

Bounds on the leakage of the input's distribution in information-hiding protocols

Abhishek Bhowmick^{1*} and Catuscia Palamidessi²

¹ Computer Science and Engineering, IIT Kanpur

² INRIA Saclay and LIX, Ecole Polytechnique

Abstract. In information-hiding, an adversary that tries to infer the secret information has a higher probability of success if it knows the distribution on the secrets. We show that if the system leaks probabilistically some information about the secrets, (that is, if there is a probabilistic correlation between the secrets and some observables) then the adversary can approximate such distribution by repeating the observations. More precisely, it can approximate the distribution on the observables by computing their frequencies, and then derive the distribution on the secrets by using the correlation in the inverse direction. We illustrate this method, and then we study the bounds on the approximation error associated with it, for various natural notions of error. As a case study, we apply our results to Crowds, a protocol for anonymous communication.

1 Introduction

The growing development of the internet and its interaction with everyday activities has triggered an unprecedented need for mechanisms to protect private information such as personal data, preferences, credit card number, etc., against potentially malicious users. Consequently, there has been an increasing interest for research on information-hiding, both at the level of developing protocols which ensure the protection of sensitive data during transactions, and at the level of studying the foundational aspects related to the leakage of classified information in programs, systems, and protocols.

Recent research on the foundations of information-hiding has been paying more and more attention to the quantitative aspects, and in particular to probability. This is because the data to be protected are often obeying the laws of some probabilistic distribution, and also because the mechanisms for ensuring their protection often rely on randomization to obfuscate the link between the hidden information and the observables. This is the case, for example, of many anonymity protocols, such as Crowds [22], Onion Routing [27], Tor [13], Tarzan [14], Mix-Nets [7], DC Nets [6], etc.

A common framework for studying the information leakage from a probabilistic point of view is to regard the correlation between the hidden information

* This paper has been developed during the visit of the first author to LIX, the laboratory of Computer Science of the Ecole Polytechnique.

and the observables as a noisy channel, in the information-theoretic sense. The hidden information is modeled as a random variable A which constitutes the input of the channel, the observables O constitute the output of the channel and are in general a random variable as well, and the channel itself represents the protocol, program or system, and establishes the correlation between secrets and observables. Such correlation, which in general is probabilistic if the protocol or program performs randomized operations, is expressed in terms of the conditional probability distribution on the observables, given the input. This distribution is in general assumed to be known, and also supposed to be the only information that matters about the channel. That is to say, a channel is represented by the matrix of the conditional probabilities.

In general an adversary is able to see the observable outcome o of the protocol or program, and it is interested in finding out the secret information, namely the element of A which has given rise to such observable. If the distribution on A (i.e. the *a priori probability* of the input) is known, then the best strategy, for the adversary, is to apply the so-called *Maximum A posteriori Probability* (MAP) rule, which consists in choosing the $a \in A$ with the maximum *a posteriori probability* $Pr(a|o)$, that can be computed, using Bayes' theorem, as:

$$Pr(a|o) = \frac{Pr(o|a) Pr(a)}{Pr(o)} \quad (1)$$

where $Pr(o)$ can be computed using the formula:

$$Pr(o) = \sum_a Pr(o|a) Pr(a) \quad (2)$$

The MAP rule is optimal in the sense that it minimizes the probability of error, i.e. the average probability of choosing the wrong a , weighted on the probabilities of all the observables [11].

If the distribution on A is not known then the above formula does not help to compute $Pr(a|o)$. If one can repeat the experiment and collect more observables while keeping the same secret as the input, however, then the MAP rule can be replaced by the so-called *Maximum Likelihood* (ML) rule, which consists in choosing the a for which $Pr(\vec{o}|a)$ is maximum, where \vec{o} is the sequence of the observables collected during the experiments. It is well known that the ML rule gives in the long term (i.e. as the number of experiments increases) the same result as the MAP rule, in the sense that the $Pr(a)$ component becomes less and less relevant for determining the a which gives the maximum $Pr(a|\vec{o})$ [11]. (The denominator of (1) is just a normalization factor and it does not need to be computed for determining such a .)

In protocols and in programs it is in general not possible to ensure that the input remains the same through different runs, especially if the adversary is *passive*. On the other hand, we show in this paper that the fact that the input may change makes it possible to approximate its distribution. The idea is the following: The adversary observes the outcomes of n experiments, and it approximates the distribution on the observables by computing their frequencies,

i.e. by assigning to each o the number of times that o has occurred, divided by n . Then, it replaces the $Pr(o)$ in (2) by its approximations, thus obtaining a system of linear equations where the unknown are the $Pr(a)$'s. Finally, by solving the system, the adversary obtains a distribution on A . We show in this paper that, under the condition that the matrix is invertible, this distribution approximates the true distribution on A , namely that the probability of error with respect to the true distribution decreases as the number of the experiments increases.

1.1 Related work

The problem of inferring a hidden information from observable events that depend on the information is known under the name of *hypothesis testing* (see for example [11]). The case in which this dependence is expressed in terms of a known conditional distribution is well studied in literature, and the MAP and ML rules are the most used decision functions. In spite of the large literature on this topic, however, we have not been able to find an investigation of the scenario in which the hidden event (the hypothesis) changes every time the experiment is performed. We think that the reason may be the fact that hypothesis testing has been considered, so far, for applications in which there is one hypothesis which is *true*, and it is not supposed to change over time. For instance, in medicine, the hypothesis is the kind of illness of the patient, and the observables are the symptoms. The tests on the patient may detect various symptoms, but the cause remains the same. The situation in which the hypothesis changes at every experiment is typical of information-hiding protocols and programs, where the hypotheses are the inputs, and the experiments are the runs. This application is new for hypothesis testing, with the exception of the recent work mentioned below. Consequently we think that, despite its relative simplicity, the method that we describe in this paper is new.

Hypothesis testing in the context of information hiding protocols has been investigated in [16, 4, 26]. In these works, however, the focus is on the inference of the true hypothesis, and not on the inference of the probability distribution.

The foundational aspects of information hiding and information flow, in a probabilistic setting, have been studied also in several other papers. We mention in particular [1] which explores the relation between probability and nondeterminism, and [3] which extends of the notion of probable innocence. A related line of work is directed at exploring the application of information-theoretic concepts [24, 12, 20, 21, 29, 5, 10, 19, 15, 8, 9, 17, 18, 2]. The relation with hypothesis testing is given by the fact that the exponential of the conditional entropy is an upper bound of the Bayes risk (the probability of error using the MAP rule) [23, 4], although [26] has pointed out that the bound can be in some case very loose.

1.2 Contribution

The contributions of this paper are:

- A method to compute the probability distribution on the hidden events from repeated executions of the protocol or program.

- The proof of correctness of this method, expressed in probabilistic terms: the probability that the error with respect to the true distribution is arbitrarily small converges to 1 as the number of experiments grows.
- The application of these concepts to the case of Crowds. The studies of Crowds so far have been assuming a fixed user as the culprit, and have ignored the problem of determining the a priori probability that an arbitrary user be the culprit.

1.3 Plan of the paper

In the next section we recall some basic notions about the systems of linear equations. In Section 3 we present our framework in which protocols and programs are seen as noisy channels, we explain our method for approximating the distribution on the hidden events, and we introduce three notions of approximation error. In Section 4 we show that, under the hypothesis that the matrix of the channel is invertible, the approximation of the probability distribution can be made as accurate as desired, provided we increase the number of experiments. In Section 5 we study the case in which the matrix is not invertible. Finally, in Section 6, we apply our study to the example of Crowds.

For reasons of space the proofs are omitted. They can be found on line at the following URL: <http://www.lix.polytechnique.fr/~catuscia/papers/Anonymity/Bhowmick/report.pdf>.

2 Preliminaries

A system of linear equations is a set of the form

$$\begin{aligned} m_{11}x_1 + m_{12}x_2 + \dots + m_{1n}x_n &= y_1 \\ m_{21}x_1 + m_{22}x_2 + \dots + m_{2n}x_n &= y_2 \\ &\vdots \\ m_{m1}x_1 + m_{m2}x_2 + \dots + m_{mn}x_n &= y_m \end{aligned}$$

where the m_{ij} 's and the y_i 's are constants and the x_j 's are the unknowns. Such a system can be represented as:

$$MX = Y$$

where Y is the $m \times 1$ vector containing the y_i 's, X is the $n \times 1$ vector containing the x_j 's and M is the $m \times n$ matrix whose element in the i^{th} row and j^{th} column is m_{ij} .

In this paper we denote by M_{ij} the (i, j) *minor*, namely the determinant of the matrix formed by removing the i^{th} row and the j^{th} column. We use c_{ij} to represent the *cofactor* of m_{ij} , namely $(-1)^{i+j}M_{ij}$. We represent by $\det A$ the determinant of the square matrix A , and by $|x|$ the absolute value of the real number x .

The inverse of M , if it exists, is the unique matrix M^{-1} such that $MM^{-1} = M^{-1}M = I$, where I is the identity matrix, i.e. the matrix with whose elements are 1 on leading diagonal and 0 otherwise.

3 Information hiding protocols modeled as matrices

In our framework, we regard an information-hiding protocol as a system where the secrets are disjoint hidden events a_1, a_2, \dots, a_n , with a probability distribution $Pr(a_1), Pr(a_2), \dots, Pr(a_n)$, and the observables are disjoint events o_1, o_2, \dots, o_m that depend probabilistically on the a_j 's. We use $Pr(o_i|a_j)$ to represent the *conditional probability* of o_i given a_j . These conditional probabilities induce a probability distribution on the o_i 's that, because of the disjointness of the a_{ij} 's, is given by:

$$Pr(o_i) = \sum_{j=1}^n Pr(o_i|a_j)Pr(a_j) \quad \text{for each } i. \quad (3)$$

For simplicity, we introduce the following notation:

$$\begin{aligned} x_j &= Pr(a_j) \\ y_i &= Pr(o_i) \\ m_{ij} &= Pr(o_i|a_j) \end{aligned}$$

and we denote by X, Y and M the matrices containing the x_j 's, y_i 's and m_{ij} 's respectively. Hence, the property (3) can be represented as the equality:

$$Y = MX \quad (4)$$

Since, $Pr(\cdot|a_j)$ is a probability distribution, we have the following properties:

$$0 \leq m_{ij} \leq 1 \quad \text{for each } i, j \quad (5)$$

$$\sum_{i=1}^m m_{ij} = 1 \quad \text{for each } j \quad (6)$$

We assume that we have a *passive adversary*, namely an entity that can observe the outcome of the protocol (the observables), and knows the behaviour of the protocol, hence the $Pr(o_i|a_j)$'s, but it cannot interfere with or change the way the protocol works. The adversary wishes to find out the $Pr(a_j)$'s. Due to the above assumptions, the only thing it can do is to estimate (an approximation of) the $Pr(o_i)$'s, and then calculate (an approximation of) the $Pr(a_j)$'s by solving the system (4) above.

The estimation of $Pr(o_i)$ can be done by observing the outcome of the protocol several times, say h , and counting the number of times $\#o_i$ that the event o_i has occurred. We know that for large h , this method gives a good approximation of $Pr(o_i)$ with high probability, because of the law of large numbers [28]:

$$\lim_{h \rightarrow \infty} Pr(|Pr(o_i) - \frac{\#o_i}{h}| < \varepsilon) = 1 \quad (7)$$

for any $\varepsilon > 0$ and for $1 \leq i \leq m$.

The real goal of the adversary, however, is to estimate the $Pr(a_j)$'s. So, we want to find out whether the method of solving the system (4) also gives an approximation of the $Pr(a_j)$'s, and how good this approximation is, namely what the bounds are on the approximation errors for the $Pr(a_j)$'s in terms of the approximation errors for the $Pr(o_i)$'s.

Let Y_h be the computed approximation of the y_i 's, namely the vector:

$$Y_h \stackrel{\text{def}}{=} \left(\frac{\#o_1}{h}, \frac{\#o_2}{h}, \dots, \frac{\#o_m}{h} \right)$$

$$\stackrel{\text{notation}}{=} (y_{h1}, y_{h2}, \dots, y_{hm})$$

Let X_h be the vector of the solutions to the system (4) with Y substituted by its approximation Y_h (if the system is solvable), namely the vector such that:

$$Y_h = MX_h \tag{8}$$

We are now going to explore the bounds on the approximation errors on X in terms of the bounds of the approximation errors on Y .

There are various possibilities for defining the notion of approximation error. We consider the following three, which seem to us the most natural.

In the first definition, we regard the error as the vector of the absolute differences on the individual components of Y and Y_h and on X and X_h respectively.

Definition 1 (Notion of error #1).

$$E_Y = (|y_1 - y_{h1}|, |y_2 - y_{h2}|, \dots, |y_m - y_{hm}|)$$

$$E_X = (|x_1 - x_{h1}|, |x_2 - x_{h2}|, \dots, |x_m - x_{hm}|)$$

In the second definition, we regard the error as the sum of all the absolute differences on the individual components.

Definition 2 (Notion of error #2).

$$e_Y = \sum_{i=1}^m |y_{hi} - y_i| \quad e_X = \sum_{j=1}^n |x_{hj} - x_j|$$

Finally, in the third definition, we regard the error as the vectorial distance between Y and Y_h , and X and X_h respectively.

Definition 3 (Notion of error #3).

$$err_Y = \sqrt{\sum_{i=1}^m |y_{hi} - y_i|^2} \quad err_X = \sqrt{\sum_{j=1}^n |x_{hj} - x_j|^2}$$

4 Analysis of the error in the case of invertible matrix

In this section, we study the bounds on the approximation error when $m = n$ and M is invertible. We use L to represent M^{-1} , and l_{ij} to represent the i^{th} row and j^{th} column element of L . Hence, (4) can be rewritten as $X = LY$.

4.1 Bound on the error for notion #1

Here we study the upper bound on E_X in terms of E_Y . We do not have any interesting lower bound for this case.

First we observe the following:

Lemma 1. $\sum_i |l_{ji}| \leq n \frac{\max_i |M_{ij}|}{|\det M|}$

The above lemma allows us to establish an upper bound on the error:

Theorem 1. *Each component of E_X is bounded by*

$$n \frac{\max_{ij} |M_{ij}|}{|\det M|} \max_i |y_{hi} - y_i|$$

Thus, we see that if each component of Y_h is error-bound by ε , then each component of X_h is error-bound by some finite multiple of ε . Hence, if the protocol matrix M is invertible, then the adversary can approximate the values of the probability of the inputs with very high probability to the desired degree of accuracy, by increasing the number of experiments.

4.2 Bounds on the error for notion #2

We now study the bounds on e_X in terms of e_Y . We start with the lower bound.

Theorem 2. $e_X \geq e_Y$

Now we show that we can give an upper bound on e_X in terms of e_Y . First we observe the following fact:

Lemma 2. $\sum_j |l_{ji}| \leq n \frac{\max_j |M_{ij}|}{|\det M|}$

The above lemma is used to establish an upper bound on the second notion of error:

Theorem 3. $e_X \leq n \frac{\max_{ij} |M_{ij}|}{|\det M|} e_Y$

4.3 Bounds on the error for notion #3

Here we study the bounds on err_X in terms of err_Y . For the lower bound, we have:

Theorem 4. $err_Y \leq \sqrt{n} err_X$

Now, we show that we can give an upper bound on err_X in terms of err_Y . First, we observe the following:

Lemma 3. $\sum_j l_{ji}^2 \leq n \frac{\max_j M_{ij}^2}{(\det M)^2}$

The above lemma allows us to derive an upper bound on the third notion of error:

Theorem 5. $err_X \leq n \frac{\max_{ij} |M_{ij}|}{|\det M|} err_Y$

4.4 Convergence to 0 of the error in the three definitions

A consequence of the bounds determined above is that, since the error in the approximation of Y tends to 0 as h increases (cfr. (7)), the error in the approximation of X also tends to 0 as h increases (for all the three notions), by the sandwich principle. In other words, if the adversary is able to repeat the experiment, his guesses about the input distribution become increasingly more accurate. Formally, this is expressed by the following theorem.

Theorem 6.

$$\begin{aligned} \lim_{h \rightarrow \infty} Pr(|x_{hj} - x_j| < \varepsilon) &= 1 && \text{for any } \varepsilon > 0 \text{ and for any } j \\ \lim_{h \rightarrow \infty} Pr(e_X < \varepsilon) &= 1 \\ \lim_{h \rightarrow \infty} Pr(err_X < \varepsilon) &= 1 \end{aligned}$$

The above result states that all the definitions of error we have considered converge to 0. The convergence speed is also the same: In fact, the coefficient factors on the bounds of all the three definition are the same despite the definitions are different.

5 Analysis of the error in the general case

In this section we consider the cases in which $m \neq n$ or M is not invertible.

We first note that the system $MX = Y$ always has a solution, because Y represents the true probability distribution on the output, hence the equation is satisfied, by definition, by the vector X which represents the true probability on the input.

It may happen, however, that the system has infinitely many solutions. This happens when the rank of M (i.e. the maximal number of linearly-independent columns of M) is strictly smaller than $\min\{m, n\}$. In this case it is not possible, for the adversary, to approximate the input distribution at an arbitrary degree of precision. Consider the following example:

Example 1. Consider a protocol represented by the following system:

$$\begin{aligned}\frac{1}{3}x_1 + \frac{1}{3}x_2 + \frac{1}{2}x_3 &= y_1 \\ \frac{1}{3}x_1 + \frac{1}{3}x_2 + \frac{3}{8}x_3 &= y_2 \\ \frac{1}{3}x_1 + \frac{1}{3}x_2 + \frac{1}{8}x_3 &= y_3\end{aligned}$$

Assume that the adversary gets to know somehow the true output distribution, and assume that it is $y_1 = \frac{5}{12}, y_2 = \frac{17}{48}, y_3 = \frac{11}{48}$. By solving the system, the adversary finds that all the tuples which satisfy $x_1 + x_2 = \frac{1}{2}, x_3 = \frac{1}{2}$ (and $x_1, x_2 \geq 0$) are possible probability distributions on the input. However, it has no way to infer how the probability $\frac{1}{2}$ distributes among x_1 and x_2 . So the approximation error on the first two components in the worst case is $\frac{1}{2}$ even in the limit.

From the above example we can conclude that in case the rank of the matrix is smaller than n , the adversary cannot approximate the true probability of the input. It is possible, however, to *approximate the combined probability* of some of the inputs, like the combination of x_1 and x_2 in the example.

Let r be the rank of M . We show how the adversary can reduce M to a matrix $r \times r$ without losing any information that can be used for computing the approximation. The idea is to remove the dependent columns, one by one, and then remove the redundant rows, again one by one. Once this reduction is done, the adversary can proceed like illustrated in the previous section for the square and invertible matrices.

5.1 Removal of the dependent columns

Consider a column of M that can be expressed as a linear combination of other columns. Let h be its index, and let T be the set of indexes of the columns which form the linear combination. Let T' denote the set of indexes of the remaining columns. Let λ_i 's be the corresponding coefficients of the linear combination. Hence, for every i ,

$$\sum_{j \in T} \lambda_j m_{ij} = m_{ih}$$

Now, Let M' be the matrix obtained from M by simply removing its h^{th} column, and let X' be a vector of variables which is same as X without the component x_h .

Proposition 1. *If $MX = Y$ has a solution $X = (x_1, x_2, \dots, x_n)$ then $M'X' = Y$ has a solution $X' = (x'_1, x'_2, \dots, x'_n)$, where*

$$x'_j = \begin{cases} x_j + \lambda_j x_h & \text{if } j \in T \\ x_j & \text{otherwise} \end{cases}$$

We continue the above procedure till we obtain a matrix M_f which has r columns.

The number of rows m of M_f is still the same as the one of M . If $r < m$, there are necessarily $m - r$ rows which are linear combinations of other rows in M_f . The corresponding system $M_f X_f = Y$ has a solution, as proved above, however when we replace Y by the approximation vector Y_h , we are not guaranteed that $M_f X_f = Y_h$ still has a solution. To fix this problem, we could simply remove the dependent rows of M_f . This would not be the best method, however, from the point of view of efficiency. In fact, the experiments which give an output corresponding to a removed row would be wasted for the calculation of the frequencies. In the rest of this section we illustrate a better method.

5.2 Elimination of the dependent rows

We rename M_f as M for simplicity. Now, consider a row of M that can be expressed as a linear combination of other rows. Let h be its index, and let S be the set of indexes of the rows which form the linear combination. We choose an arbitrary k from S and construct the new M' and Y' from M, Y respectively by removing their h -th row, and defining the other elements as follows:

$$m'_{ij} = \begin{cases} m_{ij} & \text{if } i \neq h, k \\ m_{kj} + m_{hj} & i = k \end{cases}$$

$$y'_i = \begin{cases} y_i & \text{if } i \neq h, k \\ y_k + y_h & i = k \end{cases}$$

It is important to note that, by the above construction, the crucial properties, (5) and (6) still hold.

Proposition 2. *If $X = (x_1, x_2, \dots, x_n)$ is a solution to $MX = Y$ then it is also a solution to $M'X = Y'$.*

Example 2. Consider again the system in Example 1, with the known values of the y'_i s. By removing the dependent column (the second one), we obtain:

$$\begin{aligned} \frac{1}{3}x'_1 + \frac{1}{2}x'_3 &= \frac{5}{12} \\ \frac{1}{3}x'_1 + \frac{3}{8}x'_3 &= \frac{17}{48} \\ \frac{1}{3}x'_1 + \frac{1}{8}x'_3 &= \frac{11}{48} \end{aligned}$$

Then, we observe that the first row is a linear combination of the other two, with coefficients $\frac{3}{2}$ and $-\frac{1}{2}$ respectively. By eliminating the dependent rows with the method illustrated above, ($h = 1, k = 2$) we obtain:

$$\begin{aligned} \frac{2}{3}x'_1 + \frac{7}{8}x'_3 &= \frac{37}{48} \\ \frac{1}{3}x'_1 + \frac{1}{8}x'_3 &= \frac{11}{48} \end{aligned}$$

The solution is $x'_1 = x'_3 = \frac{1}{2}$. We recall that the relation with the solutions of the original system is given by $x'_1 = x_1 + x_2, x'_3 = x_3$.

6 Application: Crowds

In this section, we apply the previously obtained bounds to the Crowds' anonymity protocol. This protocol was introduced by Reiter and Rubin [22] to the purpose of making it possible for a user to send a message to a server without revealing its identity. The idea is to send the message through a chain of users who are also participating in this protocol. The exact algorithm is as follows: First, the initiator chooses a user x randomly and forwards the message to x . Then, with probability p_f , x decides to forward it to another randomly chosen user, and with probability $1 - p_f$ he sends it to the server directly. It is easy to see that the initiator is strongly anonymous with respect to the server, as all users have the same probability of being the forwarder who finally delivers the message. However, the more interesting case is when the attacker is one of the users of the protocol (called a corrupted user) which uses his information to find out the identity of the initiator. A corrupted user has more information than the server since he sees other users forwarding the message through him. The initiator, being the first in the path, has greater probability of forwarding the message to the attacker than any other user, so strong anonymity cannot hold. However, under certain conditions on the number of corrupted users, Crowds can be shown to satisfy a weaker notion of anonymity called probable innocence [22]. In our analysis, we shall consider the clique network topology which was also used in the original presentation of Crowds. In this topology, each user can communicate with any other user. Therefore, the protocol matrix M is symmetric and easy to compute.

Let the total number of users be m , out of which n are honest and $c = m - n$ are corrupt. To construct the protocol matrix M we must define the hidden events and the visible events. Since the initiator wants to hide his identity, we choose $A = u_1, \dots, u_n$ as the set of hidden events, where u_j denotes the event that user j is the initiator. For the purpose of the analysis we consider only the honest users as possible initiators. This is because the attacker's own identity cannot be hidden from him.

Now, we have to define the set of visible events. In Crowds we assume that the attacker does not have access to the entire network (such an attacker would be too powerful for this protocol) but only to the messages that pass through a corrupted user. If a user i forwards the message to a corrupted user, we say that he is detected. As in other studies of Crowds [22, 25], we assume that an attacker will not forward a message himself, as he would not gain more information by that. Thus, we can say that at each execution of the protocol, if a corrupted user is on the path, then there is exactly one detected user. Therefore we define $O = d_1, \dots, d_n$ where d_i means that user i was detected, restricted to the cases in which there was a corrupted user in the path.

Now we need to compute the probabilities $Pr(d_i|u_j)$ for all $1 \leq i, j \leq n$. As in [22], let I be the event that the first corrupted user on the path is immediately preceded by the message initiator. Let H_k be the event that the first corrupted person on the path appears at the k^{th} position. The initiator occupies the 0^{th} position. Let $H_{k+} = \cup_{k' \geq k} H_{k'}$. It has been shown in [22] that $Pr(I|H_{1+}) =$

$1 - \frac{n-1}{m}p_f$. It is also easy to see that, for every i ,

$$Pr(d_i|u_i) = Pr(I|H_{1+})$$

Also, by (6), for every j ,

$$\sum_i Pr(d_i|u_j) = 1$$

By symmetry, we note that $Pr(d_i|u_j)$ is the same for any j except when $i = j$. Thus, by the above observations, we state the following:

$$m_{ij} = Pr(d_i|u_j) = \begin{cases} 1 - \frac{n-1}{m}p_f & \text{if } i = j \\ \frac{p_f}{m} & \text{otherwise} \end{cases}$$

6.1 Probable innocence and strong anonymity

The condition of probable innocence, proposed in [22], is that the detected user is not more likely to be the initiator than not to be. Formally:

$$Pr(I|H_{1+}) \leq \frac{1}{2} \tag{9}$$

In our case $Pr(I|H_{1+})$ is the value of the elements of the the leading diagonal of the protocol matrix. Hence, if the $m_{ii} \leq 1/2$, then the path initiator has the probable innocence protection against the c corrupted users.

Example 3. Let us consider the case in which $p_f = 0.6$, $m = 100$ and $n = 90$. The matrix M is as follows:

$$M = \begin{pmatrix} 0.466 & 0.006 & \dots & 0.006 \\ 0.006 & 0.466 & \dots & 0.006 \\ \vdots & \vdots & \ddots & \vdots \\ 0.006 & 0.006 & \dots & 0.466 \end{pmatrix}$$

Note that the condition of probable innocence is satisfied as $m_{ii} = 0.466 \leq 0.5$.

We shall now compute the bound on the approximation error in X as a function of the approximation error in Y using the three definitions introduced previously.

Notion #1 $0 \leq |x_{hj} - x_j| \leq 194.48 \max_i |y_{hi} - y_i|$, for each j .

Notion #2 $e_Y \leq e_X \leq 194.48 e_Y$.

Notion #3 $0.11 err_Y \leq err_X \leq 194.48 err_Y$.

It is evident that as the error in approximation of Y tends to 0, the errors in approximating X also tend to 0. However, as we shall observe from the graphical analysis to follow, the coefficient of the upper bound on the error in X shoots

up when probable innocence is not satisfied, and goes to infinity for the case in which the columns of the matrix are all equal, which corresponds to the case of *strong anonymity* [5]³. The condition under which the columns are equal is, by definition:

$$1 - \frac{n-1}{m} p_f = \frac{p_f}{m}$$

or, equivalently

$$p_f = \frac{m}{n}$$

Since $p_f < 1$ and $\frac{m}{n} > 1$, this condition cannot be achieved, but it can be approximated for $n = m - 1$, large values of m , and values of p_f close to 1.

6.2 Graphical analysis of the error bounds

We consider the upper bounds on the errors, which are the most interesting. In the following, we analyze the coefficients in the upper bounds as a function of the various parameters of the system.

We denote by Z the coefficient of the bound on the error according to notion $\#i$, for $i \in \{1, 2, 3\}$, as in, for instance, $e_X \leq Z e_Y$. We recall that Z is the same for all the three definitions.

Figure 1 illustrates the plot of Z obtained by varying n and p_f , while m is kept constant and equal to 100. The minimum value of Z is 1.00 at $n = 1$, $p_f * 100 = 10$. The convergence to 0 for small n is only apparent and due to the large scale of Z .

It is clear from the graph that as n increases (keeping p_f constant), also Z increases, and the network becomes safer. Thus, the chance of error is big for the attacker. Also, as p_f increases (keeping n constant), Z increases.

We now study how Z is related to the condition of probable innocence. It is easy to see (cfr. also [22]) that the condition (9), in case $p_f > \frac{1}{2}$, is equivalent to the following:

$$m \geq \frac{p_f}{p_f - 1/2} (c + 1), \text{ given } p_f > 1/2. \quad (10)$$

Let us consider the line in the graph where $p_f = 0.8$. By applying relation (10), we see that probable innocence is achieved for $n \geq 64$. As we can see from the graph, along the line $p_f=0.8$, Z increases rapidly when n increases beyond 64.

6.3 Study of the bounds in relation to the condition of probable innocence

We now plot Z as a function of $\frac{c+1}{m}$ (x -axis) and $\frac{p_f-1/2}{p_f}$ (y -axis). This plots is shown in Figure 2. Note that we are justified in taking $\frac{p_f-1/2}{p_f}$ and $\frac{c+1}{m}$ as the

³ Note that in our case we use the convention of linear algebra for the matrix, while [5] uses the convention of Information Theory, and as a consequence the roles of the rows and columns are exchanged

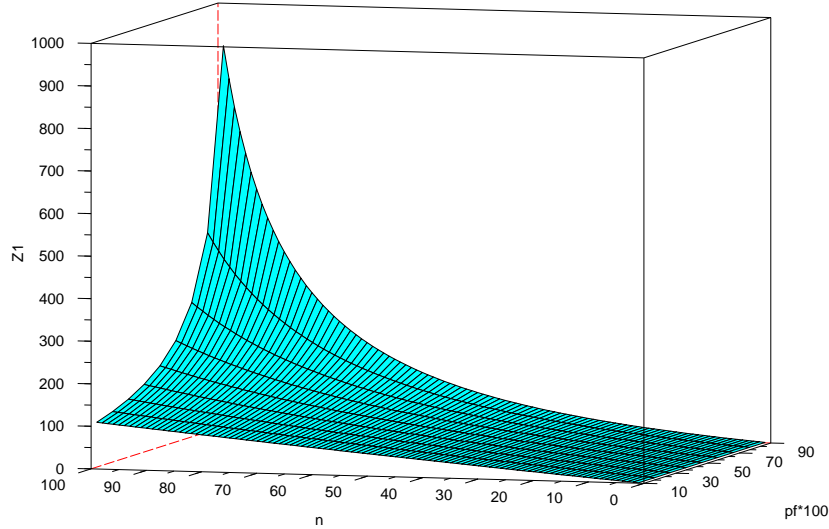


Fig. 1. The plot of Z as a function of n and p_f , and for $m = 100$.

independent variables because we keep m constant and thus Z can entirely be written in terms of these two new variables without any explicit mention of p_f and c .

In the plot in Figure 2, the minimum value of Z is 2.0132 at $x = 0.9875$, $y = 0.038$. Again, the convergence to 0 as x approaches 1 is only apparent and due to the large scale of Z .

In all the readings, $p_f > 1/2$. Thus, probable innocence is satisfied in the region where the x -value is smaller than the y -value. We observe that there is a considerable increase in the slope in this region, and that the inclination is parallel to the plane $x = y$.

References

1. Mohit Bhargava and Catuscia Palamidessi. Probabilistic anonymity. In Martín Abadi and Luca de Alfaro, editors, *Proceedings of CONCUR*, volume 3653 of *Lecture Notes in Computer Science*, pages 171–185. Springer, 2005.
2. Konstantinos Chatzikokolakis and Keye Martin. A monotonicity principle for information theory. In *Proceedings of the Twenty-fourth Conference on the Mathematical Foundations of Programming Semantics*, 2008. To appear.
3. Konstantinos Chatzikokolakis and Catuscia Palamidessi. Probable innocence revisited. *Theoretical Computer Science*, 367(1-2):123–138, 2006.

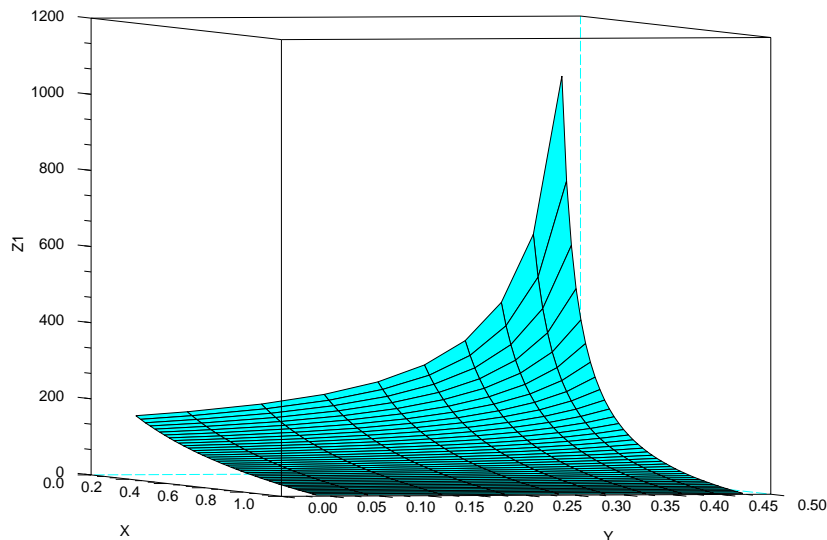


Fig. 2. A plot of Z as a function of $\frac{c+1}{m}$ and $\frac{p_f-1/2}{p_f}$, and for $m = 100$.

4. Konstantinos Chatzikokolakis, Catuscia Palamidessi, and Prakash Panangaden. Probability of error in information-hiding protocols. In *Proceedings of the 20th IEEE Computer Security Foundations Symposium (CSF20)*, pages 341–354. IEEE Computer Society, 2007.
5. Konstantinos Chatzikokolakis, Catuscia Palamidessi, and Prakash Panangaden. Anonymity protocols as noisy channels. *Information and Computation*, 206(2–4):378–401, 2008.
6. David Chaum. The dining cryptographers problem: Unconditional sender and recipient untraceability. *Journal of Cryptology*, 1:65–75, 1988.
7. David L. Chaum. Untraceable electronic mail, return addresses, and digital pseudonyms. *Communications of the ACM*, 24(2):84–90, 1981.
8. David Clark, Sebastian Hunt, and Pasquale Malacaria. Quantitative analysis of the leakage of confidential data. In *Proc. of QAPL 2001*, volume 59 (3) of *Electr. Notes Theor. Comput. Sci.*, pages 238–251. Elsevier Science B.V., 2001.
9. David Clark, Sebastian Hunt, and Pasquale Malacaria. Quantified interference for a while language. In *Proc. of QAPL 2004*, volume 112 of *Electr. Notes Theor. Comput. Sci.*, pages 149–166. Elsevier Science B.V., 2005.
10. Michael R. Clarkson, Andrew C. Myers, and Fred B. Schneider. Belief in information flow. *Journal of Computer Security*, 2008. To appear. Available as Cornell Computer Science Department Technical Report TR 2007-207.
11. Thomas M. Cover and Joy A. Thomas. *Elements of Information Theory*. John Wiley & Sons, Inc., 1991.
12. Claudia Díaz, Stefaan Seys, Joris Claessens, and Bart Preneel. Towards measuring anonymity. In Roger Dingledine and Paul F. Syverson, editors, *Proceedings of the*

- workshop on Privacy Enhancing Technologies (PET) 2002*, volume 2482 of *Lecture Notes in Computer Science*, pages 54–68. Springer, 2002.
13. Roger Dingledine, Nick Mathewson, and Paul Syverson. Tor: The second-generation onion router. In *Proceedings of the 13th USENIX Security Symposium*, August 2004.
 14. Michael J. Freedman and Robert Morris. Tarzan: A peer-to-peer anonymizing network layer. In *Proceedings of the 9th ACM Conference on Computer and Communications Security (CCS 2002)*, Washington, DC, November 2002.
 15. J. W. Gray, III. Toward a mathematical foundation for information flow security. In *Proceedings of the 1991 IEEE Computer Society Symposium on Research in Security and Privacy (SSP '91)*, pages 21–35, Washington - Brussels - Tokyo, May 1991. IEEE.
 16. Joseph Y. Halpern and Kevin R. O'Neill. Anonymity and information hiding in multiagent systems. *Journal of Computer Security*, 13(3):483–512, 2005.
 17. Gavin Lowe. Quantifying information flow. In *Proc. of CSFW 2002*, pages 18–31. IEEE Computer Society Press, 2002.
 18. Pasquale Malacaria. Assessing security threats of looping constructs. In Martin Hofmann and Matthias Felleisen, editors, *Proceedings of the 34th ACM SIGPLAN-SIGACT Symposium on Principles of Programming Languages, POPL 2007, Nice, France, January 17-19, 2007*, pages 225–235. ACM, 2007.
 19. John McLean. Security models and information flow. In *SSP'90*, pages 180–189. IEEE, 1990.
 20. Ira S. Moskowitz, Richard E. Newman, Daniel P. Crepeau, and Allen R. Miller. Covert channels and anonymizing networks. In Sushil Jajodia, Pierangela Samarati, and Paul F. Syverson, editors, *WPES*, pages 79–88. ACM, 2003.
 21. Ira S. Moskowitz, Richard E. Newman, and Paul F. Syverson. Quasi-anonymous channels. In *IASTED CNIS*, pages 126–131, 2003.
 22. Michael K. Reiter and Aviel D. Rubin. Crowds: anonymity for Web transactions. *ACM Transactions on Information and System Security*, 1(1):66–92, 1998.
 23. Nandakishore Santhi and Alexander Vardy. On an improvement over Rényi's equivocation bound, 2006. Presented at the 44-th Annual Allerton Conference on Communication, Control, and Computing, September 2006. Available at <http://arxiv.org/abs/cs/0608087>.
 24. Andrei Serjantov and George Danezis. Towards an information theoretic metric for anonymity. In Roger Dingledine and Paul F. Syverson, editors, *Proceedings of the workshop on Privacy Enhancing Technologies (PET) 2002*, volume 2482 of *Lecture Notes in Computer Science*, pages 41–53. Springer, 2002.
 25. V. Shmatikov. Probabilistic model checking of an anonymity system. *Journal of Computer Security*, 12(3/4):355–377, 2004.
 26. Geoffrey Smith. Adversaries and information leaks (tutorial). In Gilles Barthe and Cédric Fournet, editors, *Proceedings of the Third Symposium on Trustworthy Global Computing*, volume 4912 of *Lecture Notes in Computer Science*, pages 383–400. Springer, 2007.
 27. P.F. Syverson, D.M. Goldschlag, and M.G. Reed. Anonymous connections and onion routing. In *IEEE Symposium on Security and Privacy*, pages 44–54, Oakland, California, 1997.
 28. Henk Tijms. *Understanding Probability: Chance Rules in Everyday Life*. Cambridge University Press, 2007.
 29. Ye Zhu and Riccardo Bettati. Anonymity vs. information leakage in anonymity systems. In *Proc. of ICDCS*, pages 514–524. IEEE Computer Society, 2005.