Distributed Size Estimation of Dynamic Anonymous Networks

Håkan Terelius, Damiano Varagnolo, Karl Henrik Johansson

KTH - Royal Institute of Technology

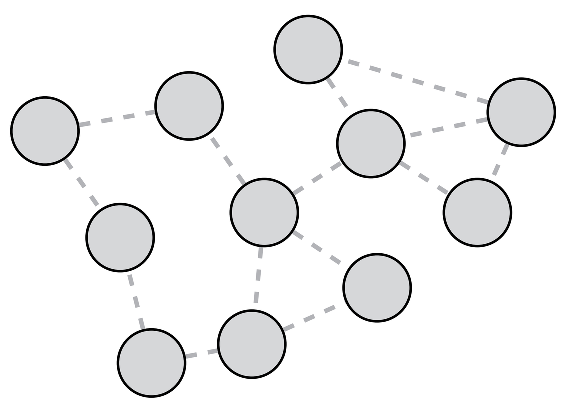
Abstract We consider the problem of estimating the size of dynamic anonymous networks. The proposed algorithm exploits max-consensus protocols and extends a previous strategy suited for static networks. A regularization term accounts a-priori assumptions on the smoothness of the estimate, and we specifically consider quadratic regularization terms since they lead to closed-form solutions. We explicitly derive an estimation scheme tailored for peer-to-peer service networks, starting from their statistical model. Numerical experiments validate the accuracy of the algorithm and show how the strategy can be implemented using finite precision arithmetics.

Problem Description

- dynamic network of $N(t)$ agents
- each agent wants to estimate $N(t)$

Constraints:

- can only use local information
- agents are anonymous (no global unique ID)



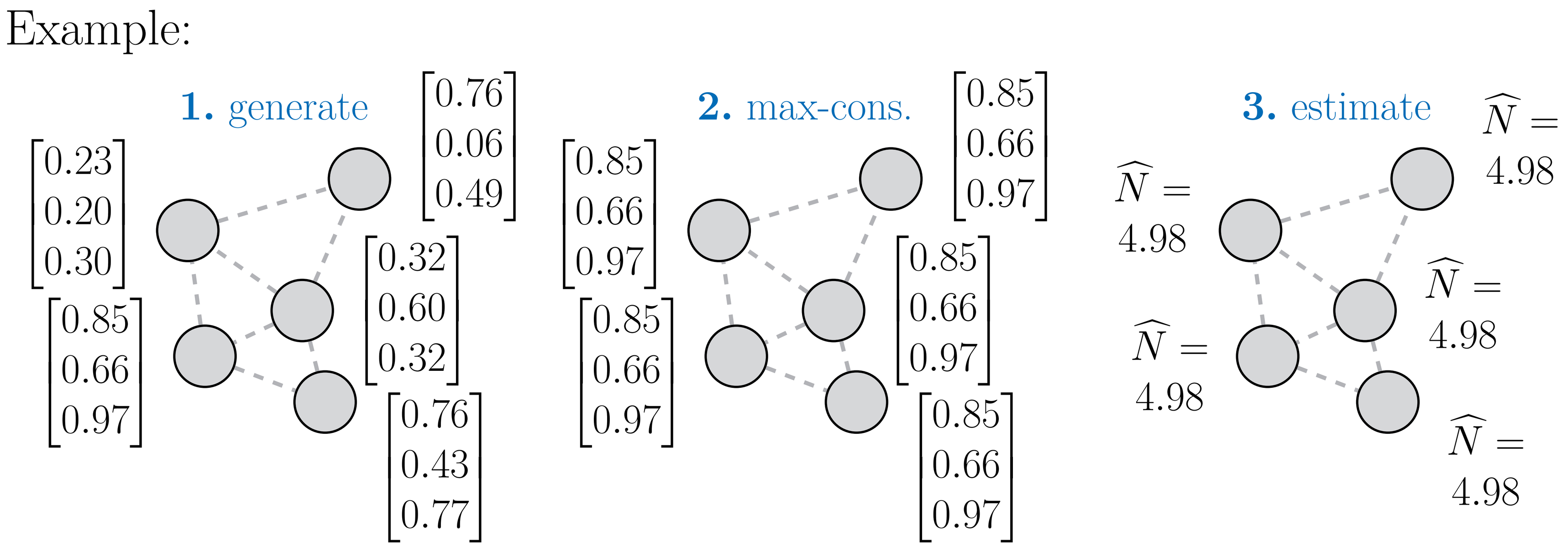
Static Estimation ($N(t) = N$)

Algorithm:

1. every agent i generates $y_{i,m} \sim \mathcal{U}[0,1]$, $m = 1, \dots, M$, i.i.d.
2. run max consensus: $f_m = \max_i y_{i,m}$, $\boldsymbol{f}(t) := [f_1, \dots, f_M]^T$,
3. estimate N through Maximum-Likelihood:

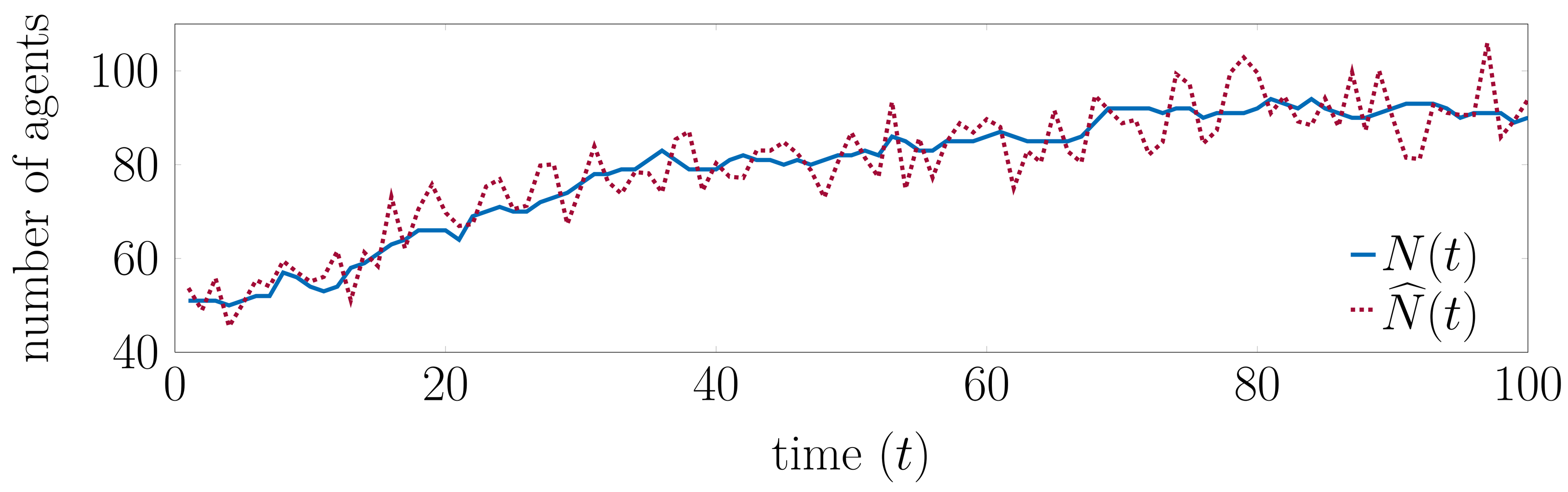
$$\widehat{N} = \arg\min_{\mathcal{N}} -\log\left(p\left(f_1, \dots, f_M\ ;\ \mathcal{N}\right)\right) = \arg\min_{\mathcal{N}} -\log\left(\prod_{m=1}^{M} \mathcal{N} \cdot f_m^{\mathcal{N}-1}\right)$$

Example:



Naïve Dynamic Estimation

Strategy = repeat the static estimator for every t:



Regularization Based Dynamic Estimation

Idea: penalize implausible hypothesis

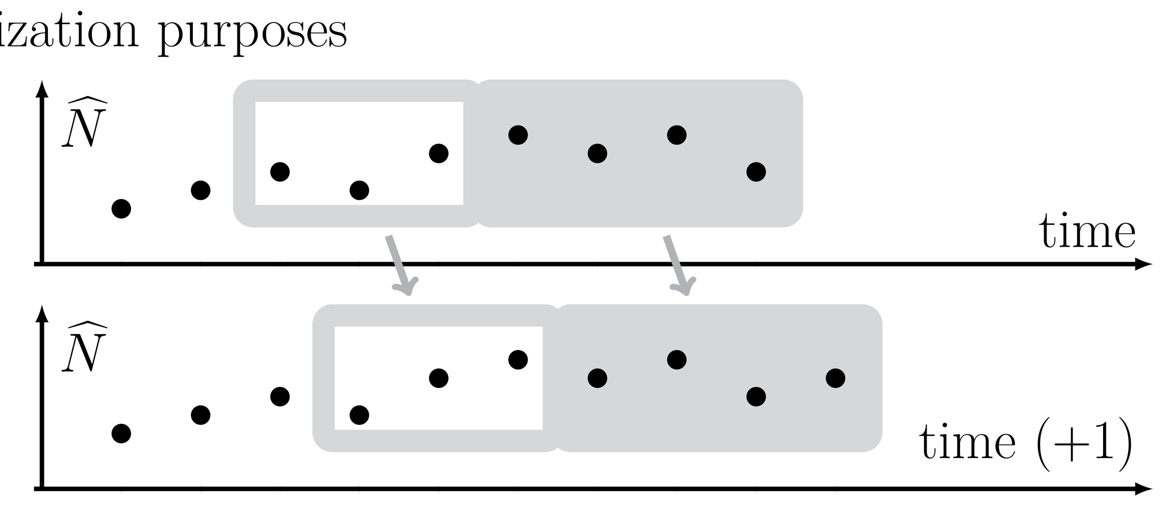
⇒ add a regularization term $\mathcal{R}$ to the naive approach

(use a *penalized log-likelihood function*):

$$\widehat{\boldsymbol{N}}(t) = \arg\min_{\boldsymbol{\mathcal{N}} \in \mathbb{R}^{\tau+1}} -\log p\left(\boldsymbol{f}(t), \dots, \boldsymbol{f}(t-\tau)\ ;\ \boldsymbol{\mathcal{N}}\right) + \gamma \mathcal{R}\left(\boldsymbol{\mathcal{N}}, \widehat{\boldsymbol{N}}_\tau^\eta(t)\right)$$

Note: two different windows!

- $\boldsymbol{\mathcal{N}}(t) := [\mathcal{N}(t), \dots, \mathcal{N}(t-\tau)]^T$ = guesses relative to interval $t : t-\tau$
- $\widehat{\boldsymbol{N}}_\tau^\eta(t) := [\widehat{N}(t-\tau-1), \dots, \widehat{N}(t-\eta)]^T$ = additional memory for regularization purposes



Quadratic regularization

(approximation: $N(t)$ = Gaussian Process)

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

Here the optimal estimate $\widehat{\boldsymbol{N}}(t) = [\widehat{N}(t), \dots, \widehat{N}(t-\tau)]$ solves the quadratic system ($s(\tau) := -\sum_{m=1}^{M} \log f_m(\tau)$, $\boldsymbol{s}(t) := [s(t), \dots, s(t-\tau)]^T$):

$$\operatorname{diag}\left(\widehat{\boldsymbol{N}}(t)\right)\left(\boldsymbol{s}(t) + 2\gamma\mathcal{Q}_{11}\left(\widehat{\boldsymbol{N}}(t) - \boldsymbol{\mu}_1\right) + 2\gamma\mathcal{Q}_{12}\left(\widehat{\boldsymbol{N}}_\tau^\eta(t) - \boldsymbol{\mu}_2\right)\right) - M\mathbb{1} = \boldsymbol{0}$$

Example: Peer-to-Peer Network

Framework: peer-to-peer network with $N_{\max}$ agents (either active or inactive)

Assumption: birth-death process with transition probabilities

$$p = \mathbb{P}\left[x(t) = 1 \mid x(t-1) = 0\right]$$
$$q = \mathbb{P}\left[x(t) = 0 \mid x(t-1) = 1\right]$$
$$\Downarrow$$
$$\boldsymbol{\mu}_1 = \boldsymbol{\mu}_2 = \mathbb{E}\left[N(t)\right] = \frac{\alpha}{1+\alpha} N_{max} \qquad \alpha := \frac{p}{q}$$
$$Q = \mathbb{E}\left[\begin{bmatrix} N(t) - \mu \\ N(t-1) - \mu \end{bmatrix}\begin{bmatrix} N(t) - \mu \\ N(t-1) - \mu \end{bmatrix}^T\right] = N_{max}\frac{\alpha}{(1+\alpha)^2}\begin{bmatrix} 1 & 1-p-q \\ 1-p-q & 1 \end{bmatrix}$$

Estimator for the 1-step memory case, $\widehat{\boldsymbol{N}}(t) = \widehat{N}(t)$, $\widehat{\boldsymbol{N}}_\tau^\eta(t) = \widehat{N}(t-1)$:

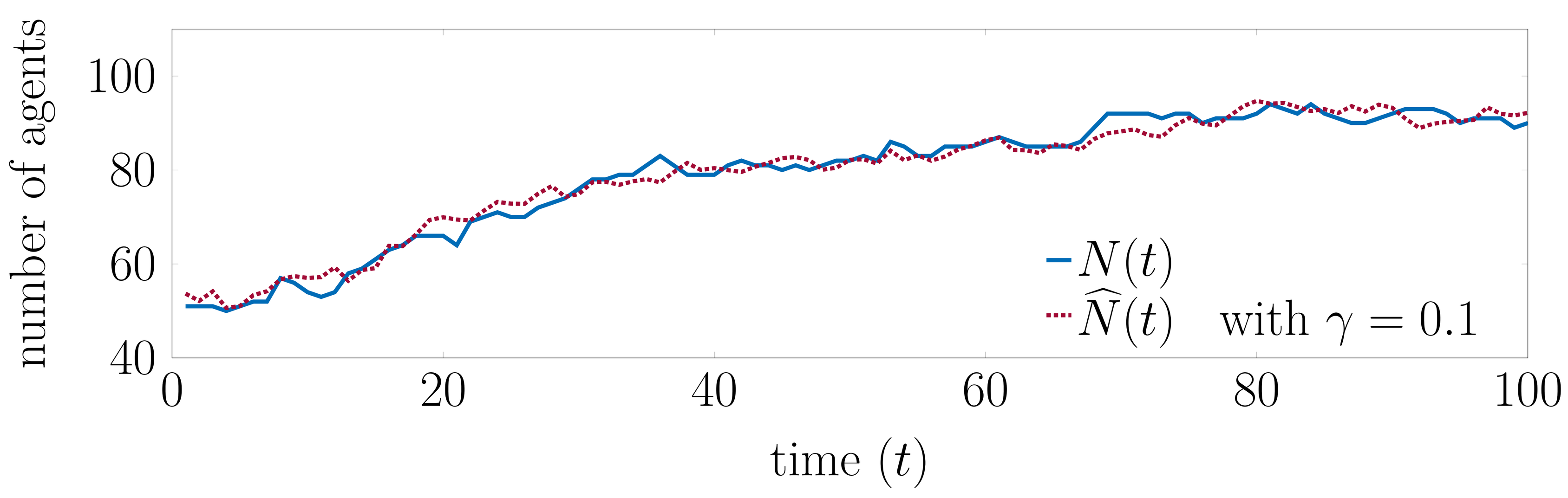
$$\widehat{N}(t) = \sqrt{\left(\frac{b\widehat{N}(t-1) + c}{2a}\right)^2 + \frac{M}{a}} - \left(\frac{b\widehat{N}(t-1) + c}{2a}\right)$$

with

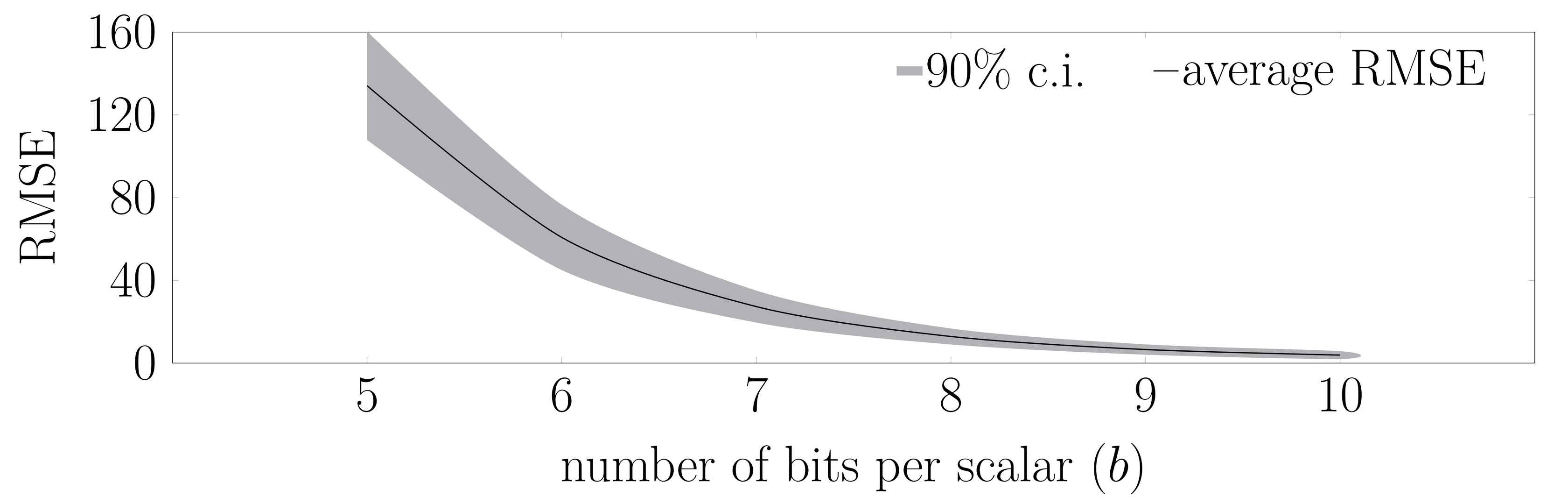
$$a := 2\gamma\mathcal{Q}_{11} \quad b := 2\gamma\mathcal{Q}_{12} \quad c := s(t) - 2\gamma\left(\mathcal{Q}_{11} + \mathcal{Q}_{12}\right)\mu$$

⇒ a nonlinear estimator, intrinsically different from a standard Kalman Filter

Numerical Experiments



Effects of quantization errors (1000 Monte Carlo runs):



References

- H. Terelius, D. Varagnolo, and K.H. Johansson. Distributed size estimation of dynamic anonymous networks. *CDC*, 2012 (submitted).
- D. Varagnolo, G. Pillonetto, and L. Schenato. Distributed statistical estimation of the number of nodes in sensor networks. *CDC*, 2010.
- D. Varagnolo, G. Pillonetto, and L. Schenato. Consensus based estimation of anonymous networks size using bernoulli trials. *ACC*, 2012.
- D. Varagnolo, G. Pillonetto, and L. Schenato. Distributed size estimation in anonymous networks. *TAC*, 2012 (submitted).