A Distributed Minimum Variance Estimator for Sensor Networks

A. Speranzon, *Member, IEEE*, C. Fischione, *Member, IEEE*,
K. H. Johansson, *Member, IEEE*, and A. Sangiovanni-Vincentelli, *Fellow, IEEE*

***Abstract*—A distributed estimation algorithm for sensor networks is proposed. A noisy time-varying signal is jointly tracked by a network of sensor nodes, in which each node computes its estimate as a weighted sum of its own and its neighbors' measurements and estimates. The weights are adaptively updated to minimize the variance of the estimation error. Both estimation and the parameter optimization is distributed; no central coordination of the nodes is required. An upper bound of the error variance in each node is derived. This bound decreases with the number of neighboring nodes. The estimation properties of the algorithm are illustrated via computer simulations, which are intended to compare our estimator performance with distributed schemes that were proposed previously in the literature. The results of the paper allow to trading-off communication constraints, computing efforts and estimation quality for a class of distributed filtering problems.**

***Index Terms*—Distributed Estimation; Sensor Networks; Convex Optimization, Parallel and Distributed Computation; In-network Processing; Cooperative Communication.**

I. Introduction

A sensor network (SN) is a network of autonomous devices that can sense their environment, make computations and communicate with neighboring devices. SNs, and in particular wireless sensor networks, have a growing domain of application in areas such as environmental monitoring, industrial automation, intelligent buildings, search and surveillance, and automotive applications [3]–[5]. The characteristics of SNs motivate the development of new classes of distributed estimation and control algorithms, which explore these systems' limited power, computing and communication capabilities. It is important that the algorithms have tuning parameters that can be adjusted according to the demands set by the applications. In this paper, we investigate a distributed estimation algorithm for tracking an unknown time-varying physical variable.

The new estimator for SNs presented in this paper belongs to a class of recently developed filtering algorithms that exploit *in-network computing* [6]. The scalability of these algorithms is based on that node operates using only local information. Suitable cooperation between neighboring nodes improves the estimation quality considerably. Using sensor readings from more than one sensor, for example, can overcome intrinsic performance limitations due to uncertainty and noise present in individual devices.

In-network computing thus differs from the traditional architecture where sensors simply provide raw data to a fusion center. By letting the network do the computations, it is possible to reach a scalable, fault tolerant and flexible design. The drawback is that such a system is more difficult to analyze, as it is an asynchronous distributed computing system [7] with inputs and dynamics coupled to a physical environment. Despite current research activity and major progress, the theoretical understanding is far from satisfactory of these systems, exposed to link and node failures, packet drops, restricted power consumption etc.

A. Main Contribution

The main contribution of this paper is a novel distributed minimum variance estimator. A time-varying signal is jointly tracked by a SN, in which each node computes an estimate as a weighted sum of its own and its neighbors' measurements and estimates. The filter weights are time varying and updated locally. The filter has a cascade structure with an inner loop producing the state estimate and an outer loop producing an estimate of the error covariance. The state estimate is obtained as the solution of an optimization problem with quadratic cost function and quadratic constraints. We show that the problem has a distributed implementation with conditions that can be locally checked. It is argued that the estimator is practically stable if the signal to track is slowly varying, so the estimate of each node converges to a neighborhood of the signal to track. The estimate in each node has consequently a small variance and a small bias. A bound on the estimation error variance, which is linear in the measurement noise variance and decays with the number of neighboring nodes, is presented. The algorithm is thus characterized by a trade-off between the amount of communication and the resulting estimation quality. Compared to similar distributed algorithms presented in the literature, the one introduced in this paper features better estimates, but at the cost of a slightly increased computational complexity. These aspects are illustrated in the implementation

The work by A. Speranzon was partially supported by the European Commission through the Marie Curie Transfer of Knowledge project BRIDGET (MKTD-CD 2005 029961). The work by C. Fischione, K. H. Johansson, and A. Sangiovanni-Vincentelli was partially supported by the HYCON NoE, contract number FP6-IST-511368. It was partially funded also by the Swedish Foundation for Strategic Research and the Swedish Research Council. A. Sangiovanni-Vincentelli and C. Fischione wish to acknowledge the support of the NSF ITR CHESS and the GSRC. C. Fischione and A. Speranzon gratefully acknowledge the support of the San Francisco Italian Institute of Culture throughout the Science & Technology Attaché T. Scapolla. Preliminary versions of the results of the paper were presented at conferences [1], [2].

A. Speranzon is with Unilever R&D Port Sunlight, Bebington, Wirral, CH63 3JW, United Kingdom. E-mail: alberto.speranzon@unilever.com. C. Fischione and A. Sangiovanni-Vincentelli are with University of California at Berkeley, CA. E-mail: {fischion,alberto}@eecs.berkeley.edu. K. H. Johansson is with School of Electrical Engineering, Royal Institute of Technology, 100-44 Stockholm, Sweden. E-mail: kallej@ee.kth.se.

discussion and computer simulations exposition in the latter part of the paper.

B. Related Work

Distributed signal processing is a very active research area due to the recent developments in networking, computer and sensor technologies.

The estimator presented in this paper has two particular characteristics: it does not rely on a model of the signal to track, and its filter coefficients are time varying. It is related to recent contributions on low-pass filtering by diffusion mechanisms, e.g., [1]–[13]. Many of these papers focus on diffusion mechanisms to have each node of the network obtain the average of the initial samples of the network nodes. Major progress has been made in understanding the convergence behavior of these consensus or state-agreement approaches. In [11], a scheme for sensor fusion based on a consensus filter is proposed. Here, each node computes a local weighted least-squares estimate that is shown to converge to the maximum-likelihood solution for the overall network. An extension of this approach is presented in [14], where the authors study a distributed average computation of a time-varying signal, when the signal is affected by a zero-mean noise. A convex optimization problem is posed to compute the edge weights, which each node uses to minimize the least mean square deviation of the estimates. The same linear filter is also considered in [15], where the weights are computed off-line to speed up the computation of the averages. Further characterization of consensus filters for distributed sensor fusion is given in [13].

Another approach to distributed estimation is based on nonlinear filters using self-synchronization and coupling functions, e.g., [16]–[19]. In this case, the estimate of each node is provided by the state of a nonlinear dynamical system. This system is coupled to some of the other nodes by a static coupling function. Some conditions on the coupling function that lead to asymptotic state synchronization are investigated in [19].

Distributed filtering using model-based approaches is studied in various wireless network contexts, e.g., [20]–[24]. Distributed Kalman filters and more recently a combination of the diffusion mechanism, discussed previously, with distributed Kalman filtering, e.g., [12], [25] have been proposed. A plausible approach is to communicate the estimates of the local Kalman filters, and then average these values using a diffusion strategy.

The originality of our approach is based on:

- Our estimator tracks a time-varying signal, while [9]–[11] are limited to averaging initial samples.
- Our approach does not require a model of the system that generates the signal to track, in contrast to model-based approaches, e.g., [12], [24].
- We do not impose a pre-assigned coupling law among the nodes as in [19].
- Compared to [11]–[13], we do not rely on the Laplacian matrix associated to the communication graph, but consider a more general model of the filter structure.
- Our filter parameters are computed through distributed algorithms, whereas for example [14] and [15] rely on centralized algorithms for designing the filters.
- With respect to our own early contributions [1], [2], where we extended the algorithms in [11]–[13] by designing the filter weights such that the variance of the estimation errors is minimized, here we improve the filter design considerably and we characterize the performance limit of the filter.

C. Outline

Section II presents the distributed estimation problem considered throughout the paper. The distributed estimator design is discussed in Section III. A distributed minimum variance optimization problem is posed and by restricting the set of admissible filter weights it is possible to obtain a solution where the error convergence is guaranteed. A bound on the estimation error variance is also computed. The latter part of Section III discusses estimation of the error covariance. Section IV presents the detail of the implementation of the estimation algorithm. Numerical results illustrating the performance of the proposed estimator and comparing it to some related proposals are also given. Finally, Section V concludes the paper.

D. Notation

We denote the set of non-negative integers as $\mathbb{N}_0 = \{0, 1, 2, \dots\}$. With $|\cdot|$ we denote either absolute value or cardinality, depending on the context. With $\|\cdot\|$ we denote the ℓ^2-norm of a vector and the spectral norm of a matrix. Given a matrix $A \in \mathbb{R}^{n\times n}$, we denote with $\lambda_r(A)$, $1 \leq r \leq n$, its r-th eigenvalue, with $\lambda_{\min}(A) = \lambda_1(A)$ and $\lambda_{\max}(A) = \lambda_n(A)$ being the minimum and maximum eigenvalue, respectively, where the order is taken with respect to the real part. We refer to its largest singular value as $\gamma_{\max}(A)$. The trace of A is denoted $\operatorname{tr} A$. With I and $\mathbb{1}$ we denote the identity matrix and the vector $(1, \dots, 1)^T$, respectively. Given a stochastic variable x we denote by $\mathbb{E}\, x$ its expected value. For the sake of notational simplicity, we disregard the time dependence when it is clear from the context. We define $\mathbb{N}_0 = \mathbb{N} \cup \{0\}$.

II. Preliminaries

A. Problem Formulation

Consider $N > 1$ sensor nodes placed at random and static positions in space. We assume that each node measures a common scalar signal $d(t)$ affected by additive noise:

$$u_i(t) = d(t) + v_i(t)\,, \quad i = 1, \dots, N\,,$$

with $t \in \mathbb{N}_0$ and where $v_i(t)$ is zero-mean white noise. Let us collect measurements and noise variables in vectors, $u(t) = (u_1(t), \dots, u_N(t))^T$ and $v(t) = (v_1(t), \dots, v_N(t))^T$, so that we can rewrite the previous equation as

$$u(t) = d(t)\,\mathbb{1} + v(t)\,, \qquad t \in \mathbb{N}_0\,.$$

The covariance matrix of $v(t)$ is assumed to be diagonal $\Sigma = \sigma^2 I$, so $v_i(t)$ and $v_j(t)$, for $i \neq j$, are uncorrelated. The additive noise, in each node, can be averaged out only if nodes communicate measurements or estimates. The communication rate of the measurements and estimates should be just fast enough to track the variations of $d(t)$. Indeed, increasing the sampling rate, in general, is not beneficial because measurements might then be affected by auto-correlated noise.

It is convenient to model the communication network as an undirected graph $\mathcal{G} = (\mathcal{V}, \mathcal{E})$, where $\mathcal{V} = \{1, \dots, N\}$ is the vertex set and $\mathcal{E} \subseteq \mathcal{V} \times \mathcal{V}$ the edge set. We assume that if $(i, j) \in \mathcal{E}$ then $(j, i) \in \mathcal{E}$, so that the direction of the edges can be dropped when representing the network. The graph $\mathcal{G}$ is said to be connected if there is a sequence of edges in $\mathcal{E}$ that can be traversed to go from any vertex to any other vertex.

In the sequel, we denote the set of neighbors of node $i \in \mathcal{V}$ plus the node itself as

$$\mathcal{N}_i = \{j \in \mathcal{V} : (j, i) \in \mathcal{E}\} \cup \{(i, i)\} .$$

The estimation algorithm we propose is such that a node i computes an estimate $x_i(t)$ of $d(t)$ by taking a linear combination of neighboring estimates and measures

$$x_i(t) = \sum_{j \in \mathcal{N}_i} k_{ij}(t) x_j(t-1) + \sum_{j \in \mathcal{N}_i} h_{ij}(t) u_j(t) . \tag{II.1}$$

We assume that neighboring estimates and measurements are always successfully received, i.e., there are no packet losses. This assumption is obviously plausible for wired connections, but it is still valid in wireless networks if certain assumptions hold true. More specifically, the designed estimator is suitable in wireless networks where the sampling time between measurements is long compared to the coherence time of the wireless channel (which is around some hundreds of milliseconds) and an Automatic Repeat Request (ARQ) protocol is used. Under such assumptions, if the wireless channel does not allow a packet to be successfully received at a given time instance, there is enough time to detect and retransmit erroneous packets until they are successfully received. These assumptions are representative of the IEEE 802.11b and IEEE 802.11g [26], which have been actually used for distributed estimation and control algorithms of unmanned aerial vehicles [27].

We assume that for each node i, the algorithm is initialized with $x_j(0) = u_i(0)$, $j \in \mathcal{N}_i$. In vector notation, we have

$$x(t) = K(t) x(t-1) + H(t) u(t) . \tag{II.2}$$

$K(t)$ and $H(t)$ can be interpreted as the adjacency matrices of two graphs with time-varying weights. These graphs are compatible with the underlying communication network represented by $\mathcal{G}$.

Given a SN modeled as a connected graph $\mathcal{G}$, we have the following design problem: find time-varying matrices $K(t)$ and $H(t)$, compatible with $\mathcal{G}$, such that the signal $d(t)$ is consistently estimated and the variance of the estimate is minimized. Moreover, the solution should be distributed in the sense that the computation of $k_{ij}(t)$ and $h_{ij}(t)$ should be performed locally by node i.

B. Convergence of the Centralized Estimation Error

In this section we derive conditions on $K(t)$ and $H(t)$ that guarantee the estimation error to converge. Define the estimation error $e(t) = x(t) - d(t)\mathbb{1}$. Introduce $\delta(t) = d(t) - d(t-1)$, so that the error dynamics can be described as

$$\begin{aligned} e(t) &= K(t) e(t-1) + d(t)(K(t) + H(t) - I)\mathbb{1} \\ &\quad - \delta(t) K(t)\mathbb{1} + H(t) v(t) . \end{aligned} \tag{II.3}$$

Taking the expected value with respect to the stochastic variable $v(t)$, we obtain

$$\begin{aligned} \mathbb{E}\, e(t) &= K(t)\,\mathbb{E}\, e(t-1) + d(t)(K(t) + H(t) - I)\mathbb{1} \\ &\quad - \delta(t) K(t)\mathbb{1} . \end{aligned} \tag{II.4}$$

Proposition II.1. *Consider the system Equation* (II.3)*. Assume that*

$$(K(t) + H(t))\mathbb{1} = \mathbb{1} , \tag{II.5}$$

and that there exists $0 \le \gamma_0 < 1$ *such that*

$$\gamma_{\max}(K(t)) \le \gamma_0 \tag{II.6}$$

for all $t \in \mathbb{N}_0$.

(i) *If* $H(t)\mathbb{1} = \mathbb{1}$, *for all* $t \in \mathbb{N}_0$, *then*

$$\lim_{t \to +\infty} \mathbb{E}\, e(t) = 0 .$$

(ii) *If* $|\delta(t)| < \Delta$, *for all* $t \in \mathbb{N}_0$, *then*

$$\lim_{t \to +\infty} \| \mathbb{E}\, e(t) \| \le \frac{\sqrt{N}\Delta\gamma_0}{1 - \gamma_0} . \tag{II.7}$$

Proof: If $(K(t) + H(t))\mathbb{1} = \mathbb{1}$ then the system equation reduces to

$$\mathbb{E}\, e(t) = K(t)\,\mathbb{E}\, e(t-1) + \delta(t)(H(t) - I)\mathbb{1} . \tag{II.8}$$

(i) If $H(t)\mathbb{1} = \mathbb{1}$, then (II.8) becomes $\mathbb{E}\, e(t) = K(t)\,\mathbb{E}\, e(t-1)$. Let us consider the function $V(t) = \|\mathbb{E}\, e(t)\|$. It follows that

$$V(t) \le \|K(t)\| V(t-1) \le \gamma_0 V(t-1) \le \gamma_0^t V(0) ,$$

which implies that $\lim_{t \to +\infty} \mathbb{E}\, e(t) = 0$.

(ii) In this case, we have $H(t)\mathbb{1} - \mathbb{1} = -K(t)\mathbb{1}$ and thus the system Equation (II.3) becomes

$$\mathbb{E}\, e(t) = K(t)\,\mathbb{E}\, e(t-1) - \delta(t) K(t)\mathbb{1} .$$

With $V(t) = \|\mathbb{E}\, e(t)\|$, we have

$$\begin{aligned} V(t) &\le \|K(t)\| V(t-1) + \|K(t)\|\sqrt{N}\Delta \\ &\le \gamma_0 V(t-1) + \gamma_0\sqrt{N}\Delta \\ &\le \gamma_0^t V(0) + \gamma_0 \frac{1 - \gamma_0^t}{1 - \gamma_0}\sqrt{N}\Delta \end{aligned}$$

Taking the limit for $t \to +\infty$ we obtained the result.

■

Proposition II.1(i) provides the condition $H(t)\mathbb{1} = \mathbb{1}$ under which the estimate is unbiased. It is possible to show that in this case the variance is minimized if $K(t) = 0$ and

$$h_{ij}(t) = h_{ji}(t) = \begin{cases} \dfrac{1}{|\mathcal{N}_i|} & \text{if } j \in \mathcal{N}_i \\ 0 & \text{otherwise} . \end{cases}$$

Note that nodes do not use any memory and that the error variance at each node is proportional to its neighborhood size. However, if $d(t)$ is slowly varying, then, under the assumptions of Proposition II.1(ii), it is possible to guarantee that $\| \mathbb{E}\, e(t)\|$ tends to a neighborhood of the origin, but the estimate might be biased. Note also that $\| \mathbb{E}\, e(t)\|$ is a cumulative bias, i.e., it is a function of the sum of the N biases of individual nodes.

The size of the cumulative bias can be kept small with respect to the signal to track by defining a proper value of γ_0. In particular, Equation (II.7) can be related to the Signal-to-Noise Ratio (SNR) of the average of the estimate as follows. Let P_d denote the average power of d and let P_b denote the desired power of the biases of the average of the estimates. Then, we define the desired SNR as $\text{SNR} = P_d/P_b$. Since there are N nodes, we consider the average SNR of each node as $\Upsilon = \text{SNR}/N$. Let us assume that we want the estimator to guarantee that the right-hand side of Equation (II.7) be equal to this desired $\sqrt{\text{SNR}}$, i.e., that

$$\gamma_0 = \frac{\sqrt{\Upsilon}}{\sqrt{\Upsilon} + \Delta} \, .$$

The right-hand side is useful in the tuning of the estimator. Hence, we denote it as $f(\Delta, \Upsilon)$. By choosing an appropriate Υ, we have a guaranteed convergence property of the estimator given by the corresponding $f(\Delta, \Upsilon)$. This function will allow us to relate the size of the bias of estimates with the variations of the signal to track, and the stability of the estimates, as we see in the next sections.

III. Distributed Estimator Design

In this section we describe how each node computes adaptive weights to minimize its estimation error variance. Notice that, in order to guarantee that the estimation error of the overall sensor network, in average, converges to a neighbor of the origin, each node needs to locally compute the row-elements of $K(t)$ and $H(t)$ so that conditions in Proposition II.1 are fulfilled. The condition $(K(t) + H(t))\, \mathbb{1} = \mathbb{1}$ is easily handled in a distributed way, as it states that the sum of the row-elements of $K(t)$ and $H(t)$ need to sum up to one. The bound on the maximum singular value of $K(t)$, however, requires some transformations so that new conditions on the row-elements of $K(t)$ fulfill $\gamma_{\max}(K(t)) \le f(\Delta, \Upsilon)$. It turns out that it is possible to determine a local condition, $\sum_{j\in\mathcal{N}_i} k_{ij}^2 \le \psi_i$, where ψ_i is a constant that can be computed locally by the nodes. We then pose a optimization problem for finding optimal weights that minimize the error variance in each node, where the previous conditions are considered as constraints. An important aspect of the distributed optimal solution is that the weights depend on the error covariance matrix, which is not available at each node. We end this section by discussing a way of estimating it.

A. Distributed Variance Minimization

Let $M_i = |\mathcal{N}_i|$, which denotes the number of neighbors of node i, including the node itself. Collect the estimation errors available at node i in the vector $\epsilon_i \in \mathbb{R}^{M_i}$. The elements of ϵ_i are ordered according to node indices:

$$\epsilon_i = (e_{i_1}, \dots, e_{i_{M_i}})^T, \qquad i_1 < \cdots < i_{M_i} \, .$$

Similarly, we introduce vectors $\kappa_i^T(t), \eta_i^T(t) \in \mathbb{R}^{M_i}$ corresponding to the non-zero elements of row i of the matrices $K(t)$ and $H(t)$, respectively, and ordered according to node indices.

The expected value of the estimation error at node i can be written as

$$\mathbb{E}\, e_i(t) = \kappa_i^T(t)\, \mathbb{E}\, \epsilon_i(t-1) - \kappa_i^T(t)\delta(t)\, \mathbb{1} \ , \tag{III.1}$$

where we used the fact that $d(t) - d(t-1) = \delta(t)$ and that $(K(t) + H(t))\, \mathbb{1} = \mathbb{1}$. Note that the latter inequality is equivalent to $(\kappa_i(t) + \eta_i(t))^T\, \mathbb{1} = 1$. We assume that $e_i(0) = u_i(0)$. Hence

$$\mathbb{E}\, (e_i(t) - \mathbb{E}\, e_i(t))^2 = \kappa_i^T(t)\Gamma_i(t-1)\kappa_i(t) + \sigma^2 \eta_i^T(t)\eta_i(t) \, ,$$

where $\Gamma_i(t) = \mathbb{E}\, (\epsilon_i(t) - \mathbb{E}\, \epsilon_i(t))(\epsilon_i(t) - \mathbb{E}\, \epsilon_i(t))^T$. To minimize the variance of the estimation error in each node, we need to determine $\kappa_i(t)$ and $\eta_i(t)$ so that the previous expression is minimized at each time instant. We have the following optimization problem:

$$\mathcal{P}_1 : \qquad \min_{\kappa_i(t), \eta_i(t)} \qquad \kappa_i^T(t)\Gamma_i(t-1)\kappa_i(t) + \sigma^2 \eta_i^T(t)\eta_i(t) \tag{III.2}$$

$$\text{s.t.} \qquad (\kappa_i(t) + \eta_i(t))^T\, \mathbb{1} = 1 \, , \tag{III.3}$$

$$\gamma_{\max}(K(t)) \le f(\Delta, \Upsilon) \, . \tag{III.4}$$

The inequality constraint (III.4) is still global, since $\gamma_{\max}(K(t))$ depends on all $\kappa_i(t)$, $i = 1, \dots, N$. We show next that it can be replaced by the local constraint

$$\|\kappa_i(t)\| \le \psi_i \, , \quad t \in \mathbb{N}_0 \, , \tag{III.5}$$

where $\psi_i > 0$ is a constant that can be computed locally. The new constraint, however, even though ensure the stability of the estimation error, leads to a distributed solution which is in general different from the centralized one.

For $i = 1, \dots, N$, define the set $\Theta_i = \{j \neq i : \mathcal{N}_j \cap \mathcal{N}_i \neq \emptyset\}$, which is the collection of nodes located at two hops distance from node i plus neighbor nodes of i. We have the following result.

Proposition III.1. *Suppose there exist $\psi_i > 0$, $i = 1, \dots, N$, such that*

$$\psi_i + \sqrt{\psi_i} \sum_{j\in\Theta_i} \sqrt{\alpha_{i,j}^{(i)} \alpha_{i,j}^{(j)} \psi_j} \le f^2(\Delta, \Upsilon) \, , \tag{III.6}$$

where $\alpha_{i,j}^{(i)}, \alpha_{i,j}^{(j)} \in (0, 1)$ are such that

$$\sum_{c\in\mathcal{N}_j\cap\mathcal{N}_i} k_{ic}^2 \le \alpha_{i,j}^{(i)} \sum_{r=1}^{M_i} \kappa_{ii_r}^2 \qquad \sum_{c\in\mathcal{N}_j\cap\mathcal{N}_i} k_{jc}^2 \le \alpha_{i,j}^{(j)} \sum_{r=1}^{M_j} \kappa_{ji_r}^2 .$$

If $\|\kappa_i(t)\|^2 \le \psi_i$, $i = 1, \dots, N$, then $\gamma_{\max}(K(t)) \le f(\Delta, \Upsilon)$.

Proof: We use Geršgorin to bound the eigenvalues of the matrix KK^T, i.e., the singular values of K. The following relations hold: $[KK^T]_{ii} = \sum_{c=1}^{M_i} k_{ic}^2$ and $[KK^T]_{ij} =$

$\sum_{c=1}^{N} k_{ic}k_{jc}$. By the Geršgorin Theorem we know that for $r = 1, \dots, N$

$$\lambda_r(KK^T) \in \bigcup_{i=1}^{N} \left\{ z \in \mathbb{R} : |z - [KK^T]_{ii}| \leq R_i(KK^T) \right\}$$

with

$$R_i(KK^T) = \sum_{\substack{j=1\\j\neq i}}^{N} \left|[KK^T]_{ij}\right| = \sum_{\substack{j=1\\j\neq i}}^{N} \left|\sum_{c=1}^{N} k_{ic}k_{jc}\right| .$$

Now the inner sum in $R_i(KK^T)$ is non-zero only for $c \in \mathcal{N}_j \cap \mathcal{N}_i$. Thus,

$$R_i(KK^T) = \sum_{j\in\Theta_i} \left| \sum_{c\in\mathcal{N}_j\cap\mathcal{N}_i} k_{ic}k_{jc} \right| .$$

Using the Cauchy-Schwarz inequality,

$$\left| \sum_{c\in\mathcal{N}_j\cap\mathcal{N}_i} k_{ic}k_{jc} \right|^2 \leq \sum_{c\in\mathcal{N}_j\cap\mathcal{N}_i} k_{ic}^2 \sum_{c\in\mathcal{N}_j\cap\mathcal{N}_i} k_{jc}^2 .$$

Then,

$$\begin{aligned}
\sum_{\substack{j=1\\j\neq i}}^{N} \left|\sum_{c=1}^{N} k_{ic}k_{jc}\right| &\leq \sum_{j\in\Theta_i} \sqrt{\sum_{c\in\mathcal{N}_j\cap\mathcal{N}_i} k_{ic}^2 \sum_{c\in\mathcal{N}_j\cap\mathcal{N}_i} k_{jc}^2} \\
&\leq \sum_{j\in\Theta_i} \sqrt{\alpha_{i,j}^{(i)} \sum_{r=1}^{M_i} \kappa_{ii_r}^2 \, \alpha_{i,j}^{(j)} \sum_{r=1}^{M_j} \kappa_{jj_r}^2} \\
&= \sqrt{\sum_{r=1}^{M_i} \kappa_{ii_r}^2} \cdot \sum_{j\in\Theta_i} \sqrt{\alpha_{i,j}^{(i)}\alpha_{i,j}^{(j)} \sum_{r=1}^{M_j} \kappa_{jj_r}^2} .
\end{aligned}$$

Hence,

$$\begin{aligned}
\lambda_r(KK^T) \in \bigcup_{i=1}^{N} \Bigg\{ z \in \mathbb{C} : \left| z - \sum_{r=1}^{M_i} \kappa_{ii_r}^2 \right| \leq \sqrt{\sum_{r=1}^{M_i} \kappa_{ii_r}^2} \cdot \\
\sum_{j\in\Theta_i} \sqrt{\alpha_{i,j}^{(i)}\alpha_{i,j}^{(j)} \sum_{r=1}^{M_j} \kappa_{jj_r}^2} \Bigg\}
\end{aligned}$$

From the hypothesis that $\|\kappa_i\| = \sum_{r=1}^{M_i} \kappa_{ii_r}^2 \leq \psi_i$ and (III.6), then

$$\sum_{r=1}^{M_i} \kappa_{ic}^2 + \sqrt{\sum_{r=1}^{M_i} \kappa_{ii_r}^2} \cdot \sum_{j\in\Theta_i} \sqrt{\sum_{r=1}^{M_j} \alpha_{i,j}^{(i)}\alpha_{i,j}^{(j)} \kappa_{jj_r}^2} \leq f^2(\Delta, \Upsilon) .$$

Hence $\gamma_{\max}(K) \leq f(\Delta, \Upsilon)$. ■

Proposition III.1 provides a simple local condition on the filter coefficients such that $\gamma_{\max}(K) \leq f(\Delta, \Upsilon)$. We can expect that Proposition III.1 is in general conservative, because no a-priori knowledge of the network topology is used, the proof relies on the Geršgorin theorem and the Cauchy-Schwartz inequality. There are many other ways to bound the eigenvalues of a matrix by its elements than the one used in the proof above, e.g., [28, pages 378–389]. However, we do not know of any other bounds requiring only local information. Further, the Perron-Frobenius theory cannot be directly applied to bound the eigenvalues, because we make no assumption on the sign of the elements of $K(t)$.

The parameters $\alpha_{i,j}^{(i)}$ and $\alpha_{i,j}^{(j)}$ in Proposition III.1 can all be set to one. However, this yields very conservative bounds on the maximum eigenvalue of KK^T. In Section IV, we show how to chose these parameters to avoid bounds that are too conservative.

B. *Optimal Weights for Variance Minimization*

Using previous results, we consider the following local optimization problem:

$$\mathcal{P}_2 : \qquad \min_{\kappa_i(t),\eta_i(t)} \qquad \kappa_i(t)^T\Gamma_i(t-1)\kappa_i(t) + \sigma^2\eta_i(t)^T\eta_i(t) \tag{III.7}$$

$$\begin{aligned}
\text{s.t.} \qquad & (\kappa_i(t) + \eta_i(t))^T \,\mathbb{1} \, = 1 \\
& \|\kappa_i\|^2 \leq \psi_i \, ,
\end{aligned} \tag{III.8}$$

We remark here that Problem $\mathcal{P}_2$ has a different solution with respect to Problem $\mathcal{P}_1$, since the constraint (III.4) has been replaced with (III.8). Problem $\mathcal{P}_2$ is convex. In fact, the cost function is convex, as $\Gamma(t-1)$ is positive definite, since it represents the covariance matrix of Gaussian random variable, and the two constraints are also convex. The problem admits a strict interior point solution, corresponding to $\kappa_i(t) = 0$ and $\eta_i(t)\,\mathbb{1} \, = 1$. Thus, Slater's condition is satisfied and strong duality holds [29, pag. 226]. The problem, however, does not have a closed form solution: we need to rely on numerical algorithms to derive the optimal $\kappa_i(t)$ and $\eta_i(t)$. The following proposition provides a specific characterization of the solution.

Proposition III.2. *For a given positive definite matrix $\Gamma_i(t-1)$, the solution to problem $\mathcal{P}_2$ is given by*

$$\kappa_i(t) = \frac{\sigma^2(\Gamma_i(t-1) + \xi_i I)^{-1}\,\mathbb{1}}{\sigma^2\,\mathbb{1}^{\,T}(\Gamma_i(t-1) + \xi_i I)^{-1}\,\mathbb{1} \, + M_i} \tag{III.9}$$

$$\eta_i(t) = \frac{\mathbb{1}}{\sigma^2\,\mathbb{1}^{\,T}(\Gamma_i(t-1) + \xi_i I)^{-1}\,\mathbb{1} \, + M_i} \, , \tag{III.10}$$

with $\xi_i \in \left[0, \max(0, \sigma^2/\sqrt{M_i\psi_i} - \lambda_{\min}(\Gamma_i(t-1)))\right]$.

Proof: Since the problem is convex and Slater's condition holds, the KKT conditions are both necessary and sufficient for optimality. The primal and dual optimal points, (κ_i^*, η_i^*) and (λ_i^*, ν_i^*) respectively, need to satisfy

$$\begin{aligned}
&(\kappa_i^*)^T\kappa_i^* - \psi_i \leq 0 \, , \qquad (\kappa_i^* + \eta_i^*)^T\,\mathbb{1} \, - 1 = 0 \\
&,\xi_i^* \geq 0 \, , \qquad \xi_i^*((\kappa_i^*)^T\kappa_i^* - \psi_i) = 0 \, , \\
&2(\Gamma_i + \lambda_i^* I)\kappa_i^* + \nu_i^*\,\mathbb{1} \, = 0 \, , \qquad 2\sigma^2\eta_i^* + \nu_i^*\,\mathbb{1} \, = 0 \, ,
\end{aligned}$$

where the last two KKT conditions follow from $\nabla_{\kappa_i} L(\kappa_i, \eta_i, \xi, \nu)$ and $\nabla_{\eta_i} L(\kappa_i, \eta_i, \xi, \nu)$ with the Lagrangian

$$\begin{aligned}
L(\kappa_i, \eta_i, \xi, \nu) = \kappa_i^T\Gamma_i\kappa_i + \sigma^2\eta_i^T\eta_i + \xi_i\big(\kappa_i^T\kappa_i - \psi_i\big) \\
+ \nu_i\big((\kappa_i + \eta_i)^T\,\mathbb{1} \, - 1\big) \, .
\end{aligned}$$

Combining these two KKT conditions with the second KKT condition we obtain the optimal values. From the fourth KKT condition we have that either $\xi^* = 0$ or $(\kappa_i^*)^T \kappa_i^* = \psi_i$, where the second equality gives

$$\frac{\sigma^4 \, \mathbb{1}^T (\Gamma_i + \xi_i^* I)^{-2} \, \mathbb{1}}{\left(\sigma^2 \, \mathbb{1}^T (\Gamma_i + \xi_i^* I)^{-1} \, \mathbb{1} + M_i\right)^2} = \psi_i \, .$$

We are not able to provide a solution ξ^* in closed form. Instead we give a bound for the variable. From the previous equation, we can enforce a $\xi \geq 0$ such that

$$(\kappa_i^*)^T \kappa_i^* \leq \frac{\sigma^4 \|(\Gamma_i + \xi I)^{-1}\|^2}{M_i} \leq \frac{\sigma^4}{M_i \lambda_{\min}^2 (\Gamma_i + \xi I)} \leq \psi_i \, .$$

from where we obtain

$$\xi \geq \frac{\sigma^2}{\sqrt{M_i \psi_i}} - \lambda_{\min}(\Gamma_i) \, .$$

and for all these values of ξ the first KKT condition is always satisfied. This implies that the optimal value of ξ must be in the interval $[0, \max(0, \sigma^2/\sqrt{M_i \psi_i} - \lambda_{\min}(\Gamma_i(t-1))]$, and the theorem is proven. ■

Proposition III.2 gives an interval within which the optimal ξ_i can be found. The first constraint in problem $\mathcal{P}_2$ is similar to the water-filling problem for power allocation in wireless networks [29]. Analogously to that problem, simple search algorithms such as a bisection algorithm, can be considered to solve for ξ_i numerically. Note that each node i needs to know the covariance matrix $\Gamma_i(t-1)$ to compute the weights. It is important to notice that the problem $\mathcal{P}_2$ does not have the same solution as the problem $\mathcal{P}_1$, as the constraints (III.4) and (III.8) are not equivalent, although if (III.8) holds then (III.4) holds as well.

C. Bounds on the Error Variance

The optimal weights from Proposition III.2 gives the following estimation error variance.

Proposition III.3. *Let $\kappa_i(t)$ and $\eta_i(t)$ be an optimal solution given by* (III.9) *and* (III.10)*. Then*

$$\mathbb{E}\,(e_i(0) - \mathbb{E}\,e_i(0))^2 = \sigma^2$$
$$\mathbb{E}\,(e_i(t) - \mathbb{E}\,e_i(t))^2 \leq \frac{\sigma^2}{M_i} \, ,$$

$t \in \mathbb{N}_0 \setminus \{0\}$.

Proof: For $t = 0$, $e_i(0) = u_i(0) = d(0) + v_i(0)$, so $\mathbb{E}\,(e_i(0) - \mathbb{E}\,e_i(0))^2 = \sigma^2$. For $t > 0$, the error variance of the i-th node with the optimal values of $\kappa_i(t)$ and $\eta_i(t)$ is

$$\begin{aligned}\mathbb{E}\,(e_i(t) - \mathbb{E}\,e_i(t))^2 &= \frac{\sigma^2}{M_i + \sigma^2 \, \mathbb{1}^T (\Gamma_i(t-1) + \xi_i I)^{-1} \, \mathbb{1}} \\ &\quad - \frac{\sigma^4 \xi_i \, \mathbb{1}^T (\Gamma_i(t-1) + \xi_i I)^{-2} \, \mathbb{1}}{\left(M_i + \sigma^2 \, \mathbb{1}^T (\Gamma_i(t-1) + \xi_i I)^{-1} \, \mathbb{1}\right)^2} \\ &\leq \frac{\sigma^2}{M_i + \sigma^2 \, \mathbb{1}^T (\Gamma_i(t-1) + \xi_i I)^{-1} \, \mathbb{1}} \, .\end{aligned}$$

Since $\Gamma_i(t-1)$ is positive definite and $\xi_i \geq 0$, it holds that $\mathbb{1}^T (\Gamma_i(t-1) + \xi_i I)^{-1} \, \mathbb{1} > 0$. Hence $\mathbb{E}\,(e_i(t) - \mathbb{E}\,e_i(t))^2 \leq \frac{\sigma^2}{M_i} \, , \quad t \in \mathbb{N}_0 \setminus \{0\}$. This concludes the proof. ■

A consequence of Proposition III.3 is that the estimation error in each node is always upper bounded by the variance of the estimator that computes the averages of the M_i measurements $u_i(t)$. The bound is obviously rather conservative, since we do not use any knowledge about the covariance matrix $\Gamma_i(t)$. Proposition III.2 helps us to improve the bound in Proposition III.3 as follows.

Corollary III.4. *The optimal value of $\kappa_i(t)$ and $\eta_i(t)$ are such that the error variance at node i satisfies*

$$\mathbb{E}\,(e_i(t) - \mathbb{E}\,e_i(t))^2 \leq \frac{\sigma^2}{M_i + \left(\sum_{j \in \mathcal{N}_i} M_j^{-1} + (M_i \psi_i)^{-1/2}\right)^{-1}} \, ,$$

$t \in \mathbb{N}_0 \setminus \{0\}$.

Proof: Using the result in Proposition III.3 we have that

$$\operatorname{tr} \Gamma_i(t-1) = \sum_{j \in \mathcal{N}_i} \mathbb{E}\,(e_{i_j}(t-1) - \mathbb{E}\,e_{i_j}(t-1))^2 \leq \sum_{j \in \mathcal{N}_i} \frac{\sigma^2}{M_j} \, .$$

Thus

$$\begin{aligned}\lambda_{\max}(\Gamma_i(t-1) + \xi_i I) &\leq \sum_{j \in \mathcal{N}_i} \frac{\sigma^2}{M_j} + \\ &\quad + \max\left(0, \frac{\sigma^2}{\sqrt{M_i \psi_i}} - \lambda_{\min}(\Gamma_i(t-1))\right) \\ &\leq \sum_{j \in \mathcal{N}_i} \frac{\sigma^2}{M_j} + \frac{\sigma^2}{\sqrt{M_i \psi_i}} \, ,\end{aligned}$$

where we used the bound on ξ determined in Proposition III.2. Since

$$\mathbb{1}^T (\Gamma_i(t-1) + \xi_i I)^{-1} \, \mathbb{1} \geq \frac{1}{\lambda_{\max}(\Gamma_i(t-1) + \xi_i I) M_i}$$

we have that

$$\begin{aligned}\mathbb{E}\,(e_i(t) - \mathbb{E}\,e_i(t))^2 &\leq \frac{\sigma^2}{M_i + \sigma^2 \, \mathbb{1}^T (\Gamma_i(t-1) + \xi_i I)^{-1} \, \mathbb{1}} \\ &\leq \frac{\sigma^2}{M_i + \left(\sum_{j \in \mathcal{N}_i} M_j^{-1} + (M_i \psi_i)^{-1/2}\right)^{-1}} \, .\end{aligned}$$

■

D. Distributed Computation of Constraints

The choice of the constants ψ_i, $i = 1, \ldots, N$, in the local constraint of problem $\mathcal{P}_2$ is critical for the performance of the distributed estimator. Next we discuss how to compute good values of ψ_i. The intuition is that ψ_i has to be upper bounded to guarantee the estimation error to converge, but ψ_i should not be too small in order to put large enough weights on the estimates. Indeed, from the proof of Proposition III.2 we see that if ψ_i is large then the Lagrangian multiplier ξ_i is small, since it must lie in the in interval $\max\left[0, \sigma^2/\sqrt{M_i \psi_i} - \lambda_{\min}(\Gamma_i(t-1))\right]$. From Proposition III.3 (and Corollary III.4) it is clear that the estimation error variance at the node i decreases as ξ_i decreases. Thus the larger the value of ψ_i the lower the error variance.

The set of nonlinear equations in Proposition III.1 provides a tool to determine suitable values of ψ_i that guarantee stability. Since we are interested in determining the largest solution of the nonlinear equations, we consider the following optimization problem:

$$\max_{\psi_1,\ldots,\psi_N} \quad \sum_{i=1}^{N} \psi_i \tag{III.11}$$

$$\text{s.t.} \quad \psi_i + \sqrt{\psi_i} \cdot \sum_{j\in\Theta_i} \sqrt{\alpha_{i,j}^{(i)}\alpha_{i,j}^{(j)}\psi_j} \le f^2(\Delta,\Upsilon) \tag{III.12}$$

$$\psi_i \ge 0\,,$$

with $i = 1,\ldots,N$. It is possible to show that previous problem has a unique solution, which is the solution to the following equations:

$$\psi_i + \sqrt{\psi_i} \sum_{j\in\Theta_i} \sqrt{\alpha_{i,j}^{(i)}\alpha_{i,j}^{(j)}\psi_j} = f^2(\Delta,\Upsilon) \qquad i = 1,\ldots,N\,. \tag{III.13}$$

Clearly the solution of such system of nonlinear equations is interesting in our setup if it can be solved in a decentralized fashion. The fact that in (III.13) only information from neighboring nodes is required, and not of the entire network, allows us to develop a decentralized algorithm to compute the solution. Following [7, Pag.181–191], we consider the iterative algorithm

$$\psi(t+1) = T(\psi(t)) = (T_1(\psi(t)),\ldots,T_N(\psi(t))) \tag{III.14}$$

with initial condition $\psi(0) > 0$ and with $T : \mathbb{R}_+^N \to R_+^N$ such that

$$T_i(\psi(t)) = \frac{1}{4}\left[\sqrt{\left(\sum_{j\in\Theta_i}\sqrt{\alpha_{i,j}^{(i)}\alpha_{i,j}^{(j)}\psi_j(t)}\right)^2 + 4f^2(\Delta,\Upsilon)} - \sum_{j\in\Theta_i}\sqrt{\alpha_{i,j}^{(i)}\alpha_{i,j}^{(j)}\psi_j(t)}\right]^2, \tag{III.15}$$

where $T_i(\psi)$ is obtained by solving (III.13) with respect to ψ_i. It is not difficult to show that $T_i(\psi)$ is a contractive function. The component solution method in [7] ensures that the fixed point solution at which the iteration converges is the solution of the nonlinear Equations (III.13). The computation of the iteration (III.14) can be done distributively. Note that node i does not need to know the thresholds ψ_j, $j \neq i$, of all the other nodes in the network, but those which concur in the definition of $T_i(\psi)$, i.e., ψ_j that are associated to the nodes of the set Θ_i. Thresholds corresponding to nodes at two hops from node i can be communicate to such node through its neighbors, with little communication overhead. Notice that, the computation of the thresholds and the associated communication takes place before the nodes start to track the signal $d(t)$. Notice also that the convergence rate of the component solution method for block contraction converges geometrically to the fixed point.

E. Estimation of Error Covariance

Estimating the error covariance matrix is in general hard for the problem considered in this paper, because the estimator is a time-varying system and the stochastic process x, and thus e, is not stationary. However, if we consider the signals in the quasi-stationary sense, estimation based on samples guarantees to give good results. We have the following definition.

Definition III.5 ([30, pag. 34])**.** *A signal $s(t) : \mathbb{R} \to \mathbb{R}$ is said to be quasi-stationary if there exists a positive constant C and a function $R_s : \mathbb{R} \to \mathbb{R}$, such that s fulfills the following conditions*

(i) $\mathbb{E}\, s(t) = m_s(t)$, $|m_s(t)| \le C$ *for all* t

(ii) $\mathbb{E}\, s(t)s(r) = R_s(t,r)$, $|R_s(t,r)| \le C$ *for all* t *and*

$$\lim_{N\to+\infty} \frac{1}{N}\sum_{t=1}^{N} R_s(t,t-\tau) = R_s(\tau)$$

for all τ.

It is easy to see that the time-varying linear system (II.2) is uniformly bounded-input bounded-output stable [31, pag. 509]. If a quasi-stationary signal is the input of such system, then its output is also quasi-stationary [32]. In our case, the measurement signal $u(t)$ is (component-wise) stationary and ergodic and thus also quasi-stationary. This implies that also $x(t)$ is quasi-stationary, since it is the output of a uniformly exponentially stable time-varying linear system. Thus, we estimate the error covariance using the sample covariance. Specifically, we have that the mean $\mathbb{E}\,\epsilon_i = m_{\epsilon_i}(t)$ and covariance $\Gamma_i(t)$ can be estimated from samples as

$$\hat{m}_{\epsilon_i}(t) = \frac{1}{t}\sum_{\tau=0}^{t} \hat{\epsilon}_i(\tau) \tag{III.16}$$

$$\hat{\Gamma}_i(\tau) = \frac{1}{\tau}\sum_{\tau=0}^{t} (\hat{\epsilon}_i(\tau) - \hat{m}_{\epsilon_i}(\tau))(\hat{\epsilon}_i(\tau) - \hat{m}_{\epsilon_i}(\tau))^T\,, \tag{III.17}$$

where $\hat{\epsilon}_i(t)$ is the an estimate of the error. Thus the problem reduces to design an estimator of $\epsilon_i(t)$. Node i has estimates $x_{i_j}(t)$ and measurements $u_{i_j}(t)$, $i_j \in \mathcal{N}_i$, available. Let $x^{(i)}(t)$ and $u^{(i)}(t)$ denote the collection of all these variables. We can model this data set as

$$x^{(i)}(t) = d(t)\,\mathbb{1} + \beta(t) + w(t)\,, \qquad u^{(i)}(t) = d(t)\,\mathbb{1} + v(t)\,,$$

where $\beta(t) \in \mathbb{R}^{M_i}$ models the bias of the estimates and $w(t)$ is zero-mean Gaussian noise modelling the variance of the estimator. Summarizing: node i has available $2M_i$ data values in which half of the data are corrupted by a small biased term $\beta(t)$ and a low variance noise $w(t)$ and the other half is corrupted by zero-mean Gaussian noise $v(t)$ with high variance. It is clear that using only $u^{(i)}(t)$ to generate an estimate $\hat{d}(t)$ of $d(t)$, which could then be used to estimate $\hat{\epsilon}_i(t) = x^{(i)}(t) - \hat{d}(t)\,\mathbb{1}$, would have the advantage of being unbiased. However, its covariance is rather large since M_i is typically small. Thus, using only measurements to estimate $d(t)$ yield to an over-estimate of the error, which results in poor performance. On the other hand, using only $x^{(i)}(t)$ we obtain an under-estimate of the error. This makes the weights $\eta_i(t)$ rapidly vanish and the signal measurements are discarded, thus tracking becomes impossible. From these arguments, in order to use both $x^i(t)$ and $u^i(t)$ we pose a linear least square problem as follows:

$$\begin{aligned}\min_{\hat{d},\hat{\beta}} &\quad \left\| \begin{pmatrix} x^i \\ u^i \end{pmatrix} - A \begin{pmatrix} \hat{d} \\ \hat{\beta} \end{pmatrix} \right\|^2 \\ \text{s.t.} &\quad \left\| B \begin{pmatrix} \hat{d} & \hat{\beta} \end{pmatrix} \right\|^2 \leq \rho \end{aligned}$$

with $A \in \mathbb{R}^{2M_i \times M_i+1}$ and $B \in \mathbb{R}^{M_i \times M_i+1}$

$$A = \begin{pmatrix} \mathbb{1} & I \\ \mathbb{1} & 0 \end{pmatrix}, \qquad B = \begin{pmatrix} 0 & I \end{pmatrix},$$

and ρ being the maxim value of the squared norm of the bias. However, the previous problem is very difficult to solve in a closed form, as it is a Quadratically Constrained Quadratic Problem and it typically requires heavy numerical algorithms to find the solution, such the transformation into a SDP problem [29, pag. 653]. Notice also that, in general, the value of ρ is not known in advance, being it a maximum value of the cumulative bias of M_i nodes. We thus consider the following regularized problem

$$\min_{\hat{d},\hat{\beta}} \quad \left\| \begin{pmatrix} x^i \\ u^i \end{pmatrix} - A \begin{pmatrix} \hat{d} \\ \hat{\beta} \end{pmatrix} \right\|^2 + \nu \left\| B \begin{pmatrix} \hat{d} \\ \hat{\beta} \end{pmatrix} \right\|^2 \tag{III.18}$$

where $\nu > 0$ is a parameter whose choice is typically rather difficult.

The solution of (III.18) is

$$(\hat{d}, \hat{\beta})^T = (x^i, u^i)^T A \left(A^T A + \nu B^T B\right)^{-1} .$$

The inverse of the matrix in the previous equation can be computed in closed form using the following result:

Proposition III.6. *If $\nu > 0$ then*

$$\begin{aligned}&\left(A^T A + \nu B^T B\right)^{-1} = \\ &= \frac{1}{M_i(1+2\nu)} \begin{pmatrix} 1+\nu & -\mathbb{1}^T \\ -\mathbb{1} & \dfrac{M_i(1+2\nu)I + \mathbb{1}\mathbb{1}^T}{1+\nu} \end{pmatrix}\end{aligned} \tag{III.19}$$

Proof: By Schur's complement we obtain

$$\begin{aligned}&\left(A^T A + \nu B^T B\right)^{-1} = \\ &= \begin{pmatrix} \left(2M_i - \dfrac{M_i}{1+\nu}\right)^{-1} & \mathbb{1}^T(\mathbb{1}\mathbb{1}^T - 2M_i(1+\nu)I)^{-1} \\ (\mathbb{1}\mathbb{1}^T - 2M_i(1+\nu)I)^{-1}\mathbb{1} & \left((1+\nu)I - \dfrac{\mathbb{1}\mathbb{1}^T}{2M_i}\right)^{-1} \end{pmatrix}\end{aligned}$$

From [33] it follows that

$$\left((1+\nu)I - \frac{\mathbb{1}\mathbb{1}^T}{2M_i}\right)^{-1} = \frac{I}{1+\nu} + \frac{\mathbb{1}\,\mathbb{1}^T}{M_i(1+2\nu)(1+\nu)},$$

It is easy from here to show that the resulting matrix is (III.19). ■

Since we are interested in estimating $\epsilon_i(t) = x(t) - d(t)\,\mathbb{1}$ we observe that such an estimate is given by $\hat{\beta}$. From the solution of (III.18), we have

$$\hat{\beta} = \frac{x^i}{1+\nu} - \frac{\nu\,\mathbb{1}^T x^i + (1+\nu)\,\mathbb{1}^T u^i}{M_i(1+2\nu)(1+\nu)}\,\mathbb{1} \tag{III.20}$$

For the choice of the parameter ν we propose to use the Generalized Cross-Validation (GCV) method [34]. This consists in choosing ν as

$$\nu = \arg\min \frac{\|(A^T A + \nu B^T B)^{-1} A^T (x^i, u^i)^T\|}{\operatorname{tr}(A^T A + \nu B^T B)^{-1}} .$$

Typically the GCV approach is computationally expensive since the trace of the matrix $(A^T A + \nu B^T B)^{-1}$ is difficult to compute, but in our case we have a closed form representation of the matrix, and thus the computation is not difficult. However, it might be computationally difficult to carry out the minimization. Observing that

$$\begin{aligned}\nu &= \arg\min \frac{\|(A^T A + \nu B^T B)^{-1} A^T (x^i, u^i)^T\|}{\operatorname{tr}(A^T A + \nu B^T B)^{-1}} \\ &\leq \arg\min \frac{\|(A^T A + \nu B^T B)^{-1} A^T\|}{\operatorname{tr}(A^T A + \nu B^T B)^{-1}} \|(x^i, u^i)^T\| .\end{aligned} \tag{III.21}$$

a sub-optimal value of ν can be computed solving the right hand side of (III.21). Notice that the first term in the right hand side of (III.21) is a function of ν that can be computed off-line and stored in a look-up table at the node. Then, for different data, the problem becomes that of searching in the table.

Using (III.20) with the parameter ν computed from (III.21) we can then estimate the error mean and covariance matrix applying (III.16) and (III.17), respectively.

IV. Implementation and Numerical Results

This section presents the estimator structure and the algorithmic implementation followed by some numerical results.

A. Estimator Structure and Implementation

Figure 1 summarizes the structure of the estimator implemented in each node. The estimator has a cascade structure with two sub-systems: the one to the left is an adaptive filter that produces the estimate of d; the one to the right computes an estimate of the error covariance matrix Γ_i. In the following, we discuss in some detail a pseudo-code implementation of the blocks in the figure. The estimator is presented as Algorithm 1. Initially, the distributed computation of the threshold is performed (lines 1–8): node i updates its threshold ψ_i until a given precision ϖ is reached. In the computations of ψ_i, we chose $\alpha_{i,j}^{(i)} = |\mathcal{N}_j \cap \mathcal{N}_i|/(M_i - 1)$ and $\alpha_{i,j}^{(j)} = |\mathcal{N}_j \cap \mathcal{N}_i|/(M_j - 1)$. This works well in practice because k_{ii_r}, $i_r = 1, \ldots, M_i$, are of similar magnitude. Indeed, the stability of the average of the estimation error established in Section II-B, and the bounds on the error variance in Section III-C, ensure that estimates among nodes have similar performance.

Numerical results show that that the while-loop (lines 4–8) converges after about 10–20 iterations.

The estimators for the local mean estimation error and the local covariance matrix are then initialized (lines 9–10). The main loop of the estimator is lines 13–24. Lines 14–19 are related to the left subsystem of Figure 1. The optimal weights are computed using Equations (III.9) and (III.10) (lines 17–18). Notice that the optimal Lagrangian multiplier ξ_i is computed

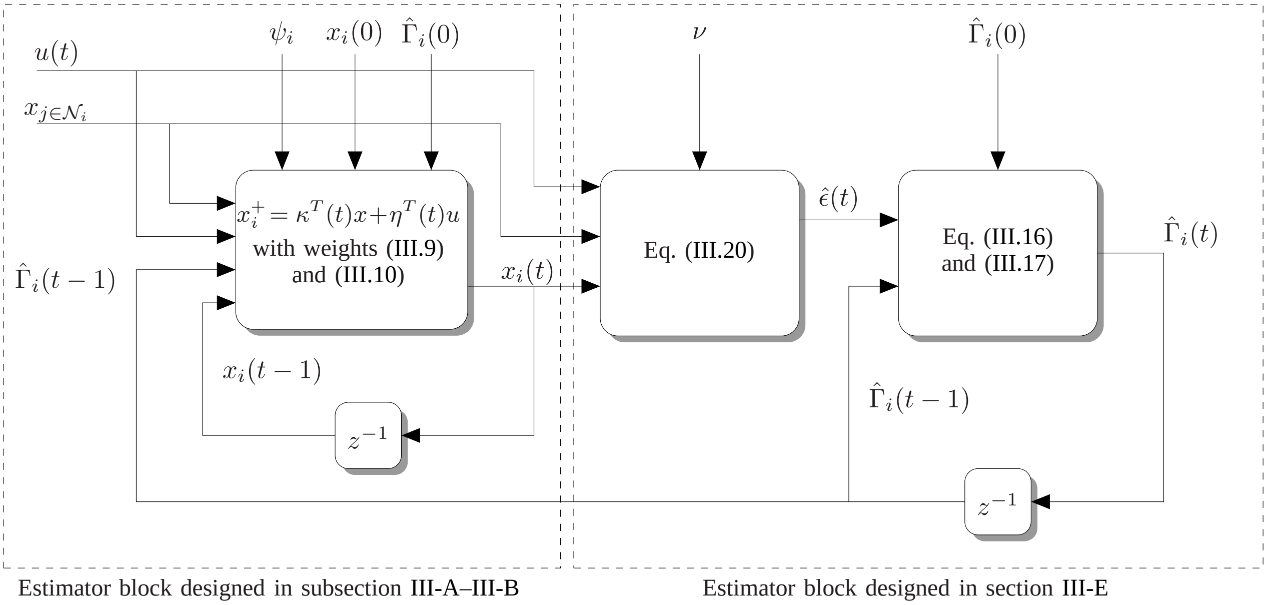


Fig. 1. Block diagram of the proposed estimator. It consists of two subsystems in a cascade coupling. The subsystem to the left is an adaptive filter that produces the estimate of $d(t)$ with small variance and bias. The subsystem to the right estimates the error covariance matrix.

Algorithm 1 Estimation algorithm for node i

```
1.  $t := 0$
2.  $\psi_i(t-1) = 0$
3.  $\psi_i(t) = 1/M_i$
4.  while $|\psi_i(t) - \psi_i(t-1)| \geq \varpi = 10^{-10}$ do
5.      $\psi_i(t+1) = T_i(\psi(t))$
6.      collect thresholds from nodes in $\Theta_i$
7.      $t := t+1$
8.  end while
9.  $t := 0$
10. $\hat{m}_{\epsilon_i}(t) := 0$
11. $\hat{\Gamma}_i(t) := \sigma^2 I$
12. $x_i(t) := u_i(t)$
13. while forever do
14.     $M_i := |\mathcal{N}_i|$
15.     $t := t+1$
16.     $\xi_i = \text{bisection}\left(\max[0, \sigma^2/\sqrt{M_i psi_i} - \lambda_{\min}(\Gamma_i(t-1))]\right)$
17.     $\kappa_i(t) := \dfrac{\sigma^2(\hat{\Gamma}_i(t-1) + \xi_i I)^{-1}\mathbb{1}}{M_i + \sigma^2\,\mathbb{1}^T(\hat{\Gamma}_i(t-1) + \xi_i I)^{-1}\mathbb{1}}$
18.     $\eta_i(t) := \dfrac{\mathbb{1}}{M_i + \sigma^2\,\mathbb{1}^T(\hat{\Gamma}_i(t-1) + \xi_i I)^{-1}\mathbb{1}}$
19.     $x_i(t) := \sum_{j\in\mathcal{N}_i} \kappa_{i_j}(t)x_j(t-1) + \sum_{j\in\mathcal{N}_i} \eta_{i_j}(t)u_j(t)$
20.     $\hat{\beta} := \dfrac{x^i}{1+\nu} - \dfrac{\nu\,\mathbb{1}^T x^i + (1+\nu)\,\mathbb{1}^T u^i}{M_i(1+2\nu)(1+\nu)}\mathbb{1}$
21.     $\hat{\epsilon}_i := \hat{\beta}$
22.     $\hat{m}_{\epsilon_i}(t) := \dfrac{t-1}{t}\hat{m}_{\epsilon_i}(t-1) + \dfrac{1}{t}\hat{\epsilon}_i(t)$
23.     $\hat{\Gamma}_i(t) := \dfrac{t-1}{t}\hat{\Gamma}_i(t-1) + \dfrac{1}{t}(\hat{\epsilon}_i(t) - \hat{m}_{\epsilon_i}(t))(\hat{\epsilon}_i(t) - \hat{m}_{\epsilon_i}(t))^T$
24. end while
```

using the function `bisection` which takes as argument the interval $[0, \max(0, \sigma^2/\sqrt{M_i\psi_i} - \lambda_{\min}(\Gamma_i(t-1)))]$ where the optimal value lays. Notice that, if the nodes have limited computational power, so that the minimum eigenvalue of the matrix $\Gamma_i(t-1)$ cannot be exactly computed, an upper-bound based on Geršgorin can be used instead. The estimate of $d(t)$ is computed in line 19. Lines 20–23 are related to the right subsystem of Figure 1. These lines implement the error covariance estimation by solving the constrained least-squares minimization problem described in subsection III-E. Sample mean and covariance of the estimation error are updated in lines 22–23. These formulas correspond to recursive implementation of (III.16) and (III.17).

Let us comment on the inversions of the estimated error covariance matrix $\hat{\Gamma}_i$ in lines 17–18. In general, the dimension of $\hat{\Gamma}_i$ is not a problem because we consider cases when the number of neighbors is small. Precautions have still to be taken, because even though the error covariance matrix Γ_i is always positive definite, its estimate $\hat{\Gamma}_i$ may not be positive definite before sufficient statistics are collected. In our implementation, we use heuristics to ensure that $\hat{\Gamma}_i$ is positive definite.

B. Numerical Results

Numerical simulations have been carried out in order to validate performance of the proposed distributed estimator. We compare the our estimator with some similar estimators related to the literature. We consider the following five estimators:

E_1: $K = H = (I - L)/2$ where L is the Laplacian matrix associated to the graph $\mathcal{G}$.

E_2: $K = 0$ and $H = [h_{ij}]$ with $h_{ij} = 1/M_i$ if node i and j are connected, and $h_{ij} = 0$ otherwise. Thus, the updated estimate is the average of the measurements.

E_3: $K = [k_{ij}]$, where $k_{ii} = 1/2M_i$, $k_{ij} = 1/M_i$ if node i and j are connected, $k_{ij} = 0$ otherwise, whereas $H = [h_{ij}]$ with $H_{ii} = 1/2M_i$, and $h_{ij} = 0$ elsewhere. This is the average of the old estimates and node's single measurement.

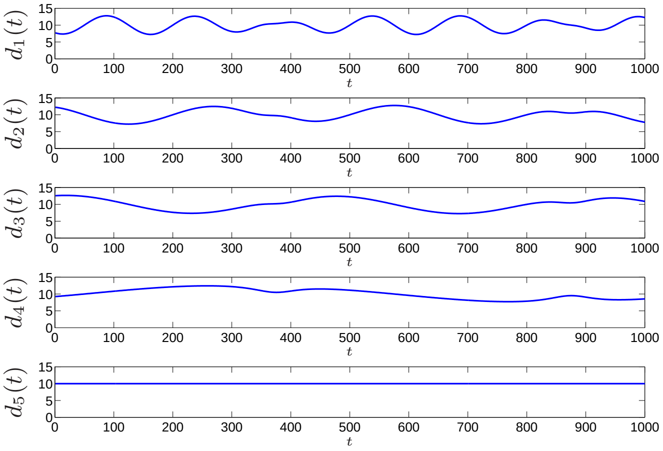
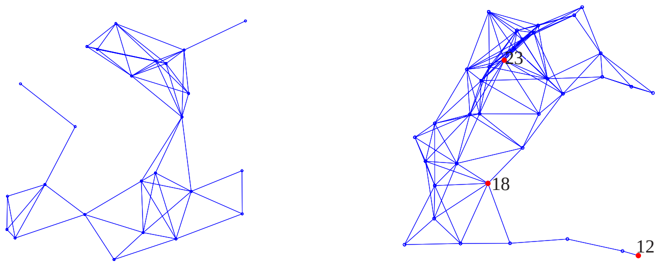


Fig. 2. Test signals used in the numerical simulations. Note that $d_2, \ldots, d_5$ are obtained from $d_1(t)$ changing the frequency.



(a) (b)

Fig. 3. Topology of the networks with $N = 25$ nodes (on the left) and $N = 35$ (on the right) used in the simulations. For the network with $N = 35$, three nodes are highlighted, corresponding to the identifier 12, 18, and 23. They have the following number of neighbors: $|\mathcal{N}_{12}| = 2$, $|\mathcal{N}_{18}| = 8$, and $|\mathcal{N}_{23}| = 15$. The node with maximum degree in all the network is node 23.

E_4: $K = H$ with $k_{ij} = 1/2M_i$ if node i and j are connected, and $i = j$. The updated estimate is the average of the old estimates and all local measurements.

E_p: The estimator proposed in this paper.

The estimators $E_1, \ldots, E_4$ are based on various heuristics. They are related to proposals in the literature, e.g., E_1 uses filter coefficients given by the Laplacian matrix, cf., [11]–[13]. It is important to note, however, that in general the weights based on Laplacian do not ensure the minimization of the variance of the estimation error.

Figure 2 shows a set of test signals $d_1, \ldots, d_5$ that we have used to benchmark the estimators. Note that the signals differ only in their frequency content.

We suppose that we know a bound for Δ based on these signals. We have set Δ to be 10% larger than its actual value for each signal. We have chosen the desired average SNR to $\Upsilon = 10$, see Section II.

We consider two networks $\mathcal{G}_{25}$ and $\mathcal{G}_{35}$ with $N = 25$ and $N = 35$ nodes, respectively, see Figure 3. These networks are obtained by distributing the nodes randomly over a squared area of size $N/3$. The graph is then obtained by letting two nodes communicate if their relative distance is less than $\sqrt{N}$.

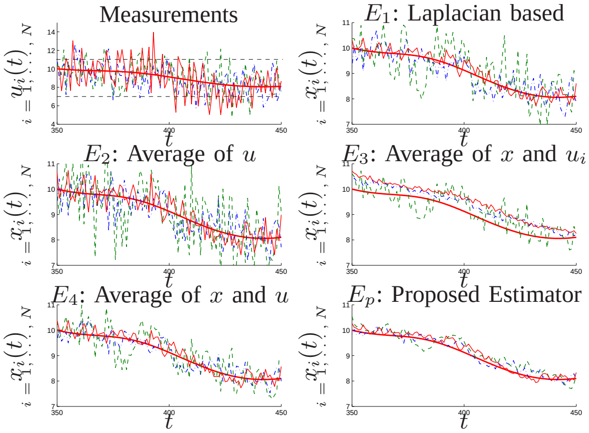


Fig. 5. Zoom of some of the curves in Figure 2. In particular, we plot the measurements and estimates of the nodes 12, 18 and 23 having the minimum degree, degree equal to the average degree of the network, and maximum, respectively (see Figure 3). In thick solid curve is shown the signal $d(t)$. The dashed curves show the measurement and estimate at node 12, in dash-dotted those at node 18 and the solid curves show those at node 23. The horizontal lines in the the top-left figure are the interval within which the estimates variate. We chose to have different scales to make more clear the estimation process.

We discuss in detail the distributed estimator over the network network $\mathcal{G}_{35}$. Measurements and estimates for all nodes are shown in Figure 4 for d_2. Clearly, the measurements are quite noisy, and in particular $\sigma^2 = 1.5$. All estimators, $E_1, \ldots, E_4$ and E_p, are able to track the signal, but the quality of the estimates are varying quite a bit. It is evident that E_1 and E_2 give the worst estimates, while E_p performs best. The relative performance between $E_1, \ldots, E_4$ is rather obvious given how their estimate is constructed, e.g., E_2 simply take the average of the measurements while E_4 averages over both measurements and estimates. By choosing the weights appropriately, we see that the proposed estimator E_p gives substantially lower estimation variance. Figure 5 shows a zoom of Figure 4 for the time interval $[350, 450]$. The figure compares the measurements and estimates of the three nodes highlighted in Figure 3. These nodes represent the node with minimum connectivity (dashed curve), average connectivity (dash-dotted curve) and maximum connectivity (solid curve). The thick line correspond to d_2. Note that the node with low connectivity is not following d_2 very well. We also see that the estimate produced by E_3 has a quite substantial bias. In general, we have observed through extensive simulations that E_3 work well for low-frequency signals to track, whereas E_4 works better for signal with higher frequency. Numerical studies of various networks confirm the type of behaviors we see in Figures 4 and 5. We summarize a set of these simulations next. In order to study performance of the estimators, we consider the mean square error of the estimates of each node. Each estimator has an initial transition phase, so to remove that we compute the mean square error after 70 steps. We average the mean square error over all nodes of the network. The average we obtain, we denote MSE. We define an improvement factor of our

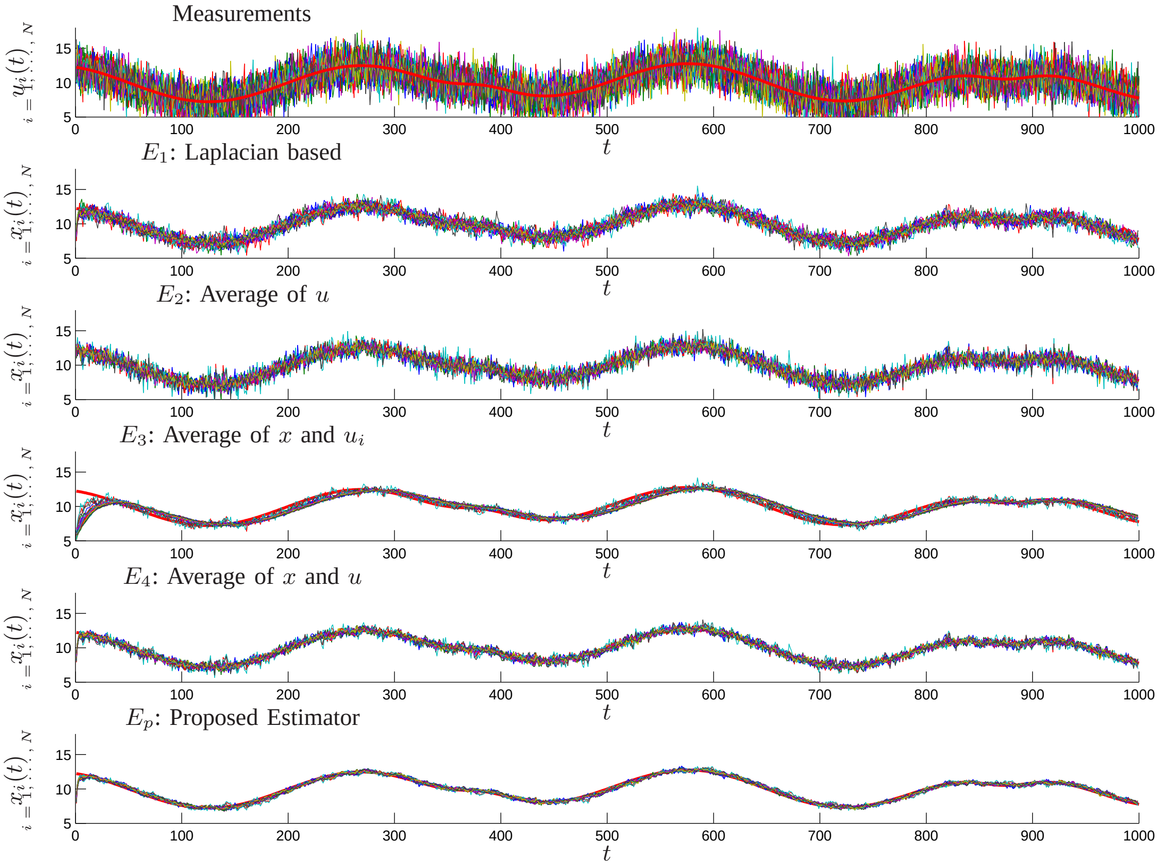


Fig. 4. Plots showing $N = 35$ realizations of the measurements and estimates at each node for each estimator and for the signal $d_2(t)$.

estimator compared to the estimators $E_1, \dots, E_4$ as

$$\mu_i = \frac{\text{MSE}(E_i) - \text{MSE}(E_p)}{\text{MSE}(E_i)}, \qquad i = 1, \dots, 4.$$

Table I reports MSE and μ_i for $\mathcal{G}_{25}$ and $\mathcal{G}_{35}$, and for all test signals $d_1, \dots, d_5$. Table I shows that E_p outperforms all other estimators in all cases. Specifically, E_p exhibits performance improvement from a minimum of 15%, in the case of network $\mathcal{G}_{35}$ nodes and d_1, up to 85% in the same network and d_4. The Laplacian-based estimator E_1 has poor performance. Note that E_1 typically puts more weights to the local estimate and measurement of a node than to the estimates and measurements received from its neighbors. Because the network is homogenous, this yields poor performance. When the frequency of the test signal is high, the performance improvement is not substantial for E_p. The reason is found in the fact that our estimator tries to keep low the bias, but paying the price of an higher variance of the estimation error. Finally, in Figure 6 we plotted the maximum singular value of the matrix $K(t)$ of E_p as a function of time for the simulations in Table I. This plot verifies the validity of our design approach.

V. Conclusions

In this paper, we have presented a fully distributed minimum variance estimator for sensor networks. The purpose of such estimator is accurate tracking of a time varying signal using noisy measurements. A mathematical framework is proposed to design a filter, which runs locally in each node. It only requires a cooperation among neighboring nodes. In order to obtain a minimum variance estimator, we started from a centralized optimization problem, and then we converted it into a decentralized problem transforming global constraints into distributed ones. The filter structure is composed by a cascade of two blocks: the first block computes the estimator coefficients at each time instance, and the second block estimates the error covariance matrix needed, by the first block, at next step. The estimator coefficients are designed such that the local behavior of a node ensures the overall estimation process to be stable. We showed that the distributed estimator is stable, with mean and variance of the estimation error bounded. Numerical results proved that our filter outperforms existing solutions proposed in literature, as well as other heuristic solutions. Future work includes stability analysis of the filter with respect to packet losses, and experimental validation.

	$d_1(t)$, Δ = 0.118			
Type of Estimator	N = 25		N = 35	
	MSE	μ_i	MSE	μ_i
E_1: Laplacian Based	0.445	32.7 %	0.456	41.1 %
E_2: Average of u	0.659	54.5 %	0.567	52.6 %
E_3: Average of x and u_i	0.665	54.9 %	0.872	69.2 %
E_4: Average of x and u	0.372	19.6 %	0.316	15.1 %
E_p: Proposed Estimator	0.299		0.270	
	$d_2(t)$ Δ = 0.059			
Type of Estimator	N = 25		N = 35	
	MSE	μ_i	MSE	μ_i
E_1: Laplacian Based	0.445	48.8 %	0.454	52.9 %
E_2: Average of u	0.658	65.4 %	0.565	62.2 %
E_3: Average of x and u_i	0.389	41.6 %	0.539	60.3 %
E_4: Average of x and u	0.366	37.7 %	0.313	31.6 %
E_p: Proposed Estimator	0.228		0.214	
	$d_3(t)$, Δ = 0.040			
Type of Estimator	N = 25		N = 35	
	MSE	μ_i	MSE	μ_i
E_1: Laplacian Based	0.442	49.0 %	0.457	59.4 %
E_2: Average of u	0.662	65.8 %	0.572	67.6 %
E_3: Average of x and u_i	0.330	31.5 %	0.391	52.5 %
E_4: Average of x and u	0.367	38.4 %	0.315	41.2 %
E_p: Proposed Estimator	0.224		0.186	
	$d_4(t)$, Δ = 0.033			
Type of Estimator	N = 25		N = 35	
	MSE	μ_i	MSE	μ_i
E_1: Laplacian Based	0.444	61.8 %	0.447	66.4 %
E_2: Average of u	0.663	74.3 %	0.564	73.4 %
E_3: Average of x and u_i	0.240	28.8 %	0.256	41.3 %
E_4: Average of x and u	0.370	53.8 %	0.308	51.2 %
E_p: Proposed Estimator	0.170		0.150	
	$d_5(t)$, Δ = 0.000			
Type of Estimator	N = 25		N = 35	
	MSE	μ_i	MSE	μ_i
E_1: Laplacian Based	0.436	73.9 %	0.446	81 %
E_2: Average of u	0.656	83.8 %	0.560	85 %
E_3: Average of x and u_i	0.195	41.6 %	0.150	44 %
E_4: Average of x and u	0.358	68.2 %	0.305	72 %
E_p: Proposed Estimator	0.114		0.080	

TABLE I
Performance (MSE and improvement factor) of the proposed estimator versus some chosen heuristics. The table has been divided into five sub-tables, one for each test signal $d_1(t)$-$d_5(t)$, and for different cases of the network size.

Acknowledgment

The authors would like to thank Håkan Hjalmarsson, Björn Johansson, Mikael Johansson and Bo Wahlberg for fruitful discussions.

References

[1] A. Speranzon, C. Fischione, and K. H. Johansson, “Distributed and collaborative estimation over wireless sensor networks,” in *IEEE Conference on Decision and Control*, 2006.

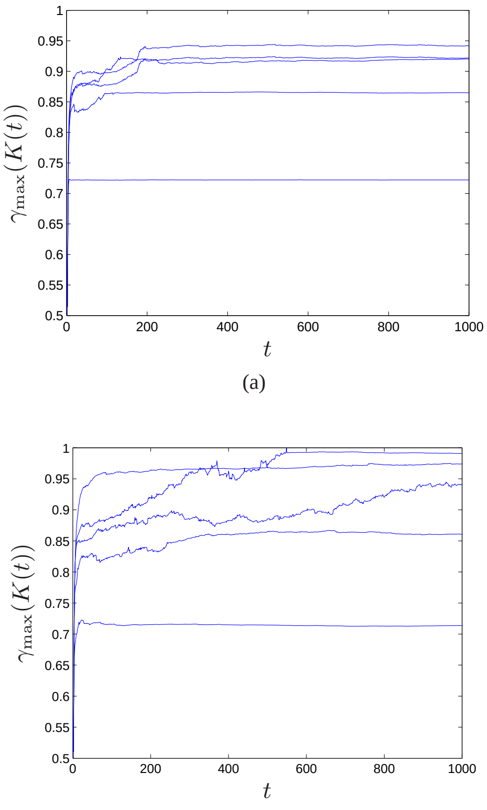
[2] ——, “A distributed estimation algorithm for tracking over wireless sensor networks,” in *IEEE International Conference on Communication*, 2007.

[3] D. Estrin, R. Govindan, J. Eidemann, and S. Kumar, “Next century challenges: Scalable coordination in sensor networks,” in *Proceedings of the ACM/IEEE International Conference on Mobile Computing and Networking*, 1999.

[4] H. Gharavi and P. R. Kumar, Eds., *Proceedings of IEEE: Special Issue on Sensor Networks and Applications*, vol. 91, 2003.

[5] Z.-Q. Luo, M. Gatspar, J. Liu, and A. Swami, Eds., *IEEE Signal Processing Magazine:Special Issue on Distributed Signal Processing in Sensor Networks*. IEEE, 2006.

[6] H. Gharavi and P. R. Kumar, “Toward a theory of in-network computation in wireless sensor networks,” *IEEE Communication Magazine*, pp. 97–107, 2006.



(a)

(b)

Fig. 6. Values of the maximum singular value $\gamma_{\max}(K(t))$ for the set of simulations summarized in Table I. On the left the case of $N = 25$ and on the right $N = 35$. As it can be seen $\gamma_{\max}(K(t)) < 1$ in all cases. Low values of $\gamma_{\max}(K(t))$ correspond to higher values of Δ (namely fast signals).

[7] D. P. Bertsekas and J. N. Tsitsiklis, *Parallel and Distributed Computation: Numerical Methods*. Athena Scientific, 1997.

[8] A. Jadbabaie, J. Lin, and A. S. Morse, “Coordination of groups of mobile autonomous agents using nearest neighbor rules,” *IEEE Transactions on Automatic Control*, vol. 48, no. 6, pp. 988–1001, 2003.

[9] R. Olfati-Saber and R. M. Murray, “Consensus problems in networks of agents with switching topology and time-delays,” *IEEE Transactions on Automatic Control*, 2004.

[10] R. Carli, F. Fagnani, A. Speranzon, and S. Zampieri, “Communication constraints in the average consensus problem,” *Automatica*, 2008, to Appear.

[11] L. Xiao, S. Boyd, and S. Lall, “A scheme for robust distributed sensor fusion based on average consensus,” *In Proceedings of IEEE IPSN*, 2005.

[12] D. P. Spanos, R. Olfati-Saber, and R. M. Murray, “Approximate distributed Kalman filtering in sensor networks with quantifiable performance,” in *IEEE Conference on Decision and Control*, 2005.

[13] R. Olfati-Saber and J. S. Shamma, “Consensus filters for sensor networks and distributed sensor fusion,” *In Proceedings of IEEE Conference on Decision and Control*, 2005.

[14] L. Xiao, S. Boyd, and S. J. Kim, “Distributed average consensus with least-mean-square deviation,” *Submitted to Journal of Parallel and Distributed Computing*, 2006.

[15] S. B. L. Xiao, “Fast linear iterations for distributed averaging,” *System Control Letter*, 2004.

[16] A. Pikovsky, M. Rosenblum, and J. Kurths, *Synchronization : A Universal Concept in Nonlinear Sciences*. Cambridge University Press, 2003.

[17] Y.-W. Hong, L. Cheow, and A. Scaglione, “A simple method to reach detection consensus in massively distributed sensor networks,” in *IEEE International Symposium on Information Theory*, 2004.

[18] S. Barbarossa and F. Celano, “Self-organizing sensor networks designed as a population of mutually coupled oscillators,” in *IEEE Signal Processing Advances in Wireless Communications*, 2005.

[19] S. Barbarossa and G. Scutari, “Decentralized maximum likelihood estimation for sensor networks composed of nonlinearly coupled dynamical system,” *IEEE Transactions on Signal Processing*, 2007.

[20] F. Zhao, J. Liu, L. Guibas, and J. Reich, “Collaborative signal and information processing: an information-directed approach,” *Proceedings of the IEEE*, vol. 91, no. 8, 2003.

[21] J. Liu, J. Reich, and F. Zhao, “Collaborative in-network processing for target tracking,” *Eurasip Journal on Applied Signal Processin*, no. 4, pp. 378–391, 2003.

[22] Y. Shang, W. Ruml, Y. Zhang, and M. Fromherz, “Localization from connectivity in sensor networks,” *IEEE Transaction on Parallel and Distributed System, Vol. 15, No. 11*, 2004.

[23] G. Di Stefano, F. Graziosi, and F. Santucci, “Distributed positioning algorithm for ad-hoc networks,” *International Workshop on Ultra Wideband Systems (IWUWBS)*, 2003.

[24] A. Giridhar and P. Kumar, “Distributed clock synchronization over wireless networks: Algorithm and analysis,” in *45th IEEE Conference on Decision and Control*, 2006.

[25] P. Alriksson and A. Rantzer, “Distributed Kalman filtering using weighted averaging,” in *In Proceedings of the 17th International Symposium on Mathematical Theory of Networks and Systems*, 2006.

[26] L.-M. S. C. of the IEEE Computer Society, “Wireless lan medium access control (MAC) and physical layer specification,” IEEE, New York, NY, USA, IEEE std 802.11-1997 edition, Tech. Rep., 1997.

[27] C. Tomlin, “STARMAC, Stanford testbed of autonomous rotorcraft for multi-agent control,” Research project. [Online]. Available: http://hybrid.stanford.edu/starmac/overview

[28] R. A. Horn and C. R. Johnson, *Matrix Analysis*. Cambridge University Press, 1985.

[29] S. Boyd and L. Vandenberghe, *Convex Optimization*. Cambridge University Press, 2004.

[30] L. Ljung, *System Identification: Theory for the User (2nd Edition)*. Prentice Hall PTR, 1998.

[31] W. J. Rugh, *Linear Sysmtem Theory*, 2nd ed. Prentice Hall, 1996.

[32] L. Ljung and B. Walhberg, “Asymptotic properties of the least-squares method for estimating transfer functions and disturbance spectra,” *Advances in Applied Probability*, vol. 24, no. 2, 1992.

[33] K. S. Miller, “On the inverse of the sum of matrices,” *Mathematics Magazine*, vol. 54, no. 2, 1981.

[34] G. H. Golub and U. Von Matt, “Tikhonov regularization for large scale problems,” Stanford SCCM, Stanford, Tech. Rep. 97-03, 1997.