^{2+\kappa q}} \tag{5.36}$$

$$\psi_+^{\mathrm{Int}}(u) = \frac{2q}{B(\kappa, \kappa[q-1])} (c + u^2) \frac{(u^2)^{\kappa[q-1]-1/2}}{(1+u^2)^{2+\kappa q}}. \tag{5.37}$$

The left tail has dominant power $(1/|u|)^{3+2\kappa[q-1]}$ *and the right tail* $(1/u)^{3+2\kappa}$*, which corresponds to the longest possible tails as listed in Table 4.2. This does not come as a surprise; the Int attack has the same tail behavior in the case of innocent users.*

Proof. In the case of the Int attack, we have $K_b = b/c$. Then the $\sum_b$ summations in Theorem 4.30 can be evaluated exactly yielding the result. □

6

Numerical results

In this chapter we apply the CSE method to obtain numerics about P_{FP} and P_{FN}. These two cases are going to be studied first separately and then combined in the ROC curve. We analyzed how the errors probability scales over changes on all the parameters involved in the scheme: c, q, m, κ and, obviously, the strategy.

6.1 Convergence properties of the CSE method

In this section we are going to discuss the convergence of the CSE method. How many terms in the series expansions must be kept in order to obtain an accurate result? If too few are taken, the result is incorrect. If many are taken, too much time is spent. As we will see, it can even happen that a series first converges and then diverges when more terms are added. The power of k where we cutoff the series will be called "the cutoff". The motivation is critical: in order to obtain computable numerics it is necessary to introduce some cutoffs that produce an important speedup. Our target is to analyze whenever these approximations can produce good or bad results.

Among all the components defined so far, it has been necessary to introduce a cutoff just in these four functions:

- $[\tilde{\varphi}(k/\sqrt{m})]^m$ defined in (4.40),
- $[\tilde{\chi}(k/\sqrt{m})]^m$ defined in (4.86),
- $R_m(\tilde{Z})$ defined in (4.41),

- $C_m(\tilde{\Delta})$ defined in (4.87).

Theorems 4.19 shows how $[\tilde{\varphi}(k/\sqrt{m})]^m$ provides the variables ω_t, ν_t and α_t necessary to compute R_m. The exact same type of relation holds between $[\tilde{\chi}(k/\sqrt{m})]^m$ and C_m as shown in Theorem 4.41. In (4.40) and (4.86) the summations represent a series expansion over k, so we can equivalently fix a cutoff $\nu_{\max}$ as degree of these series expansions (namely we remove from the series the terms k^ν s. t. $\nu > \nu_{\max}$).

To understand the accuracy of the approximated power series we analyze the effect of the cutoff on the $R_m(\tilde{Z})$ and $C_m(\tilde{\Delta})$ curves. Indeed, thanks to the CLT, we know in advance the kind of shape the two curves should have: for low $\tilde{Z}$ and $\tilde{\Delta}$ values, they follow a Gaussian curve behaviour, while for higher values they change their slopes to a power-law behaviour. The power-law behaviour has to be consistent with Figures 5.3 and 5.4 and with Theorems 4.21 and 4.30. So, we can conclude that the cutoff is accurate enough if the curves:

- do not visibly change when we increase the cutoff,
- have a Gaussian region around the expected value,
- have a power-law region in the tail,
- have values in $[0, 1]$.

The results we are going to show are mostly based on a trial-and-error approach. 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[1], in particular the attack strategy. When $\nu_{\max}$ is chosen too small, we observe one of the following problems: there exist $\tilde{Z}$ for which $R_m(\tilde{Z})$ is not in the range $[0, 1]$; the $R_m(\tilde{Z})$ is not a smooth function of $\tilde{Z}$; or is not a strictly decreasing function, e.g. containing oscillations. The most pronounced effect is around the point where the curve leaves the Gaussian curve. The numerics have been computed using Mathematica 8 on a Windows machine. The running time depends on many parameters, in particular $\nu_{\max}$. To compute $R_m(\tilde{Z})$ for 100 points it can take from few seconds to several hours. However, the code has not been fully optimized. The code

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

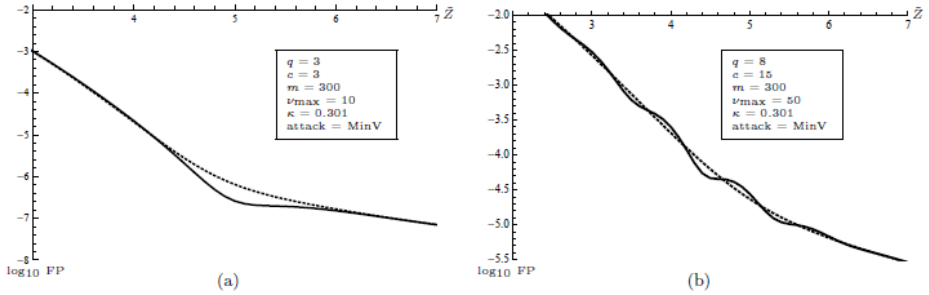


FIGURE 6.1: *Examples of incorrect $R_m(\tilde{Z})$ curves (solid line) when the cutoff $\nu_{\max}$ is chosen too small. Oscillations occur in the region where the curve departs from Gaussian behaviour. The dotted curve is the correct result.*

is available online[2]. Examples are shown in Fig. 6.1. The exact same effect happens for $C_m(\tilde{\Delta})$.

6.1.1 Convergence of the innocent user series

Table 6.1 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 [32], so they are shown together in one column.

From the table we can see that $\nu_{\max} = 30$ is in general a safe choice. As we expected, the common effect of raising $\nu_{\max}$ is to stabilize the so called "correction term" added to $\Omega(\tilde{Z})$ in (4.41). An example of this effect can be seen in Figure 6.2,

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 this effect has its origin in the 'large' value of $1/\sqrt{m}$ which is used as the expansion parameter, leading to an ill-defined series expansion in the CSE method. It is known that Edgeworth expansions and Gram-Charlier expansions are not always convergent [15], especially in the case of fat tails. Our expansion is similar to an Edgeworth expansion, but with non-integer powers.

[2]http://www.win.tue.nl/CREST/

c	q	m	MajV/ $\tilde{\mu}$-min	MinV	Int	RS
3	3	300	30	$\geq$15	$\geq$25	$\geq$25
	3	1000	$\geq$15	$\geq$15	$\geq$15	$\geq$15
	3	2000	$\geq$15	$\geq$15	$\geq$15	$\geq$15
	5	300	$\geq$15	$\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$15	$\geq$25	$\geq$15
	3	1000	$\geq$40	$\geq$15	$\geq$15	$\geq$15
	3	2000	$\geq$25	$\geq$10	$\geq$15	$\geq$15
	5	300	$\geq$20	$\geq$10	$\geq$10	$\geq$10
	5	1000,2000	$\geq$15	$\geq$10	$\geq$10	$\geq$10
	8	300	$\geq$15	$\geq$10	$\geq$10	$\geq$10
	8	1000	$\geq$15	$\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$25	$\geq$15
	3	1000	-	$\geq$10	$\geq$15	$\geq$15
	3	2000	-	$\geq$10	$\geq$15	$\geq$15
	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$15	$\geq$10	$\geq$10	$\geq$10
	15	300	$\geq$15	10 - 40	$\geq$10	$\geq$10
	15	1000	$\geq$15	$\geq$10	$\geq$10	$\geq$10
	15	2000	$\geq$10	$\geq$10	$\geq$10	$\geq$10
15	3	300	-	$\geq$10	$\geq$25	$\geq$15
	3	1000	-	$\geq$10	$\geq$15	$\geq$15
	3	2000	-	$\geq$10	$\geq$15	$\geq$10
	5	300,1000,2000	-	$\geq$10	$\geq$10	$\geq$10
	8	300	-	10 - 35	$\geq$10	$\geq$10
	8	1000	$\geq$35	$\geq$10	$\geq$10	$\geq$10
	8	2000	$\geq$25	$\geq$10	$\geq$10	$\geq$10
	15	300	25 - 35	10	$\geq$10	$\geq$10
	15	1000	$\geq$20	$\geq$10	$\geq$10	$\geq$10
	15	2000	$\geq$15	$\geq$10	$\geq$10	$\geq$10

TABLE 6.1: *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. $\kappa \approx 1/q$.*

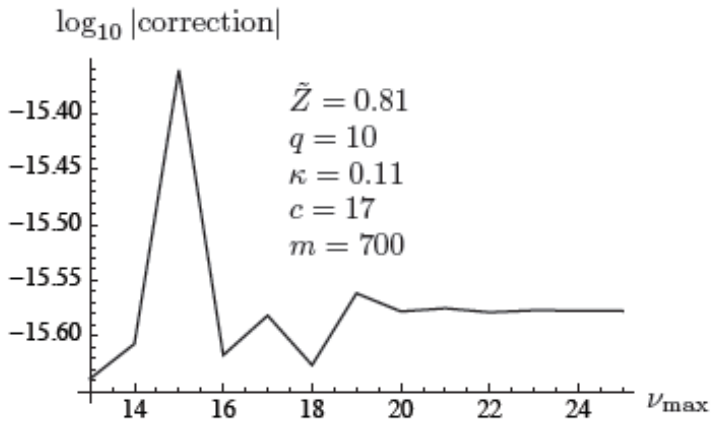


FIGURE 6.2: *Logarithmic plot of the correction to* $\Omega(\tilde{Z})$ *as a function of* $\nu_{\max}$*, the maximum power of* k *kept in the expansion. The applied strategy is MajV.*

In Table 6.1 we can also observe how the variables c, q, and m alter the convergence speed of $R_m(\tilde{Z})$.

c For MajV/$\tilde{\mu}$-min it is clear that for growing c the convergence is more difficult to achieve. This effect is not present for the other three strategies where, on the contrary, in some cases the convergence is obtained even more quickly (for example, for RS attack the case $(c = 8, q = 3, m = 2000)$ converges more slowly than $(c = 15, q = 3, m = 2000)$).

q For all the strategies a higher q speeds up the convergence.

m Increasing m facilitates the convergence. The cases mentioned beforehand, in which $\nu_{\max}$ can be *too big*, occur for low values of m combined with high c and q values and just for Class 3 strategies.

In conclusion, all the parameters studied affect the convergence speed of the CSE method. Non-ranking-based strategies seem to converge better than Class 3 strategies. We hypothesize that the presence of cases such that $K_b = 0$ for some b (high or low), typical for Class 3 strategies, can be the reason why these strategies has more convergence issues then RS and Int, where K_b is always positive for $b > 0$. A detailed analysis of the CSE convergence properties is left for future work.

6.1.2 Convergence of the guilty user series

The study of the guilty curve is less complete than the innocent one because of time limitations. We preferred to concentrate more on the innocent case being the most important.

The convergence of (4.87) turns out to be rather quick. Often it suffices to take powers only up to $\nu_t \approx 10$ in order to get good accuracy. An example is shown in Fig. 6.3. (The parameters were chosen such that we are not in the Gaussian regime but in the right tail.)

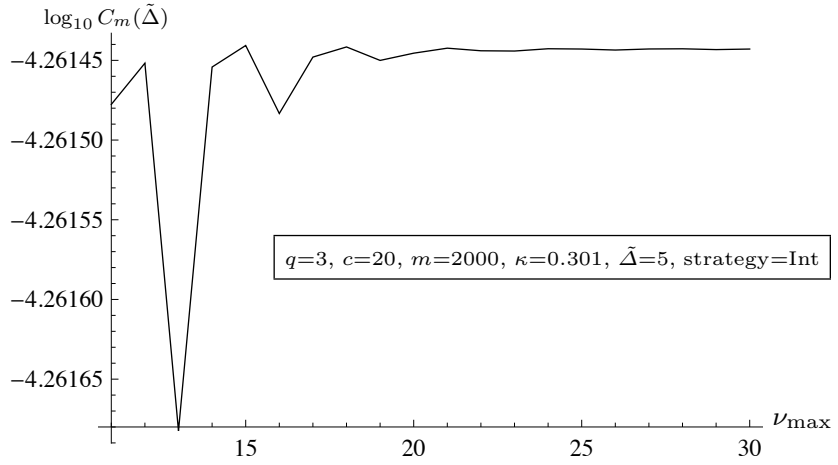


FIGURE 6.3: *Convergence example.* $C_m(\tilde{\Delta})$ *computed according to (4.85) as a function of the cutoff power* $\nu_{\max}$.

6.2 Power-law behaviour of the FP tail

In this section we present numerics showing that the tail of $R_m(\tilde{Z})$ (4.41) indeed has power law behaviour that follows directly from the dominant contribution in $\varphi(u)$ (4.45). In Section 5.2.2 we saw that all the investigated strategies, except MajV and sometimes $\tilde{\mu}$-min, have $K_1 > 0$. This leads to a dominant term proportional to $(1/u)^{5+2\kappa}$ at $u \gg 1$ (see Table 4.1). 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 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. 6.4 we show a log-log plot of $R_m(\tilde{Z})$ for several strategies, for one combination of q, c, m, κ. (Without providing further evidence we mention that the behaviour is the same for other parameter choices.) In the same graph we have also plotted the single-segment $P_{\text{FP}}\ R_1(\tilde{Z}) = \int_{\tilde{Z}}^{\infty} \mathrm{d}u\ \varphi(u)$ 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. 5.3. (Except for MajV which has a completely different tail.)

The fact that the tail of the MinV curve lies higher than the rest is shown again in Table 4.1: 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 6.1. *It holds that* $K_1^{\text{RS}} \geq K_1^{\text{Int}}$.

Proof. We have $\Psi_1^{\text{Int}}(\boldsymbol{x}) = \frac{1}{c}$ and, $\Psi_1^{\text{RS}}(\boldsymbol{x}) = \frac{1}{s(\boldsymbol{x})+1}$, 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

$$\Psi_1^{\text{RS}}(\boldsymbol{x}) = \frac{1}{s(\boldsymbol{x})+1} \geq \frac{1}{\min\{c,q\}} = \max\left\{\frac{1}{c}, \frac{1}{q}\right\} \geq \frac{1}{c} = \Psi_1^{\text{Int}}(\boldsymbol{x}). \quad (6.1)$$

Finally, from the definition of K_b (4.23) we know that $\Psi_1^{\text{RS}}(\boldsymbol{x}) \geq \Psi_1^{\text{Int}}(\boldsymbol{x})$ implies $K_1^{\text{RS}} \geq K_1^{\text{Int}}$. □

The $\tilde{\mu}$-min strategy is more difficult to analyze. As shown in [32], it behaves in a rather complicated way, sometimes being equivalent to MinV, sometimes MajV, or something in-between, depending on many parameters, mostly κ and q.

In [7] a rare event estimation technique was used to estimate FP probabilities. There no transition from Gaussian to power-law shape was observed down to $\varepsilon_1 \approx 10^{-25}$ at $m = 600$ and $c \leq 5$.

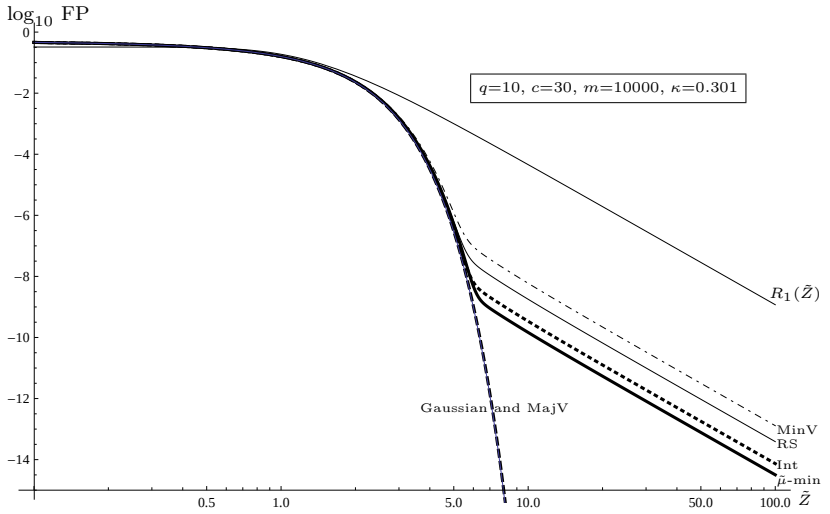


FIGURE 6.4: *Log-log plot of $R_m(\tilde{Z})$ for several strategies. The single-segment FP probability $R_1(\tilde{Z})$ for the Int is also plotted.*

6.3 Comparison of FP rates for different attacks

The precise P_{FP} is the most critical error to study. When we were developing the CSE method it was our first target and, once we manage to compute it, we faced the problem of how to make fair comparisons between the strategies, not having at that moment access to accurate FN numbers. Once we manage to compute also FN probabilities, we did not present many comparisons via ROC curve because of lack of time and because the usual plots makes more visible the dependence on the threshold. Before going through the study of the effect of the several scheme parameters on P_{FP}, we first describe the procedure used to compare the strategies.

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

In [34] we chose 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 a better understanding, see Figure 3.1.) 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} [43]; hence the FN probability is approximately $(\frac{1}{2})^c$. Other than this, very little information was available about the scores of the colluders. Fortunately, the pdf of the collective score S is 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 $Z = Z_{\text{half}}$.

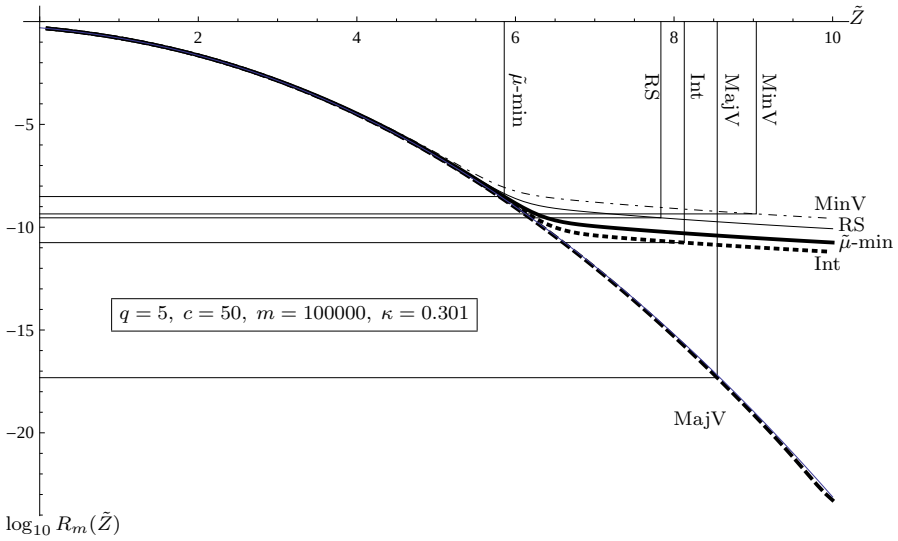


FIGURE 6.5: *FP probability as a function of the accusation threshold, for different strategies. The auxiliary lines connect each curve to its $\tilde{Z}_{\text{half}}$, allowing us to read off FP values for a fair comparison of strategies. Note that for the chosen parameter values, the $\tilde{\mu}$-min 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. 6.5 and 6.6. 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. 6.5 the $\tilde{\mu}$-min attack wins, while in Fig. 6.6 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 part of the $\tilde{\mu}$-min curve, there can be no stronger attack

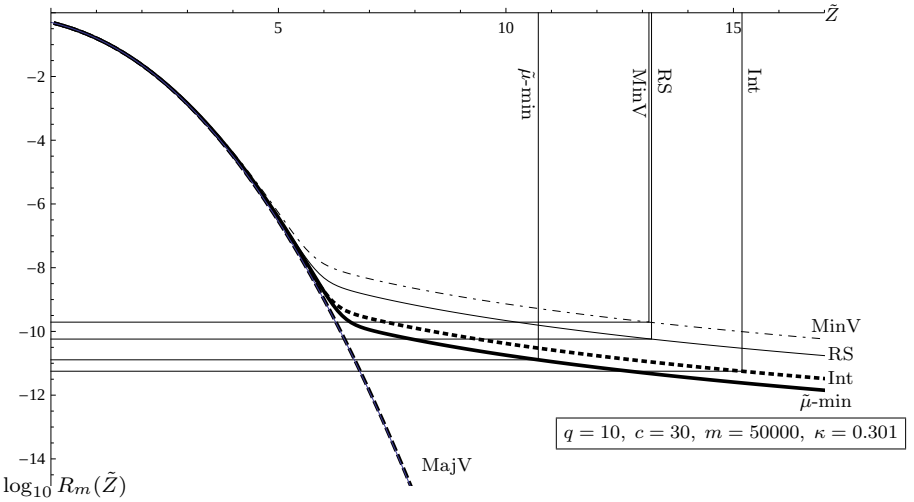


FIGURE 6.6: *Same type of plot as Fig. 6.5, but with different q, c and m. In this case the $\tilde{\mu}$-min attack has its $\tilde{Z}_{\text{half}}$ far outside the Gaussian part of the curve.*

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.

6.3.2 Study of the effect of c, q and m

We present plots for the dependency of the attacks on the three parameters c, q and m separately. The parameter κ is a bit problematic. The best choice of κ (from the tracing point of view) depends on q and c in a complicated way, which is only partially known via lower bounds on $\tilde{\mu}$ (see e.g. [40]). We have decided not to show all plots for multiple κ as this would lead to an excessive number of figures. Instead we have picked a representative κ. In [42] (Fig.2) it was shown that at 'finite' c and $q > 3$, the bound on $\tilde{\mu}$ has an optimum at $\kappa > 1/q$, where the distance between the optimal κ and $1/q$ increases with decreasing c. From this we distilled a 'compromise' $\kappa \approx 0.3$ that is not too far away from the optimum for all considered q. In [13], instead, they used $\kappa = \frac{1}{2}$ focusing at the code rate min-max game for general decoders, while we just concentrate on Tardos score.

Varying the coalition size c

Fig. 6.7 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. 6.5 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.

Notice that Pictures 6.7(b), (c) and (d) show a change of slope for many strategies passing from an almost straight increasing behaviour to a curved one. This change indicates the transaction from the power-law behaviour (high $\tilde{Z}_{\text{half}}$) to the Gaussian one (low $\tilde{Z}_{\text{half}}$). In the plot 6.7(a) just the curved behaviour is present indicating that all the studied $\tilde{Z}_{\text{half}}$'s lie in the Gaussian regime and, indeed, $\tilde{\mu}$-min is the leading strategy. (Notice that in all the plots MajV follows the Gaussian curve.) In the other of Fig. 6.7 we have the same result in the curved side, while in the straight one is MinV to win, proving that the regime is not Gaussian. The better results provided by $\tilde{\mu}$-min is expected and can be seen also from this point of view: once the Gaussian regime is entered, the strategies are basically overlapping. The contribution given by the strategies and the parameters is affecting just the $\tilde{Z}_{\text{half}}$ which is minimized by $\tilde{\mu}$-min attack.

Varying the alphabet size q

Fig. 6.8 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 [42], forcing the "read-off" point in Fig. 6.5 to the right. We see crossovers occurring as a function of q too.

Varying the code length m

Fig. 6.9 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. Notice also that for growing m the broadening of the Gaussian region is slower than the increasing of the $\tilde{Z}_{\text{half}}$'s. Indeed, for high m, MinV is always the best strategy, indicating that the $\tilde{Z}_{\text{half}}$ are outside the Gaussian region even if higher m contributes to enlarge its range.

Varying the parameter κ

Fig. 6.10 shows the dependance on κ. Apart from $\tilde{\mu}$-min, all the strategies have a smooth behaviour. As shown in Section 5.2.2, 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 redefinition of the $\tilde{\mu}$-min 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. 6.10 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 [40], attack strategy independent bounds were obtained on the error probabilities in q-ary Tardos codes. If we compare Fig. 6.10 to those bounds (in particular the rightmost part of Table 1 in [40]), we see that the bounds are far from tight. In Fig. 6.10 the FP error around $\kappa = 0.3$ for the most powerful attack in our set is more than 10^4 times smaller than the strategy-independent bound in [40]. (And we believe that there exists no attack strategy that significantly outperforms the set considered here.)

6.3.3 Transition in the $\tilde{\mu}$-min attack

We dedicate a separate section to $\tilde{\mu}$-min attack. This strategy represents the best attack in the asymptotic scenario. In the Gaussian regime it has been shown that a codelength $m = (2/\tilde{\mu}^2)c^2 \ln(1/\varepsilon_1)$ is sufficient against a coalition of size c. We want to show how the parameters influence the convergence to the Gaussian regime when the $\tilde{\mu}$-min is chosen, giving particular attention to κ.

In [33] the $\tilde{\mu}$-min attack was studied for a restricted parameter range, $\kappa \approx 1/q$. For such a choice of κ the strategy reduces to MajV. We study a *broader range*. We use Theorem 5.7 to precompute the K_b and then (4.51), (4.40) and (4.41) to compute the false accusation probability R_m as a function of the accusation threshold. We found that keeping terms in the expansion with $\nu_t \leq 37$ gave stable results.

For a comparison with [33], we set $\varepsilon_1 = 10^{-10}$, and search for the smallest codelength m_* for which it holds that $R_m(\tilde{\mu}\sqrt{m}/c) \leq \varepsilon_1$. The special choice $\tilde{Z} = \tilde{\mu}\sqrt{m}/c$ puts the threshold at the expectation value of a colluder's accusation. As a result the probability of a false negative error is $\approx \frac{1}{2}$. Our results for m_* are consistent with the numbers given in [33].

In Fig. 6.11 we present graphs of $2/\tilde{\mu}^2$ as a function of κ for various q, c.[3] If the accusation pdf is Gaussian, then the quantity $2/\tilde{\mu}^2$ is very close

[3] The $\tilde{\mu}$ can become negative. These points are not plotted, as they represent a

to the proportionality constant in the equation $m \propto c^2 \ln(1/\varepsilon_1)$. We also plot $\frac{m_*}{c^2 \ln(1/\varepsilon_1)}$ as a function of κ for various q, c. Any discrepancy between the $\tilde{\mu}$ and m_* plots is caused by non-Gaussian tail shapes.

In the plots on the left we see that the attack becomes very powerful (very large $2/\tilde{\mu}^2$) around $\kappa = \frac{1}{2}$, especially for large coalitions. This can be understood from the fact that the $T(b)$ values are decreasing, and some even becoming negative for $\kappa > \frac{1}{2}$, as discussed in Section 5.1.5. This effect becomes weaker when q increases. The plots also show a strong deterioration of the scheme's performance when κ approaches $\frac{1}{2(q-1)}$, as expected.

For small and large κ, the left and right graphs show roughly the same behaviour. In the middle of the κ-range, however, the m_* is very irregular. We think that this is caused by rapid changes in the 'ranking' of b values induced by the function $T(b)$; there is a transition from majority voting (at small κ) to minority voting (at large κ). It was shown in 5.3.1 (i) majority voting causes a more Gaussian tail shape than minority voting; (ii) increasing κ makes the tail more Gaussian. These two effects together explain the m_* graphs in Fig. 6.11: first, the transition for majority voting to minority voting makes the tail less Gaussian (hence increasing m_*), and then increasing κ gradually makes the tail more Gaussian again (reducing m_*).

In Fig. 6.12 we show the shape of the false accusation pdf of both sides of the transition in the $q = 3$, $c = 7$ plot. For the smaller κ the curve is better than Gaussian up to false accusation probabilities of better than 10^{-17}. For the larger κ the curve becomes worse than Gaussian around 10^{-8}, which lies significantly above the desired 10^{-10}. The transition from majority to minority voting is cleanest for $q = 2$, and was already shown in [42] to lie precisely at $\kappa = \frac{1}{2}$ for all c. For $q \geq 3$ it depends on c and is less easy to pinpoint.

6.4 Power-law behaviour of the FN tail

In Table 4.2 we see that the single-segment pdf has a power law $(1/u)^{3+2\kappa}$ in the right tail (provided that $K_1 \neq 0$). Hence the integrated probability mass beyond Z scales as $(1/Z)^{2+2\kappa}$. For large Z we expect to see the $(1/Z)^{2+2\kappa}$ scaling also in the $C_m(\tilde{\Delta})$ curves. (Due to the Central Limit

situation where the accusation scheme totally fails, and there exists no sufficient code length m_*.)

Theorem, the $C_m(\tilde{\Delta})$ goes to a Gaussian shape, but only for small $\tilde{\Delta}$; for large $\tilde{\Delta}$ the original single-segment tail is still there.) We use this as a consistency check on our CSE implementation. Fig. 6.13 shows a log-log plot of the right tail for various strategies. The tails in this plot indeed have the same slope as the curve for $m = 1$.

6.5 ROC curves

One of the most useful types of graph for decision-making problems is the Receiver Operating Characteristic (ROC). We take a slightly different graph, with ε_1 and (our upper bound on) P_{FN} on the axes. This way, being closer to the origin means better performance. An example is shown in Fig. 6.14 and 6.14(b). Each curve corresponds to tracing Z from very low (lots of users get accused: high FP and low FN) to very high (almost nobody gets accused: low FP and high FN).

In Fig. 6.14(a) most of the relevant Z values lie outside the Gaussian regime, i.e. for all the attacks except MajV and $\tilde{\mu}$-min Z lies in the linear tail of the *innocent*-user score pdf. The order of the curves for the different strategies is consistent with Section 6.3: the most powerful attack is MinV, then RS, Int, and MajV/$\tilde{\mu}$-min. (The MajV and $\tilde{\mu}$-min are identical for the chosen parameters.) The quick transition from almost 1 to almost 0 on the vertical axis occurs when Z passes through the peak of the guilty-user score pdf. The FP probability changes little during this transition, since Z lies in the tail of the innocent-user score pdf.

In Fig. 6.14(b) we used the same settings as in Fig. 6.13. This is an example of a choice of parameters such that the Z visits the innocent-user Gaussian regime during the FN-transition. When Z is lowered (downward and to the right in the figure) into the innocent-user Gaussian regime, $\tilde{\mu}$-min becomes the most powerful attack. This is not surprising, as $\tilde{\mu}$-min is designed to be the strongest attack under the Gaussian assumption. Note that the FN-transition is much wider than in Fig. 6.14(a). This is caused by the fact that $\tilde{\mu}$ enters the steep Gaussian part of the innocent-user score pdf. This is particularly the case for the $\tilde{\mu}$-min attack, which has the lowest $\tilde{\mu}$ value.

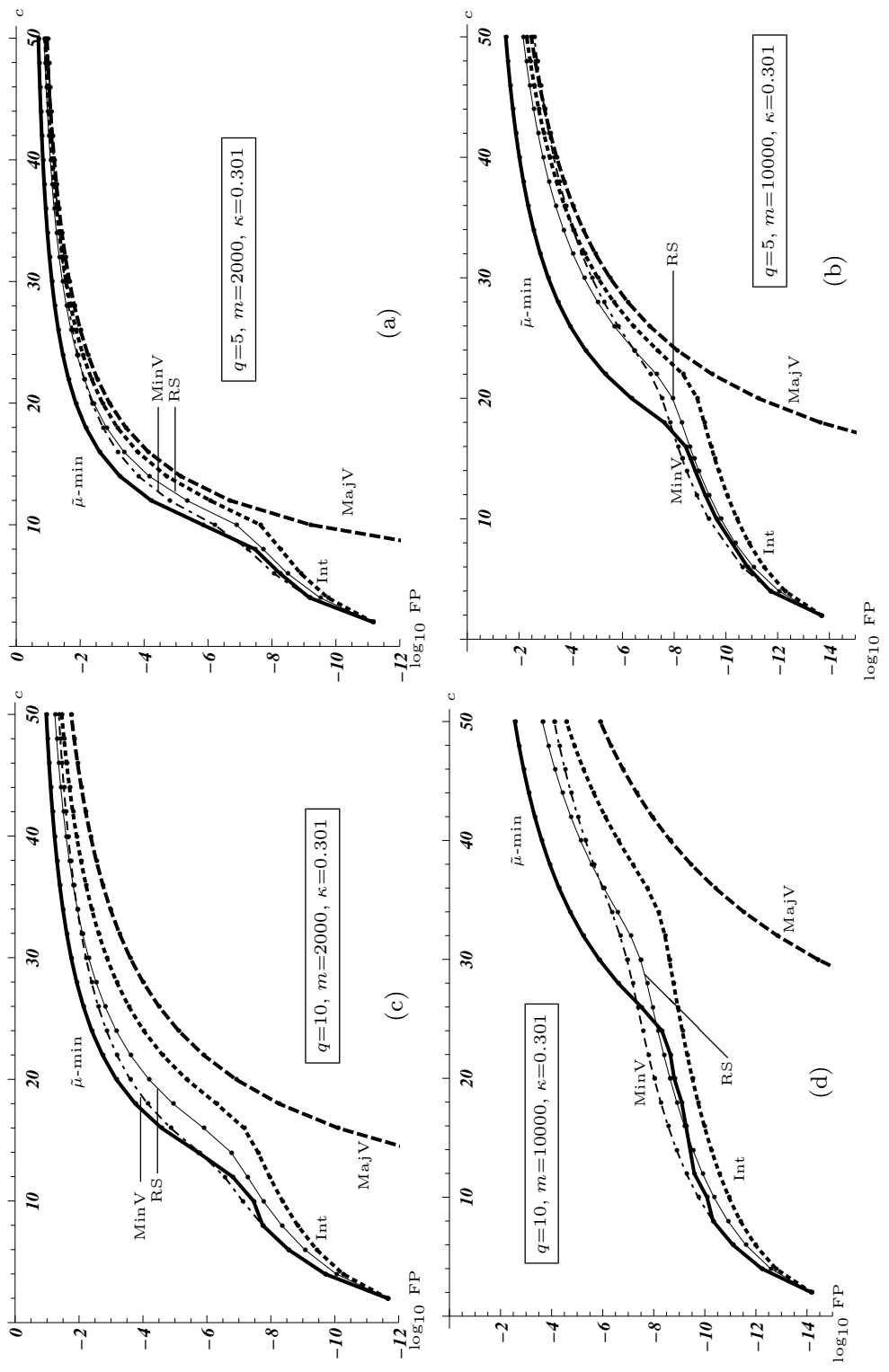


FIGURE 6.7: *FP probability $R_m(\tilde{Z}_{\text{half}})$ as a function of c for all the attack strategies. Four combinations of q and m are shown.*

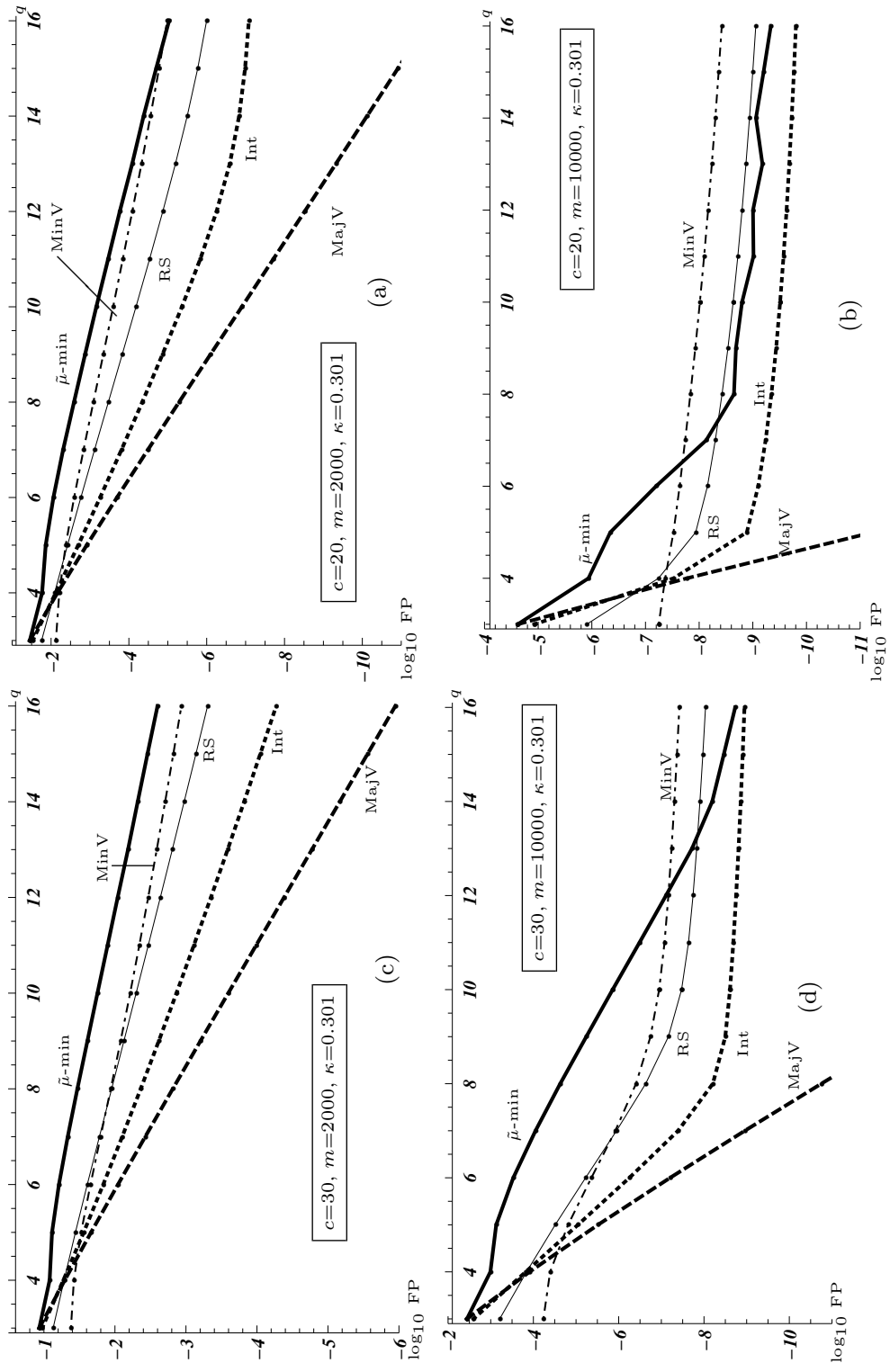


FIGURE 6.8: *FP probability $R_m(\tilde{Z}_{\text{half}})$ as a function of q for all the attack strategies. Four combinations of c and m are shown.*

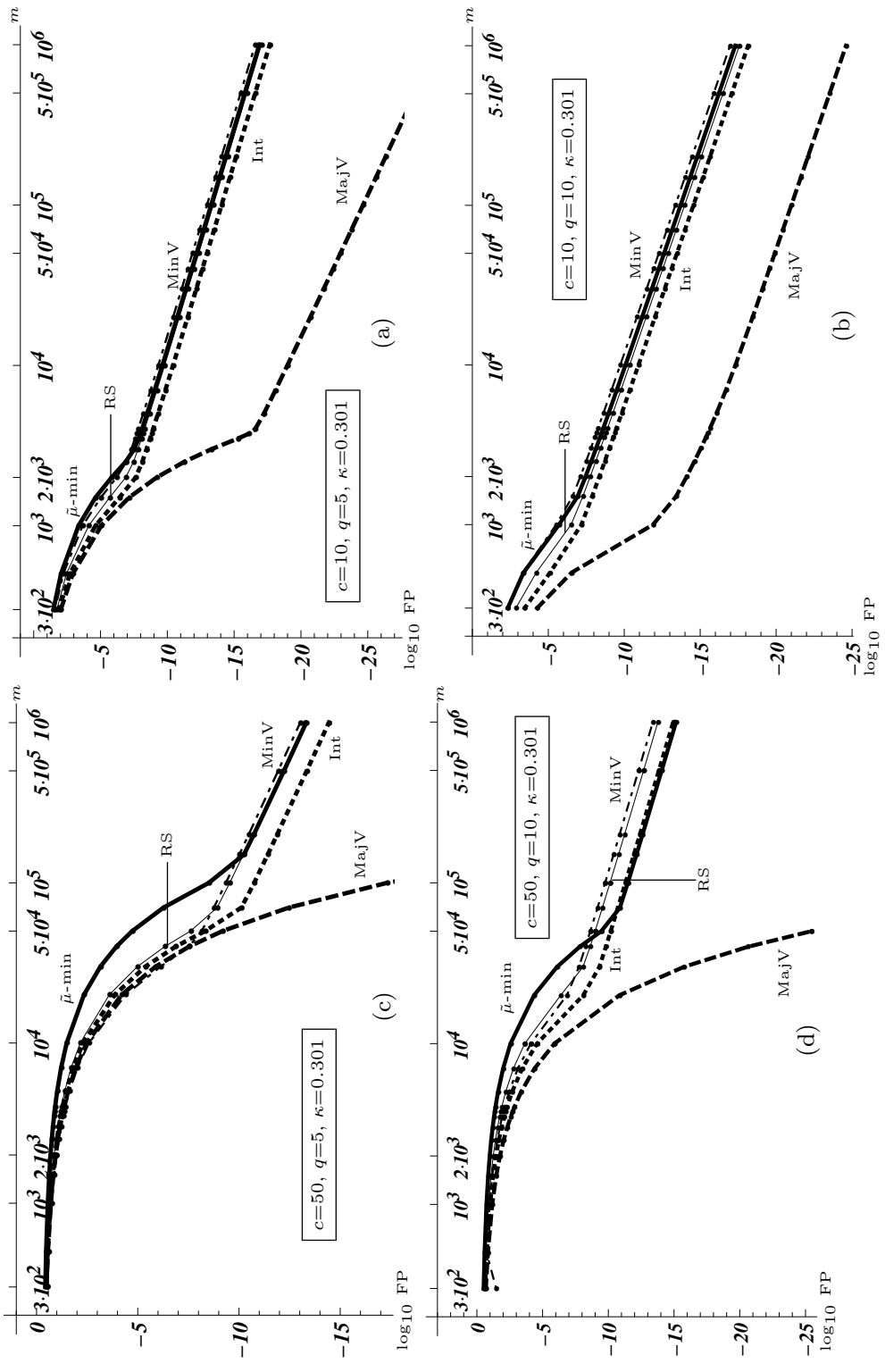


FIGURE 6.9: *FP probability $R_m(\tilde{Z}_{\text{half}})$ as a function of m for all the attack strategies. Four combinations of c and q are shown.*

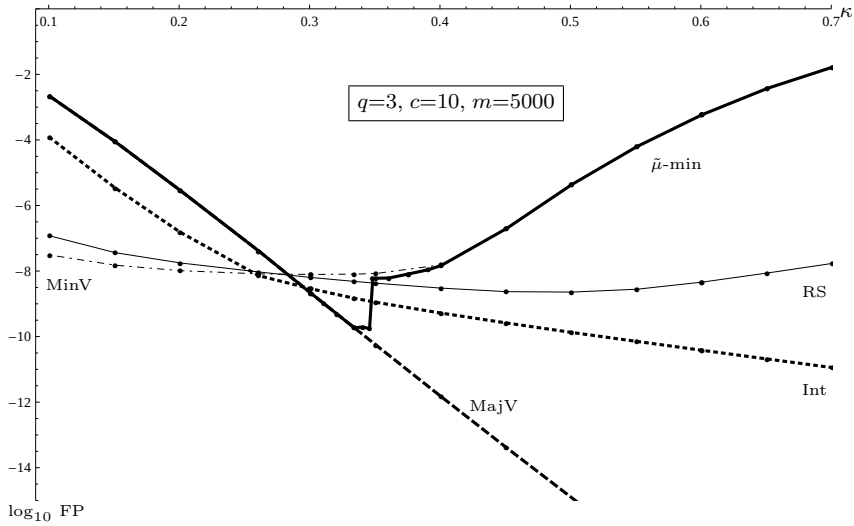


FIGURE 6.10: *FP probability* $R_m(\tilde{Z}_{\text{half}})$ *as a function of* κ.

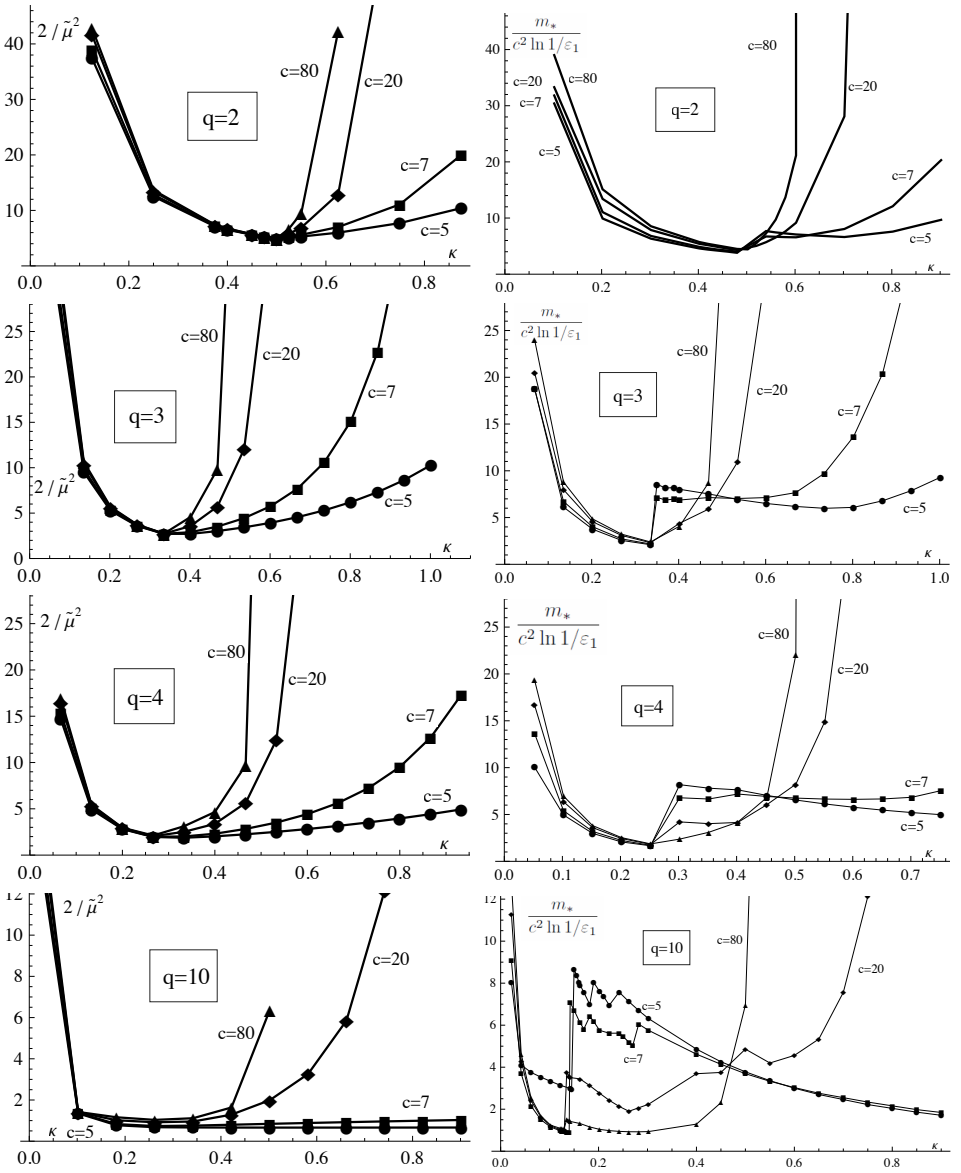


FIGURE 6.11: *Numerical results for the $\tilde{\mu}$-minimizing attack.* $\varepsilon_1 = 10^{-10}$. **Left:** *The Gaussian-limit code length constant* $\frac{2}{\tilde{\mu}^2}$ *as a function of* κ*, for various* q *and* c. **Right:** *The sufficient code length* m_**, scaled by the factor* $c^2 \ln(1/\varepsilon_1)$ *for easy comparison to the Gaussian limit.*

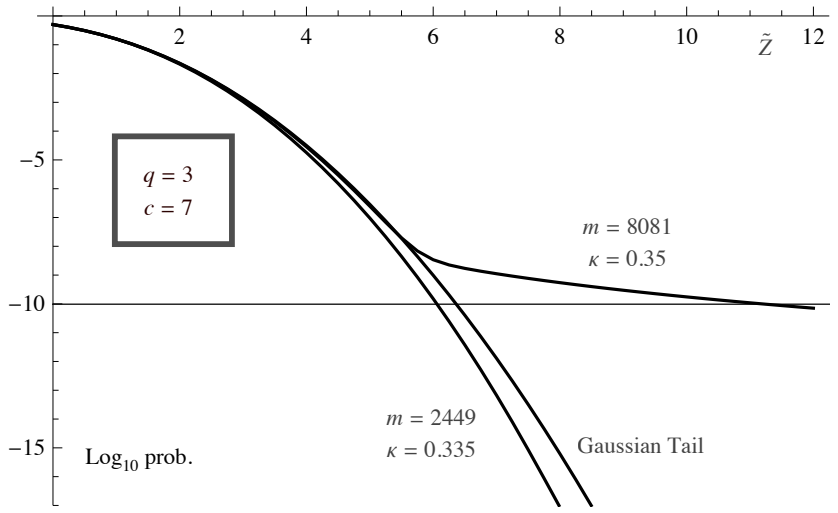


FIGURE 6.12: *Accusation probability for a fixed innocent user as a function of the (scaled) accusation threshold* $\tilde{Z} = Z/\sqrt{m}$. *The attack is the* $\tilde{\mu}$*-minimizing attack. The graph shows the Gaussian limit, and two parameter settings which correspond to 'before' and 'after' a sharp transition.*

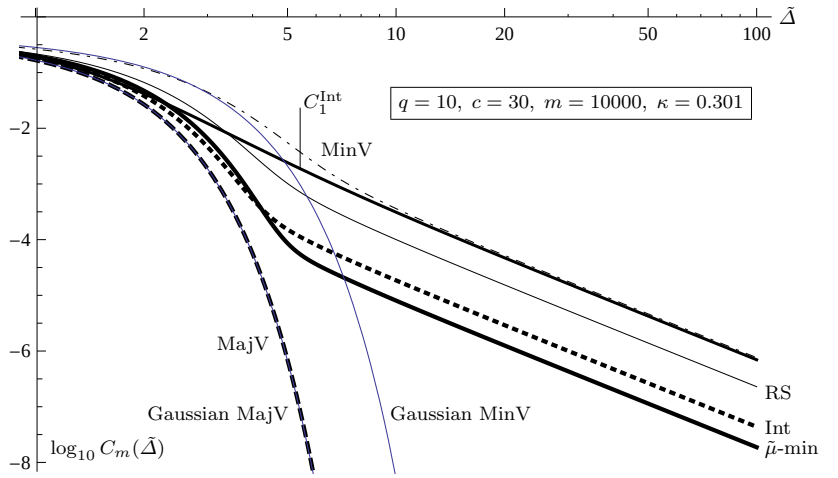
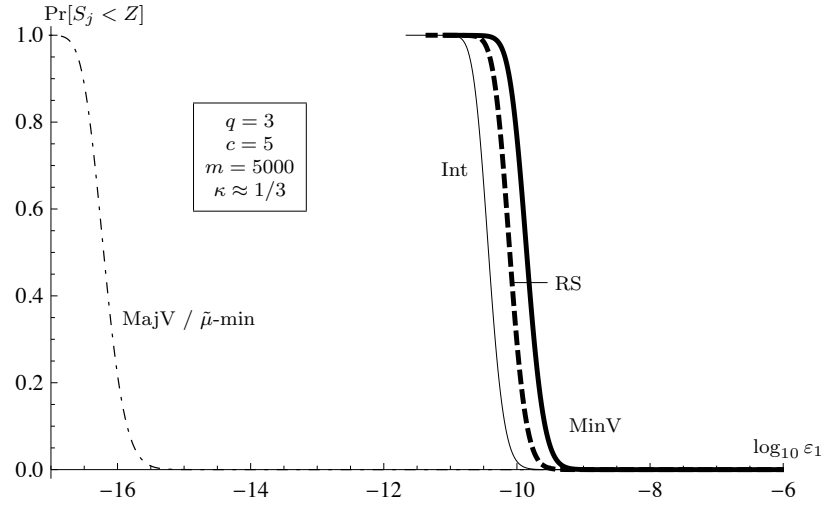
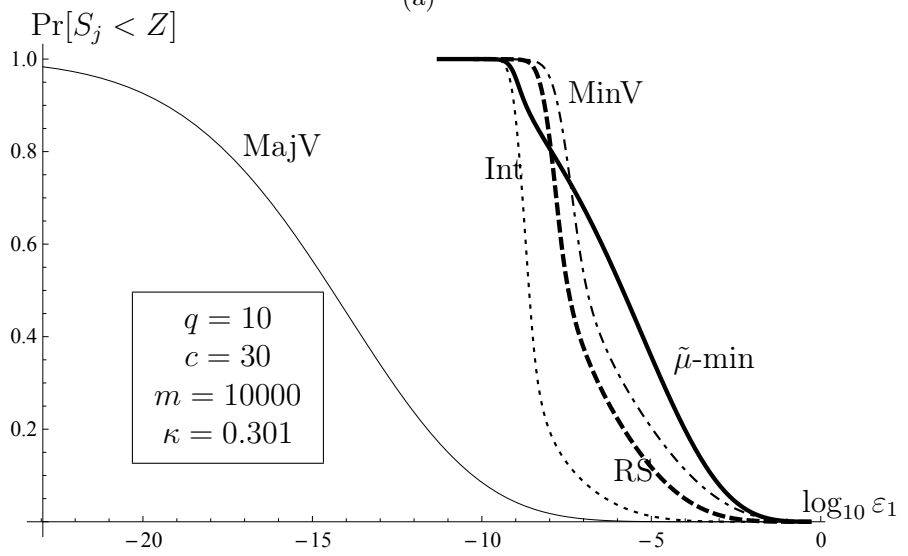


FIGURE 6.13: *Log-log plot of* $C_m(\tilde{\Delta})$ *for several strategies. The single-segment tail integral* $C_1(\tilde{\Delta})$ *for the Int attack is also shown. Two Gaussian tails are plotted: for the V value corresponding to the MajV and MinV strategies. The MajV curve coincides with its Gaussian approximation.*



(a)



(b)

FIGURE 6.14: *Example 'ROC' curves. Our upper bound* $\Pr[S_j < Z]$ *on* P_{FN} *versus the probability* ε_1 *of accusing a fixed innocent user. The* ε_1 *data is taken from [34].*

7

Conclusions

The main topic of this thesis is the performances of collusion resistant codes in the context of audio-video watermarking. Watermarking is a content protection technique that can be used independent of other protection measures. Unique watermarks are hidden in content so that unauthorized redistribution of the content can be traced. The most powerful attack against this form of watermarking is the collusion attack: a coalition of users receives differently watermarked versions of the same content; by comparing their versions they obtain partial knowledge about the embedded watermark sequences, which allows for a more targeted attack.

The aim of *collusion resistant codes* is to provide watermark sequences that can resist coalition attacks, i.e. even after such an attack has taken place enough information is still present in the damaged watermark to trace at least one coalition member. Bias-based codes, also known as Tardos codes, were proposed by G. Tardos in 2003 [38]. Their asymptotic optimality $m \propto c_0^2$ has made them a popular topic of study. Work in this field has concentrated on different issues, e.g. improvements of the code construction and the decoder algorithm. The field has reached a certain maturity. For general alphabet size the asymptotic ($c \to \infty$) channel capacity for bias-based fingerprinting has been determined, as well as the combination (attack, bias distribution) constituting the corresponding saddlepoint. A simple decoder has been found achieving this asymptotic capacity. For small c, numerical methods are available to locate the saddlepoint, and joint decoders have been developed to improve the tracing efficiency.

In spite of all this progress, it has turned out to be surprisingly dif-

ficult to establish how well a scheme performs in the non-asymptotic regime. (And this is usually the regime of interest.) The performance is usually measured in term of the False Positive and False Negative probability. The two main approaches both yield unsatisfactory results. On the one hand, simulations have difficulty handling the required low FP probability; they take time on the order of $1/P_{\mathrm{FP}}^{\mathrm{global}}$. On the other hand, provable bounds overestimate the FP probability by orders of magnitude. Because of these problems it was difficult to use a Tardos code in practice: either there would be the danger of an FP rate turning out higher than expected, or resources would not be used well. Given the limited resources in typical audio/video content, the ability to determine the error rates accurately is crucial for making a system that can be deployed in practice.

In view of the situation sketched above, the main research question of this thesis was how to accurately determine error probabilities in Tardos codes. We restricted ourselves to an attack model that lends itself to analysis, and to the best known q-ary scheme at the beginning at the project. Thus the main question was

How to determine the actual error rates of the symmetric Tardos scheme [42] in the Restricted Digit Model?

7.1 Contributions

This thesis work has resulted in a new algorithm that quickly computes error probabilities as a function of all the system parameters and the attack strategy. In many cases the code can resist twice or more the number of attackers suggested by the provable bounds on the error rates. The results are summarized below in detail.

CSE method
We have developed a new method to compute the probability distribution of the scores (for innocent as well as guilty users) in the symmetric Tardos scheme. We call it 'Convolution and Series Expansion' (CSE). It is based on the convolution property of characteristic functions (Property 4.1, page 29): the probability distribution of a sum of independent random variables is obtained simply by multiplying their

Fourier-transformed pdfs (characteristic functions) and then doing a single reverse Fourier transform. Hence, if the pdf of a user's score in a single content segment is known, the pdf of his total score can be found.

The single-segment pdf is computed in Theorem 4.21 (innocent user, 'φ') and Theorem 4.30 (colluder, 'ψ'). The expressions obtained here are interesting in their own right, because they tell us how the attack strategy influences the tails of the pdf. The right tail of the innocent user pdf is especially interesting since it has an impact on the FP rate. It turns out that, of all the strategies we studied, Minority Voting causes the longest tail, while Majority Voting maximally shortens it. Furthermore, the single-segment pdf provides us with a valuable consistency check: the total pdf must have the same power-law in its tails as the single-segment pdf.

The Fourier transform of φ and ψ is computed in Section 4.2.3 and 4.3.4. The results are analytic complex-valued expressions containing hypergeometric ${}_1F_2$ functions. The $\tilde{\varphi}$ and $\tilde{\psi}$ are not entire functions of the Fourier variable k, making further analysis nontrivial.

Performing the final reverse Fourier transform turned out to be a difficult task. Our initial attempt to straightforwardly do a one-dimensional numerical integration of $\tilde{\varphi}^m$ failed because the integrand is oscillating very fast. Instead we decided to perform a series expansion of the integrand (as correction terms on the Gaussian curve), yielding an infinite sum of integrals that can be evaluated analytically. The small parameter in the expansion is $1/\sqrt{m}$. The result is given in Theorem 4.19 and Theorem 4.41. The procedure resembles an Edgeworth expansion, but with non-integer powers. Though the expansion is presented in analytical form, the task of finding all the parameters ω_t, ν_t and α_t is best not done by hand but automated, since it is quite complicated, involving a series expansion of numerous ${}_1F_2$ functions substituted into two successive Taylor series. We automated this procedure by writing a Mathematica program.

Compact parametrization of the attack strategy and pre-computations

During the work on the innocent-user single-segment pdf, we realized that the set of parameters $\theta_{y|\boldsymbol{\sigma}}$ describing the attack strategy can be reduced to $\Psi_b(\boldsymbol{x})$ (Section 3.2) when the strategy is symbol-symmetric, i.e. when the alphabet has no natural ordering. The $\Psi_b(\boldsymbol{x})$ stands for the probability that a symbol with tally b (given that such a symbol exists)

gets chosen by the attackers, conditioned on the other tallies $\boldsymbol{x}$. Furthermore, many formulas contain the expectation $\mathbb{E}_{\boldsymbol{x}|b}\Psi_b(\boldsymbol{x})$ for which we introduced the notation K_b. K_b is the probability that the colluders choose a symbol α given that α has tally b. With this notation, amongst others the pdfs φ and ψ and the average coalition score $\tilde{\mu}$ can be written in a compact form. There is a further advantage: for all the strategies that we wanted to investigate we were able to 'pre-compute' the K_b parameters (Section 5.2). We found a sum representation for K_b that (from a certain alphabet size onward) requires far fewer terms than the $\mathbb{E}_{\boldsymbol{x}|b}$ sum. This helps to significantly speed up the CSE method when the alphabet is not small.

Testing the CSE method

We subjected the CSE method to a number of tests. Most importantly, we did two consistency checks. First, the results of the CSE method agree with simulations whenever simulation are feasible. Second, the pdf tails properly follow the same power law as the single-segment pdf (Section 6.2 and 6.4).

Furthermore we studied the convergence of the expansion (4.41). We proved (Theorem 4.19) that the parameters $\omega_t(m)$ decrease as $m^{-\nu_t/6}$ or faster. However, that does not guarantee convergence. It is known that Edgeworth expansions are not always convergent, and indeed in our overview Table 6.1 we see a few cases where adding extra terms causes the series to *diverge* from the correct result. Furthermore, when m is too small the series does not produce the desired result at all (or at least not with a feasibly computable number of terms); this problem does not come as a surprise, since the CSE method uses $1/\sqrt{m}$ as the expansion parameter. The worst convergence occurs when the attack strategy is Majority Voting. We do not fully understand why this is the case. However, Majority Voting happens to have a very wide Gaussian regime, the largest of all the strategies we tested, so actually we hardly need the CSE here and can just use the Gaussian approximation.

Regarding the running time of our CSE implementation in Mathematica, we noticed the following. The computational effort is strongly dependent on several parameters, in particular ν. The strategies, instead, did not affect directly the computational effort thanks to the pre-computation of the K_b parameter[1]. The needed time to compute $R_m(\tilde{Z})$ or $C_m(\tilde{Z})$ for a fixed $\tilde{Z}$ could vary from few seconds to few minutes.

[1]Still the strategies affect indirectly the computational effort through ν, as shown

Investigation of the Tardos scheme using the CSE method
In Sections 6.3–6.5 we applied the CSE method to a large number of parameter combinations for the Majority Voting, Minority Voting, Interleaving, Random Symbol and $\tilde{\mu}$-minimizing attack. When plotted as a function of the threshold Z, the FP rate (and the guilty-user accusation probability C_m) is seen to have a Gaussian regime at small Z and then a transition to power-law behaviour in the tail. The point of transition depends on the attack strategy and shifts to larger Z when the code length m is increased.

For understanding the performance of the Tardos code it is crucial to look at the location of the standard threshold $\tilde{Z}_{\text{half}} = \sqrt{m}\tilde{\mu}/c$ (which suffices to keep the FN rate under control) compared to the transition point in the innocent-user plot. If $\tilde{Z}_{\text{half}}$ lies in the Gaussian part, then (i) the Gaussian approximation (Section 3.5) applies, and (ii) the $\tilde{\mu}$-minimizing attack is the strongest attack. If $\tilde{Z}_{\text{half}}$ lies in the tail, then Minority Voting is the strongest attack. (This summary of events is slightly complicated by the fact that the transition point actually depends on the attack strategy.) Increasing c causes $\tilde{Z}_{\text{half}}$ to move to the left while the P_{FP} plot hardly changes; this makes the situation 'more Gaussian'. Increasing m has two counter-acting effects: $\tilde{Z}_{\text{half}}$ moves to the right, but at the same time the Gaussian region becomes wider.

In Section 6.3.2 we compared five strategies for a large part of the parameter space in order to compare their strength and to chart where the pdf transitions lie. For a better understanding about how the two errors change simultaneously, we presented results also as ROC curves (Section 6.5). The ROC representation permits to point more easily which is the strongest strategy.

Investigation of the $\tilde{\mu}$-minimizing attack
contribution. asymptotically big m. It is also optimal whenever the Gaussian regime is entered. In Section 6.3.3 we dedicated a separate study on this attack, focusing in particular on its dependency on the parameter κ. For $\kappa < \frac{1}{2(q-1)}$, $\tilde{\mu}$-minimizing attack behaves as Majority Voting, while for $\kappa > \frac{1}{2}$ it behaves as Minority Voting. In between these values the behaviour becomes more complicated, as shown in Figure 6.11. The transition from Majority to Minority voting consists of complicated intermediate 'rankings' (as defined in Section 5.2.1) of b values.

in Table 6.1.

7.2 Limitations

The proposed CSE method and the way we use it has two noteworthy limitations. The first limitation concerns the strategies that can be written in the compact K_b form. The K_b formulation is possible only for strategies that have symbol symmetry, attacker symmetry and segment symmetry. As explained in Chapter 5, relevant (i.e. strong) attacks typically satisfy these symmetries because of (respectively) the symbol symmetry of the embedding method, equal risk sharing by the attackers, and large coalition size. Our focus on strong attacks excludes some strategies that surely will not be strong but that have interesting properties in other contexts, like for instance the all-1 attack (non-symbol-symmetric), typically needed in the group testing problem (Section 2.3), and the scapegoat attack (non-attacker-symmetric) studied in the dynamic Tardos scheme [20] in which a random attacker is sacrificed to save the rest of the coalition.

The CSE method is still able to handle a certain lack of symmetry in the following cases:

- *non-segment-symmetric attacks*: as shown in Section 4.4, the CSE method can handle mixed strategies, allowing it to be used in case of some non-segment-symmetric attacks.

- *non-symbol-symmetric attacks*: Even though it is impossible to formulate K_b in this case, $\theta_{y|\boldsymbol{\sigma}}$ is still well defined. It is relatively straightforward to derive expressions for φ and ψ based on $\theta_{y|\boldsymbol{\sigma}}$: e.g. (D.3) still applies. This will of course require more work than in the symmetric case.

The second limitation is the convergence of the CSE method. As discussed in Section 6.1 and shown in Table 6.1, the expansion parameter $1/\sqrt{m}$ can lead to a ill-defined series expansion whenever m is too small.

7.3 Future work

The work done is this thesis can be extended in many directions. First of all, the CSE method can be applied under different circumstances, e.g. for different decoders, attack models and strategies than those considered here.

- Oosterwijk et al. [29] recently found a simple decoder that is asymptotically capacity-achieving. We expect that it will completely supplant the score function (3.11). It will be interesting to see how the CSE performs when applied to their new score function.

- In this thesis the Restricted Digit Model was adopted because it is relatively simple to analyze. An obvious next step is to adapt the CSE to the more realistic Combined Digit Model. The main complication will probably be the introduction of extra averages in the computation of φ (4.45) due to the additional probabilistic degrees of freedom in the detector when it is faced with noise and symbol fusion.

- We have considered symbol-symmetric strategies, and we have seen that they allow for particularly efficient and compact formulas in terms of the parameters K_b. However, it would be interesting to also study strategies that do not have the symbol symmetry, e.g. the all-1 attack which, as was discussed in Section 2.3, links traitor tracing codes to Group Testing. Though it is impossible to formulate K_b parameters for the all-1 attack, it is relatively straightforward to derive expressions for φ and ψ: e.g. (D.3) still applies, hence it is possible to apply the CSE method.

- One of the most discussed topic in this thesis is the Gaussian approximation. We have shown that in the Gaussian regime many things change, in particular how the various attacks affect the users' score and the error probabilities. We have also shown that the width of the Gaussian region depends on all the parameters involved in the system. The knowledge of the point in which the curves switch from Gaussian to power-law behaviour determines which would be the best strategy for a specific threshold $\tilde{Z}$ and then which will be the worst scenario. A deeper study on the change of slope as function of the whole parameter set would let to know the regime widths without computing the entire curves.

Future research could also concentrate on improving the CSE method, especially its convergence properties. A closer study of Edgeworth-like expansions may provide a way to get tighter provable bounds, i.e. bounds that lie close to the CSE results.

Bibliography

[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] A. Charpentier, F. Xie, C. Fontaine, and T. Furon. Expectation maximization decoding of Tardos probabilistic fingerprinting code. In *Media Forensics and Security*, volume 7254 of *SPIE Proceedings*, page 72540, 2009.

[6] W. Feller. *An Introduction to Probability Theory and Its Applications*, volume 1. Wiley, January 1968. ISBN 0471257087. URL `http://www.amazon.ca/exec/obidos/redirect?tag=citeulike04-20{&}path=ASIN/0471257087`.

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

[8] 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.

[9] I. Gradshteyn and I. Ryzhik. *Table of Integrals, Series, and Products, 5th edition*. Academic Press, 1994.

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

[11] Y.-W. Huang and P. Moulin. Saddle-point solution of the fingerprinting capacity game under the marking assumption. In *Proceedings of the 2009 IEEE international conference on Symposium on Information Theory - Volume 4*, ISIT'09, pages 2256–2260, Piscataway, NJ, USA, 2009. IEEE Press. ISBN 978-1-4244-4312-3. URL `http://dl.acm.org/citation.cfm?id=1700967.1700985`. Extra version available in `http://arxiv.org/abs/0905.1375`.

[12] Y.-W. Huang and P. Moulin. Maximin optimality of the arcsine fingerprinting distribution and the interleaving attack for large coalitions. In *Information Forensics and Security (WIFS), 2010 IEEE International Workshop on*, pages 1–6, 2010. doi: 10.1109/WIFS.2010.5711451.

[13] 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.

[14] T. Kitagawa, M. Hagiwara, K. Nuida, H. Watanabe, and H. Imai. A group testing based deterministic tracing algorithm for a short random fingerprint code. In *Information Theory and Its Applications, 2008. ISITA 2008. International Symposium on*, pages 1–5, 2008. doi: 10.1109/ISITA.2008.4895500.

[15] J. Kolassa. *Series approximation methods in statistics*. Springer, 2006.

[16] M. Kuribayashi. Tardos's fingerprinting code over awgn channel. In *Information Hiding*, pages 103–117, 2010.

[17] M. Kuribayashi, N. Akashi, and M. Morii. On the systematic generation of Tardos's fingerprinting codes. In *MMSP 2008*, pages 748–753.

[18] T. Laarhoven and B. de Weger. Optimal symmetric Tardos traitor tracing schemes, 2011. `http://arxiv.org/abs/1107.3441`.

[19] T. Laarhoven and B. de Weger. Discrete distributions in the tardos scheme, revisited. In *1st ACM Workshop on Information Hiding and Multimedia Security (IHMMSec)*, 2013.

[20] T. Laarhoven, J. Doumen, P. Roelse, B. Škorić, and B. de Weger. Dynamic tardos traitor tracing schemes. *IEEE Transactions on Information Theory*, 2013.

[21] D. MacKay. *Information Theory, Inference and Learning Algorithms*. Cambridge University Press, 2003. ISBN 9780521642989. URL `http://books.google.nl/books?id=AKuMj4PN_EMC`.

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

[23] P. Meerwald and T. Furon. Group testing meets traitor tracing. In *Acoustics, Speech and Signal Processing (ICASSP), 2011 IEEE International Conference on*, pages 4204–4207, 2011. doi: 10.1109/ICASSP.2011.5947280.

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

[25] P. Moulin and J. O'Sullivan. Information-theoretic analysis of watermarking. In *Acoustics, Speech, and Signal Processing, 2000. ICASSP '00. Proceedings. 2000 IEEE International Conference on*, volume 6, pages 3630–3633 vol.6, 2000. doi: 10.1109/ICASSP.2000.860188.

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

[27] K. Nuida, M. Hagiwara, H. Watanabe, and H. Imai. Optimization of tardos's fingerprinting codes in a viewpoint of memory amount. In T. Furon, F. Cayre, G. Doërr, and P. Bas, editors, *Information Hiding*, volume 4567 of *Lecture Notes in Computer Science*, pages 279–293. Springer Berlin Heidelberg, 2007. ISBN 978-3-540-77369-6. doi: 10.1007/978-3-540-77370-2_19. URL `http://dx.doi.org/10.1007/978-3-540-77370-2_19`.

[28] K. Nuida, S. Fujitsu, M. Hagiwara, T. Kitagawa, H. Watanabe, K. Ogawa, and H. Imai. An improvement of discrete tardos fingerprinting codes. *Designs, Codes and Cryptography*, 52(3):339–362, 2009. ISSN 0925-1022. doi: 10.1007/s10623-009-9285-z. URL `http://dx.doi.org/10.1007/s10623-009-9285-z`.

[29] J.-J. Oosterwijk, B. Škoric, and J. Doumen. Optimal suspicion functions for tardos traitor tracing schemes.

[30] C. Peikert, A. Shelat, and A. Smith. Lower bounds for collusion-secure fingerprinting. In *SODA 2003*, pages 472–478.

[31] A. Prudnikov, Y. Brychkov, and O. Marichev. *Integrals and Series, 4th printing*, volume 1. CRC, 1998.

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

[33] A. Simone and B. Škorić. Accusation probabilities in tardos codes: beyond the gaussian approximation. *Designs, Codes and Cryptography*, 63(3):379–412, 2012. Longer version available in `http://eprint.iacr.org/2010/472`.

[34] A. Simone and B. Škorić. False Positive probabilities in q-ary Tardos codes: comparison of attacks. `http://eprint.iacr.org/2012/522`, 2012.

[35] A. Simone and B. Škorić. False Negative probabilities in Tardos codes. `http://eprint.iacr.org/2012/667`, 2012.

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

[37] D. R. Stinson, T. V. Trung, and R. Wei. Secure frameproof codes, key distribution patterns, group testing algorithms and related structures. *Journal of Statistical Planning and Inference*, 86: 595–617, 1997.

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

[39] G. Tardos. Optimal probabilistic fingerprint codes. *J. ACM*, 55(2), 2008.

[40] B. Škorić and J. Oosterwijk. Binary and q-ary Tardos codes, revisited. `http://eprint.iacr.org/2012/249`.

[41] B. Škorić, S. Katzenbeisser, H. Schaathun, and M. Celik. Tardos fingerprinting codes in the combined digit model. In *IEEE Workshop on Information Forensics and Security (WIFS) 2009*, pages 41–45.

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

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

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

Appendix

A Proof of Lemma 4.2

First we write for all $\alpha \neq y$: $p_\alpha = (1-p_y)s_\alpha$, with $s_\alpha \in [0,1]$. This gives

$$\mathrm{d}^q \boldsymbol{p} = \mathrm{d}p_y (1-p_y)^{q-1} \mathrm{d}^{q-1}\boldsymbol{s}, \tag{A.1}$$

$$\prod_{\beta\in\mathcal{Q}} p_\beta^{-1+\kappa} = p_y^{-1+\kappa}(1-p_y)^{(q-1)(-1+\kappa)} \prod_{\alpha\in\mathcal{Q}\setminus\{y\}} s_\alpha^{-1+\kappa}, \tag{A.2}$$

$$\delta\left(1-\sum_{\beta\in\mathcal{Q}} p_\beta\right) = \delta\left([1-p_y]\left[1-\sum_{\alpha\in\mathcal{Q}\setminus\{y\}} s_\alpha\right]\right) \tag{A.3}$$

$$= (1-p_y)^{-1}\delta\left(1-\sum_{\alpha\in\mathcal{Q}\setminus\{y\}} s_\alpha\right). \tag{A.4}$$

Then we split the q-dimensional integration $\int \mathrm{d}^q\boldsymbol{p}\, F(\boldsymbol{p})r(\boldsymbol{p})$ as follows,

$$\mathbb{E}_{\boldsymbol{p}}[r(\boldsymbol{p})] = \frac{1}{B(\kappa\mathbf{1}_q)} \int_0^1 \mathrm{d}p_y\, p_y^{-1+\kappa} \int_0^{1-p_y} \mathrm{d}^{q-1}\boldsymbol{p}_{\setminus y} \cdot \delta\left(1-p_y-\sum_{\beta\in\mathcal{Q}\setminus\{y\}} p_\beta\right) \boldsymbol{p}_{\setminus y}^{-1+\kappa} r(\boldsymbol{p}). \tag{A.5}$$

Combining all these ingredients, we find

$$\mathbb{E}_{\boldsymbol{p}}[r(\boldsymbol{p})] = \frac{1}{B(\kappa\mathbf{1}_q)} \int_0^1 \mathrm{d}p_y\, p_y^{-1+\kappa}(1-p_y)^{-1+\kappa[q-1]} \cdot \int_0^1 \mathrm{d}^{q-1}\boldsymbol{s}\, \delta\left(1-\sum_{\gamma\in\mathcal{Q}\setminus\{y\}} s_\gamma\right) \prod_{\beta\in\mathcal{Q}\setminus\{y\}} s_\beta^{-1+\kappa} r(\boldsymbol{p}). \tag{A.6}$$

Combined with the fact that $B(\kappa \mathbf{1}_q) = B(\kappa, \kappa[q-1])B(\kappa \mathbf{1}_{q-1})$, these steps yield the end result. □

B Proof of Theorem 4.18

For an innocent user j we have $\Pr[S_j > Z] = \Pr\left[\sum_{i=1}^m S_j^{(i)} > Z\right]$. The 'Pr' refers to the whole set of random variables $\boldsymbol{p}$, $\boldsymbol{\sigma}$, y. The terms $S_j^{(i)}$ are independent, identically distributed random variables. This allows us to write

$$\Pr[S_j > Z] = \int_{-\infty}^{\infty} \mathrm{d}u_1 \varphi(u_1) \cdots \int_{-\infty}^{\infty} \mathrm{d}u_m \varphi(u_m)\, \Theta(u_1 + \cdots + u_m - Z). \tag{B.1}$$

Here Θ is the Heaviside step function. Next we use a well known integral representation of the step function,

$$\Theta(x) = \lim_{\eta \downarrow 0} \frac{1}{2\pi i} \int_{-\infty}^{\infty} \mathrm{d}\lambda\, \frac{e^{i\lambda x}}{\lambda - i\eta}. \tag{B.2}$$

Substituting (B.2) into (B.1) and rearranging the order of the integrations, we get

$$\Pr[S_j > Z] = \lim_{\eta \downarrow 0} \int_{-\infty}^{\infty} \frac{\mathrm{d}\lambda}{2\pi i}\, \frac{e^{-i\lambda Z}}{\lambda - i\eta} \prod_{a=1}^{m} \left[\int_{-\infty}^{\infty} \mathrm{d}u_a\, \varphi(u_a) e^{i\lambda u_a} \right] \tag{B.3}$$

$$= \lim_{\eta \downarrow 0} \int_{-\infty}^{\infty} \frac{\mathrm{d}\lambda}{2\pi i}\, \frac{e^{-i\lambda Z}}{\lambda - i\eta} \left[\tilde{\varphi}(-\lambda)\right]^m \tag{B.4}$$

$$= -\lim_{\eta \downarrow 0} \int_{-\infty}^{\infty} \frac{\mathrm{d}k}{2\pi i}\, \frac{e^{ikZ/\sqrt{m}}}{k + i\eta} \left[\tilde{\varphi}(\frac{k}{\sqrt{m}})\right]^m. \tag{B.5}$$

In the last line of (B.5) we changed the integration variable to $k = -\lambda\sqrt{m}$ in order to get the 'scaled' threshold $Z/\sqrt{m}$ in the integrand, which makes it easier to visualize the result using Fig. 3.1.

We define $D(k) = (2\pi)^{-1} e^{ikZ/\sqrt{m}} \left[\tilde{\varphi}(\frac{k}{\sqrt{m}})\right]^m$ for brevity and write $D(k) = D_{\text{even}}(k) + D_{\text{odd}}(k)$. The power expansion of D_{odd} around $k = 0$ has dominant term k^a, where $a > 0$ (Corollary 4.17). We write

$$\lim_{\eta \downarrow 0} \int_{-\infty}^{\infty} \mathrm{d}k\, \frac{D(k)}{k + i\eta} = \lim_{\eta \downarrow 0} \int_{-\infty}^{\infty} \mathrm{d}k\, \frac{(k - i\eta) D(k)}{k^2 + \eta^2} \tag{B.6}$$

$$= \lim_{\eta \downarrow 0} \int_{-\infty}^{\infty} \mathrm{d}k\, \frac{k D_{\text{odd}}(k)}{k^2 + \eta^2} - i\pi D(0). \tag{B.7}$$

Here we made use of a standard representation of the delta function,

$$\delta(k) = \frac{1}{\pi} \lim_{\eta \to 0} \eta/(k^2 + \eta^2). \tag{B.8}$$

We also used the fact that in the remaining integration the D_{even} vanishes since it gets multiplied by an odd function of k. Then we use that $a > 0$ in the power series of D_{odd} (from Corollary 4.17 together with Corollary 4.15). This causes the integrand to behave like k^{-1+a} in the limit $\eta \to 0$, i.e. the integral near $k = 0$ is convergent even when η is precisely zero. Thus we can set $\eta = 0$ in this integral.

$$\Pr[S_j > Z] = i \lim_{\eta \downarrow 0} \int_{-\infty}^{\infty} \mathrm{d}k \, \frac{D(k)}{k + i\eta} = i \int_{-\infty}^{\infty} \mathrm{d}k \, \frac{D(k)}{k} + \pi D(0). \tag{B.9}$$

After substituting $D(k)$ with its definition and $D(0)$ with $(2\pi)^{-1}$ (obtained easily with Corollary 4.17) the result is given. □

C Proof of Theorem 4.19

We start from Corollary 4.17 and write a general power series expansion,

$$\tilde{\varphi}(k) = 1 - \tfrac{1}{2}k^2 + \sum_{t=0}^{\infty} \gamma_t |k|^{r_t}, \tag{C.1}$$

where the $r_t \geq 3$ are powers and the $\gamma_t \in \mathbb{C}$ are coefficients of the form $i^{\beta_t \operatorname{sgn} k}$ times a real factor. In this expression the desired relation $\tilde{\varphi}(-k) = [\tilde{\varphi}(k)]^*$ evidently holds, and the properties $\tilde{\varphi}(0) = 1$, $\tilde{\varphi}'(0) = 0$, $\tilde{\varphi}''(0) = -1$, $|\tilde{\varphi}'''(0)| < \infty$ are clearly present. Then we write

$$\left[\tilde{\varphi}(\frac{k}{\sqrt{m}})\right]^m = \exp\left[m \ln \tilde{\varphi}(\frac{k}{\sqrt{m}})\right] = e^{-\frac{1}{2}k^2} \exp\left[m \sum_{t=0}^{\infty} (\frac{|k|}{\sqrt{m}})^{r'_t} \delta_t\right], \tag{C.2}$$

where the powers $r'_t \geq 3$ and coefficients $\delta_t \propto i^{\beta'_t \operatorname{sgn} k}$ are obtained (laboriously) by substituting (J.1) into the Taylor series for the logarithm, $\ln(1+\varepsilon) = \varepsilon - \varepsilon^2/2 + \varepsilon^3/3 - \varepsilon^4/4 + \cdots$. It is worth noting that m disappears from the k^2 term, but not from the others. Eq. (4.40) is obtained from (J.2) by using the Taylor series for the exp function,

$$\exp \varepsilon = 1 + \varepsilon + \varepsilon^2/2! + \varepsilon^3/3! + \cdots \tag{C.3}$$

and (again laboriously) collecting terms with equal powers of k.

Since we started out with powers $r_t \geq 3$, we end up with powers $\nu_t \geq 3$. A power $|k|^{\nu_t}$ may occur together with many different powers of m. This is seen as follows. The series expansion of $\ln \tilde{\varphi}(k/\sqrt{m})$ is a power series in $|k|/\sqrt{m}$. Then the logarithm is multiplied by m, and a power $|k|^{r'}$ always occurs together with $m^{1-r'/2}$. Next, the k-expansion of exp mixes up the powers of m. For instance, the power k^6 occurs as $m\delta_{(6)}(|k|/\sqrt{m})^6 \propto k^6 m^{-2}$ but also as a term $[m\delta_{(3)}(|k|/\sqrt{m})^3]^2/2! \propto k^6 m^{-1}$. Here we defined $\delta_{(x)}$ such that $r'_{(x)} = x$.

The 'worst case' (many factors m resulting from high powers of ε in (J.3)) occurs when ν_t is a multiple of 3, say $\nu_t = 3j$; there the power k^{3j} can be built up from a term $[m\delta_{(3)}(|k|/\sqrt{m})^3]^j/j!$, which is proportional to $k^{3j}m^{j-3j/2} = k^{\nu_t}m^{-\nu_t/6}$. All the j factors scale as $m(|k|/\sqrt{m})^3 = |k|^3/\sqrt{m}$. This is the least negative power of m that can occur relative to the power of k. For other powers ν_t, the 'building blocks' from which k^{ν_t} is built up cannot all scale in this way; at least one of the factors has faster decay.[2] This proves the statement about the at least $m^{-\nu_t/6}$ decay.

Finally, (4.41) follows by applying Lemma 4.11 and Corollary 4.13 to evaluate the integrals that arise when (4.40) is substituted into Theorem 4.18. □

D Proof of Theorem 4.21

We start by considering the probability of a certain accusation value u occurring for an innocent user, for fixed $\boldsymbol{p}$ and y. (We omit all column indices.) There are only two discrete possibilities: (i) $g_1(p_y)$ if the user's symbol is y; this occurs with probability p_y; (ii) $g_0(p_y)$ if the user's symbol is not y; this occurs with probability $1 - p_y$. Hence we can write this distribution as a sum of two delta peaks as follows,

$$\varphi(u|\boldsymbol{p}, y) = p_y\delta(u - g_1(p_y)) + (1 - p_y)\delta(u - g_0(p_y)). \tag{D.1}$$

The full $\varphi(u)$, without conditioning, is obtained by taking the expectation over y and $\boldsymbol{p}$. Since the expectation over y involves the parameters $\theta_{y|\boldsymbol{\sigma}}$,

[2] For instance, the least negative power of m multiplying k^7 is obtained from the ε^2 term in (J.3) and is given by $2[m\delta_{(3)}(|k|/\sqrt{m})^3][m\delta_{(4)}(|k|/\sqrt{m})^4]/2! \propto [|k|^3/\sqrt{m}][|k|^4/m]$.

the expectation over $\boldsymbol{\sigma}$ has to be done as well.

$$\varphi(u) = \mathbb{E}_{\boldsymbol{p}} \mathbb{E}_{\boldsymbol{\sigma}|\boldsymbol{p}} \sum_{y \in \mathcal{Q}} \theta_{y|\boldsymbol{\sigma}} \; \varphi(u|\boldsymbol{p}, y). \tag{D.2}$$

Next we note that $\varphi(u|\boldsymbol{p}, y)$ depends only on p_y. Hence we can write $\varphi(u|p_y)$, and

$$\varphi(u) = \sum_{y \in \mathcal{Q}} \mathbb{E}_{p_y} \mathbb{E}_{\boldsymbol{\sigma}|p_y} \theta_{y|\boldsymbol{\sigma}} \; \varphi(u|p_y) = \sum_{y \in \mathcal{Q}} \mathbb{E}_{p_y} \mathbb{E}_{\sigma_y|p_y} \mathbb{E}_{\boldsymbol{\sigma}_{\setminus y}|\sigma_y} \theta_{y|\boldsymbol{\sigma}} \; \varphi(u|p_y). \tag{D.3}$$

Now we use $\mathbb{E}_{\boldsymbol{\sigma}_{\setminus y}|\sigma_y} \theta_{y|\boldsymbol{\sigma}} = K_{\sigma_y}$, the binomial form (4.16) of $\mathbb{E}_{\sigma_y|p_y}$ and the marginal distribution of p_y (Lemma 4.4). The dummy summation variable σ_y is replaced by the notation b in order to stress the fact that it does not depend on y. Substitution of all these ingredients gives

$$\varphi(u) = \sum_{y \in \mathcal{Q}} \int_0^1 \mathrm{d}p_y \; f(p_y) \sum_{b=0}^{c} \binom{c}{b} p_y^b (1-p_y)^{c-b} K_b \; \varphi(u|p_y) \tag{D.4}$$

$$= \frac{q}{B(\kappa, \kappa[q-1])} \sum_{b=1}^{c} \binom{c}{b} K_b \int_0^1 \mathrm{d}p_y \; p_y^{-1+\kappa+b} (1-p_y)^{-1+\kappa[q-1+]c-b} \varphi(u|p_y). \tag{D.5}$$

In the last line we have used that $K_0 = 0$ and that the integral over p_y yields the same result for every y. In order to evaluate the p_y-integral we have to rewrite the delta functions of (D.1) into the form $\delta(p_y - \cdots)$. We use the rule

$$\delta(u - w(p)) = \frac{\delta(p - w^{\mathrm{inv}}(u))}{|\mathrm{d}w/\mathrm{d}p|} \tag{D.6}$$

for any monotonic function $w(p)$, which yields

$$\begin{aligned} \delta(u - g_1(p)) &= \Theta(u) \frac{2u}{(1+u^2)^2} \delta\left(p - \frac{1}{1+u^2}\right), \\ \delta(u - g_0(p)) &= \Theta(-u) \frac{2|u|}{(1+u^2)^2} \delta\left(p - \frac{u^2}{1+u^2}\right). \end{aligned} \tag{D.7}$$

After some algebra, it is then seen that the p_y-integral evaluates to

$$\frac{2}{(1+u^2)^{c+\kappa q+1}} \left[\Theta(u) (u^2)^{\kappa[q-1]+c-\sigma_y-\frac{1}{2}} + \Theta(-u) (u^2)^{\kappa+\sigma_y-\frac{1}{2}} \right]. \tag{D.8}$$

Splitting φ into a part containing $\Theta(u)$ and a part containing $\Theta(-u)$ finally yields the end result. □

E Proof of Theorem 4.29

The guilty user's symbol is denoted as X. The one-segment score is either $g_0(p_y)$ (when $X \neq y$) or $g_1(p_y)$ (when $X = y$). Since no other values are possible, the probability distribution at given $\boldsymbol{p}$ will consist of delta-function peaks. Each peak is multiplied by the probability that the corresponding event occurs

$$\psi_-(u|\boldsymbol{p}) = \textstyle\sum_{y\in\mathcal{Q}} \delta\left(u - g_0(p_y)\right) \Pr[u = g_0(p_y)|\boldsymbol{p}] \tag{E.1}$$
$$\psi_+(u|\boldsymbol{p}) = \textstyle\sum_{y\in\mathcal{Q}} \delta\left(u - g_1(p_y)\right) \Pr[u = g_1(p_y)|\boldsymbol{p}]. \tag{E.2}$$

Notice that

$$\Pr[u = g_0(p_y)|\boldsymbol{p}] = \Pr[X \neq y \wedge Y = y|\boldsymbol{p}] \tag{E.3}$$
$$\Pr[u = g_1(p_y)|\boldsymbol{p}] = \Pr[X = y \wedge Y = y|\boldsymbol{p}] \tag{E.4}$$

and that

$$\Pr[X \neq y \wedge Y = y|\boldsymbol{p}] + \Pr[X = y \wedge Y = y|\boldsymbol{p}] = \Pr[Y = y|\boldsymbol{p}] = \tau_{y|\boldsymbol{p}}. \tag{E.5}$$

Next step is to compute $\Pr[u = g_1(p_y)|\boldsymbol{p}]$ in (E.2). Let be $\boldsymbol{e}_y$ a q-ary vector entirely set to 0 except for the y-th element that is instead equal to 1.

$$\begin{aligned}\Pr[u = g_1(p_y)|\boldsymbol{p}] &= \Pr[X_{ji} = y]\Pr[Y = y|X_{ji} = y, \boldsymbol{p}] \\ &= p_y \sum_{\boldsymbol{\sigma}\in\mathcal{S}_{qc}} \binom{c-1}{\boldsymbol{\sigma} - \boldsymbol{e}_y} \boldsymbol{p}^{\boldsymbol{\sigma}-\boldsymbol{e}_y} \theta_{y|\boldsymbol{\sigma}}.\end{aligned} \tag{E.6}$$

The last equation is obtained as follows: $\Pr[X_{ji} = y] = p_y$; $\Pr[Y = y|X_{ji} = y, \boldsymbol{p}]$ is equal to the sum over all the possible $\boldsymbol{\sigma}$ vectors that have at least one occurrence of y (expressed with the condition $\sigma_y > 0$). Knowing that $X_{ji} = y$, the multinomial factor is needed to count the remaining $c - 1$ attacker symbols in $\boldsymbol{\sigma}$, subtracting 1 from σ_y (using the $\boldsymbol{e}_y$ vector).

$$\Pr[u = g_1(p_y)|\boldsymbol{p}] = \sum_{\boldsymbol{\sigma}\in\mathcal{S}_{qc}} \frac{\sigma_y}{c} \binom{c}{\boldsymbol{\sigma}} \boldsymbol{p}^{\boldsymbol{\sigma}} \theta_{y|\boldsymbol{\sigma}}. \tag{E.7}$$

In the last equation we used $\boldsymbol{p}^{\boldsymbol{\sigma}} = p_y \boldsymbol{p}^{\boldsymbol{\sigma}-\boldsymbol{e}_y}$ and $\binom{c-1}{\boldsymbol{\sigma}-\boldsymbol{e}_y} = \frac{\sigma_y}{c}\binom{c}{\boldsymbol{\sigma}}$. Then the condition $\sigma_y > 0$ becomes superfluous and (4.63) trivially follows.

Notice that

$$p_y \frac{\partial T_{y|\boldsymbol{p}}}{\partial p_y} = p_y \frac{\partial}{\partial p_y} \sum_{\boldsymbol{\sigma} \in \mathcal{S}_{qc}} \binom{c}{\boldsymbol{\sigma}} \boldsymbol{p}^{\boldsymbol{\sigma}} \theta_{y|\boldsymbol{\sigma}} \tag{E.8}$$

$$= \sum_{\boldsymbol{\sigma} \in \mathcal{S}_{qc}} \binom{c}{\boldsymbol{\sigma}} \theta_{y|\boldsymbol{\sigma}} p_y \frac{\partial \boldsymbol{p}^{\boldsymbol{\sigma}}}{\partial p_y} \tag{E.9}$$

$$= \sum_{\boldsymbol{\sigma} \in \mathcal{S}_{qc}} \binom{c}{\boldsymbol{\sigma}} \theta_{y|\boldsymbol{\sigma}} \sigma_y \boldsymbol{p}^{\boldsymbol{\sigma}} \tag{E.10}$$

proving that (4.64)=(4.63) and (4.62)=(4.61). Finally, from (E.5) combined with (E.4) we have

$$\psi_-(u|\boldsymbol{p}) = \sum_{y \in \mathcal{Q}} \delta\left(u - g_0(p_y)\right) \left(\tau_{y|\boldsymbol{p}} - \Pr[X = y \wedge Y = y|\boldsymbol{p}]\right). \tag{E.11}$$

This, together with (E.7), completes the proof. □

F Proof of Theorem 4.30

The full $\psi(u)$, without conditioning, is obtained by taking the expectation over $\boldsymbol{p}$ of (4.61)+(4.63).

$$\psi(u) = \mathbb{E}_{\boldsymbol{p}}[\psi(u|\boldsymbol{p})] = \Theta(-u)\mathbb{E}_{\boldsymbol{p}}[\psi_-(u|\boldsymbol{p})] + \Theta(u)\mathbb{E}_{\boldsymbol{p}}[\psi_+(u|\boldsymbol{p})]. \tag{F.1}$$

We first prove (4.65) starting from $\mathbb{E}_{\boldsymbol{p}}[\psi_-(u|\boldsymbol{p})]$ with $\psi_-(u|\boldsymbol{p})$ as given in (4.61).

$$\mathbb{E}_{\boldsymbol{p}}[\psi_-(u|\boldsymbol{p})] = \mathbb{E}_{\boldsymbol{p}} \left[\sum_{y \in \mathcal{Q}} \delta\left(u - g_0(p_y)\right) \sum_{\boldsymbol{\sigma} \in \mathcal{S}_{qc}} \binom{c}{\boldsymbol{\sigma}} \left(1 - \frac{\sigma_y}{c}\right) \boldsymbol{p}^{\boldsymbol{\sigma}} \theta_{y|\boldsymbol{\sigma}} \right] \tag{F.2}$$

$$= \sum_{y \in \mathcal{Q}} \sum_{\boldsymbol{\sigma} \in \mathcal{S}_{qc}} \binom{c}{\boldsymbol{\sigma}} \left(1 - \frac{\sigma_y}{c}\right) \theta_{y|\boldsymbol{\sigma}} \mathbb{E}_{\boldsymbol{p}} \left[\delta\left(u - g_0(p_y)\right) \boldsymbol{p}^{\boldsymbol{\sigma}}\right]. \tag{F.3}$$

From Lemma 4.2 and $\boldsymbol{p}_{\setminus y}^{\boldsymbol{\sigma}_{\setminus y}} = (1-p_y)^{c-\sigma_y} \prod_{\alpha\in\mathcal{Q}\setminus\{y\}} t_\alpha^{\sigma_\alpha}$ we have that

$$\mathbb{E}_{\boldsymbol{p}}\left[\delta\left(u-g_0(p_y)\right)\boldsymbol{p}^{\boldsymbol{\sigma}}\right] = \frac{1}{B(\kappa\mathbf{1}_q)}\int_0^1 \mathrm{d}p_y\,\delta\left(u-g_0(p_y)\right)p_y^{\sigma_y+\kappa-1}$$
$$\cdot\,(1-p_y)^{c-\sigma_y+\kappa[q-1]-1}\int_0^1 \mathrm{d}^{q-1}\boldsymbol{t}\;\delta(1-\sum_{\beta\in\mathcal{Q}\setminus\{y\}} t_\beta)\prod_{\alpha\in\mathcal{Q}\setminus\{y\}} t_\alpha^{\sigma_\alpha+\kappa-1}. \quad \text{(F.4)}$$

The second integral in (F.4) evaluates to $B(\boldsymbol{\sigma}_{\setminus y}+\kappa\mathbf{1}_{q-1})$, having the structure shown in Def. 3.1. In order to evaluate the p_y-integral we have to rewrite the delta function into the form $\delta\left(p_y-\cdots\right)$. We use the rule

$$\delta\left(u-w(p)\right) = \frac{\delta\left(p-w^{\mathrm{inv}}(u)\right)}{|\mathrm{d}w/\mathrm{d}p|} \quad \text{(F.5)}$$

for any monotonic function $w(p)$. This gives

$$\delta\left(u-g_0(p)\right) = \Theta(-u)\frac{2|u|}{\left(1+u^2\right)^2}\delta\left(p-\frac{u^2}{1+u^2}\right). \quad \text{(F.6)}$$

We substitute (F.6) into (F.4) and solve the integral

$$\begin{aligned}\mathbb{E}_{\boldsymbol{p}}\left[\delta\left(u-g_0(p_y)\right)\boldsymbol{p}^{\boldsymbol{\sigma}}\right] &= 2|u|\Theta(-u)\left(\frac{1}{1+u^2}\right)^2\frac{B(\boldsymbol{\sigma}_{\setminus y}+\kappa\mathbf{1}_{q-1})}{B(\kappa\mathbf{1}_q)}\\ &\quad\cdot\int_0^1\mathrm{d}p_y\,\delta\left(p_y-\frac{u^2}{1+u^2}\right)p_y^{\sigma_y+\kappa-1}(1-p_y)^{c-\sigma_y+\kappa[q-1]-1}\\ &= 2|u|\Theta(-u)\left(\frac{1}{1+u^2}\right)^2\frac{B(\boldsymbol{\sigma}_{\setminus y}+\kappa\mathbf{1}_{q-1})}{B(\kappa\mathbf{1}_q)}\\ &\quad\cdot\left(\frac{u^2}{1+u^2}\right)^{\sigma_y+\kappa-1}\left(\frac{1}{1+u^2}\right)^{c-\sigma_y+\kappa[q-1]-1}\\ &= 2\Theta(-u)\frac{B(\boldsymbol{\sigma}_{\setminus y}+\kappa\mathbf{1}_{q-1})}{B(\kappa\mathbf{1}_q)}\frac{(u^2)^{\sigma_y+\kappa-1/2}}{(1+u^2)^{c+\kappa q}}. \end{aligned} \quad \text{(F.7)}$$

Substituting (F.7) into (F.3) we have

$$\mathbb{E}_{\boldsymbol{p}}[\psi_-(u|\boldsymbol{p})] = 2\sum_{y\in\mathcal{Q}}\sum_{\boldsymbol{\sigma}\in\mathcal{S}_{qc}}\binom{c}{\boldsymbol{\sigma}}\left(1-\frac{\sigma_y}{c}\right)\frac{B(\boldsymbol{\sigma}_{\setminus y}+\kappa\mathbf{1}_{q-1})}{B(\kappa\mathbf{1}_q)}\frac{(u^2)^{\sigma_y+\kappa-1/2}}{(1+u^2)^{c+\kappa q}}\theta_{y|\boldsymbol{\sigma}}. \quad \text{(F.8)}$$

Now we change the summations as follows: the $\sum_{\boldsymbol{\sigma}}$ can be written as $\sum_b \sum_{\boldsymbol{x}}$ with $b = \sigma_y$ and $\boldsymbol{x} = \boldsymbol{\sigma}_{\setminus y}$, so $\theta_{y|\boldsymbol{\sigma}} = \Psi_b(\boldsymbol{x})$. Then the summand is a function of only b and $\boldsymbol{x}$, which allows us to write

$$\sum_y \sum_{\boldsymbol{\sigma}} \binom{c}{\boldsymbol{\sigma}} \to q \sum_{b=0}^{c} \sum_{\boldsymbol{x}} \binom{c}{b} \binom{c-b}{\boldsymbol{x}}. \tag{F.9}$$

Now we have

$$\mathbb{E}_{\boldsymbol{p}}[\psi_-(u|\boldsymbol{p})] = 2q \sum_{b=0}^{c} \sum_{\boldsymbol{x}} \binom{c}{b} \binom{c-b}{\boldsymbol{x}} \frac{c-b}{c} \frac{B(\boldsymbol{x} + \kappa \mathbf{1}_{q-1})}{B(\kappa \mathbf{1}_q)} \frac{(u^2)^{b+\kappa-1/2}}{(1+u^2)^{c+\kappa q}} \Psi_b(\boldsymbol{x}) \tag{F.10}$$

where

$$\begin{aligned}
&\sum_{\boldsymbol{x}} \binom{c}{b} \binom{c-b}{\boldsymbol{x}} \frac{B(\boldsymbol{x} + \kappa \mathbf{1}_{q-1})}{B(\kappa \mathbf{1}_q)} \Psi_b(\boldsymbol{x}) \\
&\qquad = \binom{c}{b} \sum_{\boldsymbol{x}} \binom{c-b}{\boldsymbol{x}} \frac{B(\boldsymbol{x} + \kappa \mathbf{1}_{q-1})}{B(\kappa \mathbf{1}_{q-1}) B(\kappa, \kappa[q-1])} \Psi_b(\boldsymbol{x}) && \text{(F.11)} \\
&\qquad = \binom{c}{b} \frac{1}{B(\kappa, \kappa[q-1])} \sum_{\boldsymbol{x}} \mathbb{P}_{q-1}(\boldsymbol{x}|b) \Psi_b(\boldsymbol{x}) && \text{(F.12)} \\
&\qquad = \binom{c}{b} \frac{K_b}{B(\kappa, \kappa[q-1])}. && \text{(F.13)}
\end{aligned}$$

In the last line we used K_b Definition (4.23). Substituting (F.13) into (F.10) and removing 0 and c from the b-range, we have (4.65).

We can use exactly the same steps to obtain (4.66) from (4.63). The only significant difference is the delta function which in this case will be

$$\delta\left(u - g_1(p)\right) = \Theta(u) \frac{2u}{\left(1+u^2\right)^2} \delta\left(p - \frac{1}{1+u^2}\right). \tag{F.14}$$

□

G Proof of Consistency Check 1

Integration of (4.65) and (4.66) gives

$$\int_{-\infty}^{\infty} \mathrm{d}u\, \psi(u) = \frac{2q}{B(\kappa, \kappa[q-1])} \sum_{b=1}^{c} \binom{c}{b} K_b \left[\left(1 - \frac{b}{c}\right) \int_{-\infty}^{0} \mathrm{d}u\, \frac{(u^2)^{b+\kappa-\frac{1}{2}}}{(1+u^2)^{c+\kappa q}} + \frac{b}{c} \int_{0}^{\infty} \mathrm{d}u\, \frac{(u^2)^{c-b+\kappa[q-1]-\frac{1}{2}}}{(1+u^2)^{c+\kappa q}} \right]. \quad \text{(G.1)}$$

Let be $\lambda := b + \kappa$ and $w := c - b + \kappa[q-1]$. Applying Lemma 4.8 we have

$$\frac{2q}{B(\kappa, \kappa[q-1])} \sum_{b=1}^{c} \binom{c}{b} K_b \left[\left(1 - \frac{b}{c}\right) \frac{1}{2} B(\lambda, w) + \frac{b}{c} \frac{1}{2} B(w, \lambda) \right] = \frac{q}{B(\kappa, \kappa[q-1])} \sum_{b=1}^{c} \binom{c}{b} K_b B(\lambda, w). \quad \text{(G.2)}$$

The result follows applying Lemma 4.5 followed by Lemma 4.7. □

H Proof of Consistency Check 2

Taking (4.65) and (4.66), the integral $\int_{-\infty}^{\infty} \mathrm{d}u\, u\psi(u)$ can be written as

$$\frac{2q}{B(\kappa, \kappa[q-1])} \sum_{b=1}^{c} \binom{c}{b} K_b \left[\left(1 - \frac{b}{c}\right) \int_{-\infty}^{0} \mathrm{d}u\, \frac{u\,(u^2)^{b+\kappa-\frac{1}{2}}}{(1+u^2)^{c+\kappa q}} + \frac{b}{c} \int_{0}^{\infty} \mathrm{d}u\, \frac{u\,(u^2)^{c-b+\kappa[q-1]-\frac{1}{2}}}{(1+u^2)^{c+\kappa q}} \right]. \quad \text{(H.1)}$$

Let $\lambda := b + \kappa - \frac{1}{2}$ and $w := c - b + \kappa[q-1] - \frac{1}{2}$. Applying Lemma 4.8 and the property $\Gamma(x+1) = x\Gamma(x)$ we have

$$\int_{-\infty}^{\infty} \mathrm{d}u\, u\psi(u) = \frac{2q}{B(\kappa, \kappa[q-1])} \sum_{b=1}^{c} \binom{c}{b} K_b \left[\left(\frac{b}{c} - 1\right) \frac{\Gamma(\lambda)\Gamma(w)\lambda}{2\Gamma(c+\kappa q)} + \frac{b}{c} \frac{\Gamma(\lambda)\Gamma(w)w}{2\Gamma(c+\kappa q)} \right]. \quad \text{(H.2)}$$

To obtain $\tilde{\mu}$ as in (4.59) we use Lemma 4.5 to substitute $\binom{c}{b} \frac{1}{B(\kappa, \kappa[q-1])}$ with $\frac{\mathbb{P}_1(b)}{B(\lambda+1/2, w+1/2)}$. After some simplifications, the result follows. □

I Proof of Lemma 4.31

The integral $\int_{-\infty}^{\infty} \mathrm{d}u\, u^2\psi(u)$ can be written as

$$\frac{2q}{B(\kappa,\kappa[q-1])}\sum_{b=1}^{c}\binom{c}{b}K_b\left[\left(1-\frac{b}{c}\right)\int_{-\infty}^{0}\mathrm{d}u\,\frac{u^2\,(u^2)^{b+\kappa-\frac{1}{2}}}{(1+u^2)^{c+\kappa q}}+\right.$$
$$\left.\frac{b}{c}\int_{0}^{\infty}\mathrm{d}u\,\frac{u^2\,(u^2)^{c-b+\kappa[q-1]-\frac{1}{2}}}{(1+u^2)^{c+\kappa q}}\right]. \quad \text{(I.1)}$$

Let $\lambda := c-b+\kappa[q-1]$ and $w := b+\kappa$. Applying Lemma 4.8 with (3.3) and the property $\Gamma(x+1) = x\Gamma(x)$, we get

$$\frac{2q}{B(\kappa,\kappa[q-1])}\sum_{b=1}^{c}\binom{c}{b}K_b\left[\left(1-\frac{b}{c}\right)\frac{\Gamma(\lambda-1)\Gamma(w-1)w(w-1)}{2\Gamma(c+\kappa q)}+\right.$$
$$\left.\frac{b}{c}\frac{\Gamma(\lambda-1)\Gamma(w-1)\lambda(\lambda-1)}{2\Gamma(c+\kappa q)}\right]. \quad \text{(I.2)}$$

Then using (4.20) we have

$$\int_{-\infty}^{\infty}\mathrm{d}u\,u^2\psi(u) = q\sum_{b=1}^{c}K_b\mathbb{P}_1(b)\left[\left(1-\frac{b}{c}\right)\frac{w}{\lambda-1}+\frac{b}{c}\frac{\lambda}{w-1}\right] \quad \text{(I.3)}$$

and (4.69) follows after some rewriting. □

J Proof of Theorem 4.19

We start from Corollary 4.36 and write a general power series expansion,

$$\tilde{\chi}(k) = 1-(V/2)k^2+\textstyle\sum_{t=0}^{\infty}\gamma_t|k|^{r_t}, \quad \text{(J.1)}$$

where the $r_t \geq 2+2\kappa$ are powers and the $\gamma_t \in \mathbb{C}$ are coefficients of the form $i^{\beta_t \operatorname{sgn} k}$ times a real factor. In this expression the desired relation $\tilde{\chi}(-k) = [\tilde{\chi}(k)]^*$ evidently holds, and the properties $\tilde{\chi}(0) = 1$, $\tilde{\chi}'(0) = 0$, $\tilde{\chi}''(0) = -V$ are clearly present. Then we write

$$[\tilde{\chi}(k/\sqrt{m})]^m = \exp[m\ln\tilde{\chi}(k/\sqrt{m})] = e^{-\frac{V}{2}k^2}\exp\left[m\textstyle\sum_{t=0}^{\infty}(\frac{|k|}{\sqrt{m}})^{r'_t}\delta_t\right], \quad \text{(J.2)}$$

where the powers $r'_t \geq 2 + 2\kappa$ and coefficients $\delta_t \propto i^{\beta'_t \operatorname{sgn} k}$ are obtained (laboriously) by substituting (J.1) into the Taylor series for the logarithm, $\ln(1+\varepsilon) = \varepsilon - \varepsilon^2/2 + \varepsilon^3/3 - \varepsilon^4/4 + \cdots$. It is worth noting that m disappears from the k^2 term, but not from the others. Eq. (4.86) is obtained from (J.2) by using the Taylor series for the exp function,

$$\exp \varepsilon = 1 + \varepsilon + \varepsilon^2/2! + \varepsilon^3/3! + \cdots \tag{J.3}$$

and (again laboriously) collecting terms with equal powers of k. Since we started out with powers $r_t \geq 2 + 2\kappa$, we end up with powers $\nu_t \geq 2 + 2\kappa$. Finally, (4.41) follows by applying Lemma 4.11 and Lemma 4.39 to evaluate the integrals that arise when (4.86) is substituted into Theorem 4.18. □

K Proof of Theorem 5.5

We start from (4.23), with $\mathbb{P}_{q-1}$ defined in (4.22), and reorganize the $\boldsymbol{x}$-sum to take the multiplicity ℓ into account:

$$\begin{aligned}\sum_{\boldsymbol{x}} [\cdots] &\to \sum_{\ell=0}^{\ell_{\max}} \binom{q-1}{\ell} \sum_{\boldsymbol{z} \in (\{0,\ldots,c-b\} \setminus \{b\})^r} \delta_{0, c-b(\ell+1)-\sum_{k=1}^r z_k} [\cdots] \\ &= \sum_{\ell=0}^{\ell_{\max}} \binom{q-1}{\ell} \sum_{z_1 \in \{0,\ldots,c-b\} \setminus \{b\}} \cdots \sum_{z_r \in \{0,\ldots,c-b\} \setminus \{b\}} \delta_{0, c-b(\ell+1)-\sum_{k=1}^r z_k} [\cdots]\end{aligned}$$

where δ is the Kronecker delta, and $\ell_{\max} = \min\{q-1, \lfloor \frac{c-b}{b} \rfloor\}$. The factor $\binom{q-1}{\ell}$ pops up because the summand in (4.23) is fully symmetric under permutations of $\boldsymbol{x}$. The Kronecker delta takes care of the constraint that the components of $\boldsymbol{z}$ add up to $c - b - \ell b$.

If $\ell_{\max} = \lfloor \frac{c-b}{b} \rfloor$ and the sum over ℓ is extended beyond $\ell_{\max}$, then all the additional terms are zero, because the Kronecker delta condition cannot be satisfied. (The $\sum_k z_k$ would have to become negative.) Hence we are free to replace the upper summation bound $\ell_{\max}$ by $q-1$ without changing the result of the sum.

Next we use a sum representation of the Kronecker δ as follows,

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

with $s = c-b(l+1)-\sum_k z_k$. This is a correct representation only if N_b is larger than the maximum $|s|$ that can occur. The most positive possible value of s is attained at ($\ell = 0$, $\boldsymbol{z} = 0$), namely $s = c - b$. The most negative value (s_{neg}) is attained when $z_k = c-b$ for all k. Since there are $r = q-1-\ell$ components in $\boldsymbol{z}$, we have $s_{\text{neg}} = \min_\ell[c-b(\ell+1)-(q-1-\ell)(c-b)]$. The function is linear in ℓ, so there are only two candidates: the extreme values $\ell = 0$ (that minimize s if $c-2b > 0$) and $\ell = q-1$ (that minimize s if $c-2b < 0$), which yield $|s_{\text{neg}}| = (q-2)(c-b)$ and $|s_{\text{neg}}| = |c-bq|$ respectively. In the second result, the condition $c-2b < 0$ yield that also $c-bq < 0$ allowing to write $|s_{\text{neg}}| = bq-c$. Hence N_b has to be larger than $\max\{c-b, (q-2)(c-b), bq-c\}$.

Our expression for K_b now contains sums over ℓ, z_k and a. We shift the a-sum completely to the left. Next we write

$$B(\kappa\mathbf{1}_{q-1}+\boldsymbol{x}) = \frac{[\Gamma(\kappa+b)]^\ell \prod_{k=1}^{q-1-\ell}\Gamma(\kappa+z_k)}{\Gamma(c-b+\kappa[q-1])}, \tag{K.2}$$

$$\binom{c-b}{\boldsymbol{x}} = \frac{(c-b)!}{[b!]^\ell \prod_{k=1}^{q-1-\ell} z_k!}. \tag{K.3}$$

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

$$\prod_{k=1}^{q-1-\ell}\left[\sum_{z_k\in\{0,\ldots,c-b\}\setminus\{b\}} \frac{W(b,\ell,z_k)\Gamma(\kappa+z_k)}{z_k!\;\tau_b^{az_k}}\right] = (G_{ba\ell})^{q-1-\ell}. \tag{K.4}$$

Theorem 5.5 follows after some elementary rewriting. □

L Proof of Theorem 5.6

We start from K_b as given by Theorem 5.5. The $G_{ba\ell}$ becomes G_{ba}, so the factor G_{ba}^{q-1} can be moved out of the ℓ-sum. The $w(b,\ell)$ becomes $w(b)/(\ell+1)$ and $w(b)$ can also be moved out of the ℓ-sum. The remaining sum is $\sum_{\ell=0}^{q-1}\binom{q-1}{\ell}\frac{1}{\ell+1}(v_{ba}/G_{ba})^\ell$ which evaluates to $[(G_{ba}+v_{ba})^q - G_{ba}^q]G_{ba}^{1-q}/(qv_{ba})$. Theorem 5.6 follows after substituting the definition of v_{ba} and some rewriting. □

M Proof of Theorem 5.7

In (5.11) the $W(b, \ell, z)$ becomes $W(b, z)$. The definition of class 3 specifies that $W(b, z)$ is either 1 or 0. The result (5.14) trivially follows. □

N Proof of Lemma 5.8

N.1 The case $b < c/q$

A symbol that occurs fewer than c/q times cannot have the majority. Consider the extreme case where all the other symbols also occur b times: then the total number of symbols received by the coalition would be $q \cdot b < c$.

N.2 The case $b > c/2$

Since the colluder strategy is majority voting, we have $\Psi_b(\boldsymbol{x}) = 1$ for $b > c/2$. (This follows from the fact that none of the components x_a can exceed $c/2$ due to the sum rule $\sum_a x_a = c - b < c/2$.) The result (5.20) follows after substitution of $\Psi_b(\boldsymbol{x}) = 1$ into (4.23), summing up $(\sum_{\boldsymbol{x}})$ the probabilities to 1, and finally writing the Beta functions in terms of Gamma functions according to (3.3).

N.3 The case $b = c/2$

Now $\Psi_b(\boldsymbol{x}) = 1$ unless $x_\beta = c/2$ for some $\beta \in \{1, \cdots, q-1\}$; in that case $\Psi_b(\boldsymbol{x}) = 1/2$ since there are two equivalent symbols to choose from. We have

$$\begin{aligned} K_{c/2} &= \sum_{\boldsymbol{x}: x_\beta \neq c/2} \mathbb{P}_{q-1}(\boldsymbol{x}|\frac{c}{2}) + \sum_{a=1}^{q-1} \binom{c/2}{c/2} \frac{B(\kappa \mathbf{1}_{q-1} + \frac{c}{2}\boldsymbol{e}_a)}{B(\kappa \mathbf{1}_{q-1})} \cdot \tfrac{1}{2} \\ &= \sum_{\boldsymbol{x}} \mathbb{P}_{q-1}(\boldsymbol{x}|\frac{c}{2}) - \frac{1}{2} \sum_{a=1}^{q-1} \frac{B(\kappa \mathbf{1}_{q-1} + \frac{c}{2}\boldsymbol{e}_a)}{B(\kappa \mathbf{1}_{q-1})} \\ &= 1 - \frac{q-1}{2} \frac{B(\kappa \mathbf{1}_{q-1} + \frac{c}{2}\boldsymbol{e}_a)}{B(\kappa \mathbf{1}_{q-1})}. \end{aligned} \tag{N.1}$$

In the last line we used the fact that the a is arbitrary. Finally, without loss of generality we can set $a = 1$.

N.4 The case $c/q \leq b < c/2$

MinV can be described as a Class 3 strategy whose ranking is a decreasing sequence of tallies (the higher the better). This is equivalent to defining $W(b, z)$ as follows:

$$W(b,z) = \begin{cases} 1 & \text{if } b > z \\ 0 & \text{otherwise} \end{cases} . \tag{N.2}$$

Using this definition in Theorem 5.7, the G_{ba} summation range can be shrunk in $z \in \{0, 1, \ldots, b-1\}$, and (5.17) immediately follows.

N_b computation: The N_b value introduced in Theorem 5.5 is defined as a positive integer s. t. $N_b > |s| = |c - b(l+1) - \sum_k z_k|$. The range of solutions for N_b given in the theorem is safe for any strategy contained in Class 1, 2 or 3. However, in some particular cases, it is possible to obtain lower bounds due to specific strategies and smaller variable ranges. One of these cases is the actual one, where the strategy is MajV and $c/q \leq b < c/2$. Because of the strategy, we have that $0 \leq x_j \leq b-1$. The most positive possible value of s is attained at ($\ell = 0$, $\boldsymbol{z} = 0$), namely $s = c - b$. The most negative value (s_{neg}) is attained when $z_k = b - 1$ for all k. Since there are $r = q - 1 - \ell$ components in $\boldsymbol{z}$, we have $s_{\text{neg}} = \min_\ell[c + q(1-b) - \ell - 1]$. Then trivially we set $\ell = q - 1$ which yield $|s_{\text{neg}}| = |c - bq|$. Being $b \geq c/q$, we attain $|s_{\text{neg}}| = bq - c$. Hence N_b has to be larger than $\max\{c - b, bq - c\}$.

□

O Proof of Lemma 5.10

O.1 The case $b < c/q$

MinV can be described as a Class 3 strategy whose ranking is an increasing sequence of tallies (the lower the better). This is equivalent to defining $W(b, z)$ as follows:

$$W(b,z) = \begin{cases} 1 & \text{if } b < z \\ 0 & \text{otherwise} \end{cases} . \tag{O.1}$$

Using this definition in Theorem 5.7, the G_{ba} summation range can be shrunk in $z \in \{b+1, \ldots, c-b\}$, and (5.23) immediately follows.

N_b computation: As for MajV, also for MinV is possible to obtain lower bounds for N_b. We recall that N_b needs to be bigger than $|s| =$

$|c - b(l+1) - \sum_k z_k|$. We start looking for the highest possible value of s. We first of all set $z_k = b+1$ for each k. Since there are $r = q-1-\ell$ components in $\boldsymbol{z}$, we attain $s = c-q(b+1)+1+\ell$. Notice that this is the lowest z_k value being $b+1 \leq z_k \leq c-b$. Then for $\ell = q-1$ we end with $s = c - bq$, that is always positive for the actual range of b values. The most negative value (s_{neg}) is attained when $z_k = c-b$ for all k. Then we have $s_{\text{neg}} = \min_\ell[\ell(c-2b)-(q-2)(c-b)]$. Being $c-2b > 0$, the best result is given setting $\ell = 0$, obtaining $s_{\text{neg}} = -(q-2)(c-b) \leq 0$. So $|s_{\text{neg}}| = (q-2)(c-b)$. Hence N_b has to be larger than $\max\{c-bq, (q-2)(c-b)\}$. The first function is bigger than the second just when $q = 2$, then we can split the two cases obtaining (5.21).

O.2 The case $b > c/q$

A symbol that occurs more than c/q times cannot have the minority. Consider the extreme case where all the other symbols also occur b times: then the total number of symbols received by the coalition would be $q \cdot b > c$.

□

P Proof of Theorem 5.14

We first need the following Lemma:

LEMMA P.1.

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

PROOF. We define

$$A(\alpha) := \sum_{w=0}^{q-1} \binom{q-1}{w} \alpha^w \beta^{q-1-w} = (\alpha+\beta)^{q-1}. \tag{P.2}$$

Integrating A we have:

$$\int_0^\alpha A(\alpha')\mathrm{d}\alpha' = \sum_{w=0}^{q-1} \binom{q-1}{w} \frac{1}{w+1} \alpha^{w+1} \beta^{q-1-w} = \frac{(\alpha+\beta)^q - \beta^q}{q}. \tag{P.3}$$

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

Starting from the general definition of K_b (4.23) we have

$$K_b^{\mathrm{RS}} = \mathbb{E}_{\boldsymbol{x}|b}\Psi_b^{\mathrm{RS}}(\boldsymbol{x}) = \sum_{\boldsymbol{x}} \binom{c-b}{\boldsymbol{x}} \frac{B(\kappa \mathbf{1}_{q-1} + \boldsymbol{x})}{B(\kappa \mathbf{1}_{q-1})} \Psi_b^{\mathrm{RS}}(\boldsymbol{x}). \tag{P.4}$$

Given that the strategy can be defined as

$$\Psi_b^{\mathrm{RS}}(\boldsymbol{x}) = \frac{1}{w+1}, \qquad w = |\{i : x_i > 0\}| \tag{P.5}$$

we need to rewrite the $\boldsymbol{x}$-sum in (P.4) 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 $z_1, \cdots, z_w$.

$$\sum_{\boldsymbol{x}} \{\cdots\} \to \sum_{w=1}^{q-1} \binom{q-1}{w} \sum_{\boldsymbol{z} \in \{1,\ldots,c-b\}^w} \delta_{0,c-b-\sum_{i=1}^w z_i} \{\cdots\} \tag{P.6}$$

$$= \sum_{w=1}^{q-1} \binom{q-1}{w} \sum_{z_1 \in \{1,\ldots,c-b\}} \cdots \sum_{z_w \in \{1,\ldots,c-b\}} \delta_{0,c-b-\sum_{i=1}^w z_i} \{\cdots\} \tag{P.7}$$

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

$$\delta_{0,s} = \frac{1}{N_b} \sum_{a=0}^{N_b-1} \left(e^{i2\pi/N_b}\right)^{as} \tag{P.8}$$

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 $s = -(c-b)(q-2)$. Being $q > 2$, N_b has just 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

$$B(\kappa \mathbf{1}_{q-1} + \boldsymbol{x}) = \frac{[\Gamma(\kappa)]^{q-1-w} \prod_{i=1}^w \Gamma(\kappa + z_i)}{\Gamma(c - b + \kappa[q-1])} \tag{P.9}$$

$$\binom{c-b}{\boldsymbol{x}} = \frac{(c-b)!}{\prod_{k=1}^w z_k!}. \tag{P.10}$$

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

$$\prod_{k=1}^{w}\left[\sum_{z_k=1}^{c-b}\frac{\Gamma(\kappa+z_k)}{z_k!\tau_b^{az_k}}\right]=(G_{ba})^w. \tag{P.11}$$

After some elementary rewriting we have

$$K_b^{\mathrm{RS}}=\frac{(c-b)!\Gamma(\kappa(q-1))}{N_b\Gamma(c-b+\kappa(q-1))}\sum_{a=0}^{N_b-1}\tau_b^{a(c-b)}\sum_{w=1}^{q-1}\frac{\binom{q-1}{w}}{w+1}\left[\frac{G_{ba}}{\Gamma(\kappa)}\right]^w. \tag{P.12}$$

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

$$\sum_{w=1}^{q-1}\frac{\binom{q-1}{w}}{w+1}\left[\frac{G_{ba}}{\Gamma(\kappa)}\right]^w=\frac{(G_{ba}/\Gamma(\kappa)+1)^q-1}{qG_{ba}/\Gamma(\kappa)}-1. \tag{P.13}$$

Substituting (P.13) into (P.12) we obtain

$$K_b^{\mathrm{RS}}=\frac{(c-b)!\Gamma(\kappa(q-1))}{N_b\Gamma(c-b+\kappa(q-1))}\left[\sum_{a=0}^{N_b-1}\tau_b^{a(c-b)}\frac{(G_{ba}/\Gamma(\kappa)+1)^q-1}{qG_{ba}/\Gamma(\kappa)}-\sum_{a=0}^{N_b-1}\tau_b^{a(c-b)}\right] \tag{P.14}$$

The second summation yields $\delta_{0,c-b}$ which is zero because we are looking at $b<c$. The result (5.27) follows. □

Nomenclature

$\mathcal{C}$	Set of colluding users
χ	Shifted version of ψ
$\mathcal{L}$	List of accused users
$\mathcal{Q}$	Alphabet
$\mathcal{S}_{qc}$	Set containing all the possible $\boldsymbol{\sigma}$ for a given alphabet size q and a coalition size c
$\boldsymbol{X}$	Matrix $n \times m$ filled with symbol in $\mathcal{Q}$ containing all the watermarks embedded in the digital content. The element X_{ji} indicates the symbol received by the user j in position i, while with X_j we indicate the entire codeword received by user j
$\boldsymbol{X}_{\mathcal{C}}$	Portion of $\boldsymbol{X}$ observed by the coalition
$\tilde{\mu}$	Expectation value of the collective coalition accusation sum on one segment. Formally, $\tilde{\mu} = \mathbb{E}[S_{\mathcal{C}}]/m$
P_{FN}	False negative probability
P_{FP}	One-user false positive probability
$P_{\text{FP}}^{\text{global}}$	Global false positive probability. It indicates the probability that any innocent user gets accused. It is approximatively nP_{FP}
$\mathbb{P}(\boldsymbol{\sigma}\|\boldsymbol{p})$	Probability that c users receive symbol occurrences $\boldsymbol{\sigma}$ given $\boldsymbol{p}$
$\mathbb{P}_1(b\|p)$	Marginal distribution for a single component σ_α. It gives the probability for the attackers to receive b occurrences of a symbol α when $p_\alpha = p$
$\mathbb{P}_1(b)$	The overall marginal probability distribution for one component of $\boldsymbol{\sigma}$

$\mathbb{P}_{q-1}(\boldsymbol{x}|b)$ Probability distribution of $\boldsymbol{x} = \boldsymbol{\sigma}_{\setminus\alpha}$ conditioned on $b = \sigma_\alpha$

$\delta(x)$ Dirac delta. It is equal to $+\infty$ for $x = 0$ and 0 otherwise.

δ_x Kronecker delta. It is equal to 1 for $x = 0$ and 0 otherwise.

ε_1 Probability to accuse an innocent user

ε_2 Maximum acceptable error probability to accuse none of the colluders

$\varphi(u)$ Probability distribution of one-segment contribution to innocent's accusation

$\Psi_b(\boldsymbol{x})$ Probability that the attackers output a symbol that occurs b times in a segment and the other $q-1$ symbols occur as indicated in $\boldsymbol{x}$

κ Shape parameter contained in F

$\Omega(z)$ Probability mass in the right tail of the normal distribution beyond point z

$\psi(u)$ Probability distribution of one-segment contribution to guilty's accusation

ρ_m Probability distribution of the quantity $S_j/\sqrt{m}$ for innocent j

$\sigma_\alpha^{(i)}$ Tally of symbol α in attackers' segment i

τ_m Probability distribution of the quantity $S_{\mathcal{C}}/(c\sqrt{m})$ normalized to 0 mean and variance 1

Θ Heaviside step function

$\theta_{y|\boldsymbol{\sigma}}$ Probability that attackers output symbol y given $\boldsymbol{\sigma}$

$\tilde{Z}$ $Z/\sqrt{m}$

$\tilde{Z}_{\text{half}}$ $\tilde{Z}_{\text{half}} = \tilde{\mu}\frac{\sqrt{m}}{c}$

$\mathbf{1}_q$ Vector of length q filled just with 1s

$\boldsymbol{p}^{(i)}$ Probability vector used to generate the symbols in the i-th segment of matrix $\boldsymbol{X}$. The symbols are drawn randomly according to $\mathbb{P}[X_{ji} = \alpha|\boldsymbol{p}^{(i)}] = p_\alpha^{(i)}$, where $\alpha \in \mathcal{Q}$

$\boldsymbol{p}_{\setminus y}$ Bias vector $\boldsymbol{p}$ without the element p_y

$\boldsymbol{\sigma}_{\setminus\alpha}$ Tally vector $\boldsymbol{\sigma}$ without the element σ_α

$\boldsymbol{y}$ Unauthorised watermark created by the coalition $\mathcal{C}$. With y_i we indicate the symbol in position i

B Generalized Beta function

c Number of colluders $|\mathcal{C}|$

c_0 Maximum number of colluders the scheme can resist

C_m Area function for the right-hand tail of one-guilty user score pdf

$F(\boldsymbol{p})$ Dirichlet distribution. It is used to draw randomly the vectors $\boldsymbol{p}^{(i)}$

$f(p)$ Marginal probability distribution for a single component of the vector $\boldsymbol{p}$

$g_0(p_{y_i}^{(i)})$ Score function applied when user's symbol in position i is not y_i. It gives a negative value to the user's final score

$g_1(p_{y_i}^{(i)})$ Score function applied when user's symbol in position i is y_i. It gives a positive value to the user's final score

K_b Probability that attackers output a symbol that occurs b times. It is made averaging over all the possible occurrences $\boldsymbol{x}$ given b

m Codelength

M_2 Second moment of the pdf ψ

n Number of users

q Alphabet size $|\mathcal{Q}|$

R_m Area function for the right-hand tail of ρ_m

S_j User j accusation sum

$S_j^{(i)}$ Accusation score obtained from i-th symbol of user j

$S_{\mathcal{C}}$ Coalition accusation sum. Formally, $S_{\mathcal{C}} := \sum_{j \in \mathcal{C}} S_j$

T_m Cumulative distribution function for τ_m

V Variance of the pdf ψ

Z_{half} Specific value of Z such that the $P_{\text{FN}} \approx \frac{1}{2}$. It is defined as $Z_{\text{half}} = m\tilde{\mu}/c$

Summary

Error probabilities in Tardos codes

The digital piracy is a phenomenon that has become very popular with the growth of the Internet. The constant digitalization of our goods (like movies, books and money) and the creation of peer-to-peer platforms made possible to share and find unauthorized copies of copyrighted contents and, in most of the cases, without disclosing the identity of the pirates. The vendors are the main victims of this behaviour together with the authors of the digital contents (singers, movie directors, software houses,...): the original copies sold become less and so the income. One of the ways to fight piracy consists on denying the pirates to hide their identities. To do so, the digital watermarking technique fits with the purpose (in particular with movies). A watermark can be seen as a string of data stored into the original content that contains extra information about the content itself. The watermark has to follow two important constraints: (i) its presence should not damage the original content; (ii) it has to be hard to detect. Then, a vendor can use watermarks to identify the owners of the original contents and, whenever a plain copy is shared, it is easy to trace the pirate (that could not find and remove the watermark because well hidden). The pirates can still try to corrupt the watermarks with the so called collusion attack: a group of pirates compares their original watermarked copies to locate part of the watermark. In this way they can create a new watermark and avoid the tracing. In the literature it is possible to find several solutions against the collusion attack. Certainly, the Tardos code (or Tardos scheme) is the most popular since it is the first to achieve optimal performances against big coalitions. Despite its popularity, Tardos code performances were not fully understood: both the analytical and numerical approaches could not provide a complete explanation of the error probabilities (false positive and false negative) that are bound to the code, and consequently

its real performances. How to determine the real error rates of the Tardos code? In this dissertation it has been studied therefore the error probabilities associated to the Tardos code, in particular its fully symmetric and non-binary version introduced by Škorić et al. The aim was to obtained a preferably analytical method to compute both the false positive and the false negative probabilities at any parameter setting (codelength, alphabet size, coalition size,...). To achieve this target we develop a new procedure, called CSE method (Convolution and Series Expansion) that succeeds on computing semi-analytically the error probabilities. This method has shown to be consistent with the theory and with the simulations (when simulations are doable). The CSE method required a new parameterization of the attack strategies to describe in a better way the symmetry of the scheme. This new parameterization permits also to do pre-computations, which speeds up the whole process. Thanks to the CSE method it has been possible to study the real performances of the Tardos code against the most popular attacks. We used the ROC (receiver operating characteristic) curves to have a better representation of both the error probability behaviours at the same time. This shows clearly which are the best attacks and in which cases they should be used. Finally, the CSE method can be applied also to other schemes and attacks. Also, future works can be addressed to non-symmetric scenarios.

Curriculum Vitae

Antonino Simone was born on July 4, 1983 in Venice, Italy. In 2002 he studied Computer Science at Universitá Ca' Foscari in Venice. In 2005 he obtained his Bachelor's degree cum laude with a thesis on hypergraph clustering. In 2008 he graduated cum laude with a Master thesis on machine learning entitled *Apprendimento di modelli strutturali di grafi*. From 2009 he started a PhD project on traitor tracing codes at Technische Universiteit Eindhoven of which the results are presented in this dissertation. Since 2014 he is employed at Quintiq.