Do Dummies Pay Off? Limits of Dummy Traffic Protection in Anonymous Communications

Simon Oya*, Carmela Troncoso†, and Fernando Pérez-González*

* Signal Theory and Communications Dept., University of Vigo, Spain,
{simonoya,fperez}@gts.uvigo.es
† Gradiant (Galician R&D Center in Advanced Telecommunications)
ctroncoso@gradiant.org

Abstract. Anonymous communication systems ensure that correspondence between senders and receivers cannot be inferred with certainty. However, when patterns are persistent, observations from anonymous communication systems enable the reconstruction of user behavioral profiles. Protection against profiling can be enhanced by adding dummy messages, generated by users or by the anonymity provider, to the communication. In this paper we study the limits of the protection provided by this countermeasure. We propose an analysis methodology based on solving a least squares problem that permits to characterize the adversary's profiling error with respect to the user behavior, the anonymity provider behavior, and the dummy strategy. Focusing on the particular case of a timed pool mix we show how, given a privacy target, the performance analysis can be used to design optimal dummy strategies to protect this objective.

Keywords: anonymous communications, disclosure attacks, dummies

1 Introduction

Anonymization is a popular mechanism to provide private communications. Anonymous communication [1] ensures that relationships between senders and receivers of messages cannot be inferred with certainty by the adversary. These schemes hide communication patterns by delaying and changing the appearance of messages [2] in such a way that sent messages can be ascribed to a set of potential receivers, often denoted as anonymity set. In practice, user behavior and latency constrain the composition of anonymity sets, which in turn enables an adversary observing the anonymous communication system to reconstruct persistent user behavioral profiles [3,4,5,6].

A common approach to improve users' protection against profiling is to introduce dummy traffic, either generated by users [7] or by the anonymity provider [8]. The effectiveness of this countermeasure has been studied theoretically from the perspective of individual messages in [9]. With respect to profiling, dummy traffic has been tackled in [10,5], where the authors empirically compute the number of rounds that the attacker takes to correctly identify some or all recipients of a sender. The analyses in [10,5] are limited in two aspects. On the one hand, the results strongly depend on the specific cases considered in the experiments, and it is difficult to get insight on their applicability to other scenarios. On the other hand, the analyses only consider the ability of the adversary in identifying communication partners, but not her accuracy at estimating the intensity of the communication; i.e., the users' profiles.

In this paper we propose an analysis methodology based on the least squares approach introduced in [6] that permits system designers to characterize the adversary's profiling error with respect to the user behavior, the anonymity provider behavior, and the dummy strategy. Our estimator can be used to characterize the error for bilateral relationships, individual user profiles, or the population as a whole. Our approach can accommodate a wide range of high-latency anonymous communication schemes providing the analyst with a bound on the protection achievable through the use of dummy traffic.

Another shortcoming of previous works [9,10,5] is that the proposed evaluation strategies cannot be used to guide the design of effective dummy generation strategies, which is recognized to be a hard problem [11]. This has lead the deployed high latency anonymous communication systems to either implement arbitrary dummy strategies [12] or no dummy traffic at all [11]. Our methodology can be used to support the design of dummy strategies by approaching strategy selection as an optimization problem in which the error of the adversary is maximized. The optimization criteria can be chosen by the designer to satisfy different privacy objectives, e.g., balancing the protection among users, or favoring individual users or relationships.

We illustrate the operation of our methodology using a timed binomial pool mix. We provide a performance analysis of this mixing strategy in presence of both sender-based and mix-based dummy traffic, showing that their contribution to the adversary's error can be decoupled and analyzed independently. Departing from this analysis, we design dummy traffic strategies according to two privacy criteria: increasing the estimation error for all the relationships by a constant factor, and guaranteeing a minimum estimation error for any relationship. By hiding relationships, both criteria hinder adversary's effort to infer user profiles.

Next section describes an abstract model of an anonymous communication system with dummies. Section 3 introduces a least squares-based profile estimator for dummy-based anonymization systems. We analyze in Sect. 4 the performance of this estimator when the anonymous channel is a timed binomial pool mix. The result of this analysis is used in Sect. 5 to design optimal dummy strategies, evaluated in Sect. 6. We discuss practical aspects of our method in Sect. 7 and finally conclude in Sect. 8.

2 System and Adversary Model

In this section we introduce the system and adversary model considered in the paper, as well as the general notation of the paper (summarized in Table 1). Throughout the document we use capital letters to denote random variables and lower-case letters to denote realizations of those variables. Vectors and matrices are denoted by boldface characters. Vectors of random variables are upper-cased, while their realizations are lower cased. Matrices are always denoted by upper-case boldface characters; whether they are random matrices or realizations will be clear from the context. Furthermore, we use $\mathbf{1}_n$ to denote the all-ones column vector of size n, $\mathbf{1}_{n\times m}$ to denote the all-ones $n \times m$ matrix and $\mathbf{I}_n$ for the $n \times n$ identity matrix.

System Model. Our system consists of a population of N senders, designated by index $i \in \{1, 2, \cdots, N\}$, which exchange messages with a set of M receivers, designated by index $j \in \{1, 2, \cdots, M\}$, through a high-latency mix-based anonymous communication channel. Messages in the system may be real or dummy messages: decoy messages indistinguishable from real traffic. We consider two types of *dummy traffic*:

- **Sender-based dummies:** senders may send *dummy* messages to the mix along with their *real* messages. Sender-based dummies can be recognized and discarded by the mix.
- **Mix-based dummies:** the mix-based system may send *dummy* messages to the receivers along with the *real* messages from the senders. Receivers are able to identify dummy messages and discard them.

Mix-based anonymous communication channels protect profiles by delaying messages and outputting them in batch in what are called *rounds* of mixing. We consider that the total number of messages generated by user i in round r is modeled by the random variable X_i^r. User messages can be real, modeled by random variable $X_{\lambda,i}^r$, or dummy, modeled by random variable $X_{\delta,i}^r$. These messages are sent to an anonymous communication channel in which a round of mixing consists of the following sequence of four stages, shown in Fig. 1. In the first stage, dummy messages are identified and discarded (Stage 1), while the real messages go inside the pool (Stage 2). Messages inside the pool are delayed until a specific *firing condition* is fullfilled, and then a number of them, chosen according to a *batching strategy*, exit the pool. Messages leaving the pool (modeled by random variable $X_{s,i}^r$) traverse a mixing block (Stage 3), which changes their appearance cryptographically to avoid bit-wise linkability. Messages staying in the pool are mixed with incoming real messages from subsequent rounds until they are fired. Finally, in Stage 4, mix-based dummies are added the output traffic and messages are delivered to their recipients. The number of mix-based dummies sent in round r is modeled by $X_{\mathtt{MIX}}^r$, and random variables $Y_{\lambda,j}^r$, $Y_{\delta,j}^r$ and Y_j^r model the number of real, dummy, and total messages received by receiver j in round r, respectively.

We also define the following vectors and matrices, which shall come handy later: matrix $\mathbf{U}$ is an $\rho \times N$ matrix which contains all the input observations, i.e., its (r, i)-th element is X_i^r. Similarly, matrix $\mathbf{U}_s$ contains in its (r, i)-th position the random variable $X_{s,i}^r$. Moreover, $\mathbf{H} \doteq \mathbf{I}_M \otimes \mathbf{U}$ and $\mathbf{H}_s \doteq \mathbf{I}_M \otimes \mathbf{U}_s$, where $\otimes$ denotes the Kronecker product. Vectors $\mathbf{Y}_j \doteq [Y_j^1, \cdots, Y_j^\rho]^T$ and $\hat{\mathbf{Y}}_{\delta,j} \doteq [Y_{\delta,j}^1, \cdots, Y_{\delta,j}^\rho]^T$ contain the random variables modeling the total (or just dummy)

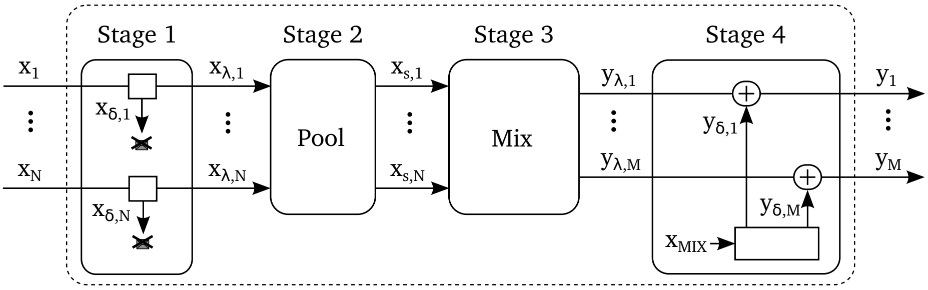


Fig. 1: Abstract model of a round in a mix-based anonymous communications channel (we omit the subscript r for the sake of clarity.)

number of messages received by j in each round. Finally, $\mathbf{Y} \doteq [\mathbf{Y}_1^T, \cdots, \mathbf{Y}_M^T]^T$ and $\hat{\mathbf{Y}}_\delta \doteq [\hat{\mathbf{Y}}_{\delta,1}^T, \cdots, \hat{\mathbf{Y}}_{\delta,M}^T]^T$.

We model the sending behavior of users in our population with two parameters:

- **Probability of real message:** the probability of real messages models how frequently users send real messages, and is denoted by $P_{\lambda_i}, i = 1, \cdots, N$. In other words, a message sent by i is real with probability P_{λ_i}, dummy otherwise. We make no assumptions on the values of P_{λ_i} other than $0 \leq P_{\lambda_i} \leq 1$, and that the probabilities of real messages are stationary during the observation period. Note that P_{λ_i} does not constrain the distributions that model the number of messages sent by users (X_i^r, $X_{\lambda,i}^r$ and $X_{\delta,i}^r$).
- **Sender profile:** the sender profile of user i models this sender's choice of recipients for her messages. It is defined as the vector $\mathbf{q}_i \doteq [p_{1,i}, p_{2,i}, \cdots, p_{M,i}]^T$, where $p_{j,i}$ denotes the probability that sender i sends a real message to receiver j. We also define the unnormalized receiver profile $\mathbf{p}_j \doteq [p_{j,1}, \cdots, p_{j,N}]^T$ and the vector containing all transition probabilities $\mathbf{p} \doteq [\mathbf{p}_1^T, \cdots, \mathbf{p}_M^T]^T$. We make no assumptions on the shape of the sender profiles other than $\mathbf{q}_i$ is in $\mathcal{P}$, the probability simplex in $\mathbb{R}^M$, i.e., $\mathcal{P} \doteq \left\{\mathbf{r} \in \mathbb{R}^M : r_i \geq 0, \sum_{i=1}^M r_i = 1\right\}$. We assume, nevertheless, that users' behavior is stationary during the observation period (the transition probabilities $p_{j,i}$ do not change between rounds), independent (the behavior of a user does not affect the behavior of the others) and memoryless (the messages sent by a user in a round do not affect the behavior of that user in subsequent rounds). We discuss the implications of the hypotheses above being false in Sect. 7.

The behavior of the mix-based anonymous communication channel is modeled by four parameters:

- **Firing condition:** the firing condition is an event, e.g., the arrival of a message (theshold mix) or the expiration of a timeout (timed mix), that causes the mix to forward some of the messages it has stored in its pool to their recipients.

– **Batching strategy:** the batching strategy models how messages are chosen to leave the pool. This strategy is determined by the function $F_{r,k}$, which models the probability that a message arriving in round k leaves the mix in round r ($r \geq k$). We do not make any assumption on the values of these parameters, other than $\sum_{r=k}^{\infty} F_{r,k} = 1$, i.e., every message will eventually leave the pool and get to its recipient. This function can for instance model a threshold mix ($F_{k,k} = 1$), or a binomial pool mix [10,5].

– **Average mix-based dummies:** this parameter, denoted as $\delta_{\texttt{MIX}}$, defines the average number of dummy messages generated by the mix each round. Note that our model does not assume any specific distribution for the number of mix-based dummies that are generated each round.

– **Mix dummy profile:** we denote by $\mathbf{q}_{\texttt{MIX}}$ the vector modeling the distribution of mix-based dummies among the receivers, $\mathbf{q}_{\texttt{MIX}} \doteq \{p_{1,\texttt{MIX}}, p_{2,\texttt{MIX}}, \cdots, p_{M,\texttt{MIX}}\}$ where $p_{j,\texttt{MIX}}$ is the probability that a dummy message generated by the mix is sent to receiver j ($\mathbf{q}_{\texttt{MIX}} \in \mathcal{P}$).

Adversary Model. We consider a global passive adversary that observes the system during ρ rounds. The adversary is able to see the identity of each sender and receiver communicating through the mix, but she is not able to link any two messages by their content nor distinguish between real and dummy messages. We assume that the adversary knows all the parameters of the system (e.g., the batching strategy determined by $F_{r,k}$, the parameters modeling the generation of dummy messages P_{λ_i} and $\delta_{\texttt{MIX}}$, the mix dummy profile $\mathbf{q}_{\texttt{MIX}}$). The goal of the adversary is to infer the sending profiles of the users in the system from the observations, i.e., to obtain an estimator $\hat{p}_{j,i}$ of the probabilities $p_{j,i}$ given the input and output observations x_i^r and y_j^r, for every $i \in \{1, 2, \cdots, N\}$, $j \in \{1, 2, \cdots, M\}$ and $r \in \{1, 2, \cdots, \rho\}$.

3 A Least Square Profile Estimator for Dummy-based Anonymization Systems

We aim here at deriving a least squares estimator for the probabilities $p_{j,i}$ for every $i = 1, 2, \cdots, N$ and $j = 1, 2, \cdots, M$, given the the observation of ρ rounds of mixing, x_i^r and y_j^r for $r = 1, \cdots, \rho$ and $\forall i, j$. Following the methodology in [13], we derive the estimator of $p_{j,i}$ by looking for the vector of probabilities $\mathbf{p}$ which minimizes the Mean Squared Error (MSE) between the random vector $\mathbf{Y}$ and the observed realization $\mathbf{y}$:

$$\hat{\mathbf{p}} = \underset{\mathbf{q}_i \in \mathcal{P},\, i=1,\cdots,N}{\arg\min} \mathrm{E}\left\{ ||\mathbf{y} - \mathbf{Y}(\mathbf{p})||^2 \right\} \tag{1}$$

where we have written $\mathbf{Y}(\mathbf{p})$ to stress the fact that the output distribution depends on all the transition probabilities $\mathbf{p}$. Note that, for notational simplicity, we are dropping the conditioning on $\mathbf{U}$ here. Even though the estimator in (1) minimizes the average error in the outputs, this does not mean it necessarily

Table 1: Summary of notation

Symbol	**Meaning**
N	Number of senders in the population, denoted by $i \in \{1, 2, \cdots, N\}$
M	Number of receivers in the population, denoted by $j \in \{1, 2, \cdots, M\}$
$F_{r,k}$	Probability that a message arriving in round k leaves the mix-based system in round r
$p_{j,i}$	Probability that user i sends a real message to receiver j
$p_{j,\mathtt{MIX}}$	Probability that the mix sends a mix-based dummy message to receiver j
$\mathbf{q}_i$	Sender profile of sender i, $\mathbf{q}_i \doteq [p_{1,i}, p_{2,i}, \cdots, p_{M,i}]^T$
$\mathbf{q}_{\mathtt{MIX}}$	Mix dummy profile, $\mathbf{q}_{\mathtt{MIX}} \doteq [p_{1,\mathtt{MIX}}, p_{2,\mathtt{MIX}}, \cdots, p_{M,\mathtt{MIX}}]^T$
$\mathbf{p}_j$	Unnormalized receiver profile for receiver j, $\mathbf{p}_j \doteq [p_{j,1}, p_{j,2}, \cdots, p_{j,N}]^T$
$\mathbf{p}$	Vector of transition probabilities, $\mathbf{p} \doteq [\mathbf{p}_1^T, \mathbf{p}_2^T, \cdots, \mathbf{p}_M^T]^T$
P_{λ_i}	Probability that user i sends a real message instead of a dummy
$\delta_{\mathtt{MIX}}$	Average number of mix-based dummies generated by the mix each round
ρ	Number of rounds observed by the adversary
$x^r_{\lambda,i}(x^r_{\delta,i})$	Number of real (dummy) messages sent by user i in round r
x^r_i	Total number of messages sent by user i in round r, $x^r_i \doteq x^r_{\lambda,i} + x^r_{\delta,i}$
$x^r_{s,i}$	Number of real messages sent by user i that leave the pool in round r
$y^r_{\lambda,j}(y^r_{\delta,j})$	Number of real (dummy) messages received by j in round r
y^r_j	Total number of messages received by j in round r, $y^r_j \doteq y^r_{\lambda,j} + y^r_{\delta,j}$
$x^r_{\mathtt{MIX}}$	Number of mix-based dummies generated by the mix in round r
$\mathbf{U}$ ($\mathbf{U}_s$)	$\rho \times N$ matrix containing all input observations $(\mathbf{U})_{r,i} = x^r_i$ $((\mathbf{U}_s)_{r,i} = x^r_{s,i})$
$\mathbf{H}$ ($\mathbf{H}_s$)	$\mathbf{I}_M \otimes \mathbf{U}$ $(\mathbf{I}_M \otimes \mathbf{U}_s)$
$\mathbf{y}_j$	Column vector containing the values y^r_j for $r = 1, \cdots, \rho$
$\mathbf{y}_{\delta,j}$	Column vector containing the values $y^r_{\lambda,j}$ for $r = 1, \cdots, \rho$
$\mathbf{y}$	Column vector containing all the output messages $\mathbf{y} \doteq [\mathbf{y}_1^T, \mathbf{y}_2^T, \cdots, \mathbf{y}_M^T]^T$
$\mathbf{y}_\delta$	Vector of output dummies $\mathbf{y}_\delta \doteq [\mathbf{y}_{\delta,1}^T, \mathbf{y}_{\delta,2}^T, \cdots, \mathbf{y}_{\delta,M}^T]^T$
$\hat{p}_{j,i}$, $\hat{\mathbf{p}}_j$, $\hat{\mathbf{p}}$,	Adversary's estimation of $p_{j,i}$, $\mathbf{p}_j$ and $\mathbf{p}$, respectively.
$\hat{\mathbf{y}}_\delta$, $\hat{\mathbf{y}}_{\delta,j}$	Adversary's estimation of $\mathbf{y}_{\delta,j}$ and $\mathbf{y}_\delta$.
$\hat{\mathbf{U}}_s$, $\hat{\mathbf{H}}_s$	Adversary's estimation of $\mathbf{U}_s$ and $\mathbf{H}_s$.

minimizes the error in the estimation of the probabilities $\mathbf{p}$. As shown in the derivations in [13], one can set the alternative problem

$$\hat{\mathbf{p}} = \underset{\mathbf{q}_i \in \mathcal{P},\, i=1,\cdots,N}{\arg\min} \left\{ ||\mathbf{y} - \mathrm{E}\left\{\mathbf{Y}(\mathbf{p})\right\}||^2 \right\} \tag{2}$$

in order to get an estimator $\hat{\mathbf{p}}$ that is not only unbiased, but also *asymptotically efficient*, i.e., the vector of estimated probabilities $\hat{\mathbf{p}}$ converges to the true value as the number of observations increases $\rho \to \infty$.

From the relations among the variables in Fig. 1, we can compute the expected value of the output $\mathbf{Y}(\mathbf{p})$ given the input observations $\mathbf{U}$ obtaining $\mathrm{E}\left\{\mathbf{Y}(\mathbf{p})\right\} = \hat{\mathbf{H}}_s \cdot \mathbf{p} + \hat{\mathbf{y}}_\delta$ (see Appendix), where

– $\hat{\mathbf{H}}_s \doteq \mathbf{I}_M \otimes \hat{\mathbf{U}}_s$, and $\hat{\mathbf{U}}_s$ (see (31)) is the matrix containing the attacker's estimation of the hidden random variables $X^r_{s,i}$, which model the number of messages from user i that leave the mix in round r (cf. Fig. 1).

- $\hat{\mathbf{y}}_\delta$ is the adversary's estimation of the number of mix-based dummies that are sent to each receiver in each round, and is given by $\hat{\mathbf{y}}_\delta = (\mathbf{I}_M \otimes \delta_{\mathtt{MIX}} \mathbf{1}_\rho) \cdot \mathbf{q}_{\mathtt{MIX}}$.

Interestingly, removing the constraints from (2) leads to an estimator which is still unbiased and asymptotically efficient, as proven in [13], and also makes a detailed performance analysis manageable as we show in Sect. 4. In the rest of this section we focus on the unconstrained estimator and refer to [13] for further information about the constrained variant. The solution to the unconstrained problem

$$\hat{\mathbf{p}} = \underset{\mathbf{q}_i\ i=1,\cdots,N}{\arg\min} \left\{ ||\mathbf{y} - \hat{\mathbf{H}}_s \cdot \mathbf{p} - \hat{\mathbf{y}}_\delta||^2 \right\} \tag{3}$$

is given by the Moore-Penrose pseudo-inverse, i.e., $\hat{\mathbf{p}} = \left(\hat{\mathbf{H}}_s^T \hat{\mathbf{H}}_s\right)^{-1} \hat{\mathbf{H}}_s^T \left(\mathbf{y} - \hat{\mathbf{y}}_\delta\right)$. This solution can be decoupled [13] resulting in a more tractable and efficient equation,

$$\hat{\mathbf{p}}_j = \left(\hat{\mathbf{U}}_s^T \hat{\mathbf{U}}_s\right)^{-1} \hat{\mathbf{U}}_s^T \left(\mathbf{y}_j - \hat{\mathbf{y}}_{\delta,j}\right) \qquad j = 1, \cdots, M \tag{4}$$

where $\hat{\mathbf{y}}_{\delta,j} \doteq \delta_{\mathtt{MIX}} p_{j,\mathtt{MIX}} \mathbf{1}_\rho$ contains the expected number of mix-based dummies sent to receiver j in each round. Given the system parameters as well as the input and output observations $\mathbf{U}$ and $\mathbf{y}$, the adversary can use (4) to get an estimation of the users' sending profiles.

4 Performance Analysis of a Dummy-based Timed Pool Mix Anonymous Communication System

In this section, we assess the performance of the least squares estimator in (4) with respect to its profiling accuracy, measured as the Mean Squared Error of estimated transition probabilities $p_{j,i}$ ($\mathtt{MSE}_{j,i} = |\hat{p}_{j,i} - p_{j,i}|^2$) representing users' behaviour. We consider the particular case when the anonymous communication channel is a *binomial timed pool mix* [14], and the number of messages sent by the users, as well as the dummies generated by the mix, are Poisson-distributed. In a binomial timed pool mix, the firing condition is a timeout and the batching strategy mandates that individual messages leave the pool with probability α every round, i.e., $F_{r,k} = \alpha(1-\alpha)^{r-k}$. The behavior of this mix is stationary, since the value of $F_{r,k}$ only depends on the difference $r - k$. This scenario can be summarized as

$$\begin{gathered} X^r_{\lambda,i} \sim \text{Poiss}\left(\lambda_i\right), \quad X^r_{\delta,i} \sim \text{Poiss}\left(\delta_i\right), \quad X^r_{\mathtt{MIX}} \sim \text{Poiss}\left(\delta_{\mathtt{MIX}}\right) \\ P_{\lambda_i} = \lambda_i/(\lambda_i + \delta_i), \quad F_{r,k} = \alpha(1-\alpha)^{r-k} \end{gathered} \tag{5}$$

where λ_i is the *sending rate*, and δ_i is the *dummy rate*, representing the average number of real messages, respectively dummies, sent by user i. Even though the results we provide correspond to the above case we must stress that the reasoning followed in the derivation is applicable to any other system that can be represented by the model in Sect. 2.

4.1 Profiling error of the least squares estimator

Under the hypotheses stated in (5), the least squares estimator is unbiased and the $\mathtt{MSE}_{j,i}$ of a single transition probability estimated is given by [15]:

$$\mathtt{MSE}_{j,i} \approx \frac{1}{\rho} \cdot \frac{1}{\alpha_q} \cdot \frac{1}{\lambda_i} \cdot \left(1 + \frac{\delta_i}{\lambda_i}\right) \cdot \left(1 - \frac{\lambda_i + \delta_i}{\sum_{k=1}^{N}(\lambda_k + \delta_k)}\right) \cdot \left(\sum_{k=1}^{N} \lambda_k p_{j,k} + \delta_{\mathtt{MIX}} p_{j,\mathtt{MIX}} - \frac{\alpha_q}{\alpha_r} \sum_{k=1}^{N} \lambda_k P_{\lambda_k} p_{j,k}^2\right) \quad (6)$$

where $\alpha_q \doteq \dfrac{\alpha}{2-\alpha}$ and $\alpha_r \doteq \dfrac{\alpha(2-\alpha)}{2-\alpha(2-\alpha)}$. This result holds when: i) the probability that each sender sends a message to receiver j is negligible when compared to the rate at which receiver j receives messages from all users ($p_{j,i} \ll \sum_k \lambda_k p_{j,k}$), ii) the number of rounds observed is large enough ($\rho \to \infty$), and iii) $\lambda_i + \delta_i \ll (\sum_k (\lambda_k + \delta_k))^2$.

Interestingly, the terms in (6) that depend on i and j in can be decoupled:

$$\mathtt{MSE}_{j,i} \approx \frac{1}{\rho} \cdot \frac{1}{\alpha_q} \cdot \epsilon_s(i) \cdot \epsilon_r(j) \quad (7)$$

where $\epsilon_s(i)$ and $\epsilon_r(j)$ denote functions that only depend on the sender i and the receiver j respectively. This property proves to be very useful when designing strategies to distribute the dummy traffic as we later see in Sect. 5.

The latter expression allows to extract qualitative conclusions on the protection dummy traffic offers to senders and receivers. As it was already shown in [13], the MSE decreases with the number of rounds observed as $1/\rho$, and delaying messages in the pool increases the $\mathtt{MSE}_{j,i}$ by a factor $(2-\alpha)/\alpha$ with respect to an scenario with no delay (i.e., $\alpha = 1$).

We now analyze the contribution to the MSE of the users' behavior. The sender-side contribution $\epsilon_s(i)$ consists of three terms:

$$\epsilon_s(i) = \frac{1}{\lambda_i} \cdot \left(1 + \frac{\delta_i}{\lambda_i}\right) \cdot \left(1 - \frac{\lambda_i + \delta_i}{\sum_{k=1}^{N}(\lambda_k + \delta_k)}\right) \quad (8)$$

1. The term $1/\lambda_i$ implies that the error when estimating the profile $\mathbf{q}_i = [p_{1,i}, \cdots, p_{M,i}]^T$ decreases as that user participates in the system more often. Naturally, when more information about the user becomes available to the adversary, it becomes easier to accurately estimate her behavior.
2. The second term, $1 + \delta_i/\lambda_i$, is always larger or equal than one, meaning that sender-based dummies always hinder the attacker's estimation. The weight of this component depends on the ratio between the dummy rate and the sending rate. Hence, a user who sends real messages very often would need to send a many more dummies to get the same level of protection than a user who rarely participates in the system.

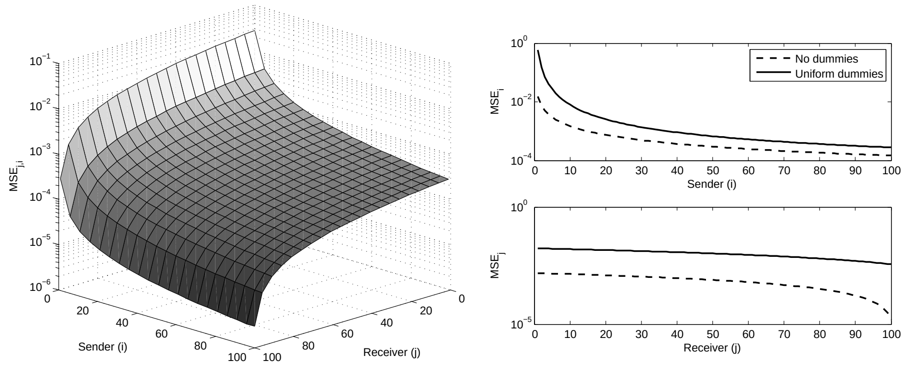
3. The last term is in general negligible since, in a normal scenario, the participation of a single user is negigible when compared to the total traffic, i.e., $\lambda_i + \delta_i \ll \sum_{k=1}^{N}(\lambda_k + \delta_k)$. However, when user i's traffic is clearly dominant among the others, this term decreases the overall gain i gets from dummies. Therefore, although sender-based dummies always increase the protection of a user, they offer diminishing returns when only one user is trying to protect herself by sending dummies.

On the other hand, receiver-side contribution, $\epsilon_r(j)$, consists of three summands:

$$\epsilon_r(j) = \sum_{k=1}^{N} \lambda_k p_{j,k} + \delta_{\mathtt{MIX}} p_{j,\mathtt{MIX}} - \frac{\alpha_q}{\alpha_r} \sum_{k=1}^{N} \lambda_k P_{\lambda_k} p_{j,k}^2 \tag{9}$$

1. The first summand is the rate at which j receives real messages from the senders. We call this term *receiver rate* and denote it by λ'_j. It is interesting to note that, contrary to the sending rates where large values of λ_i compromise the anonymity of the senders; large values of receiver rates increase the protection of the receivers. In other words, it is harder for the attacker to estimate probabilities related to a receiver which is contacted by a large number of senders than related to one receiving few messages.
2. The second summand is the rate at which j receives dummy messages from the mix. The interesting part about this summand is that it can be adjusted by the mix, to give more protection to a specific receiver j by increasing the number of dummies addressed to that recipient, i.e., increasing $p_{j,\mathtt{MIX}}$.
3. The last summand depends on the mix parameters and the users' behavior. Since $\alpha_q/\alpha_r \leq 1$ and $P_{\lambda_k} \leq 1$, when users do not focus their messages in few others, i.e., $p_{j,i} \ll 1$, this summand becomes negligible. However, if there is no dummy traffic ($P_{\lambda_k} = 1$ and $\delta_{\mathtt{MIX}} = 0$) and no pool is implemented ($\alpha_q/\alpha_r = 1$), this term must be taken into account. In this case $\epsilon_r(j)$ depends on the variance of the outputs, i.e. $\sum_{k=1}^{N} \lambda_k p_{j,k}(1 - p_{j,k})$, meaning that it would easier for the attacker to estimate probabilities $p_{j,k}$ of receivers that get messages from senders whose behavior has low variance (i.e., senders that always choose the same receiver, $p_{j,k} = 1$, or users that never send to a receiver, $p_{j,k} = 0$). Adding delay or introducing dummy traffic increases the variance of the output, thus reducing the dependency of the error on the sending profiles.

The fact that we can differentiate the contribution of i and j in (6) also allows for a graphic interpretation of the adversary's estimation error. Figure 2a represents the values of $\mathtt{MSE}_{j,i}$ as a function of i and j, in an scenario without dummies where for simplicity we have assumed that the sending rates are distributed in ascending order according to the senders' index i, and the receiving rates are distributed in descending order according to the receivers' index j. Fig. 2b shows the average $\mathtt{MSE}_{j,i}$ over j and i, offering a comparison with a system where the distribution of the dummies is uniform in both the input and output flows: $\epsilon_s(i)$ determines the evolution of $\mathtt{MSE}_{j,i}$ with i (top) and $\epsilon_r(j)$ the evolution with j



(a) (b)

Fig. 2: (a) $\texttt{MSE}_{j,i}$ as a function of i and j in an scenario where λ_i are sorted in ascending order and λ'_j in descending order. (b) Comparison of the average $\texttt{MSE}_{j,i}$ along j and i with and without dummies. ($N = 100$, $M = 100$, $\rho = 10\,000$, $\alpha = 0.5$, $\sum \lambda_k = 500$. In (b), $\delta_{\texttt{SEND}} = \delta_{\texttt{MIX}} = 250$).

(bottom). This means that by distributing dummies among sender-based and mix-based dummies, which in turn modify the value of $\epsilon_s(i)$ and $\epsilon_r(j)$, we can shape the $\texttt{MSE}_{j,i}$. We use this idea in the next section to design dummy strategies that satisfy different privacy criteria.

5 Designing Dummy Traffic Strategies

In this section, we study how to distribute dummy traffic in order to guarantee different privacy criteria. In other words, we aim at finding the values of the parameters δ_i for $i \in \{1, \cdots, N\}$ and $p_{j,\texttt{MIX}}$ for $j \in \{1, \cdots, M\}$ that maximize a certain cost function representing some privacy objective. We assume that the *total number of dummies* δ_{TOT} that can be sent *on average* per round is constrained. We denote the average number of sender-based dummies on each round as $\delta_{\texttt{SEND}} \doteq \sum_{i=1}^{N} \delta_i$, and the average number of mix-based dummies as $\delta_{\texttt{MIX}}$. We put no restriction on the distribution of dummies among senders and mix other than $\delta_{\texttt{SEND}} + \delta_{\texttt{MIX}} \leq \delta_{\texttt{TOT}}$. For notational simplicity, in the reminder of the section we omit the constraints $0 \leq p_{j,\texttt{MIX}} \leq 1$, $\sum_{j=1}^{M} p_{j,\texttt{MIX}}$, $\delta_i \geq 0$ and $\sum_{i=1}^{N} \delta_i = \delta_{\texttt{SEND}}$ in the equations.

In order to keep the optimization problems tractable, we assume that the contribution of a single user to the total input traffic is negligible (i.e., $\lambda_i + \delta_i \ll \sum_{k=1}^{N}(\lambda_k + \delta_k)$) and that users do not focus their traffic in a specific receiver (i.e., $p_{j,i} \ll 1$). In this case, we can approximate (6) as:

$$\widetilde{\texttt{MSE}}_{j,i} = \frac{1}{\rho} \cdot \frac{1}{\alpha_q} \cdot \frac{1}{\lambda_i} \cdot \left(1 + \frac{\delta_i}{\lambda_i}\right) \cdot \left(\lambda'_j + \delta_{\texttt{MIX}} p_{j,\texttt{MIX}}\right) = \frac{1}{\rho} \cdot \frac{1}{\alpha_q} \cdot \tilde{\epsilon}_s(i) \cdot \tilde{\epsilon}_r(j) \quad (10)$$

where $\lambda'_j \doteq \sum_{k=1}^{N} \lambda_k p_{j,k}$ is the receiver rate of j.

5.1 Increasing the protection of every sender-receiver relation by the largest factor $\boldsymbol{\beta}$ given a budget of dummies $\boldsymbol{\delta_{\texttt{TOT}}}$

In this section we design a dummy strategy that, given a budget of dummies $\delta_{\texttt{TOT}}$, increases $\texttt{MSE}_{j,i}$ of each transition probability $p_{j,i}$ by a factor $\beta \geq 1$ as large as possible with respect to the MSE when there are no dummies, denoted by $\texttt{MSE}^0_{j,i}$. Departing from (10) we can formalize this problem as:

$$\begin{aligned} \underset{\delta_1,\cdots,\delta_N,\mathbf{q}_{\texttt{MIX}}}{\text{maximize}} \quad & \widetilde{\texttt{MSE}}_{j,i}, \qquad \forall i,j \\ \text{subject to} \quad & \widetilde{\texttt{MSE}}_{j,i} = \beta \cdot \widetilde{\texttt{MSE}}^0_{j,i}, \qquad \forall i,j \\ & \delta_{\texttt{SEND}} + \delta_{\texttt{MIX}} = \delta_{\texttt{TOT}} \end{aligned} \tag{11}$$

Since the effects of the sender-based and mix-based dummies can be decoupled, we can split the optimization into three subproblems:

1. Find the distribution of δ_i that increases $\tilde{\epsilon}_s(i)$ by a factor $\beta_{\texttt{SEND}}$ for all i.
2. Find the distribution of $p_{j,\texttt{MIX}}$ that increases $\tilde{\epsilon}_r(j)$ by a factor $\beta_{\texttt{MIX}}$ for all j.
3. Find the distribution of $\delta_{\texttt{TOT}}$ between $\delta_{\texttt{SEND}}$ and $\delta_{\texttt{MIX}}$ that maximizes the overall increase $\beta = \beta_{\texttt{SEND}} \cdot \beta_{\texttt{MIX}}$.

Optimal distribution of sender-based dummies We want to find the distribution of δ_i among senders that increases $\tilde{\epsilon}_s(i)$ by a factor $\beta_{\texttt{SEND}}$ compared to the dummy-free case. Since $\tilde{\epsilon}_s(i) = 1/\lambda_i\left(1 + \frac{\delta_i}{\lambda_i}\right)$, sending δ_i dummies increases the MSE in a factor $\beta_{\texttt{SEND}} = 1 + \delta_i/\lambda_i$. We can now obtain the sender based dummy distribution, ensuring the that $\sum_{i=1}^{N} \delta_i = \delta_{\texttt{SEND}}$, as follows:

$$\beta_{\texttt{SEND}} = 1 + \frac{\delta_{\texttt{SEND}}}{\sum_{k=1}^{N} \lambda_k} \Longrightarrow \delta_i = \frac{\lambda_i}{\sum_{k=1}^{N} \lambda_k} \cdot \delta_{\texttt{SEND}}, \qquad \forall i \tag{12}$$

This confirms the intuition given in Sect. 4, that the number of dummies a user should send to achieve a certain level of protection is proportional to her sending rate of real messages.

Optimal distribution of mix-based dummies Similarly, we want to find the distribution of $p_{j,\texttt{MIX}}$ among receivers that increases $\tilde{\epsilon}_r(j)$ by a factor $\beta_{\texttt{MIX}}$ compared to the dummy-free case. Since $\tilde{\epsilon}_r(j) = \lambda'_j + \delta_{\texttt{MIX}} p_{j,\texttt{MIX}}$, assigning sending dummies with probability $p_{j,\texttt{MIX}}$ to receiver j increases the MSE by a factor $\beta_{\texttt{MIX}} = 1 + \delta_{\texttt{MIX}} p_{j,\texttt{MIX}}/\lambda'_j$. We can now obtain the sender-based dummy distribution, ensuring that $\sum_{j=1}^{M} p_{j,\texttt{MIX}} = 1$, as follows:

$$\beta_{\texttt{MIX}} = 1 + \frac{\delta_{\texttt{MIX}}}{\sum_{m=1}^{M} \lambda'_m} \Longrightarrow p_{j,\texttt{MIX}} = \frac{\lambda'_j}{\sum_{m=1}^{M} \lambda'_m}, \qquad \forall j \tag{13}$$

As said in Sect. 4, the protection that receivers enjoy is proportional to their receiving rate. Therefore, to increase all $\texttt{MSE}_{j,i}$s by the same factor, more mix-based dummies have to be given to those receivers that receive more real messages.

Optimal distribution of the overall amount of dummies Using the distributions obtained, and since $\sum_{k=1}^{N} \lambda_k = \sum_{m=1}^{M} \lambda'_m$, we can write $\widetilde{\mathtt{MSE}}_{j,i}$ as

$$\widetilde{\mathtt{MSE}}_{j,i} = \widetilde{\mathtt{MSE}}^{0}_{j,i} \cdot \beta_{\mathtt{SEND}} \cdot \beta_{\mathtt{MIX}} = \widetilde{\mathtt{MSE}}^{0}_{j,i} \left(1 + \frac{\delta_{\mathtt{SEND}}}{\sum_{k=1}^{N} \lambda_k}\right) \left(1 + \frac{\delta_{\mathtt{MIX}}}{\sum_{k=1}^{N} \lambda_k}\right) \tag{14}$$

The distribution of the total amount of dummies that maximizes the increase in $\widetilde{\mathtt{MSE}}_{j,i}$ is therefore $\delta_{\mathtt{SEND}} = \delta_{\mathtt{MIX}} = \delta_{\mathtt{TOT}}/2$. This result is particularly interesting: if we are to increase the relative protection of each user equally, the protection we get from sender-based and mix-based dummies is the same regardless of the system parameters. That is, assigning all our available dummies to the senders or to the mix is equivalent in terms of MSE, and distributing the dummies evenly between the input and output flow is optimal, being the maximum achievable gain $\beta \approx \left(1 + \frac{\delta_{\mathtt{TOT}}}{2\sum_k \lambda_k}\right)^2$.

5.2 Increasing the minimum protection to every sender-receiver relation given a budget of dummies $\delta_{\mathtt{TOT}}$

Our second design strategy consists in ensuring that, given a budget of dummies $\delta_{\mathtt{TOT}}$, the distribution maximizes the minimum level of protection for all relationships in the system. This implies that dummies are assigned to senders i and receivers j in relationships whose estimation error $\mathtt{MSE}_{j,i}$ is low, in order to increase the minimum $\mathtt{MSE}_{j,i}$ in the system. From a graphical point of view, we can see this as a two-dimensional waterfilling problem: we need to increase the lower $\mathtt{MSE}_{j,i}$ in Fig. 2a up to a minimum, which can be larger as more dummies $\delta_{\mathtt{TOT}}$ are available. More formally, we want to solve:

$$\begin{aligned} \underset{\delta_1,\cdots,\delta_N,\mathbf{q}_{\mathtt{MIX}}}{\text{maximize}} \quad & \min_{i,j} \widetilde{\mathtt{MSE}}_{j,i} \\ \text{subject to} \quad & \delta_{\mathtt{SEND}} + \delta_{\mathtt{MIX}} = \delta_{\mathtt{TOT}} \end{aligned} \tag{15}$$

As in the previous problem, we can separate the problem in three steps:

1. Find the distribution of δ_i that maximizes $\min_i \tilde{\epsilon}_s(i)$.
2. Find the distribution of $p_{j,\mathtt{MIX}}$ that maximizes $\min_j \tilde{\epsilon}_r(j)$.
3. Find the distribution of $\delta_{\mathtt{TOT}}$ among $\delta_{\mathtt{SEND}}$ and $\delta_{\mathtt{MIX}}$ that maximizes the minimum $\mathtt{MSE}_{j,i}$ in the system.

Optimal distribution for sender-based dummies We aim at finding the distribution of δ_i among senders that increases the minimum value of $\tilde{\epsilon}_s(i) = \frac{1}{\lambda_i}\left(1 + \frac{\delta_i}{\lambda_i}\right)$ over i, making it as large as possible given the budget of dummies. This subproblem can be formulated as

$$\begin{aligned} \underset{\delta_1,\cdots,\delta_N}{\text{maximize}} \quad & \min_{i} \tilde{\epsilon}_s(i) \\ \text{subject to} \quad & \sum_{i=1}^{N} \delta_i = \delta_{\mathtt{SEND}} \end{aligned} \tag{16}$$

Let $\mathcal{A}$ be the set containing the indices of those senders to whom we assign dummies, i.e., $\mathcal{A} \doteq \{i : \delta_i > 0\}$. Let $\tilde{\epsilon}_{s,\texttt{MIN}}$ be the minimum value of $\tilde{\epsilon}_s(i)$ we achieve with this strategy. Then, the following statements are true:

- We do not assign sender-based dummies to those users k whose $\tilde{\epsilon}_s(k) \geq \tilde{\epsilon}_{s,\texttt{MIN}}$ without dummies; i.e., we only use sender-based dummies to help users achieve that minimum.
- There is no gain in assigning dummies to a user k if by doing so we are increasing $\tilde{\epsilon}_s(k)$ above any other $\tilde{\epsilon}_s(i)$; i.e., every user $k \in \mathcal{A}$ fullfills $\tilde{\epsilon}_s(k) = \tilde{\epsilon}_{s,\texttt{MIN}}$.

Given $\tilde{\epsilon}_s(k) = \tilde{\epsilon}_{s,\texttt{MIN}}$, and to ensure $\sum_{k=1}^{N} \delta_k = \sum_{k\in\mathcal{A}} \delta_k = \delta_{\texttt{SEND}}$ we can get an expression for $\tilde{\epsilon}_{s,\texttt{MIN}}$:

$$\tilde{\epsilon}_{s,\texttt{MIN}} = \frac{1}{\lambda_k}\left(1 + \frac{\delta_k}{\lambda_k}\right) \Longrightarrow \tilde{\epsilon}_{s,\texttt{MIN}} = \frac{\delta_{\texttt{SEND}} + \sum_{k\in\mathcal{A}} \lambda_k}{\sum_{k\in\mathcal{A}} \lambda_k^2} \tag{17}$$

In order to compute $\mathcal{A}$, we assume w.l.o.g. that the indices are given to users such that their sending frequencies are sorted in *descending* order, $\lambda_1 \geq \lambda_2 \geq \cdots \geq \lambda_N$ and we let $\mathcal{A}_i \doteq \{1, 2, \cdots, i\}$. Then, $\mathcal{A} = \mathcal{A}_n$ where n is the minimum value that meets[1]

$$\frac{1}{\lambda_n} \leq \frac{\delta_{\texttt{SEND}} + \sum_{k\in\mathcal{A}_n} \lambda_k}{\sum_{k\in\mathcal{A}_n} \lambda_k^2} \leq \frac{1}{\lambda_{n+1}} \tag{18}$$

Finally, we assign

$$\delta_i = \begin{cases} \lambda_i \left(\lambda_i \tilde{\epsilon}_{s,\texttt{MIN}} - 1\right), & \text{if } i \in \mathcal{A}_n \\ 0, & \text{otherwise.} \end{cases} \tag{19}$$

Optimal distribution for mix-based dummies Similarly, we aim at finding the distribution of $p_{j,\texttt{MIX}}$ among receivers that increases the minimum value of $\tilde{\epsilon}_r(j)$, making it as large as possible given the budget of dummies. The problem can be formulated as:

$$\begin{aligned} \underset{p_{1,\texttt{MIX}},\cdots,p_{M,\texttt{MIX}}}{\text{maximize}} \quad & \min_j \tilde{\epsilon}_r(j) \\ \text{subject to} \quad & \sum_{j=1}^{M} p_{j,\texttt{MIX}} = 1 \end{aligned} \tag{20}$$

where $\tilde{\epsilon}_r(j) = \lambda'_j + \delta_{\texttt{MIX}} p_{j,\texttt{MIX}}$.

We define the set $\mathcal{B}$ as the send of receivers that get mix-based dummies, $\mathcal{B} \doteq \{j : p_{j,\texttt{MIX}} > 0\}$ and the minimum value of our optimization function we achieve with this strategy as $\tilde{\epsilon}_{r,\texttt{MIN}}$. Then, following the procedure described above, we get

$$\tilde{\epsilon}_{r,\texttt{MIN}} = \frac{\delta_{\texttt{MIX}} + \sum_{j\in\mathcal{B}} \lambda'_j}{|\mathcal{B}|} \tag{21}$$

[1] If the condition is not met because all $1/\lambda_n \leq \tilde{\epsilon}_{s,\texttt{MIN}}(\mathcal{A}_n)$, then we can assume that $n = N$, i.e., all users will send dummies.

where $|\mathcal{B}|$ denotes the number of elements of $\mathcal{B}$. If the receiver rates are sorted in *ascending* order, $\lambda'_1 \leq \lambda'_2 \leq \cdots \leq \lambda'_M$ and $\mathcal{B}_j \doteq \{1, 2, \cdots, j\}$, then the set of receivers that receive dummy messages is $\mathcal{B} = \mathcal{B}_n$ where the value of n is the smallest that meets

$$\lambda'_n \leq \frac{\delta_{\texttt{MIX}} + \sum_{j \in \mathcal{B}_n} \lambda'_j}{|\mathcal{B}_n|} \leq \lambda'_{n+1} \tag{22}$$

Finally, we assign

$$p_{j,\texttt{MIX}} = \begin{cases} \dfrac{1}{\delta_{\texttt{MIX}}} \left(\tilde{\epsilon}_{r,\texttt{MIN}} - \lambda'_j\right), & \text{if } j \in \mathcal{B}_n \\ 0, & \text{otherwise.} \end{cases} \tag{23}$$

Optimal distribution of the overall amount of dummies In this case we cannot get a closed-form expression for the optimal distribution of $\delta_{\texttt{TOT}}$ among $\delta_{\texttt{SEND}}$ and $\delta_{\texttt{MIX}}$, since it depends on the sizes of the sets $\mathcal{A}$ and $\mathcal{B}$. The minimum $\widetilde{\texttt{MSE}}_{j,i}$ we achieve is for relationships where both sender and receiver are allocated dummies, i.e., $i \in \mathcal{A}$ and $j \in \mathcal{B}$. Hence we can obtain this minimum by plugging the distributions (19) and (23) into (10), obtaining

$$\min_{j,i} \widetilde{\texttt{MSE}} = \frac{1}{\rho} \cdot \frac{1}{\alpha_q} \cdot \frac{\delta_{\texttt{SEND}} + \sum_{k \in \mathcal{A}} \lambda_k}{\sum_{k \in \mathcal{A}} \lambda_k^2} \cdot \frac{\delta_{\texttt{MIX}} + \sum_{m \in \mathcal{B}} \lambda'_m}{|\mathcal{B}|} \tag{24}$$

Optimal values for $\delta_{\texttt{SEND}}$ and $\delta_{\texttt{MIX}}$ can be computed by performing an exhaustive search along $\delta_{\texttt{SEND}} + \delta_{\texttt{MIX}} = \delta_{\texttt{TOT}}$, computing each time the sets $\mathcal{A}$ and $\mathcal{B}$ as explained above. It is interesting to note that, if the number of dummies available is large enough, i.e., $\delta_{\texttt{TOT}} \to \infty$, every sender and receiver is assigned dummies. In this case, since $\sum_{k=1}^{N} \lambda_k = \sum_{m=1}^{M} \lambda'_m$, the optimal strategy would be to distribute the total amount of dummies evenly between the input and the output traffics, i.e., $\delta_{\texttt{SEND}} = \delta_{\texttt{MIX}} = \delta_{\texttt{TOT}}/2$.

6 Evaluation

In this section we evaluate the performance of the dummy traffic design strategies designed in Sect. 5, and validate them against the theoretical bound for the adversary's error in (6) through a simulator written in the Matlab language.[2] The scope of this analysis is focused on supporting our theoretical findings rather than comparing our estimator with existing attacks. The only attack in the literature extended to cover dummy traffic is the Statistical Disclosure Attack (SDA) [11,10] and it is already shown in [**?**,13] that the least squares-based approach performs asymptotically better than SDA. It must be noted that the Bayesian inference estimator (Vida) in [4] may return a better estimation than our least squares estimator. However, its computational cost is huge even for a threshold mix [13] and it would become prohibitive in a pool mix with dummies.

[2] The code will be available upon request.

Experimental Setup. We simulate a system with $N = 100$ senders and $M = 100$ receivers. The sending frequencies of the users are sorted in ascending order, in such a way that λ_i is proportional to i, and the average total number of real messages sent by all users is $\sum \lambda_i = 500$. The sending profiles $\mathbf{q}_i$ are set such that user i sends messages to herself and all other users $k < i$ with the same probability, i.e., $p_{j,i} = 1/i$ if $j \leq i$ and $p_{j,i} = 0$. This ensures that receiving rates λ'_j are sorted in descending order. The probability that a message is fired after each round is set to $\alpha = 0.5$, and the number of rounds observed by the attacker is $\rho = 10\,000$. The theoretical $\texttt{MSE}_{j,i}$ for this scenario without dummies is shown in Fig. 2a. Though not realistic, this experiment is sufficient to illustrate the operation of the strategies in Sect. 5. The amount of dummies that users and mix send and their distribution change between experiments. We run four experiments, two for each dummy strategy in Sect. 5. We repeat each experiment 200 times and plot the average results.

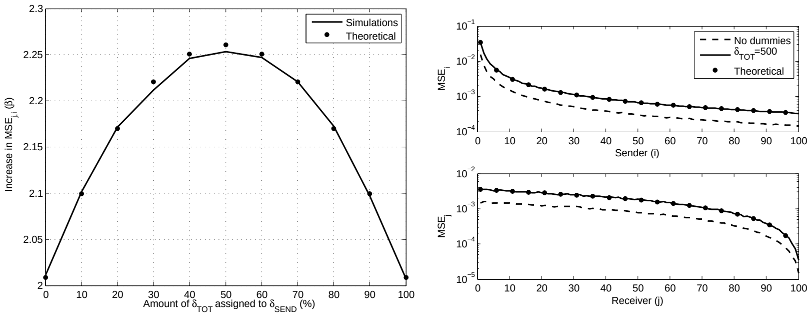
6.1 Increasing the protection of every sender-receiver relation by the largest factor β given a budget of dummies $\delta_{\texttt{TOT}}$

First, we study the influence of the distribution of dummies among senders and mix in the factor β that can be achieved with this strategy, when on average $\delta_{\texttt{TOT}} = 500$ dummies per round are available. Figure 3a shows the evolution of β for different distributions of dummy messages between senders ($\delta_{\texttt{SEND}}$) and mix ($\delta_{\texttt{MIX}}$). We see that the maximum increase is achieved when dummies are divided equally between the senders and the mix, as predicted in Sect. 5.1. We note that the maximum β in the figure is slightly higher than $\beta = 2.25$ that would be obtained using the approximation (10) used to design the dummy traffic strategy, meaning that the adversary estimation is worse than predicted by the theory.

For the particular case where $\delta_{\texttt{SEND}} = \delta_{\texttt{MIX}} = \delta_{\texttt{TOT}}/2$, we plot in Fig. 3b the average $\texttt{MSE}_{j,i}$ over i (top) and j (bottom) with and without dummies (note the vertical axis logarithmic scale). We see that indeed all $\texttt{MSE}_{j,i}$ increase by a constant factor, $\beta = 2.261$. The figure also shows that (6) accurately models the profiling error.

6.2 Maximizing the minimum protection to every sender-receiver relation given a budget of dummies $\delta_{\texttt{TOT}}$

First, we study the influence of the distribution of dummies among senders and mix on the maximum minimum $\texttt{MSE}_{j,i}$ that can be achieved with this strategy, when on average $\delta_{\texttt{TOT}} = 500$ dummies per round are available. Fig. 4a shows the evolution of the average minimum $\texttt{MSE}_{j,i}$ depending on the distribution of dummies between the senders and the mix. In the scenario considered in our experiment, the maximum minimum $\texttt{MSE}_{j,i}$ achievable is obtained when approximately 40% of the dummies are assigned to the senders and the remaining 60% to the mix. This is because, in this strategy, the rate of sender-based dummies depends quadratically on the real sending rate (c.f. (19)), while the number of



(a) (b)

Fig. 3: (a) Evolution of β with the fraction of dummies distributed among senders and mix. (b) Average $\mathtt{MSE}_{j,i}$ evolution over i (top) and j (bottom) when dummies are distributed uniformly among senders and mix. ($N = 100$, $M = 100$, $\rho = 10\,000$, $\alpha = 0.5$, $\delta_{\mathtt{TOT}} = 500$)

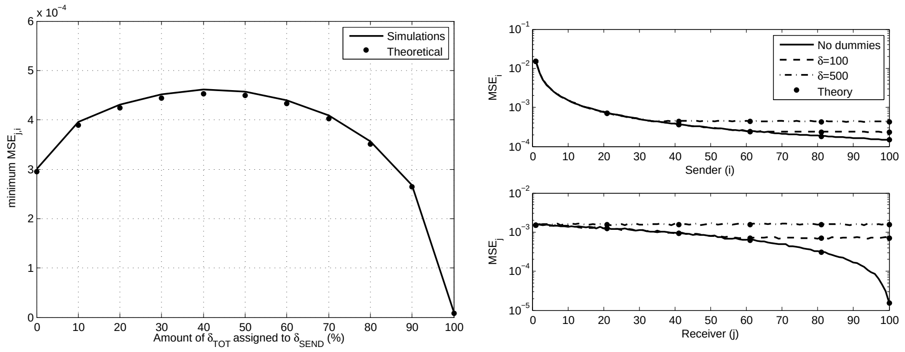
mix-based dummies depends linearly on the real receiving rate (c.f. (23)). Hence, mix-based dummies can be distributed more efficiently and it is preferable to assign the mix a larger budget than to the senders. We note that this result depends strongly on the users behavior. In fact, if the real traffic is distributed uniformly among receivers but few senders generate the majority of the traffic, allocating a large fraction of dummy traffic to the senders becomes the best option.

This is better shown in Fig. 4b. The top plot shows the $\mathtt{MSE}_{j,i}$ along i when there are no dummies, and when only sender-based dummies are available ($\delta_{\mathtt{SEND}} = \delta_{\mathtt{TOT}}$; $\delta_{\mathtt{MIX}} = 0$). As expected, more dummies increase the minimum $\mathtt{MSE}_{j,i}$, but, since the average number of sender-based dummies depends quadratically on the real sending rate, few senders with high rates exhaust the budget, which constrains the maximum minimum error achievable in the system. On the other hand, allocating all the dummies to the mix (Fig. 4b, bottom) allows to spread the distribution of dummies among more relationships, which in turn provides better overall protection than the previous case.

7 Discussion

In this section we discuss how to adapt the derivation of the least squares estimator in Sect. 3 to scenarios where pool and users' behavior are outside of the model considered throughout the document.

Non-stationary sending profiles. In practice users' behavior is expected to change over time. Our estimator can be adapted to account for dynamic profiles by implementing the Recursive Least Squares algorithm [16]. This algorithm includes a *forgetting factor*, which determines how fast the algorithm "forgets" past observations. Tuning this parameter, one can choose between getting a high-



(a) (b)

Fig. 4: (a) Evolution of the minimum $\mathtt{MSE}_{j,i}$ with the fraction of dummies distributed among senders and mix. (b) Average $\mathtt{MSE}_{j,i}$ evolution over i when only sender-based dummies are available (top), and j when only mix-based dummies are available (bottom). ($N = 100$, $M = 100$, $\rho = 10\,000$, $\alpha = 0.5$, $\delta_{\mathtt{TOT}} = 100, 500$)

variance estimator of the recent users' sending profile or obtaining a more stable long-term sending profile.

Non-independent users with memory. Although our model considers disjoint sets of senders and receivers, it can easily accommodate the case where users both send and receive messages. In this scenario, users' sending behavior may be dependent on messages sent or received in the past (e.g., email replies). Given a model of these interactions between users one can compute the expected value of the output observations given the inputs, and then proceed with the derivation of the estimator as in Sect. 3.

Non-stationary dummy strategies. If the probability of sending a real message (P_{λ_i}) changes over time, a per-round probability $P^r_{\lambda_i}$ could be defined. This dynamic probability can be used in the derivations in the Appendix (c.f. (30)) to account for the effect of this variation on the attacker's estimation of the hidden variables $X^r_{s,i}$. When the average mix-based dummies ($\delta_{\mathtt{MIX}}$) or the mix profile ($\mathbf{q}_{\mathtt{MIX}}$) vary over time, an aware attacker can include this behavior in (26), modifying the expected value of the outputs.

Complex batching strategies. Our anonymous channel model does not cover pool mixes whose batching strategy depends on the number of messages in the pool, such as that used by Mixmaster [12]. However, extending our model to this scenario is straightforward: the adversary can estimate the average number of messages in the pool by discarding a percentage of the incoming messages that are expected to be dummy, and therefore she can get an estimate of the average number of messages from each user that leave in each round, $X^r_{s,i}$. The estimator would still be formulated as (4).

8 Conclusions

In this paper, we have proposed a methodology to analyze mix-based anonymous communication systems with dummy traffic. Following a least squares approach, we derive an estimator of the probability that a user sends messages to a receiver. This estimator allows us to characterize the error of the adversary when recovering user profiles, or individual probabilities, with respect to the system parameters. Furthermore, it can be used to design dummy strategies that satisfy a wide range of privacy criteria.

As an example, we have studied the performance of the least squares estimator on a timed binomial pool mix, which enables us to derive qualitative conclusions about the effects of dummy traffic on the adversary's error. We have used this estimator to design dummy strategies that, given a budget of dummies, achieve two privacy targets: increase the protection of each sender and receiver relationship equally, and maximize the minimum protection provided to any relationship between users. The empirical evaluation of these strategies validates our theoretical results and confirms the qualitative intuitions drawn in the performance analysis.

Our methodology improves our understanding on the effect of dummy traffic on privacy in anonymous communication systems. It can be seen as a step forward towards the development of a systematic method do design dummy traffic, especially important to evaluate and improve privacy protection in deployed mix-based systems such as [11,12].

Appendix A: Derivation of the expected value of the output messages given the inputs.

We aim here at deriving an expression for the expected value of the random vector of the output observations $\mathbf{Y}(\mathbf{p})$ given the input observations $\mathbf{U}$, i.e., $\mathrm{E}\{\mathbf{Y}(\mathbf{p})|\mathbf{U}\}$. For simplificy, we assume that by the time the adversary starts observing the system the pool is empty. In practice, the initial messages in the pool would appear as noise in the initial output observations and its effect can be disregarded when the number of observations in large, as explained in [13]. For notational simplicity, we also omit writing the conditioning on $\mathbf{U}$ explicitly.

In order to relate in a statistical way the input and output flows of the mix, we follow the abstract model for the timed pool mix in Fig. 1. The different variables in this model can be *related backwards* in the following way:

- The number of output messages for receiver j in round r is $Y_j^r \doteq Y_{\lambda,j}^r + Y_{\delta,j}^r$. We can model the components refering to the real and dummy messages as:
  - Given the messages exiting the pool block $x_{s,i}^r$ for every sender i, the number of real messages leaving the mix $Y_{\lambda,j}^r$ for each receiver j is the sum of N multinomials:

$$\left\{ Y_{\lambda,1}^r, \cdots, Y_{\lambda,M}^r \middle| x_{s,1}^r, \cdots, x_{s,N}^r \right\} \sim \sum_{i=1}^{N} \mathrm{Multi}\left(x_{s,i}^r, \mathbf{q}_i\right) \tag{25}$$

where $\mathbf{q}_i \doteq [p_{1,i}, \cdots, p_{M,i}]^T$.

- Likewise, given the number of mix-based dummies generated in round r, $x^r_{\texttt{MIX}}$, $Y^r_{\delta,j}$ for $j = 1, \cdots, M$ can be modeled as:

$$\left\{ Y^r_{\delta,1}, \cdots, Y^r_{\delta,M} \middle| \, x^r_{\texttt{MIX}} \right\} \sim \text{Multi}\left(x^r_{\texttt{MIX}}, \mathbf{q}_{\texttt{MIX}}\right) \tag{26}$$

where $\mathbf{q}_{\texttt{MIX}} \doteq [p_{1,\texttt{MIX}}, \cdots, p_{M,\texttt{MIX}}]^T$. Later, we use the following result: $\text{E}\left\{Y^r_{\delta,j}\right\} = \text{E}\left\{X^r_{\texttt{MIX}}\right\} \cdot p_{j,\texttt{MIX}} = \delta_{\texttt{MIX}} \cdot p_{j,\texttt{MIX}}$.

- The messages leaving the pool from user i in round r, $X^r_{s,i}$, may come from any of the real messages sent by that user in the current and previous rounds. We can write $X^r_{s,i} = \sum_{k=1}^r X^{r,k}_{s,i}$, where $X^{r,k}_{s,i}$ is the random variable modeling the number of messages from user i that were sent in round k and leave the mix in round r ($r \geq k$). These random variables can be modeled, given the number of real messages sent by i in round r, $x^r_{\lambda,i}$, as:

$$\left\{ X^{k,k}_{s,i}, X^{k+1,k}_{s,i}, \cdots, X^{k+l,k}_{s,i}, \cdots \middle| \, x^k_{\lambda,i} \right\} \sim \text{Multi}\left(x^k_{\lambda,i}, \{F_{k,k}, F_{k+1,k}, \cdots, F_{k+l,k}, \cdots\cdots\}\right) \tag{27}$$

- Finally, given the total number of messages from user i that were sent in round r, x^r_i, we can model the number of real messages sent in that round $X^r_{\lambda,i}$ as

$$\left\{ X^r_{\lambda,i} \middle| \, x^r_i \right\} \sim \text{Bin}\left(x^r_i, P_{\lambda_i}\right) \tag{28}$$

We now compute $\text{E}\{\mathbf{Y}(\mathbf{p})\}$. First of all, from (25) and (26), we get

$$\text{E}\{\mathbf{Y}_j(\mathbf{p}_j)|\mathbf{U}_s\} = \mathbf{U}_s \cdot \mathbf{p}_j + \delta_{\texttt{MIX}}\mathbf{1}_\rho \cdot p_{j,\texttt{MIX}}$$

and, therefore, $\text{E}\{\mathbf{Y}(\mathbf{p})|\mathbf{U}_s\} = (\mathbf{I}_M \otimes \mathbf{U}_s) \cdot \mathbf{p} + (\mathbf{I}_M \otimes \delta_{\texttt{MIX}}\mathbf{1}_\rho) \cdot \mathbf{q}_{\texttt{MIX}}$. Using this last equality together with the law of total expectation, we can write

$$\text{E}\{\mathbf{Y}(\mathbf{p})\} = \text{E}\{\text{E}\{\mathbf{Y}(\mathbf{p})|\mathbf{U}_s\}\} = (\mathbf{I}_M \otimes \text{E}\{\mathbf{U}_s\}) \cdot \mathbf{p} + (\mathbf{I}_M \otimes \delta_{\texttt{MIX}}\mathbf{1}_\rho) \cdot \mathbf{q}_{\texttt{MIX}} \tag{29}$$

For notational simplicity, let $\hat{\mathbf{y}}_\delta \doteq \text{E}\{\mathbf{Y}_\delta\} = (\mathbf{I}_M \otimes \delta_{\texttt{MIX}}\mathbf{1}_\rho) \cdot \mathbf{q}_{\texttt{MIX}}$ be the attacker's estimation of the number of mix-based dummies sent each round. Likewise, let $\hat{\mathbf{U}}_s \doteq \text{E}\{\mathbf{U}_s\}$ be the estimation the attacker makes of the non-observable random matrix $\mathbf{U}_s$ and $\hat{\mathbf{H}}_s \doteq \mathbf{I}_M \otimes \text{E}\{\mathbf{U}_s\}$. In order to compute an element of $\hat{\mathbf{U}}_s$, i.e., $\hat{x}^r_{s,i}$, we use the law of total expectation repeatedly applying the relations above

$$\begin{aligned}\hat{x}^r_{s,i} \doteq \text{E}\left\{ X^r_{s,i} \middle| \, \mathbf{U} \right\} &= \sum_{k=1}^r \text{E}\left\{ X^{r,k}_{s,i} \middle| \, X^i_k = x^i_k \right\} = \sum_{k=1}^r \text{E}\left\{ \text{E}\left\{ X^{r,k}_{s,i} \middle| \, X^k_{\lambda,i} \right\} \middle| \, X^i_k = x^k_i \right\} \\ &= \sum_{k=1}^r \text{E}\left\{ X^k_{\lambda,i} \middle| \, X^i_k = x^k_i \right\} \cdot F_{r,k} = \sum_{k=1}^r x^k_i P_{\lambda_i} F_{r,k}\end{aligned} \tag{30}$$

For compactness, we define the $\rho \times \rho$ matrix $\mathbf{B}$, which contains in its (r,k)-th position the value $F_{r,k}$ if $r \geq 0$ and 0 otherwise; and the diagonal matrix $\mathbf{P}_\lambda \doteq \text{Diag}\{P_{\lambda_1}, P_{\lambda_2}, \cdots, P_{\lambda_N}\}$. Then, we can write

$$\hat{\mathbf{U}}_s = \mathbf{B} \cdot \mathbf{U} \cdot \mathbf{P}_\lambda \tag{31}$$

Plugging (31) into (29), we get $\mathrm{E}\left\{\mathbf{Y}(\mathbf{p})\right\} = \left(\mathbf{I}_M \otimes \hat{\mathbf{U}}_s\right) \cdot \mathbf{p} + \hat{\mathbf{y}}_\delta$; with $\hat{\mathbf{y}}_\delta = (\mathbf{I}_M \otimes \delta_{\mathtt{MIX}} \mathbf{1}_\rho) \cdot \mathbf{q}_{\mathtt{MIX}}$ and $\hat{\mathbf{U}}_s$ in (31), which concludes the proof.

References

1. Danezis, G., Diaz, C., Syverson, P.: Systems for anonymous communication. In Rosenberg, B., ed.: Handbook of Financial Cryptography and Security. Cryptography and Network Security Series. Chapman & Hall/CRC (2009) 341–389
2. Chaum, D.: Untraceable electronic mail, return addresses, and digital pseudonyms. Communications of the ACM **24**(2) (1981) 84–90
3. Agrawal, D., Kesdogan, D.: Measuring anonymity: The disclosure attack. IEEE Security and Privacy **1**(6) (2003) 27–34
4. Danezis, G., Troncoso, C.: Vida: How to use Bayesian inference to de-anonymize persistent communications. In Goldberg, I., Atallah, M.J., eds.: Privacy Enhancing Technologies Symposium. Volume 5672 of LNCS., Springer (2009) 56–72
5. Mathewson, N., Dingledine, R.: Practical traffic analysis: Extending and resisting statistical disclosure. In Martin, D., Serjantov, A., eds.: Privacy Enhancing Technologies Workshop. Volume 3424 of LNCS., Springer (2004) 17–34
6. Pérez-González, F., Troncoso, C.: Understanding statistical disclosure: A least squares approach. In Wright, M., Fischer-Hübner, S., eds.: Privacy Enhancing Technologies Symposium. Volume 7384 of LNCS., Springer-Verlag (2012) 38–57
7. Berthold, O., Langos, H.: Dummy traffic against long term intersection attacks. In Dingledine, R., Syverson, P.F., eds.: Privacy Enhancing Technologies Workshop. Volume 2482 of LNCS., Springer (2002) 110–128
8. Diaz, C., Preneel, B.: Taxonomy of mixes and dummy traffic. In: Working Conference on Privacy and Anonymity in Networked and Distributed Systems, Kluwer Academic Publishers (2004) 215–230
9. Diaz, C., Preneel, B.: Reasoning about the anonymity provided by pool mixes that generate dummy traffic. In Fridrich, J.J., ed.: Workshop on Information Hiding. Volume 3200 of LNCS., Springer-Verlag (2004) 309–325
10. Mallesh, N., Wright, M.: Countering statistical disclosure with receiver-bound cover traffic. In Biskup, J., Lopez, J., eds.: European Symposium On Research In Computer Security. Volume 4734 of LNCS., Springer (2007) 547–562
11. Danezis, G., Dingledine, R., Mathewson, N.: Mixminion: Design of a type iii anonymous remailer protocol. In: IEEE Symposium on Security and Privacy, IEEE Computer Society (2003) 2–15
12. Möller, U., Cottrell, L., Palfrader, P., Sassaman, L.: Mixmaster Protocol — Version 2. IETF Internet Draft (July 2003)
13. Anonymized for submission: A least squares approach to the traffic analysis of high-latency anonymous communication systems. https://www.dropbox.com/s/96pa2c4waxw1ca4/techreport.pdf
14. Díaz, C., Serjantov, A.: Generalising mixes. In Dingledine, R., ed.: Privacy Enhancing Technologies Workshop. Volume 2760 of LNCS., Springer (2003) 18–31
15. Anonymized for submission: Derivation of the mean squared error of the least squares disclosure attack in binomial timed pool mixes with dummy traffic. https://www.dropbox.com/s/w2awb78ind6k26b/MSEproof.pdf
16. Oya, S., Troncoso, C., Pérez-González, F.: Meet the family of statistical disclosure attacks. IEEE Global Conference on Signal and Information Processing (2013) 4p
17. Haykin, S.: Adaptive Filter Theory, 4/e. Prentice Hall (2002)