

Distributed statistical estimation of the number of nodes in Sensor Networks

Damiano Varagnolo, Gianluigi Pillonetto, Luca Schenato

Abstract—The distributed estimation of the number of active sensors in a network can be important for estimation and organization purposes. We propose a design methodology based on the following paradigm: some locally randomly generated values are exchanged among the various sensors and thus modified by known consensus-based strategies. Statistical analysis of the a-consensus values allows estimation of the number of participant sensors. The main features of this approach are: algorithms are completely distributed, since they do not require leader election steps; sensors are not requested to transmit authenticative information (for example identificative numbers or similar data), and thus the strategy can be implemented whenever privacy problems arise. After a rigorous formulation of the paradigm we analyze some practical examples, fully characterize them from a statistical point of view, and finally we provide some general theoretical results and asymptotic analyses.

Index Terms—sensor networks, distributed estimation, number of sensors, consensus algorithms

I. INTRODUCTION

During the last years distributed systems have been attracting more and more interests due to their intrinsic relevance in some real-world applications, as energy management and ambient monitoring [1] [2]. There has consequently been a wide interest on how to perform distributed computations in order to take the maximal possible advantage from the peculiarities of such systems [3] [4].

Recently many research groups focused on a certain kind of fully distributed computation techniques (i.e. that not require hierarchized network structures) known as consensus algorithms [5]. Their importance is given by the fact that they require low computational and communication resources, they are robust to link and nodes failures, and generally they do not require clock synchronization. Despite their simple structure, they have been proven to be able to compute a wide class of functions [6], estimate important physical parameters [7], or even to be used for synchronization purposes [8].

Yet sometimes it could be useful to know (or have an estimate of) the number of active agents / sensors composing the network, for example in parameter estimation [9] or in non-parametric regression [10]. In a distributed computation framework it is mandatory to estimate this number without using centralized paradigms, and at the best of our knowledge this has been partially addressed in literature.

This work was supported by the Italian CaRiPaRo Foundation under the Wise-Wai funded project and by European Community's Seventh Framework Programme under agreement n. FP7-ICT-223866 - FeedNetBack.

Damiano Varagnolo, Gianluigi Pillonetto and Luca Schenato are with the Department of Information Engineering, University of Padova, Via Gradenigo, 6/b, Padova, Italy. E-mails: { varagnolo | giapi | schenato }@dei.unipd.it

A common way of performing this task is to use a mobile access point moving through the network. In this context, authors of [11] analyze an algorithm based on the Good-Turing estimator of the missing mass [12] given vectors of observed sensors IDs, while in [13] other authors propose a probabilistic sequential polling protocol associated to a sensor identification mechanism, and show that the number of transmissions per sensor required to obtain an arbitrarily desired level of estimation accuracy is logarithmically bounded. In [14] authors consider underwater communications networks, and provide a probabilistic estimation procedure for counting the number of neighbours of the various nodes with a certain accuracy, but the algorithm works only locally and does not provide an estimation of the number of the active sensors in the whole network. An other interesting field that has been studied is the resource inventory application. Usually in this scenario the structure of the counting algorithm is hierarchical: a certain hand-portable sensor is moved through the environment, polling for certain kinds of objects and then returning the information to a centralized server [15]. There have been proposed also estimators based on the physical properties of the medium within information is transmitted (as in [16]).

Here we propose a fully distributed procedure with novel properties with respect to the previously cited papers. The first is the independence on the transmission medium once it is assumed that, once correctly transmitted, the information is also correctly received. The procedure can also be continuously running, allowing the development of topology changes algorithms (using well known change detection procedures [17]). Furthermore sensors are in general not required to authenticate, allowing to be insensible to privacy problematics. Finally, the procedure is asymptotically insensitive to packet loss effects, as soon as they do not affect the results of the consensus algorithms.

The paper is structured as follows: in Sec. II we formally state the analyzed problem, give two examples where analytical solutions are known, and discuss their structure and properties. In Sec. III and IV we then generalize (when possible) the results obtained for the given examples, while in Sec. V we obtain solutions to other examples using the same concepts developed before. Finally we draw some conclusions and propose future extensions in Sec. VI.

II. DISTRIBUTED NUMBER OF SENSORS ESTIMATION

A. Problem formulation

Formally, we model a network with a graph $\mathcal{G} = \{\mathcal{N}, \mathcal{E}\}$, where the set of nodes $\mathcal{N} = \{1, \dots, S\}$ is the set of the sensors composing the network, while the set of edges

$\mathcal{E} \subseteq \mathcal{N} \times \mathcal{N}$ is the set of the communication links between the sensors. We assume that the graph \mathcal{G} is undirected, i.e. $(i, j) \in \mathcal{E}$ implies that also $(j, i) \in \mathcal{E}$, and not time-varying.

Our objective is to devise a distributed strategy so that each sensor is able to estimate the total number of sensors in the network S only through local communications and with limited coordination among sensors. We propose a simple strategy based on three-steps: first sensors locally generate a set of random data, then they distributively compute a function based on the locally generated data, and finally they locally estimate S based on the value of the distributively computed function. More formally, these three steps are formalized as follows:

- 1) each sensor $i = 1, \dots, S$ locally generates a vector of $M \in \mathbb{N}_+$ i.i.d. random values $y_{i,m} \in \mathbb{R}$, $m = 1, \dots, M$, using a probability density $p(\cdot)$ that is the same among all sensors in the network; does *not* depend on the actual number of sensors S , does *not* depend on the number of generated values M ;
- 2) sensor distributively compute the vector $\mathbf{f} \in \mathbb{R}^M$ through the function $F : \mathbb{R}^S \rightarrow \mathbb{R}$ as follows:

$$\mathbf{f} := [f_1, \dots, f_M] \quad f_m := F(y_{1,m}, \dots, y_{S,m}) . \quad (1)$$

The function F must involve only computationally simple operations and local communications among the sensors. Some examples of such computable functions are: the arithmetic mean, the maximum, the minimum and the variance (of the set of data $y_{1,m}, \dots, y_{S,m}$);

- 3) each sensor locally computes an estimate \widehat{S}^{-1} of S^{-1} based on the vector $\mathbf{f} \in \mathbb{R}^M$, through a function $\Psi : \mathbb{R}^M \rightarrow \mathbb{R}_+$:

$$\widehat{S}^{-1} = \Psi(f_1, \dots, f_M) . \quad (2)$$

The reason for estimating S^{-1} rather than S in the third step of the algorithm is motivated by the fact that under general conditions, the performance results will be more natural, as will be shown below. Nonetheless, we will give performance results also for estimators of S rather than S^{-1} . This three-step strategy is illustrated in Fig. 1.

Hypothesizing a lack of knowledge of a prior on S , a natural measure of performance is given by the conditioned Mean Square Error (MSE), namely:

$$J(p, F, \Psi) := \mathbb{E} \left[\left(S^{-1} - \widehat{S}^{-1} \right)^2 \right] \quad (3)$$

where we explicitly indicated that this performance is a function of the generating p.d.f. $p(\cdot)$, the consensus function F and the estimator Ψ . Ideally we would like to minimize such error over all the possible choices of the triple (p, F, Ψ) . Obviously this is a formidable infinite dimensional problem, given the hypotheses previously posed in points 1), 2) and 3). In this work we focus on special classes of the triple (p, F, Ψ) and study the behavior of index (3) in this subset of possible choices, to get some insights on the optimization problem for the general case. We start by looking at two simple examples of such triples.

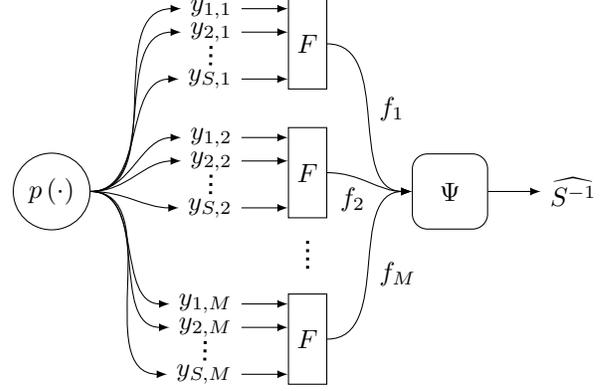


Fig. 1: Graphical representation of the estimation strategy for the inverse of the number of sensors S^{-1} .

B. Motivating example 1: Gaussian + average + Maximum Likelihood

Consider a zero-mean normal distribution for the generation of the data $y_{i,m}$, i.e. $p(y_{i,m}) = \mathcal{N}(0, 1)$; the average for the consensus function [18] [19], i.e.:

$$F(y_{1,m}, \dots, y_{S,m}) := \frac{1}{S} \sum_{i=1}^S y_{i,m} =: f_m ; \quad (4)$$

the Maximum Likelihood (ML) estimate for S^{-1} as the estimation function $\widehat{S}^{-1} = \Psi(f_1, \dots, f_M)$, i.e.:

$$\Psi(f_1, \dots, f_M) := \arg \max_{S^{-1}} p(f_1, \dots, f_M | S^{-1}) . \quad (5)$$

Clearly $f_m \sim \mathcal{N}(0, S^{-1}) \quad \forall m$ since all the $y_{i,m}$ are i.i.d. This imply that also all the f_m are i.i.d., therefore:

$$p(f_1, \dots, f_M | S^{-1}) = \frac{1}{\sqrt{2\pi} (S^{-1})^M} \exp \left(-\frac{\sum_{m=1}^M f_m^2}{2S^{-1}} \right) \quad (6)$$

and thus, after some simple computations:

$$\Psi := \arg \max_{S^{-1}} p(f_1, \dots, f_M | S^{-1}) = \frac{1}{M} \sum_{m=1}^M f_m^2 . \quad (7)$$

Considering $\widehat{S}^{-1} = \Psi(f_1, \dots, f_M)$, since $\sqrt{S} f_m \sim \mathcal{N}(0, 1)$, we have that $\sum_{m=1}^M (\sqrt{S} f_m)^2 \sim \chi^2(M)$, that can be finally traduced in:

$$\frac{M}{S^{-1}} \widehat{S}^{-1} \sim \chi^2(M) \quad (8)$$

This provides the analytic expression for the density $p(\widehat{S}^{-1} | S)$, from which we obtain the mean and variance as:

$$\mathbb{E}[\widehat{S}^{-1}] = S^{-1}, \quad \text{var}(\widehat{S}^{-1}) = S^{-2} \frac{2}{M} . \quad (9)$$

Hence, the estimator (7) is unbiased and its performance index (3) coincides with its variance, namely:

$$J(p, F, \Psi) = \mathbb{E} \left[\left(S^{-1} - \widehat{S}^{-1} \right)^2 \right] = S^{-2} \frac{2}{M} . \quad (10)$$

It is also important to remark that the previous expression implies that the relative estimation error $\frac{S^{-1} - \widehat{S}^{-1}}{S^{-1}}$ is independent of S .

For this example it is also possible to easily compute the performance of the ML estimator of S rather than S^{-1} . In fact, one has:

$$\widehat{S} := \arg \max_S p(f_1, \dots, f_M | S) = \frac{M}{\sum_{m=1}^M f_m^2} = \frac{1}{\widehat{S}^{-1}} \quad (11)$$

therefore:

$$\frac{1}{SM} \widehat{S} \sim \text{Inv-}\chi^2(M) \quad (12)$$

and thus:

$$p(\widehat{S} | S) = \Gamma\left(\frac{M}{2}\right)^{-1} \frac{1}{\widehat{S}} \left(\frac{MS}{2\widehat{S}}\right)^{\frac{M}{2}} \exp\left(-\frac{MS}{2\widehat{S}}\right) \quad (13)$$

where $\Gamma(\cdot)$ is the Gamma function. From this it follows that:

$$\mathbb{E}[S - \widehat{S}] = \frac{S}{M-2}, \quad (14)$$

$$\text{var}(\widehat{S}) = \frac{2S^2}{M} \frac{M^3}{(M-2)^2(M-4)} \quad (15)$$

and therefore the mean square error for \widehat{S} is:

$$\mathbb{E}\left[(S - \widehat{S})^2\right] = S^2 \frac{2M^3 + M(M-4)}{M(M-2)^2(M-4)}. \quad (16)$$

Notice now that asymptotically:

$$\lim_{M \rightarrow +\infty} \mathbb{E}\left[(S - \widehat{S})^2\right] = S^2 \frac{2}{M} \quad (17)$$

thus the relative estimation error of the estimator \widehat{S} and \widehat{S}^{-1} are the same.

C. Motivating example 2: Uniform + maximum + Maximum Likelihood

We start this example recalling two basic results relating order statistics [20]. Assume S to be the number of elements of the sample $y_{1,m}, \dots, y_{S,m}$, and $f_m^{(k)}$ to be its statistic of order k . Let every $y_{i,m}$ be i.i.d. and let $p(a)$ be its probability density evaluated in a , and $P(a)$ be its probability distribution evaluated in a . Then:

$$p_{f_m^{(k)}}(a) = \frac{S! P(a)^{(k-1)} (1 - P(a))^{(S-k)} p(a)}{(k-1)! (S-k)!} \quad (18)$$

while the joint density $p_{f_m^{(k)} f_m^{(j)}}(a_1 a_2)$ is given by:

$$p_{f_m^{(k)} f_m^{(j)}}(a_1 a_2) = \frac{S!}{(k-1)! (j-k-1)! (S-j)!} \cdot (P(a_2) - P(a_1))^{(j-k-1)} \cdot (1 - P(a_2))^{(S-j)} P(a_1)^{(k-1)} \cdot p(a_1) p(a_2). \quad (19)$$

Consider now a uniform distribution for the generation of the data $y_{i,m}$, i.e. $p(y_{i,m}) = \mathcal{U}[0, 1]$ and use the maximum to define the consensus function, i.e.:

$$F(y_{1,m}, \dots, y_{S,m}) := \max_i \{y_{i,m}\} =: f_m. \quad (20)$$

Again the ML estimator for S^{-1} is used to define Ψ (see Equ. (5)). The probability density of the S -th order statistic f_m is known and in general given by Equ. (18). In this case:

$$p(f_m | S) = S f_m^{S-1} \quad \forall m. \quad (21)$$

Therefore:

$$p(f_1, \dots, f_M | S) = \prod_{m=1}^M p(f_m | S) = S^M \prod_{m=1}^M f_m^{S-1} \quad \forall m. \quad (22)$$

Again, after some simple computations:

$$\Psi := \arg \max_{S^{-1}} p(f_1, \dots, f_M | S^{-1}) = -\frac{1}{M} \sum_{m=1}^M \log(f_m). \quad (23)$$

Now, defining $z := -\log(f_m)$, it is immediate to check that z is an exponential random variable with rate S , i.e.:

$$p(z | S) = \begin{cases} S \exp(-Sz) & \text{if } z \geq 0 \\ 0 & \text{otherwise.} \end{cases} \quad (24)$$

The sum of M i.i.d. exponential random variables with rate S is a Gamma random variable with shape M and scale $\frac{1}{S}$. Considering then that $\widehat{S}^{-1} = \Psi(f_1, \dots, f_M)$ is thus a scaled version of this sum of exponentials, it follows that:

$$\frac{M}{S^{-1}} \widehat{S}^{-1} \sim \text{Gamma}(M, 1) \quad (25)$$

from which it is immediate to compute mean and variance:

$$\mathbb{E}[\widehat{S}^{-1}] = S^{-1}, \quad \text{var}(\widehat{S}^{-1}) = S^{-2} \frac{1}{M}. \quad (26)$$

This implies that the estimator \widehat{S}^{-1} is unbiased and that its performance index (3) coincides with its variance, namely:

$$J(p, F, \Psi) = \mathbb{E}\left[\left(S^{-1} - \widehat{S}^{-1}\right)^2\right] = S^{-2} \frac{1}{M}. \quad (27)$$

By considerations similar to those of Sec. II-B, one obtains that \widehat{S} is asymptotically unbiased, with asymptotic variance equal to that of \widehat{S}^{-1} .

Comparing Equ. (10) and (27), it is remarkable to notice that, given a fixed M , the performance of the estimator based on the current strategy is exactly twice as large as that obtainable adopting the previous strategy.

D. Discussion on the motivating examples

A number of points regarding the previous two examples are now in order. First, notice that the function $\widehat{S}^{-1} = \Psi(f_1, \dots, f_M)$ can be decomposed into simpler blocks, as shown in Fig. 2. First, all the quantities f_m are passed through the same nonlinear function $\psi: \mathbb{R} \rightarrow \mathbb{R}$ which transform each f_m into an unbiased estimate of S^{-1} , i.e.:

$$\widehat{S}_m^{-1} = \psi(f_m) \quad m = 1, \dots, M. \quad (28)$$

Now, due to the independence of the various f_m , also the \widehat{S}_m^{-1} are uncorrelated. This implies that, in order to obtain the global estimate using all the available information,

the various \widehat{S}_m^{-1} have simply to be combined through an arithmetic mean:

$$\widehat{S}^{-1} = \frac{1}{M} \sum_{m=1}^M \widehat{S}_m^{-1}. \quad (29)$$

In fact, in Sec. II-B we had $\psi(\cdot) = (\cdot)^2$, while in Sec. II-C we had $\psi(\cdot) = -\log(\cdot)$.

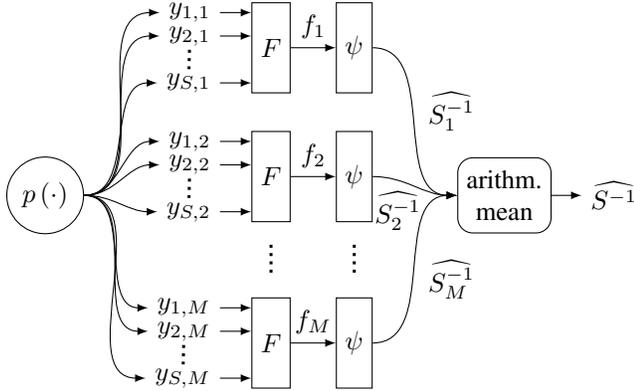


Fig. 2: Alternative graphical representation of the estimation strategies for the inverse of the number of sensors S^{-1} proposed in the motivating examples.

The second point is that being each \widehat{S}_m^{-1} an unbiased estimate, the variance of the combined estimate \widehat{S}^{-1} will decrease as $\frac{1}{M}$, and the quality of this variance will depend on the variance of the single estimates \widehat{S}_m^{-1} . By comparing the indexes (3) of the two previous examples we have also seen that the strategy of Sec. II-C gives better performance than the one of Sec. II-B (the variance is twice as small for a fixed M). This is particularly positive since the networked distributed computation of the maximum of a set of values is much faster than the computation of its average. In fact the maximum can be computed by the following simple strategy: each node i initializes its estimate of the maximum to its own value $z_{i,m}(0) = y_{i,m}$ (0 indicates a time index) and then broadcasts to its neighbors this value. When a node i receives a message from a neighbor j , i sets its own estimate to the maximum of local and the received values, i.e. $z_{i,m}(t+1) = \max\{z_{i,m}(t), z_{j,m}(t)\}$, and broadcasts again the value. It is easy to see that after a finite number of steps T , namely the largest minimum path distance between any couple of nodes, each node will have correctly computed the maximum, i.e. $z_{i,m}(t) = \max\{y_{1,m}, \dots, y_{S,m}\} \quad \forall t \geq T, \quad \forall m$. Differently, the average can still be computed in a distributed fashion through average consensus algorithms, but its rate of convergence to the actual average is exponential and depends on the size of the network. For example, in a circular network where each node has only two neighbors (left and right), the number of steps necessary to correctly compute the maximum is $T_{\max} = S/2$, while the number of steps required

to achieve a 1%-error in the estimation of the average is:

$$T_{\text{ave}} = \frac{\log(0.01)}{\log(1 - 2\pi^2/S^2)} \gg T_{\max}. \quad (30)$$

In the next sections we will show that the scheme shown in Fig. 2 is more general than the two toy examples we just presented. In particular we will concentrate on two special functions F , namely the average and the maximum, and we will study when the ML estimate is optimal in the sense of minimization of the performance index (3) for a large class of p.d.f. $p(\cdot)$. Also we would like to understand if there exist p.d.f. $p(\cdot)$'s that provide smaller estimation error, and if there are classes of equivalence in terms of achievable performance.

III. SPECIAL CASE: $F = \text{AVERAGE}$

Let:

$$F_{\text{ave}} := F(y_{1,m}, \dots, y_{S,m}) = \frac{1}{S} \sum_{i=1}^S y_{i,m} =: f_m. \quad (31)$$

Now, let us assume that data are generated by a generic gaussian r.v., i.e. $y_{i,m} \sim p(y_{i,m}) = \mathcal{N}(\mu, \sigma^2)$. It is easy to show, following the same steps of Sec. II-B, that the ML estimator for S^{-1} in this case is given by:

$$\widehat{S}^{-1} := \Psi_{\text{ML}}(f_1, \dots, f_M) = \frac{1}{M} \sum_{m=1}^M \frac{(f_m - \mu)^2}{\sigma^2} \quad (32)$$

$$\widehat{S}^{-1} \sim \frac{S^{-1}}{M} \chi^2(M). \quad (33)$$

From this observation it is possible to derive the following proposition:

Proposition 1. Let \mathcal{N} be the class of all gaussian random variables with positive variance, i.e. $p \in \mathcal{N}$ if $p = \mathcal{N}(\mu, \sigma^2)$ for some μ and $\sigma^2 > 0$. Then Ψ_{ML} is the Minimum-Variance Unbiased Estimator (MVUE) for S^{-1} within this class. Moreover we have:

$$\begin{aligned} \min_{p(\cdot) \in \mathcal{N}} J(p, F_{\text{ave}}, \Psi) &= J(\mathcal{N}(0, 1), F_{\text{ave}}, \Psi_{\text{ML}}) = \frac{2}{M}. \\ \Psi \text{ s.t. } \mathbb{E}[\Psi] &= S^{-1} \end{aligned} \quad (34)$$

Given the definition of Equ. (3) and a generic density $p(\cdot)$, it is not obvious whether the ML strategy is minimizing J . Even if we restrict Ψ to be the ML estimator (again for a given $p(\cdot)$), it is not easy to find an analytic expression for \widehat{S}^{-1} nor its distribution. One little step forward we can make is to notice that translation and scaling of a certain random variable do not affect the performance of the optimal estimator, as formally stated in Prop. 2. This allows us to restrict to distributions $p(\cdot)$ with zero mean and unit variance.

Proposition 2. Let $p_x(a)$ be a generic probability density with mean μ and variance $\sigma^2 > 0$, and consider the zero-mean unit-variance random variable:

$$y = \frac{x - \mu}{\sigma}, \quad x \sim p_x \quad (35)$$

with corresponding density $p_y(a) = \sigma p_x(\sigma a + \mu)$. Then:

$$\min_{\Psi} J(p_x, F_{\text{ave}}, \Psi) = \min_{\Psi} J(p_y, F_{\text{ave}}, \Psi) . \quad (36)$$

When we choose the average F_{ave} as the network function, it is not evident how to optimally choose the density $p(\cdot)$ and the estimator function Ψ to minimize the index J . In case of large networks, with large S , we can still exploit the central limit theorem: if $y_{i,m}$ s.t. $y_{i,m} \sim p(\cdot)$ and $\mathbb{E}[y_{i,m}] = 0$, $\text{var}(y_{i,m}) = 1$, then $f_m = F_{\text{ave}}(y_{1,m}, \dots, y_{S,m})$ has in general the following probability distribution:

$$p_{f_m}(a) = \underbrace{(p * \dots * p)}_{S \text{ times}}(a) \quad (37)$$

where the symbol $*$ indicates the convolution operator. But from the central limit theorem it follows that:

$$\lim_{S \rightarrow +\infty} p_{f_m}(\cdot) = \mathcal{N}(0, S^{-1}), \quad \text{in distribution.} \quad (38)$$

As a consequence, it is likely that for large S there is no advantage of using probability distributions $p(\cdot)$ and estimator functions Ψ different from the unit normal distribution and the ML, respectively, i.e.:

$$\lim_{S \rightarrow +\infty} \min_{p, \Psi} J(p, F_{\text{ave}}, \Psi) = J(\mathcal{N}(0, 1), F_{\text{ave}}, \Psi_{\text{ML}}) \quad (39)$$

although this claim should be rigorously proven. Even if the previous claim holds, it is still interesting to ask whether for finite S there are distributions $p(\cdot)$ that can provide better estimation performance than the Gaussian does. For example, it would be interesting to evaluate the performance, at least numerically, of highly asymmetrical distributions like the exponentials or discrete distributions like the binary distribution. We are currently exploring these directions.

IV. SPECIAL CASE: $F = \text{MAXIMUM}$

Let:

$$F_{\text{max}} := F(y_{1,m}, \dots, y_{S,m}) = \max_i \{y_{i,m}\} =: f_m . \quad (40)$$

We can notice immediately that, if $y_{i,m}$ has probability density $p(a)$ and distribution $P(a)$, then Equ. (18) leads to a joint density on the random vector f_1, \dots, f_M of the form:

$$p_{f_1, \dots, f_M}(a_1, \dots, a_M) = S^M \prod_{m=1}^M P(a_m)^{S-1} p(a_m) . \quad (41)$$

Maximization of the previous likelihood leads to the generic ML estimator for S^{-1} :

$$\widehat{S}^{-1} = \Psi_{\text{ML}}(f_1, \dots, f_M) := -\frac{1}{M} \sum_{m=1}^M \log(P(f_m)) . \quad (42)$$

It is immediate to show that the relative ML estimator \widehat{S} of S is given by $\widehat{S} = 1/\Psi_{\text{ML}} = 1/\widehat{S}^{-1}$. Define \mathcal{P} as the class of densities $p(\cdot)$ whose relative distribution $P(\cdot)$ is strictly monotonic and continuous. Then the estimators \widehat{S}^{-1} and \widehat{S} are characterized by the following propositions:

Proposition 3. For any $P(a) \in \mathcal{P}$, Ψ_{ML} is the MVUE of S^{-1} .

This means that if we restrict Ψ to be unbiased, then $\widehat{S}^{-1} = \Psi_{\text{ML}}$ is optimal with respect to index (3). In addition, the performance of the estimator is independent of the adopted density. This is made precise in the following proposition.

Proposition 4. It holds that:

$$\begin{aligned} \min_{\substack{p(\cdot) \in \mathcal{P} \\ \Psi \text{ s.t. } \mathbb{E}[\Psi] = S^{-1}}} J(p, F_{\text{max}}, \Psi) &= J(\mathcal{U}[0, 1], F_{\text{max}}, \Psi_{\text{ML}}) . \end{aligned} \quad (43)$$

V. DISCUSSION FOR DIFFERENT F : MINIMUM AND RANGE

A. Uniform generation with min. consensus

For symmetry reasons, the usage of min. consensus strategies lead to the same performance results achievable with max. consensus ones. In this case the general expression for ML estimators of S^{-1} is:

$$\Psi(f_1, \dots, f_M) = -\frac{1}{M} \sum_{m=1}^M \log(1 - P(f_m)) . \quad (44)$$

B. Uniform generation with range consensus

Running max. and min. consensus in parallel it is possible to find simultaneously the statistics of order 1 and S of $\{f_m\}$:

$$\overline{f_m} := \max_i \{y_{i,m}\} \quad \underline{f_m} := \min_i \{y_{i,m}\} . \quad (45)$$

If $y_{i,m}$ has probability density $p(a)$ and distribution $P(a)$, then Equ. (19) leads to a joint density on $\overline{f_m}, \underline{f_m}$ of the form:

$$\begin{aligned} p_{\overline{f_m}, \underline{f_m}}(a_1, a_2) &= (S^2 - S) (P(a_1) - P(a_2))^{S-2} \\ &\cdot p(a_1) p(a_2) \end{aligned} \quad (46)$$

whenever $a_1 \geq a_2$, while $p_{\overline{f_m}, \underline{f_m}}(a_1, a_2) = 0$ otherwise. The joint density on $\overline{f_1}, \dots, \underline{f_M}$ can be immediately computed and minimized in S , in order to obtain a general ML estimator of the form:

$$\widehat{S} = \frac{1}{2} - L^{-1} + \sqrt{\frac{1}{4} + L^{-2}} \quad (47)$$

where:

$$L := \frac{1}{M} \sum_{m=1}^M \log(P(\overline{f_m}) - P(\underline{f_m})) . \quad (48)$$

Considering again $y_{i,m} \sim \mathcal{U}[0, 1]$, the joint density is thus:

$$p(\overline{f_1}, \dots, \underline{f_M} | S) = S^M (S-1)^M \prod_{m=1}^M (\overline{f_m} - \underline{f_m})^{S-2} \quad (49)$$

and Equ. (48) must be refined in:

$$L := \frac{1}{M} \sum_{m=1}^M \log(\overline{f_m} - \underline{f_m}). \quad (50)$$

Notice that the random variable $r_m := \overline{f_m} - \underline{f_m}$ has probability density function:

$$\begin{aligned} p(r_m | S) &= \frac{\partial}{\partial r_m} \left(1 - \int_{r_m}^1 \int_0^{\overline{f_m} - r_m} p(\overline{f_m}, \underline{f_m} | S) d\overline{f_m} d\underline{f_m} \right) \\ &= S(S-1) (r_m^{S-2} - r_m^{S-1}) \end{aligned} \quad (51)$$

which transformation $\log(r_m)$ is anymore a r.v. which probability density is analytically known. For this reason we are not able to find the density of L (and thus of $\widehat{S}|S$) in a closed form. In this case $p(\widehat{S}|S)$ shall be estimated with Monte Carlo simulations, as we made in Fig. 3. Rather surprisingly, simulations show that, with the same amount of information exchanged among the sensors and with $y_{i,m} \sim \mathcal{U}[0, 1]$, the range consensus approach performs slightly better than the max. consensus one (notice that for a given M the range-consensus scheme actually computes $2M$ sensible data).

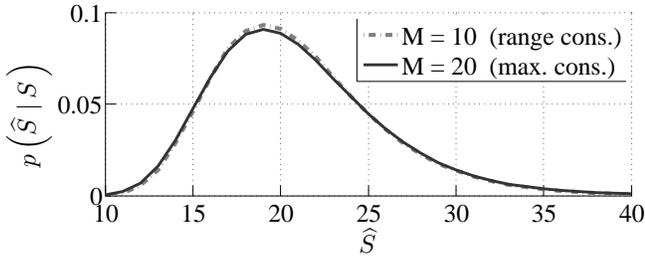


Fig. 3: Comparisons between the empirical $p(\widehat{S}|S)$ relative to max. consensus and range consensus ($2 \cdot 10^6$ number of samples, $S = 20$).

VI. CONCLUSIONS

We presented how to design consensus-based distributed algorithms for the probabilistic estimation of the number of active sensors that do not require leader election steps, are suitable for on-line estimations and that do not require sensor authentication features. These algorithms are based on the fact that when consensus algorithms are applied to randomly generated data, the probability distributions of the consensus results are in general parametrized by the actual number of participants in the consensus procedure. We shown that for max. consensus algorithms, ML estimators have a general fixed structure and their performances are the same for a large class of data generation schemes. Moreover we shown that this structure is asymptotically optimal also whenever average consensus are used.

A first branch of the future works is to analyze issues on the design also in non-asymptotic frameworks, specially for small networks, and to analyze the effects of quantizations on the results. Another branch is to use these procedures to develop network topology change algorithms, focusing on

the analysis of the tradeoffs between velocity of convergence and accuracy of the estimators.

APPENDIX

Proof of Prop. 1, 3 and 4. We start considering Prop. 3. $T(f_1, \dots, f_M) := -\sum_{m=1}^M \log(P(f_m))$ and the relative little abuse of notation:

$$\widehat{S}^{-1}(T) := \widehat{S}^{-1}(T(f_1, \dots, f_M)) := \widehat{S}^{-1}(f_1, \dots, f_M). \quad (52)$$

Since $p_{f_1, \dots, f_M}(a_1, \dots, a_M | S)$ can be rewritten as:

$$\left(\prod_{m=1}^M p(a_m) \right) (S^M \exp(-(S-1)T(a_1, \dots, a_M))) \quad (53)$$

for the Fisher-Neyman factorization theorem $T(f_1, \dots, f_M)$ is a sufficient statistic for S . From the Lehmann-Scheffé theorem, we know that if T is also complete and $\mathbb{E}[\widehat{S}^{-1}] = S^{-1}$, then \widehat{S}^{-1} is MVUE for S^{-1} . Defining $z_m := P(f_m)$, we have that $z_m \sim \mathcal{U}[0, 1]$, thus, for the same reasonings made in Equ. (24), we have that $-\log(P(f_m))$ is an exponential random variable. This implies that:

$$M\widehat{S}^{-1} = -\sum_{m=1}^M \log(P(f_m)) \sim \text{Gamma}\left(M, \frac{1}{S}\right) \quad (54)$$

and thus condition $\mathbb{E}[\widehat{S}^{-1}] = S^{-1}$ is satisfied. The completeness of $T(f_1, \dots, f_M)$ can be proved showing that if $g(T)$ is a generic measurable function s.t. $\mathbb{E}[g(T)|S] = 0$ independently of S , then it must be $g(\cdot) \equiv 0$ almost everywhere (a.e.). Considering that T is Gamma($M, \frac{1}{S}$), the previous condition on the expectation can be rewritten as:

$$\Gamma(M)^{-1} S^M \int_0^{+\infty} g(T) T^{M-1} \exp(-TS) dT \equiv 0. \quad (55)$$

This is equivalent to say that the Laplace transform of $g(T) T^{M-1}$ has to be zero a.e., and this happens if and only if $g(T)$ is zero a.e. This proves the completeness of T and thus the proposition.

For what concerns the proof of Prop. 4, it is a consequence of (55) that shows that the distribution of the estimator is independent of the density p which is adopted. Finally, the proof of Prop. 1 is omitted since can be obtained with reasonings similar to those followed above (e.g., completeness can be proved just repeating the same argument relative to (55) except that the chi-square in place of the Gamma distribution has to be considered). \square

Proof of Prop. 2. Assume a_1^x, \dots, a_S^x i.i.d. and extracted from p_x . Define:

$$a_i^y := \frac{a_i^x - \mu}{\sigma} \quad f_x := \frac{1}{S} \sum_{i=1}^S a_i^x \quad f_y := \frac{1}{S} \sum_{i=1}^S a_i^y. \quad (56)$$

From $p_x(a_i^x) = p_y(a_i^y)$, and using the invariance of the average function F_{ave} with respect to translation and scaling (it is a linear function), it follows immediately that:

$$p_{f_x}(F_{\text{ave}}(a_1^x, \dots, a_S^x) | S) = p_{f_y}(F_{\text{ave}}(a_1^y, \dots, a_S^y) | S). \quad (57)$$

This means that the conditioned statistics from which the optimal Ψ is designed are exactly the same in both cases x and y , and this implies Equ. (36). \square

REFERENCES

- [1] Steven D. Glaser. Some real-world applications of wireless sensor nodes. In *In SPIE Symposium on Smart Structures and Materials*, pages 344–355. SPIE Press, 2004.
- [2] D. Puccinelli and M. Haenggi. Wireless Sensor Networks: Applications and Challenges of Ubiquitous Sensing. *Circuits and Systems Magazine, IEEE*, 5(3), 2005.
- [3] Pramod K. Varshney. *Distributed Detection and Data Fusion*. Springer-Verlag New York, Inc., Secaucus, NJ, USA, 1996.
- [4] D.P. Bertsekas and J.N. Tsitsiklis. *Parallel and Distributed Computation: Numerical Methods*. Athena Scientific, Belmont, MA, 1997.
- [5] Reza Olfati-Saber, J.A. Fax, and R.M. Murray. Consensus and cooperation in multi-agent networked systems. *Proceedings of IEEE*, 95(1):215–233, January 2007.
- [6] Jorge Cortés. Distributed algorithms for reaching consensus on general functions. *Automatica*, 44(3):726–737, March 2008.
- [7] S. Bolognani, S. Del Favero, L. Schenato, and D. Varagnolo. Consensus-based distributed sensor calibration and least-square parameter estimation in WSNs,. *International Journal of Robust and Nonlinear Control*, 20(2):176–193, 2010.
- [8] Saverio Bolognani, Ruggero Carli, and Sandro Zampieri. A PI consensus controller with gossip communication for clock synchronization in wireless sensors networks. *NECSYS 2009*, September 2009.
- [9] Damiano Varagnolo, Gianluigi Pillonetto, and Luca Schenato. Distributed consensus-based Bayesian estimation: sufficient conditions for performance characterization. *2010 American Control Conference*, June 2010.
- [10] Damiano Varagnolo, Gianluigi Pillonetto, and Luca Schenato. Distributed function and time delay estimation using nonparametric techniques. *48th IEEE Conference on Decision and Control*, December 2009.
- [11] C. Budianu, S. Ben-David, and L. Tong. Estimation of the number of operating sensors in large-scale sensor networks with mobile access. *Signal Processing, IEEE Transactions on*, 54(5):1703–1715, May 2006.
- [12] I. J. Good. The population frequencies of species and the estimation of population parameters. *Biometrika*, 40(3):237–264, December 1953.
- [13] A. Leshem and L. Tong. Estimating sensor population via probabilistic sequential polling. *Signal Processing Letters, IEEE*, 12(5):395–398, May 2005.
- [14] M. Howlader, M.R. Frater, and M.J. Ryan. Estimating the number and distribution of the neighbors in an underwater communication network. In *Sensor Technologies and Applications, 2008. SENSORCOMM '08. Second International Conference on*, pages 693–698, August 2008.
- [15] Jiun-Long Huang, Shih-Chuan Chid, and Xin-Mao Huang. GPE: A Grid-based Population Estimation Algorithm for Resource Inventory Applications over Sensor Networks. *Journal of Information Science and Engineering*, January 2009.
- [16] Y.-D. Huang and M. Barket. On estimating the number of sources with a frequency-hopped signaling sensor array. *Antennas and Propagation, IEEE Transactions on*, 39(9):1384–1390, Sep 1991.
- [17] Michéle Basseville and Igor V. Nikiforov. *Detection of Abrupt Changes - Theory and Application*. Prentice-Hall, Englewood Cliffs, N.J., April 1993.
- [18] Fabio Fagnani and Sandro Zampieri. Randomized consensus algorithms over large scale networks. *Selected Areas in Communications, IEEE Journal on*, 26(4):634–649, May 2008.
- [19] Fabio Fagnani and Sandro Zampieri. Asymmetric randomized gossip algorithms for consensus. *Proceedings of the 17th World Congress, the International Federation of Automatic Control*, pages 9051–9056, July 2008.
- [20] H. A. David and H. N. Nagaraja. *Order Statistics*. Wiley series in Probability and Statistics, 2003.