

Distributed Size Estimation of Dynamic Anonymous Networks

Håkan Terelius, Damiano Varagnolo and Karl Henrik Johansson

Abstract—We consider the problem of estimating the size of dynamic anonymous networks, motivated by network maintenance. The proposed algorithm is based on max-consensus information exchange protocols, and extends a previous algorithm for static anonymous networks. A regularization term is accounting for a-priori assumptions on the smoothness of the estimate, and we specifically consider quadratic regularization terms since they lead to closed-form solutions and intuitive design laws. We derive an explicit estimation scheme for a particular peer-to-peer service network, starting from its statistical model. To validate the accuracy of the algorithm, we perform numerical experiments and show how the algorithm can be implemented using finite precision arithmetics as well as small communication burdens.

Index Terms—anonymous networks, distributed estimation, dynamic networks, size estimation, sensor networks.

I. INTRODUCTION

The importance of distributed computation is reflected by the variety of applications where agents interact and cooperate to reach a common goal. Examples of these systems include environmental monitoring [1], management of the electrical grid [2] and the public transportation system [3].

In most cases the collaborating agents need to preserve the properties and working conditions of the network, and also perform opportune restorative actions. To this regard, size estimation of the network is a key function, and it is indispensable for topological change detection or automatic network reconfiguration. An indication of the importance of the network size estimation problem is the abundance of literature on the topic, briefly reviewed in the following.

A common approach to network size estimation is to use *random walks* [4], [5], [6], relying on a token being passed around the network to collect information each time it visits an agent. Another strategy is to use randomly generated numbers [7], and then exploit classical results on order statistics to infer the number of participants [8], [9], [10], [11], [12], [13]. These probabilistic techniques have been analyzed from a statistical point of view, [14], [15], and are extensions of the methodologies proposed in [16], [17], for estimating sums over networks. Other procedures use the *capture-recapture* concept [18], [19], where the idea is to randomly disseminate a number of seeds through the network, then check how many seeds are in a given subset, and

from this infer the size of the network. We also notice that some works, e.g., [20], [21], exploit probabilistic counting algorithms [22], [23] usually implemented in non-distributed contexts to analyze data fluxes over given channels. Some other techniques take advantage of their ad-hoc framework and are thus not implementable in general settings [24], [25], [26], [27].

We notice that the previous works mainly deal with static networks, or where the dynamics of the network are sufficiently slow and do not affect the estimation process. There are extensions of the previous procedures to dynamical cases: for example, [28] uses order statistics, [29] considers random walks, [30] exploits opportune derivations of probabilistic counting algorithms and [31], [32], [33] all deal with various dynamic scenarios.

Here we consider a scenario of *dynamic anonymous networks*, where the network size is not constant in time and the uniqueness of the node IDs is not guaranteed [34]. The anonymity is motivated for maintaining users' privacy (e.g., where users may not want to disclose information about their identity) and also beneficial when the estimation strategy must be simple and with limited resources requirements (e.g., where generating/storing/exchanging IDs may be infeasible due to computational/memory/communication constraints). Here we assume that the nodes have very limited knowledge of the network topology, and narrowly bounded computational, memory and bandwidth resources. Our aim is then to *obtain purely distributed strategies, where all nodes execute the same operations and where no leader nor overlay structure is present*.

With respect to the previously analyzed literature, we derive a distributed estimator that extends order statistics based techniques by means of regularization theory [35], [36], that to the best of our knowledge has not been proposed before. More precisely, we introduce a regularization term that allows the designer to opportunistically take into account not only the empirical evidence of the data, but also a-priori beliefs on the typical behavior of the quantity to be estimated. In this paper we then provide a full analysis of the quadratic regularization function, and offer detailed descriptions on the effects of the choice of its design parameters.

The paper is structured as follows: Sec. II introduces some preliminaries. In Sec. III we propose a generic regularization based network size estimator for dynamic networks, and consider especially quadratic regularization terms. In Sec. IV we derive an explicit estimator from a Bayesian network model, which effectiveness is tested in Sec. V by means of numerical experiments. We then draw some conclusions and future research directions in Sec. VI. For ease of readability,

All the authors are with the ACCESS Linnaeus Centre, School of Electrical Engineering, KTH Royal Institute of Technology, SE-100 44 Stockholm, Sweden. Emails: { hakante, damiano, kallej }@kth.se.

The research leading to these results has received funding from the European Union Seventh Framework Programme [FP7/2007-2013] under grant agreement n°257462 HYCON2 Network of Excellence and n°223866 FeedNetBack. The research was also supported by the Swedish Research Council and the Knut and Alice Wallenberg Foundation.

proofs are collected in Appendix.

II. PRELIMINARIES

We consider a network model of interconnected agents (nodes), where agents can join or leave at any time. The goal is to estimate the network size, i.e., count the number of agents as the network is evolving. Each agent has a limited memory and processing power, and can only communicate with its direct neighbors. Further, the agents are assumed to be anonymous so that no global unique identifiers can be used for estimation purposes.

A. Max-consensus

Max-consensus algorithms are procedures that allow a set of agents $i = 1, \dots, N$, each owning a local scalar value f_i , to distributively compute the maximum of the set $\{f_1, \dots, f_N\}$ with either gossip or broadcast communications. In the latter case, agents sequentially broadcast their local values, and whoever receives this information updates its local f_i using the rule “if $f_{\text{received}} > f_i$ then $f_i = f_{\text{received}}$ ”. Under mild assumptions on the communication process, the max-consensus protocols are proven to converge to the true maximum in a finite amount of time, see, e.g., [37]. It is immediate to extend scalar max-consensus algorithms to component-wise max-consensus procedures on vectorial quantities.

B. Notation

In the following, plain italics indicate scalars, while bold italics indicate vectors. $N(t)$ represents the number of agents in the network at time $t \in \mathbb{N}$, $\hat{N}(t)$ corresponds to an estimate of this quantity and $\bar{N}(t)$ represents a generic hypothesis on the value of $N(t)$. We extensively use the following vectorized versions of the previous quantities:

$$\mathbf{N}(t) := [N(t), \dots, N(t - \tau)]^T \quad (1)$$

$$\bar{\mathbf{N}}(t) := [\bar{N}(t), \dots, \bar{N}(t - \tau)]^T \quad (2)$$

$$\hat{\mathbf{N}}(t) := [\hat{N}(t), \dots, \hat{N}(t - \tau)]^T \quad (3)$$

$$\hat{\mathbf{N}}_\tau^\eta(t) := [\hat{N}(t - \tau - 1), \dots, \hat{N}(t - \eta)]^T \quad (4)$$

where again $\mathbf{N}(t)$ refers to actual values, $\bar{\mathbf{N}}(t)$ to a generic hypothesis on the value of $\mathbf{N}(t)$, $\hat{\mathbf{N}}(t)$ and $\hat{\mathbf{N}}_\tau^\eta(t)$ to estimates. Notice that $\tau, \eta \in \mathbb{N}$ are fixed design parameters. The uniform distribution over the interval $[0, 1]$ is denoted by $\mathcal{U}[0, 1]$.

III. NETWORK SIZE ESTIMATION ALGORITHM

We consider the simplified framework where the effects of clocks synchronization, packets loss and quantization issues can be neglected.

The network size estimation scheme in Alg. 1 can be summarized as follows: agents periodically generate some random values, share them with their neighbors, and eventually compute estimates of $N(t)$ through a penalized Maximum Likelihood (ML) approach.

Remark 1 The time index t does *not* denote physical quantities (e.g., seconds), but rather *epochs*, defined as the time

Algorithm 1 Dynamic Network Size Estimation Algorithm

- 1: **for** $t = 1, 2, \dots$ **do**
- 2: (Generation) Each agent $i = 1, \dots, N(t)$ generates M i.i.d. random values $y_{i,m}(t) \sim \mathcal{U}[0, 1]$, $m = 1, \dots, M$;
- 3: (Communication) Agents compute, through max consensus strategies, the M -dimensional max vector $\mathbf{f}(t) := [f_1(t), \dots, f_M(t)]^T$, where $f_m(t) = \max_i y_{i,m}(t)$;
- 4: (Computation) Each agent estimates the total number of agents in the network as

$$\hat{N}(t) = \arg \min_{\bar{N} \in \mathbb{R}^{\tau+1}} J(\bar{N}; \mathbf{f}(t), \dots, \mathbf{f}(t - \tau), \hat{\mathbf{N}}_\tau^\eta(t)) \quad (5)$$

necessary to complete each iteration in Alg. 1. We thus implicitly assume that agents always reach consensus on the locally generated quantities.

We define the penalized likelihood function J in (5) as follows:

$$J(\bar{N}; \mathbf{f}(t), \dots, \mathbf{f}(t - \tau), \hat{\mathbf{N}}_\tau^\eta(t)) := -\log p(\mathbf{f}(t), \dots, \mathbf{f}(t - \tau); \bar{N}) + \gamma \mathcal{R}(\bar{N}, \hat{\mathbf{N}}_\tau^\eta(t)) . \quad (6)$$

This allows us to estimate the network size $N(t)$ penalizing the hypotheses \bar{N} that deviate from expected behaviors by means of the regularization term $\mathcal{R} : \mathbb{R}^{\tau+1} \times \mathbb{R}^{\eta-\tau} \rightarrow \mathbb{R}_+$. Thus, given a hypothesis \bar{N} , (5) evaluates both its plausibility and its empirical evidence [38, Chap. 4]. γ in (6) is called the regularization parameter, and captures the trade-off between the empirical evidence of \bar{N} and its plausibility.

We notice that the hypothesis \bar{N} correspond to a time-window of fixed length $\tau+1$, while the regularization term \mathcal{R} also explicitly depends on the memory of the past estimates $\hat{\mathbf{N}}_\tau^\eta(t)$ up to time $t - \eta$ ($\eta \geq \tau$), defined in (4). The past estimates $\hat{\mathbf{N}}_\tau^\eta(t)$ are not changed by the estimator, and are used as extra parameters. A pictorial description of how these time windows shift in time is given in Fig. 1.

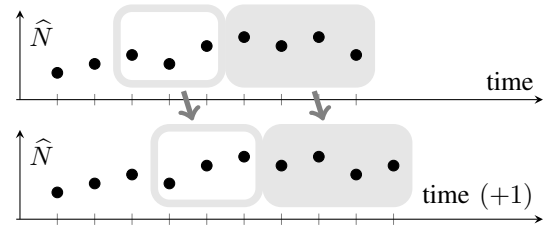


Fig. 1. Example of the time behavior of the estimation scheme (6). The white rectangle indicates the $\bar{N}_\tau^\eta(t)$, playing the role of parameters, while the gray rectangle indicates the time-window where the optimization problem (5) acts to obtain novel estimates. As time increases these windows are shifted.

Remark 2 If $\mathcal{R} = 0$, then Alg. 1 reduces to sequentially computing the estimates as

$$\begin{aligned}\widehat{N}(t) &:= \arg \min_{\overline{N} \in \mathbb{R}} \left(-\log p(\mathbf{f}(t); \overline{N}) \right) \\ &= -\left(\frac{1}{M} \sum_{i=1}^M \log f_i \right)^{-1}.\end{aligned}\quad (7)$$

In this case, the various $\widehat{N}(t)$'s are estimated independently.

(7) corresponds to the ML approach used in static anonymous network frameworks [39]. In this case, the statistical properties of \widehat{N} can be summarized as, when $M > 2$ then

$$\mathbb{E} \left[\frac{\widehat{N}(t)}{N(t)} ; M \right] = \frac{M}{M-1}, \quad (8)$$

$$\mathbb{E} \left[\left(\frac{N(t) - \widehat{N}(t)}{N(t)} \right)^2 ; M \right] = \frac{M^2 + M - 2}{(M-1)^2(M-2)}. \quad (9)$$

A. Parameter design constraints

Intuitively the estimation accuracy is non-decreasing in M, τ and η . However, M is bounded by transmission constraints (in the max-consensus step), τ is bounded by computational constraints (in the optimization step (5)), while η is bounded by memory constraints.

The following states that $\mathbf{f}(t), \mathbf{f}(t-1), \dots$ can be compressed to scalar values without loss of information:

Proposition 3 Let $s(\tau) := -\sum_{m=1}^M \log f_m(\tau)$. Then $s(\tau)$ is a complete and minimal sufficient statistic for $N(\tau)$.

By introducing $\mathbf{s}(t) := [s(t), \dots, s(t-\tau)]^T$, the penalized likelihood (6) can be rewritten as

$$J(\overline{N} ; \mathbf{s}(t), \widehat{N}_\tau^\eta(t)) = -\log p(\mathbf{s}(t) ; \overline{N}) + \gamma \mathcal{R}(\overline{N}, \widehat{N}_\tau^\eta(t))$$

with a memory saving of $M \cdot \tau$ scalars (notice that to compute the current $\mathbf{s}(t)$, agents need to run the max consensus on all the various $f_m(t)$, $m = 1, \dots, M$).

B. Quadratic regularization

Adding a regularization term \mathcal{R} in empirical risk minimization problems, as we did in (6), generally improves their conditioning properties [38, Chap. 4]. The presence of these terms can also be motivated by Bayesian perspectives, where the penalty \mathcal{R} reflects a-priori beliefs on typical behaviors.

Here we explicitly consider quadratic regularization terms

$$\mathcal{R}(\overline{N}, \widehat{N}_\tau^\eta) = \begin{bmatrix} \overline{N} - \boldsymbol{\mu}_1 \\ \widehat{N}_\tau^\eta - \boldsymbol{\mu}_2 \end{bmatrix}^T \underbrace{\begin{bmatrix} Q_{11} & Q_{12} \\ Q_{12}^T & Q_{22} \end{bmatrix}}_{Q^{-1}} \begin{bmatrix} \overline{N} - \boldsymbol{\mu}_1 \\ \widehat{N}_\tau^\eta - \boldsymbol{\mu}_2 \end{bmatrix} \quad (10)$$

where $\boldsymbol{\mu}$ is a nominal behavior of N , and Q^{-1} is a symmetric positive definite matrix. With this choice the following result holds:

Proposition 4 Given a quadratic regularization term (10), the optimal estimator $\widehat{N}(t)$ in (5) satisfies the quadratic equation system

$$\begin{aligned} & \text{diag}(\widehat{N}(t)) \cdot \\ & \cdot \left(\mathbf{s}(t) + 2\gamma Q_{11}(\widehat{N}(t) - \boldsymbol{\mu}_1) + 2\gamma Q_{12}(\widehat{N}_\tau^\eta(t) - \boldsymbol{\mu}_2) \right) \\ & - M\mathbf{1} = \mathbf{0}. \quad (11) \end{aligned}$$

Quadratic regularization terms, as in (10), especially capture the design strategies where \mathcal{R} penalizes just the differences between the various $N(t), \dots, N(t-\eta)$. In fact, by defining $\Omega_{ij} := (\mathbf{e}_i - \mathbf{e}_j)(\mathbf{e}_i - \mathbf{e}_j)^T$, where $\{\mathbf{e}_i\}$ is the standard basis of \mathbb{R}^n , and $\mathbf{x} = [x_1, \dots, x_n]^T$, and letting $Q^{-1} = \sum_{i,j} q_{ij} \Omega_{ij}$ with $q_{ij} > 0$ then $\|\mathbf{x} - \boldsymbol{\mu}\|_Q^2 = \sum_{i,j} q_{ij} (x_i - x_j)^2$. In this case, choices for η different from $\eta = \tau$ or $\eta = \tau + 1$ are meaningless, since larger values would just add a constant value to the regularization term.

IV. PROPERTIES UNDER A MARKOVIAN MODEL

We now derive the quadratic regularization term as an approximation of the probabilistic model for a simple but practical network of agents.

Consider an anonymous peer-to-peer file sharing network, where a certain file is only located at a subset of the peers, and the goal is to estimate how many peers that have the file. At any time, a user can either decide to download or to delete the file. The peers that have the file (and only those) generates new random values, the max-consensus procedure is then run in the background to estimate how many peers that have the file. We assume that:

- there exists a bound on the maximal number of peers¹, say N_{max} ;
- downloading and deleting files happen independently among the peers;
- the stochastic process that agent a downloads or deletes the file is a Markov process with (known) probabilities:

$$\begin{aligned} p &:= \mathbb{P}[x_a(t) = 1 \mid x_a(t-1) = 0] \\ q &:= \mathbb{P}[x_a(t) = 0 \mid x_a(t-1) = 1] \end{aligned} \quad (12)$$

where $x_a(t) = 1$ (0) corresponds to agent a (not) having the file at time t .

Given these assumptions, we derive the one-step estimator ($\tau = 0$) with two-steps of regularization memory ($\eta = 1$).

A. Derivation of the regularization term

Let us first consider the Bayesian interpretation of the quadratic regularization term as log-Gaussian priors on $[N(t), N(t-1)]$, and given the independence assumptions stated before, we need to compute the nominal behavior $\boldsymbol{\mu} := \mathbb{E}[N(t)]$ and variance

$$Q := \mathbb{E} \left[\begin{bmatrix} N(t) - \boldsymbol{\mu} \\ N(t-1) - \boldsymbol{\mu} \end{bmatrix} \begin{bmatrix} N(t) - \boldsymbol{\mu} \\ N(t-1) - \boldsymbol{\mu} \end{bmatrix}^T \right]. \quad (13)$$

¹As stated in the following Sec. IV-C, this assumption is not strictly required and can easily be removed.

Lemma 5 Let $\alpha := \frac{p}{q}$ be the ratio between the probabilities. Then,

$$\mu := \mathbb{E}[N(t)] = \frac{\alpha}{1+\alpha} N_{max} \quad (14)$$

$$\text{var}(N(t)) = \frac{\alpha}{(1+\alpha)^2} N_{max} \quad (15)$$

$$\text{cov}(N(t), N(t-1)) = (1-p-q) \frac{\alpha}{(1+\alpha)^2} N_{max} \quad (16)$$

Thus,

$$Q = N_{max} \frac{\alpha}{(1+\alpha)^2} \begin{bmatrix} 1 & 1-p-q \\ 1-p-q & 1 \end{bmatrix}. \quad (17)$$

B. Derivation of the estimator

Consider $\tau = 0$ and $\eta = 1$, so that $\widehat{N}(t) = \widehat{N}(t)$, $\widehat{N}_\tau^\eta(t) = \widehat{N}(t-1)$,

$$\begin{aligned} \mu_1 = \mu_2 = \mu &= \frac{\alpha}{1+\alpha} N_{max} \\ \mathcal{Q}_{11} = \mathcal{Q}_{22} &= \frac{1}{\mu q (2 - q(1+\alpha))} \\ \mathcal{Q}_{12} = \mathcal{Q}_{21} &= \frac{q(1+\alpha) - 1}{\mu q (2 - q(1+\alpha))}. \end{aligned} \quad (18)$$

In this case, the condition on the optimal estimator (11) simplifies into the quadratic form

$$a \widehat{N}^2(t) + (b \widehat{N}(t-1) + c) \widehat{N}(t) - M = 0 \quad (19)$$

where

$$\begin{aligned} a &:= 2\gamma \mathcal{Q}_{11} \\ b &:= 2\gamma \mathcal{Q}_{12} \\ c &:= s(t) - 2\gamma (\mathcal{Q}_{11} + \mathcal{Q}_{12}) \mu. \end{aligned}$$

The unique admissible solution for $\widehat{N}(t)$ is given by

$$\widehat{N}(t) = \sqrt{\left(\frac{b \widehat{N}(t-1) + c}{2a} \right)^2 + \frac{M}{a}} - \left(\frac{b \widehat{N}(t-1) + c}{2a} \right). \quad (20)$$

Remarkably, our penalized ML approach leads to a recursive estimator that is nonlinear but still easy to be implemented in devices with small computational capabilities. The fact that the obtained smoother is nonlinear accords to that even when we derived the regularization term using Gaussian assumptions on $[N(t), N(t-1)]$, the likelihood term in J is non-Gaussian. If the likelihood were Gaussian, the estimator would have been a linear smoother, leading to a Kalman filtering strategy.

Notice that the derivation of Q using Gaussian assumptions is formally incorrect, since it undertakes the probability that $N(t), N(t-1)$ can admit negative values. A formally correct probabilistic interpretation would require \mathcal{R} to be derived from the actual prior distribution, but this would lead to a non-quadratic \mathcal{R} , and thus to non-closed-form solutions of (5). Despite this error, the effects of this approximation vanish as N_{max} increase since $N(t) = \sum_{a=1}^{N_{max}} x_a(t)$ is approximatively Gaussian because of central limit effects.

C. The role of the regularization parameter γ

In (6), $-\log p(s(t); \overline{N})$ takes into account the experimental evidence, while \mathcal{R} reflects the a-priori information about the regularity of the solution. The regularization parameter γ then captures the trade-off between these two components and represents how much one trusts the regularity assumptions. Notice that the γ maximizing the predictive capabilities of the filter strongly depends on M , i.e., on the amount of available information.

If N_{max} is not known a-priori, or if its knowledge is vague, γ can also be tuned online, e.g., with simple cross-validation methods [40, Chap. 7.10]. In this case tuning γ , assuming the knowledge of the probabilities q and p , corresponds to estimate N_{max} given q, p and M .

V. NUMERICAL EXPERIMENTS

We start by noticing the beneficial effects of our regularization approach (20) in Fig. 2, where we compare the outcomes with point-wise estimation ($\gamma = 0$). The network is chosen as in Sec. IV, with $N_{max} = 1000$, $p = q = 0.01$, and the estimation parameters are $M = 200$ and $\gamma = 0.001$.

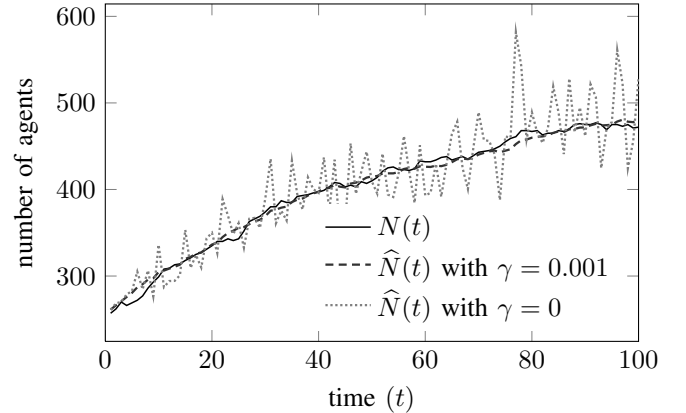


Fig. 2. Comparison of the results from (20) and point-wise estimation for the same set of $s(\tau)$ for a network as in Sec. IV. $N_{max} = 1000$, $p = q = 0.01$, $\gamma = 0.001$, $M = 200$.

In Fig. 3 we show the effects of the parameters p, q, γ, N_{max} and M by considering 4 different scenarios: $p = q = 0.1$ or 0.01 ; $N_{max} = 1000$ or 2000 . For each of these scenarios we independently generate 1000 trajectories $N_j(t)$, $t = 1, \dots, 100$, $j = 1, \dots, 1000$ from the network model in Sec. IV. For each trajectory $N_j(t)$ we compute the estimate (20) using different M 's in $[10, 200]$ and different γ 's in $[10^{-6}, 10^{-2}]$. Each of the 4 subplots then shows the dependency on M and γ of the following average Root-Mean-Square Error (RMSE)

$$\text{RMSE}(M, \gamma) := \sqrt{\frac{1}{10^5} \sum_{j=1}^{1000} \sum_{t=1}^{100} (N_j(t) - \widehat{N}_j(t; M, \gamma))^2}, \quad (21)$$

used as an estimation performance index.

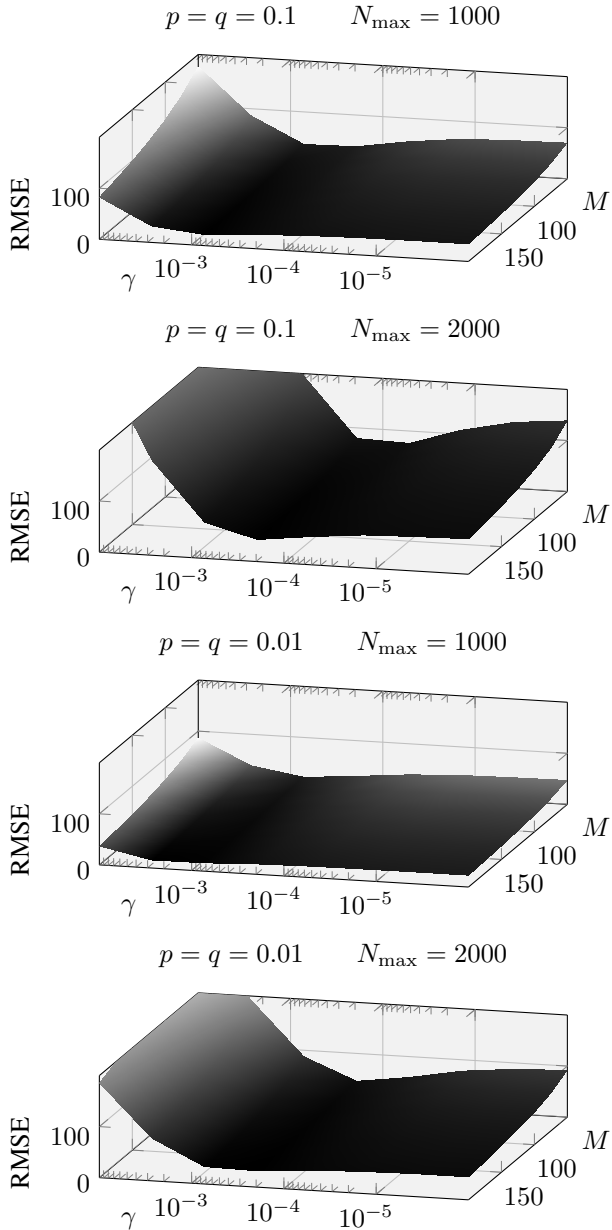


Fig. 3. Dependency of the average RMSE (21) on the parameters M and γ for various values of p, q and N_{\max} . $M \in [10, 200]$, $\gamma \in [10^{-6}, 10^{-2}]$ (logarithmic axis). $\text{RMSE}(M, \gamma)$ is plotted in the z -axis.

The behaviors of the 4 surfaces induce the following rules-of-thumb, supported also by intuition. Assuming that p, q, M and N_{\max} are fixed, there exists an optimal regularization parameter γ^* minimizing the RMSE (21). Then:

- if p, q and N_{\max} are fixed, then increasing M leads to a smaller optimal regularization parameter;
- if M and N_{\max} are fixed, then increasing p and q leads to a smaller optimal regularization parameter;
- if p, q and M are fixed, then increasing N_{\max} leads to a smaller optimal regularization parameter.

We finally analyze numerically how finite representations using b -bits of the random samples $y_{i,m}(t)$ in Alg. 1 can

affect the estimation performances, i.e.,²

$$y_{i,m}(t) \in \{0, \alpha, 2\alpha, \dots, 1\} \quad \text{with} \quad \alpha = \frac{1}{2^b - 1}. \quad (22)$$

Considering again the network model in Sec. IV, with $N_{\max} = 1000$, $p = q = 0.01$, $M = 200$ and $\gamma = 0.001$ as in Fig. 2, we independently generate 1000 trajectories $N_j(t)$, $t = 1, \dots, 100$, $j = 1, \dots, 1000$. In the communication step we use b -bits precision, but in the local computation of the estimate (20), we use 64-bits precision. The average RMSE performance index shows (Fig. 4) that for small networks it is sufficient to represent the samples $y_{i,m}(t)$ with 12 bits.

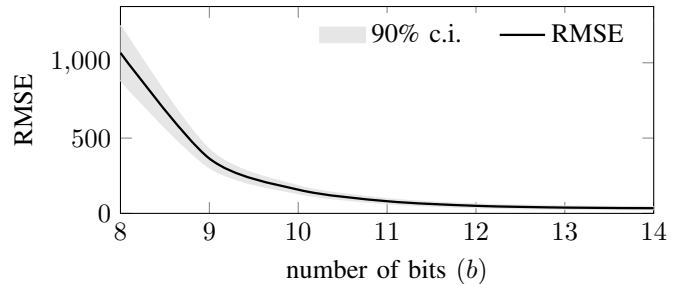


Fig. 4. Dependency of the RMSE (21) on the number of bits used to represent the samples $y_{i,m}(t)$, assuming (20) and (21) to be computed using 64-bits precisions. $N_{\max} = 1000$, $p = q = 0.01$, $M = 200$ and $\gamma = 0.001$.

Remark 6 Experiments in Figures 2 and 3 have been computed with the discretization scheme in Fig. 4 using 12 bits. Ignoring communication protocol overheads, with this choice $M = 200$ leads to data packets of 300 bytes.

VI. CONCLUSIONS

We proposed to estimate the size of anonymous dynamic networks using stochastic inference with a max-consensus protocol and a regularization-based estimator. Regularization approaches naturally penalize hypotheses conflicting with a-priori assumptions on the network's behavior, encoded in the regularization term. We explicitly considered and characterized the class of quadratic regularization terms, and also applied the strategy to a particular peer-to-peer network model, showing how the performance of the estimation strategy is influenced by the design parameters. Interestingly, the derived estimator corresponds to a nonlinear smoother.

Indeed, the algorithm has been derived exploiting some simplifying assumptions: convergence of the max-consensus protocols, reliable communications, infinite numerical precision and a time synchronized network. We nonetheless remark that if the algorithm does not converge to consensus, then this naturally leads to an estimation of the subset of the network with whom one had the possibility of exchanging information with, and this represents an interesting extension of the current estimator. We have also shown with numerical experiments that quantization effects seem to play a minor

²The problem of designing the optimal alphabet is not developed here because of space constraints.

role, and that for networks of hundreds of agents, numbers can be represented with just a few bits even though a more precise analysis should be devised. The synchronization assumption is instead crucial for the current algorithm, and corrupted communication schedules can lead to severe estimation biases intuitively corresponding to delays in the arrival of information. Future investigation directions are thus to evaluate the fragility of the scheme, and endow the algorithm with consensus-based clock-synchronization strategies.

REFERENCES

- [1] K. M. Lynch, I. B. Schwartz, P. Yang, and R. A. Freeman, "Decentralized environmental modeling by mobile sensor networks," *IEEE Transactions on Robotics*, vol. 24, no. 3, pp. 710 – 724, June 2008.
- [2] S. Bolognani and S. Zampieri, "A distributed control strategy for reactive power compensation in smart microgrids," *arXiv*, vol. arXiv:1106.5626v2 [math.OA], October 2011.
- [3] R. Herring, A. Hofleitner, S. Amin, T. Nasr, A. Khalek, P. Abbeel, and A. Bayen, "Using mobile phones to forecast arterial traffic through statistical learning," in *Transportation Research Board Annual Meeting*, Washington D.C., USA, January 2010.
- [4] B. Ribeiro and D. Towsley, "Estimating and sampling graphs with multidimensional random walks," in *Proceedings of the 10th annual conference on Internet measurement*, 2010.
- [5] C. Gkantsidis, M. Mihail, and A. Saberi, "Random walks in peer-to-peer networks: algorithms and evaluation," *Performance Evaluation*, vol. 63, no. 3, pp. 241 – 263, March 2006.
- [6] L. Massoulié, E. L. Merrer, A.-M. Kermarrec, and A. Ganesh, "Peer counting and sampling in overlay networks: random walk methods," in *25th annual ACM symposium on Principles of distributed computing*, 2006.
- [7] D. Kostoulas, D. Psaltoulis, I. Gupta, K. P. Birman, and A. J. Demers, "Active and passive techniques for group size estimation in large-scale and dynamic distributed systems," *The Journal of Systems and Software*, vol. 80, no. 10, pp. 1639 – 1658, October 2007.
- [8] C. Baquero, P. S. Almeida, R. Menezes, and P. Jesus, "Extrema propagation: Fast distributed estimation of sums and network sizes," *IEEE Transactions on Parallel and Distributed Systems*, vol. 23, no. 4, pp. 668 – 675, April 2012.
- [9] J. C. S. Cardoso, C. Baquero, and P. S. Almeida, "Probabilistic estimation of network size and diameter," in *Latin-American Symposium on Dependable Computing*, João Pessoa, Brasil, September 2009, pp. 33 – 40.
- [10] D. Varagnolo, G. Pilonetto, and L. Schenato, "Distributed size estimation in anonymous networks," *IEEE Transactions on Automatic Control*, vol. (submitted), 2011.
- [11] P. Chassaing and L. Gerin, "Efficient estimation of the cardinality of large data sets," in *4th Colloquium on Mathematics and Computer Science*, 2006, pp. 419 – 422.
- [12] F. Giroire, "Order statistics and estimating cardinalities of massive data sets," *Discrete Applied Mathematics*, vol. 157, pp. 406 – 427, 2009.
- [13] J. Lumbroso, "An optimal cardinality estimation algorithm based on order statistics and its full analysis," in *International Meeting on Probabilistic, Combinatorial, and Asymptotic Methods in the Analysis of Algorithms*, 2010.
- [14] J. Cichoń, J. Lemiesz, and M. Zawada, "On message complexity of extrema propagation techniques," Wroclaw University of Technology, Tech. Rep., 2012.
- [15] P. Clifford and I. A. Cosma, "A statistical analysis of probabilistic counting algorithms," *Scandinavian Journal of Statistics*, vol. 1, pp. 1 – 14, March 2011.
- [16] E. Cohen, "Size-estimation framework with applications to transitive closure and reachability," *Journal of Computer and System Sciences*, vol. 53, no. 3, pp. 441 – 453, December 1997.
- [17] D. Mosk-Aoyama and D. Shah, "Fast distributed algorithms for computing separable functions," *IEEE Transactions on Information Theory*, vol. 7, no. 7, pp. 2997 – 3007, July 2008.
- [18] S.-L. Peng, S.-S. Li, X.-K. Liao, Y.-X. Peng, and N. Xiao, "Estimation of a population size in large-scale wireless sensor networks," *Journal of Computer Science and Technology*, vol. 24, no. 5, pp. 987 – 997, September 2009.
- [19] S. Petrovic and P. Brown, "A new statistical approach to estimate global file populations in the eDonkey P2P file sharing system," in *21st International Teletraffic Congress*, September 2009.
- [20] J. Cichoń, J. Lemiesz, W. Szpankowski, and M. Zawada, "Two-phase cardinality estimation protocols for sensor networks with provable precision," in *IEEE Wireless Communications and Networking Conference*, Paris, France, April 2012.
- [21] J. Cichoń, J. Lemiesz, and M. Zawada, "On cardinality estimation protocols for wireless sensor networks," in *Ad-hoc, mobile, and wireless networks*, ser. Lecture Notes in Computer Science. Springer, 2011, vol. 6811/2011, pp. 322 – 331.
- [22] P. Flajolet and G. N. Martin, "Probabilistic counting algorithms for data base applications," *Journal of Computer and System Sciences*, vol. 31, no. 2, pp. 182 – 209, 1985.
- [23] P. Flajolet, E. Fusy, O. Gandouet, and F. Meunier, "Hyperloglog: the analysis of a near-optimal cardinality estimation algorithm," in *Analysis of Algorithms*, 2007.
- [24] D. Dolev, O. Mokryn, and Y. Shavit, "On multicast trees: Structure and size estimation," *IEEE/ACM Transactions on Networking*, vol. 14, no. 3, pp. 557 – 567, June 2006.
- [25] M. Howlader, M. R. Frater, and M. J. Ryan, "Estimating the number and distribution of the neighbors in an underwater communication network," in *SENSORCOMM*, August 2008.
- [26] R. Ali, S. S. Lor, and M. Rio, "Two algorithms for network size estimation for master/slave ad hoc networks," in *IEEE 3rd International Symposium on Advanced Networks and Telecommunication Systems*, December 2009.
- [27] A. Leshem and L. Tong, "Estimating sensor population via probabilistic sequential polling," *IEEE Signal Processing Letters*, vol. 12, no. 5, pp. 395 – 398, May 2005.
- [28] E. Fusy and F. Giroire, "Estimating the number of active flows in a data stream over a sliding window," in *Workshop on Analytic Algorithmics and Combinatorics*, 2007.
- [29] D. Psaltoulis, D. Kostoulas, I. Gupta, K. Birman, and A. Demers, "Practical algorithms for size estimation in large and dynamic groups," University of Illinois, Urbana-Champaign, Tech. Rep., February 2004.
- [30] Y. Chabchoub and G. Hébrail, "Sliding hyperloglog: Estimating cardinality in a data stream over a sliding window," in *IEEE International Conference on Data Mining Workshops*, Sydney, Australia, December 2010.
- [31] S. Alouf, E. Altman, C. Barakat, and P. Nain, "On the dynamic estimation of multicast group sizes," in *MTNS*, Leuven, Belgium, July 2004.
- [32] S. Alouf, E. Altman, and P. Nain, "Optimal on-line estimation of the size of a dynamic multicast group," in *Joint Conference of the IEEE Computer and Communications Societies*, vol. 2, November 2002, pp. 1109 – 1118.
- [33] T. M. Shafaat, A. Ghodsi, and S. Haridi, "A practical approach to network size estimation for structured overlays," *Self-organizing Systems*, vol. 5343, pp. 71 – 83, 2008.
- [34] M. Yamashita and T. Kameda, "Computing on an anonymous network," in *Proceedings of the seventh annual ACM Symposium on Principles of distributed computing*, 1988, pp. 117 – 130.
- [35] G. Wahba, *Spline models for observational data*. SIAM, 1990.
- [36] V. N. Vapnik, *Statistical Learning Theory*. New York: Springer-Verlag, 1998.
- [37] F. Iutzeler, P. Ciblat, and J. Jakubowicz, "Analysis of max-consensus algorithms in wireless channels," *IEEE Transactions on Signal Processing*, 2012 (submitted).
- [38] B. Schölkopf and A. J. Smola, *Learning with kernels: Support vector machines, regularization, optimization, and beyond*. The MIT Press, 2002.
- [39] D. Varagnolo, G. Pilonetto, and L. Schenato, "Distributed statistical estimation of the number of nodes in sensor networks," in *IEEE Conference on Decision and Control*, Atlanta, USA, December 2010.
- [40] T. Hastie, R. Tibshirani, and J. Friedman, *The Elements of Statistical Learning: Data Mining, Inference, and Prediction*, 2nd ed. New York: Springer, 2001.

VII. APPENDIX

Proof (Prop. 3) Due to the independence of the variables $y_{i,m}(\tau)$, it follows that $p(\mathbf{f}(t), \dots, \mathbf{f}(1); \bar{N}) = \prod_{\tau=1}^t p(\mathbf{f}(\tau); \bar{N}\tau)$. To prove the proposition it is then sufficient to show that $s(\tau)$ is a complete and minimal sufficient statistic for $N(\tau)$.

Since

$$p(\mathbf{f}(\tau); \bar{N}) = \bar{N}^M e^{-(M-1)s(\tau)},$$

$s(\tau)$ is a sufficient statistic for $N(\tau)$ because of the Fisher-Neyman factorization theorem. It is also clearly minimal because it is a scalar.

To show the completeness of $s(\tau)$, we must show that if $g(s(\tau))$ is a generic measurable function s.t. $\mathbb{E}[g(s(\tau)) | N] = 0$ independently of N , then it must be $g(\cdot) \equiv 0$ (a.e.). Consider now that $-\log f_i(\tau)$ is an exponential random variable with rate N . Thus $s(\tau)$ is the sum of i.i.d. exponential random variables, i.e., $s(\tau) \sim \text{Gamma}(M, \frac{1}{N})$. $\mathbb{E}[g(s(\tau)) | N] = 0$ can then be rewritten as

$$\Gamma(M)^{-1} N^M \int_0^{+\infty} g(s) s^{M-1} \exp(-sN) ds \equiv 0.$$

This is equivalent to the fact that the Laplace transform of $g(s)s^{M-1}$ has to be zero a.e., and this happens if and only if $g(s)$ is zero a.e. \diamond

Proof (Prop. 4) Since, for all t ,

$$p(f_1(t), \dots, f_M(t); \bar{N}(t)) = \prod_{m=1}^M \bar{N}(t) \cdot f_m(t)^{\bar{N}(t)-1}$$

and since $s(t), \dots, s(t-\tau)$ are independent, it follows that

$$\begin{aligned} -\log p(s(t), \dots, s(t-\tau); \bar{N}) &= \\ &= \sum_{i=t-\tau}^t ((\bar{N}(i)-1)s(i) - M \log \bar{N}(i)). \end{aligned}$$

We can thus rewrite the estimator (5) as

$$\begin{aligned} \arg \min_{\bar{N}} \sum_{i=t-\tau}^t ((\bar{N}(i)-1)s(i) - M \log \bar{N}(i)) &+ \\ &+ \gamma (\bar{N} - \boldsymbol{\mu}_1)^T \mathcal{Q}_{11} (\bar{N} - \boldsymbol{\mu}_1) \\ &+ 2\gamma (\bar{N} - \boldsymbol{\mu}_1)^T \mathcal{Q}_{12} (\widehat{N}_\tau^\eta - \boldsymbol{\mu}_2) \\ &+ \gamma (\widehat{N}_\tau^\eta - \boldsymbol{\mu}_2)^T \mathcal{Q}_{22} (\widehat{N}_\tau^\eta - \boldsymbol{\mu}_2). \end{aligned}$$

Setting the gradient w.r.t. \bar{N} equal to zero yields, for each $i = t-\tau, \dots, t$,

$$s(i) - \frac{M}{\bar{N}(i)} + 2\gamma \left(\mathcal{Q}_{11}^{(i)} (\bar{N} - \boldsymbol{\mu}_1) + \mathcal{Q}_{12}^{(i)} (\widehat{N}_\tau^\eta - \boldsymbol{\mu}_2) \right) = 0,$$

where $\mathcal{Q}_{11}^{(i)}$ is the i -th row of \mathcal{Q}_{11} (same for $\mathcal{Q}_{12}^{(i)}$). Multiplying by $\bar{N}(i)$ and vectorizing the previous equation leads to (11). \diamond

Proof (Lem. 5) Notice that $N(t) = \sum_{a=1}^{N_{max}} x_a(t)$, where the processes x_a are i.i.d. Let us first compute the expected value, variance and covariance of x_a for a single agent.

The Markov process in (12) is described by the transition matrix P given by

$$P = \begin{bmatrix} 1-p & p \\ q & 1-q \end{bmatrix}.$$

The equilibrium distribution, $\boldsymbol{\pi} = \boldsymbol{\pi}P$, for the Markov process is $\boldsymbol{\pi} = \frac{1}{1+\alpha} [1 \quad \alpha]$, thus the expected value is

$$\mathbb{E}[x_a(t)] = \frac{\alpha}{1+\alpha}.$$

Further, the variance is

$$\begin{aligned} \text{var}(x_a(t)) &= \mathbb{E}[x_a(t)^2] - \mathbb{E}[x_a(t)]^2 = \\ &= \frac{\alpha}{1+\alpha} - \left(\frac{\alpha}{1+\alpha} \right)^2 = \frac{\alpha}{(1+\alpha)^2}. \end{aligned}$$

Finally, for a single agent we have the covariance

$$\begin{aligned} \text{cov}(x_a(t), x_a(t-1)) &= \\ &= \mathbb{E}[x_a(t)x_a(t-1)] - \mathbb{E}[x_a(t)]\mathbb{E}[x_a(t-1)] = \\ &= \frac{\alpha}{1+\alpha}(1-q) - \left(\frac{\alpha}{1+\alpha} \right)^2 = (1-p-q) \frac{\alpha}{(1+\alpha)^2}. \end{aligned}$$

For the entire system $N(t) = \sum_{a=1}^{N_{max}} x_a(t)$ we utilize the fact that the different agents are i.i.d., and the linearity of the expected value, variance and covariance to simply multiply the results for a single agent by N_{max} . \diamond