Improved Distributed Estimation Method for Environmental time-variant Physical variables in Static Sensor Networks

Magdi S. Mahmoud, Haris M. Khalid and Muhammad Sabih *†

March 15, 2013

Abstract

In this paper, an improved distributed estimation scheme for static sensor networks is developed. The scheme is developed for environmental time-variant physical variables. The main contribution of this work is that the algorithm in [1]-[3] has been extended, and a filter has been designed with weights, such that the variance of the estimation errors is minimized, thereby improving the filter design considerably and characterizing the performance limit of the filter, and thereby tracking a time-varying signal. Moreover, certain parameter optimization is alleviated with the application of a particular finite impulse response (FIR) filter. Simulation results are showing the effectiveness of the developed estimation algorithm.

Keywords: Distributed estimation, static sensor networks, minimum variance, performance limit, parameter optimization, estimation algorithm, time-variant.

1 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 (WSNs), have a growing domain of application in areas such as environmental monitoring, industrial automation, intelligent buildings, search and surveillance, and automotive applications [4]-[11]. 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.

Today, an increasing number of applications demands remote control of plants over unreliable networks. In these systems issues of communication delay, data loss and time–synchronization play critical roles. It is noted in [12] that several approaches have focused 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 [1], a scheme

*Manuscript **MsM-KFUPM-HK-MS[R-III].tex**

†Systems Engineering Department, King Fahd University of Petroleum and Minerals, P. O. Box 5067, Dhahran 31261, **Saudi Arabia**, e-mail: msmahmoud, mhariskh, msabih@kfupm.edu.sa.

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.

This paper is organized as follows. Section 2 presents the related works. Section 3 presents the distributed estimation problem formulation considered throughout the paper. The distributed estimator design is discussed in Section 4. 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 4 discusses estimation of the error covariance. Section 5 presents the detail of the implementation of the estimation algorithm. Numerical results and implementation structure illustrating the performance of the proposed estimator and comparing it to some related proposals are also given. Finally, Section 6 concludes the paper.

2 Related Works

An extension of the approach for development of consensus filter based on sensor fusion is presented in [13], 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 [14], where the weights are computed off-line to speed up the computation of the averages. Another approach towards estimation is using H_∞ estimation with the case of limited communication capacity [15] and asynchronous filtering in [16]. Further characterization of consensus filters for distributed sensor fusion is given in [3]. Another approach to distributed estimation is based on nonlinear filters using self-synchronization and coupling functions, see [17]-[20]. 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 [20]. Distributed filtering using model-based approaches is studied in various wireless network contexts, see for example [21]-[25]. Distributed Kalman filters and more recently a combination of the diffusion mechanism, discussed previously, with distributed Kalman filtering, see [2] and [26] 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.

Distributed signal processing is a very active research area due to the recent developments in networking, computer and sensor technologies. 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 [27] 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, to name a few.

Considering distributed consensus-based estimation, iterative consensus protocols is shown in [28], local average consensus algorithms can be seen in [29], consensus strategies-based algorithms can be seen in [30], consensus iterations-based algorithms are proposed in [31], and convergence speed of consensus strategies are shown in [32]. Then, distributed expectation maximization algorithm over sensor networks are also proposed in the literature, where consensus filter used to diffuse local sufficient statistics to neighbors and estimate global sufficient statistics in each node is shown in [33], distributed expectation maximization algorithm over sensor networks, consensus filter used to diffuse local sufficient statistics to neighbors and estimate global sufficient statistics in each node are proposed in this paper, consensus filter diffusion of local

sufficient statistics over the entire network through communication with neighbor nodes is shown in [34].

Ignoring this, time delay is an obvious violation of the physics behind the propagation of signal, being envisioned to be best estimated from measurements of distantly placed sensor nodes. Some recent results on various topics of distributed estimation have been reported in [35]-[41]. If we consider results on time-varying state delay on network systems, an approach can be seen in [42]. Also, a robust H_∞ estimation with signal transmission delay and data packet dropouts can be seen in [43]. Moreover, considering the work on time-delay a new delay system on networked system is proposed in [44], a robust stability for uncertain delay in [45], a stabilization approach with delay in [46] and [47], a filtering technique for mixed random delays is shown in [48], a feedback control approach with mixed delays is working in [49] and a disturbance rejection technique for discrete-time delay systems is shown in [50].

The problem statement for estimation physical variables in SNs can be stated as follows. Consider the estimation of an available SN is to be based on the available sensor measurements. In sensor networks, due to the constraints on communication and computation, we cannot broadcast all the sensor measurements to implement the real-time estimation of the field at the locations of interest. Rather, we have to determine the relevant sensors and use their particular measurements only in the field estimation at a particular location of interest.

The new proposed estimator for SNs presented in this paper belongs to a class of recently developed filtering algorithms that exploit in-network computing [12]. 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 this paper, we adopt the concept of minimum variance to develop an improved distributed estimation algorithm by considering the propagation delay of the physical quantity to be measured. This estimation algorithm has the following features:

- With a consideration of time-delay appearing in the measurements of different sensor nodes, it is assumed that the source of data is located at a known position.
- It incorporates an FIR filter aims at simplifying the signal estimation and alleviating the parameter optimization.
- It tracks a time-varying signal, while the estimators of [51] and [1] 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 [2] and [25].
- There is no pre-assigned coupling law imposed among the nodes as in [20].
- Compared to the methods of [1]–[3], our approach does not rely on the Laplacian matrix associated with the communication graph. Rather, it considers a more general model of the filter structure.
- Our filter parameters are computed through distributed algorithms, whereas, for example, the estimators of [13] and [14] rely on centralized algorithms for designing the filters.
- By extending the algorithms in [1]–[3], we design the filter weights such that the variance of the estimation errors is minimized, thereby improving the filter design considerably and we characterize the performance limit of the filter.

Remark 2.1 *In this paper an improved distributed estimation method will be developed based on the concept of minimum variance with a consideration of time delay appearing in the measurements of different sensor nodes. The scheme is developed on a static SN. This is attributed to the time taken by physical quantity to be measured (for example, temperature, acoustic waves, to name few) to propagate from one point in the space to another one. In this regard, it is assumed that source of the data is located at a known position and the spatial ordinates of the sensor nodes are known. This knowledge is applied to calculate the time delay between the source of the data and any node in the network. The spectrum of the signal of interest is assumed to be known a–priori. Based on this knowledge the data is first passed through a low pass FIR filter. The signal is then jointly tracked by a SN, in which each node computes an estimate as a delay adjusted weighted sum of its own and its neighbors measurements and estimates. The filter weights are time varying and updated locally and it has a cascade structure comprised of two loops: an inner loop producing the state estimate and an outer loop producing an estimate of the error covariance. The state estimate is thus obtained as the solution of an optimization problem with quadratic cost function and quadratic constraints. As will be shown in later section, incorporation of FIR filter helps to alleviate the optimal estimation of a parameter, at a cost of acceptable deviation from the otherwise regular estimate. 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 discussion and computer simulations exposition in the latter part of the paper.*

2.1 Notation

We denote the set of non-negative integers as $N_0 = \{0, 1, 2, \dots\}$ and N the number of nodes. 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, where l_2-norm is the length of the vector. Given a matrix $A \in \mathbf{R}^{n\times n}$, 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 as Tr(A). With I and $\mathbf{1}$ we denote the identity matrix and the vector $(1, \dots, 1)^T$, respectively. Given a stochastic variable x we denote $\mathbf{E}[x]$ as the expected value of x. We denote $N_0 = N \cup \{0\}$. For the sake of notational simplicity, we discard the time dependence when it is clear from the context. In symmetric block matrices or complex matrix expressions, we use the symbol $\bullet$ to represent a term that is induced by symmetry.

3 Problem Formulation and Preliminaries

Consider $N > 1$ sensor nodes with known static positions in space. It is assumed that the source of the signal is located at a known position and thus the radial r_i distance between an ith node and the signal source is also known. The time τ_i taken by a particular wavefront of the signal to travel from the source to an ith node can be obtained from the following relationship.

$$\tau_i = \frac{c}{r_i} \tag{1}$$

where c is the propagation speed of the signal, assumed to be known. An ith node measures a common scalar signal $d(t)$ affected by additive noise and delayed by τ_i:

$$u_{(\tau_i,v_i')}(t) = d(t-\tau_i) + v_i'(t), i = 1,\ldots,N, \tag{2}$$

with $t \in N_0$ and where $v_i'(t)$ is zero-mean white noise. With the knowledge of τ_i, data from an ith node and its neighbors can be appropriately *time adjusted* (either delayed or advanced, depending upon whether a neighbor node is closer to or farther from the source with respect to the ith node). We assume that this operation on the data is performed at the very beginning of the distributed estimation scheme under consideration and the resulting delay adjusted noisy signal is denoted as $u_{v_i'}(t)$.

Since the signal spectrum is assumed known, a significant amount of noise can be filtered out with the application of a linear phase FIR filter. This processing of the signal can be carried out at each individual node. As a result the signal will be left contaminated only with that part of the noise which has a spectrum overlapping with that of the signal of interest. Let the filtered noisy signal and vestige of the noise at node i are denoted as $u_i(t)$ and $v_i(t)$, respectively.

Let us collect measurements and noise variables in vectors, $u(t) = (u_1(t), \ldots, u_N(t))^T$ and $v(t) = (v_1(t), \ldots, v_N(t))^T$, so that we can rewrite the previous equation as:

$$u(t) = d(t)\mathbf{1} + v(t), \quad t \in N_0. \tag{3}$$

The covariance matrix of $v(t)$ is assumed to be diagonal (See remark 3.1) $\Sigma = \sigma^2 I$, 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 $G = (V, E)$, where $V = \{1,\ldots,N\}$ is the vertex set and $E \subseteq V \times V$ the edge set. We assume that if $(i,j) \in E$ then $(j,i) \in E$, so that the direction of the edges can be dropped when representing the network. The graph G is said to be connected if there is a sequence of edges in 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 where $i \in V$ plus the node itself as:

$$N_i = \{j \in V : (j,i) \in E\} \cup \{(i,i)\}. \tag{4}$$

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 measurements.

$$x_i(t) = \sum_{j \in N_i} k_{ij}(t) x_j(t-1) + \sum_{j \in N_i} h_{ij}(t) u_j(t). \tag{5}$$

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

[52], which have been actually used for distributed estimation and control algorithms of unmanned aerial vehicles [53].

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

$$x(t) = K(t)x(t-1) + H(t)u(t). \tag{6}$$

$K(t)$ 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 G.

Given an SN modeled as a connected graph G, we have the following design problem: find time-varying matrices $K(t)$ and $H(t)$, compatible with 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.

Remark 3.1 *It should be noted here that it is assumed to be diagonal as there are multiple sensors at each location, sensing, for example, waves at different frequencies. Again under the assumption that well-separated sensors have uncorrelated measurement noise, and sensors at the same location have correlated measurement noise, the sensor noise covariance matrix would be block-diagonal. As such the covariance of all of the sensor measurements would decompose as the sum of a low-rank matrix (with rank equal to the total number of sources over all measured frequencies) and a block-diagonal matrix. A block-diagonal and low-rank decomposition problem also arises if the second Diagonal and Low-Rank Matrix Decompositions order statistics of the noise have certain symmetries.*

3.1 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 $e(t) = x(t) - d(t)1$. Introduce $\delta(t) = d(t) - d(t-1)$, so that the error dynamics can be described as:

$$\begin{aligned} e(t) &= K(t)e(t-1) + d(t)(K(t) + H(t) - I)\mathbf{1} \\ &\quad -\delta(t)K(t)\mathbf{1} + H(t)v(t). \end{aligned} \tag{7}$$

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

$$\begin{aligned} \mathbf{E}[e(t)] &= K(t)E[e(t-1)] + d(t)(K(t) + H(t) - I)\mathbf{1} \\ &\quad -\delta(t)K(t)\mathbf{1}. \end{aligned} \tag{8}$$

Proposition 3.1 *Consider system (7) and assume that*

$$(K(t) + H(t))\mathbf{1} = 1, \tag{9}$$

and that there exists $0 \leq \gamma_0 < 1$

$$s.t.\ \gamma_{\max}(K(t)) \leq \gamma_0, \quad \forall\ t \in N_0$$

(1) If $H(t)\mathbf{1} = 1$*, for all* $t \in N_0$*, then*

$$\lim_{t \to +\infty} \boldsymbol{E}[e(t)] = 0. \tag{10}$$

(2) If $|\delta(t)| < \Delta$, for all $t \in N_0$, then

$$\lim_{t\to+\infty} \|\boldsymbol{E}[e(t)]\| \leq \frac{\sqrt{N}\Delta\gamma_0}{1-\gamma_0}. \tag{11}$$

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

$$\mathbf{E}[e(t)] = K(t)\mathbf{E}[e(t-1)] + \delta(t)(H(t) - I)\mathbf{1}. \tag{12}$$

(1) If $H(t)\mathbf{1} = 1$, then (12) becomes $\mathbf{E}[e(t)]$ = $K(t)\mathbf{E}[e(t-1)]$. Let us consider the function $\mathrm{V}(t) = \|\mathbf{E}[e(t)]\|$. It follows that

$$V(t) \leq \|K(t)\|V(t-1) \leq \gamma_0 V(t-1) \leq \gamma_0^t V(0), \tag{13}$$

which implies that $\lim_{t\to+\infty} \|\mathbf{E}[e(t)]\| = 0$. It should be noted here that a finite difference technique is used in (13) of a mathematical expression of the form, lets say $f(x + b)$ - $f(x + a)$. If a finite difference is divided by b - a, one gets a difference quotient. The approximation of derivatives by finite differences plays a central role in finite difference methods for the numerical solution of differential equations, especially boundary value problems. In this case, a forward difference has been used as $\Delta_h[f](x) = f(x + h)$ - $f(x)$, where depending on the application, the spacing h may be variable or constant.

(2) In this case, we have $H(t)\mathbf{1} - 1 = -K(t)\mathbf{1}$ and thus the system (7) becomes

$$\mathbf{E}[e(t)] = K(t)\mathbf{E}[e(t-1)] - \delta(t)K(t)\mathbf{1}. \tag{14}$$

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

$$\begin{aligned} V(t) \leq \quad & \|K(t)\|V(t-1) + \|K(t)\|\sqrt{N}\Delta \\ \leq \quad & \gamma_0 V(t-1) + \gamma_0\sqrt{N}\Delta \\ \leq \quad & \gamma_0^t V(0) + \gamma_0\frac{1-\gamma_0^t}{1-\gamma_0}\sqrt{N}\Delta. \end{aligned} \tag{15}$$

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

Proposition 3.1 (i) provides the condition $H(t)\mathbf{1} = 1$ under which the estimate is unbiased. It is possible to show that in this case the variance is minimized if $K(t) = 0$ and

$$h_{ij}(t) = h_{ji}(t) = \begin{cases} \frac{1}{|N_i|} & if\ j \in N_i \\ 0 & otherwise. \end{cases} \tag{16}$$

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* 3.1 (ii), it is possible to guarantee that $\|\mathbf{E}[e(t)]\|$ tends to be a neighborhood of the origin, but the estimate might be biased. Note also that $\|\mathbf{E}[e(t)]\|$ is a cumulative bias, that is, 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, (11) 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 SNR = P_d/P_b. Since there are N nodes,

we consider the average SNR of each node as $\Upsilon = SNR/N$. Let us assume that we want the estimator to guarantee that the right-hand side of Equation (11) be equal to this desired $\sqrt{SNR}$, i.e., that

$$\gamma_0 = \frac{\sqrt{\Upsilon}}{\sqrt{\Upsilon} + \Delta}. \tag{17}$$

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.

4 Distributed Estimation 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* 3.1 are fulfilled. The condition $(K(t) + H(t))\mathbf{1} = 1$ 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 $\gamma_{\max}(K(t)) \leq f(\Delta, \Upsilon)$. It turns out that it is possible to determine a local condition, $\sum_{j \in N_i} \kappa_{ij}^2 \leq \psi_i$, where ψ_i is a constant that can be computed locally by the nodes. We then pose an 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.

4.1 Distributed variance minimization

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

$$\epsilon_i = (e_{i1}, \ldots, e_{i_{M_i}})^T, \quad i_1 < \cdots < i_{M_i}. \tag{18}$$

Similarly, we introduce vectors $\kappa_i^T(t), \eta_i^T(t) \in R^{M_i}$ 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

$$\mathbf{E}[e_i(t)] = \kappa_i^T(t)\mathbf{E}[\epsilon_i(t-1)] - \kappa_i^T(t)\delta(t)1, \tag{19}$$

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

$$\mathbf{E}[e_i(t) - \mathbf{E}[e_i(t)]]^2 = \kappa_i^T(t)\Gamma_i(t-1)\kappa_i(t) + \sigma^2\eta_i^T(t)\eta_i(t), \tag{20}$$

where $\Gamma_i(t) = \mathbf{E}[\epsilon_i(t) - \mathbf{E}[\epsilon_i(t)]][\epsilon_i(t) - \mathbf{E}[\epsilon_i(t)]]^T$. 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:

$$\mathsf{P}_1 : \min_{\kappa_i(t), \eta_i(t)} \kappa_i^T(t)\Gamma_i(t-1)\kappa_i(t) + \sigma^2\eta_i^T(t)\eta_i(t) \tag{21}$$

$$s.t. \quad (\kappa_i(t) + \eta_i(t))^T \mathbf{1} = 1 \ and \ \gamma_{\max}(K(t)) \leq f(\Delta, \Upsilon)$$

The inequality constraint (22) is still global, since $\gamma_{\max}(K(t))$ depends on all $\kappa_i(t), i = 1, \ldots, N$. We show next that it can be replaced by the local constraint:

$$\|\kappa_i(t)\| \leq \psi_i, \quad t \in N_0, \tag{22}$$

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 $i = 1, \ldots, N$ define the set $\Theta_i = \{j \neq i : N_j \cap N_i \neq 0\}$, 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 4.1 *Suppose there exist $\psi_i > 0, i = 1, \ldots, N$, such that*

$$\psi_i + \sqrt{\psi_i} \sum_{j \in \Theta_i} \sqrt{\alpha_{i,j}^{(i)} \alpha_{i,j}^{(j)} \psi_j} \leq f^2(\Delta, \Upsilon), \tag{23}$$

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

$$\sum_{c \in N_j \cap N_i} \kappa_{ic}^2 \leq \alpha_{i,j}^{(i)} \sum_{r=1}^{M_i} \kappa_{ii_r}^2 \qquad \sum_{c \in N_j \cap N_i} \kappa_{jc}^2 \leq \alpha_{i,j}^{(j)} \sum_{r=1}^{M_j} \kappa_{ji_r}^2. \tag{24}$$

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

Proof: We use Gershgorin's theorem (See Remark 4.1) to bound the eigenvalues of the matrix KK^T i.e. the singular values of K. The following relations hold: $[KK^T]_{ii} = \sum_{c=1}^{M_i} \kappa_{ic}^2$ and $[KK^T]_{ij} = \sum_{c=1}^{N} \kappa_{ic}\kappa_{jc}$. By the Gershgorin's theorem we know that for $r = 1, \ldots, N$

$$\lambda_r(KK^T) \in \cup_{i=1}^{N} \{z \in R : |z - [KK^T]_{ii}| \leq R_i(KK^T)\}, \tag{25}$$

with

$$R_i(KK^T) = \sum_{j=1, j \neq i}^{N} |[KK^T]_{ij}| = \sum_{j=1, j \neq i}^{N} \Big| \sum_{c=1}^{N} k_{ic} k_{jc} \Big| \tag{26}$$

Now the inner sum in $R_i(KK^T)$ is non-zero only for $c \in N_j \cap N_i$. Thus,

$$R_i(KK^T) = \sum_{j \in \Theta_i} \Big| \sum_{c \in N_j \cap N_i} k_{ic} k_{jc} \Big|. \tag{27}$$

Using the Cauchy-Schwartz inequality,

$$\Big| \sum_{c \in N_j \cap N_i} k_{ic} k_{jc} \Big|^2. \leq \sum_{c \in N_j \cap N_i} k_{ic}^2 \sum_{c \in N_j \cap N_i} k_{jc}^2. \tag{28}$$

Then,

$$\sum_{j=1,j\neq i}^{N}\Big|\sum_{c=1}^{N}k_{ic}k_{jc}\Big| \leq \sum_{j\in\Theta_i}\sqrt{\sum_{c\in N_j\cap N_i}k_{ic}^2\sum_{c\in N_j\cap N_i}k_{jc}^2}$$

$$\leq \sum_{j\in\Theta_i}\sqrt{\alpha_{i,j}^{(i)}\sum_{r=1}^{M_i}\kappa_{ii_r}^2\alpha_{i,j}^{(j)}\sum_{r=1}^{M_j}\kappa_{ji_r}^2}$$

$$\sqrt{\sum_{r=1}^{M_i}\kappa_{ii_r}^2}\cdot\sum_{j\in\Theta_i}\sqrt{\alpha_{i,j}^{(i)}\alpha_{i,j}^{(j)}\sum_{r=1}^{M_j}\kappa_{jj_r}^2}. \tag{29}$$

Hence,

$$\lambda_r(KK^T)\in\cup_{i=1}^{N} \left\{ z\in C:\Big|z-\sum_{r=1}^{M_i}\kappa_{jj_r}^2\Big|\leq\sqrt{\sum_{r=1}^{M_i}\kappa_{ii_r}^2}. \right. \tag{30}$$

$$\left. \sum_{j\in\Theta_i}\sqrt{\alpha_{i,j}^{(i)}\alpha_{i,j}^{(j)}\sum_{r=1}^{M_j}\kappa_{jj_r}^2} \right\} \tag{31}$$

From the hypothesis that $\|\kappa_i\|=\sum_{r=1}^{M_i}\kappa_{ii_r}^2\leq\psi_i$ and (23), then

$$\sum_{r=1}^{M_i}\kappa_{ic}^2+\sqrt{\sum_{r=1}^{M_i}\kappa_{ii_r}^2}\cdot\sum_{j\in\Theta_i}\sqrt{\sum_{r=1}^{M_j}\alpha_{i,j}^{(i)}\alpha_{i,j}^{(j)}\kappa_{jj_r}^2}\leq f^2(\Delta,\Upsilon). \tag{32}$$

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

Proposition 4.1 provides a simple local condition on the filter coefficients such that $\gamma_{\max}(K)-f(\Delta,\Upsilon)$. We can expect that *Proposition* 4.1 is in general conservative, because no *a–priori* knowledge of the network topology is used, the proof relies on the Gershgorin's 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 of proposition 4.1, for example, ([54], pages 378.389). However, we do not know of any other bounds requiring only local information. Further, the Peron-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* 4.1 can all be set to one. However, this yields very conservative bounds on the maximum eigenvalue of KK^T. In Section 5, we show how to chose these parameters to avoid bounds that are too conservative.

Remark 4.1 *Gershgorins Theorem is used for estimating eigenvalues of a certain matrix. It estimates the range of the eigenvalues. This cannot be determined by trace of the matrix. The trace merely tells us what all the eigenvalues add up to. It doesn't give us any range for the eigenvalues. Even if we have a very small trace we can still theoretically have two eigenvalues whose absolute values are very large but have an opposite sign.*

4.2 Optimal weights for variance minimization

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

$$\mathsf{P}_2 : \min_{\kappa_i(t),\eta_i(t)} \kappa_i(t)^T \Gamma_i(t-1)\kappa_i(t) + \sigma^2 \eta_i(t)^T \eta_i(t) \tag{33}$$

$$s.t. \qquad (\kappa_i(t) + \eta_i(t))^T 1 = 1 \ and \ \|\kappa_i\|^2 \leq \psi_i, \tag{34}$$

We remark here that Problem P_2 has a different solution with respect to Problem P_1, because the constraint (21) has been replaced with (34). Problem 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 $\eta_i(t)1 = 1$. Thus, Slater's condition is satisfied and strong duality holds ([55], p. 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 4.2 *For a given positive definite matrix $\Gamma_i(t-1)$, the solution to problem P_2 is given by*

$$\kappa_i(t) = \frac{\sigma^2(\Gamma_i(t-1) + \xi_i I)^{-1}\mathbf{1}}{\sigma^2 \mathbf{1}^T(\Gamma_i(t-1) + \xi_i I)^{-1}\mathbf{1} + M_i}, \tag{35}$$

$$\eta_i(t) = \frac{1}{\sigma^2 \mathbf{1}^T(\Gamma_i(t-1) + \xi_i I)^{-1}\mathbf{1} + M_i}, \tag{36}$$

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

Proof: Since the problem is convex and Slater's condition holds, the **KKT** conditions (See remark 4.2)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

$$(\kappa_i^*)^T\kappa_i^* - \psi_i \leq 0, \quad (\kappa_i^* + \eta_i)^T 1 - 1 = 0, \tag{37}$$
$$\xi_i^* \geq 0, \quad \xi_i^*((\kappa_i^*)^T\kappa_i^* - \psi_i) = 0, \tag{38}$$
$$2(\Gamma_i + \lambda_i^* I)\kappa_i^* + \nu_i^* 1 = 0, \quad 2\sigma^2\eta_i^* + \nu_i^* 1 = 0, \tag{39}$$

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

$$L(\kappa_i, \eta_i, \xi, \nu) = \kappa_i^T\Gamma_i\kappa_i + \sigma^2\eta_i^T\eta_i + \xi_i(\kappa_i^T\kappa_i - \psi_i) \tag{40}$$
$$+ \nu_i((\kappa_i + \eta_i)^T 1 - 1). \tag{41}$$

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

$$\frac{\sigma^4 1^T(\Gamma_i + \xi_i^* I)^{-2}1}{(\sigma^1 1^T(\Gamma_i + \xi_i^* I)^{-1}1 + M_i)^2} = \psi_i \tag{42}$$

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

$$(\kappa_i^*)^T \kappa_i^* \leq \frac{\sigma^4 \|(\Gamma_i + \xi I)^{-1}\|^2}{M_i} \leq \frac{\sigma^4}{M_i \lambda_{\min}^2(\Gamma_i + \xi I)} \leq \psi_i, \tag{43}$$

from where we obtain

$$\xi \geq \frac{\sigma^2}{\sqrt{M_i \psi_i}} - \lambda_{\min}(\Gamma_i), \tag{44}$$

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. $\nabla\nabla\nabla$

Proposition 4.2 gives an interval within which the optimal ξ_i can be found. The first constraint in Problem P_2 is similar to the water-filling problem for power allocation in wireless networks [29]. Analogous 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 P_2 does not have the same solution as the Problem P_1, as the constraints (21) and (34) are not equivalent, although if (34) holds then (21) holds as well.

Remark 4.2 *KKT conditions (also known as the KuhnTucker conditions) are first order necessary conditions for a solution to be optimal, provided that some regularity conditions are satisfied. Allowing inequality constraints, the KKT approach to nonlinear programming generalizes the method of Lagrange multipliers, which allows only equality constraints. It should be noted that KKT conditions become necessary when the problem is convex and Slater's conditions hold. Slater's theorem provides a sufficient condition for strong duality to hold. Namely, if:*

- *The primal problem is convex,*
- *It is strictly feasible, that is, there exists $x_0 \in \boldsymbol{R}^n$ such that $Ax_0 = b$, $f_i(x_0) < 0$, $i = 1, \ldots m$, then strong duality holds: $p^* = d^*$, and the dual problem is attained.*

4.3 Bounds on the error variance

The optimal weights from *Proposition* 4.2 gives the following estimation error variance.

Proposition 4.3 *Let $\kappa_i(t)$ and $\eta_i(t)$ be an optimal solution given by (35) and (36). Then*

$$\boldsymbol{E}[e_i(0) \quad - \quad \boldsymbol{E}[e_i(0)]]^2 = \sigma^2, \tag{45}$$

$$\boldsymbol{E}[e_i(t) \quad - \quad \boldsymbol{E}[e_i(t)]]^2 \leq \frac{\sigma^2}{M_i}, \quad t \in N_0 \{0\}. \tag{46}$$

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

$$\begin{aligned} \mathbf{E}[e_i(t) - \mathbf{E}[e_i(t)]]^2 &= \frac{\sigma^2}{M_i + \sigma^2 \mathbf{1}^T (\Gamma_i(t-1) + \xi_i I)^{-1} \mathbf{1}} \\ &- \frac{\sigma^4 \xi_i \mathbf{1}^T (\Gamma_i(t-1) + \xi_i I)^{-2} \mathbf{1}}{(M_i + \sigma^2 \mathbf{1}^T (\Gamma_i(t-1) + \xi_i I)^{-1} \mathbf{1})^2} \\ &= \frac{\sigma^2}{M_i + \sigma^2 \mathbf{1}^T (\Gamma_i(t-1) + \xi_i I)^{-1} \mathbf{1}}. \end{aligned} \tag{47}$$

Since $\Gamma_i(t-1)$ is positive definite and $\xi_i \geq 0$, it holds that $\mathbf{1}^T(\Gamma_i(t-1)+\xi_i I)^{-1}\mathbf{1} > 0$. Hence $\mathbf{E}[e_i(t) - \mathbf{E}[e_i(t)]]^2 \leq \frac{\sigma^2}{M_i},\ t \in N_0\ \{0\}$. This concludes the proof. $\nabla\nabla\nabla$

A consequence of *Proposition* 4.3 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* 4.2 helps us to improve the bound in *Proposition* 4.3 as follows.

Corollary 4.1 *The optimal value of $\kappa_i(t)$ and $\eta_i(t)$ are such that the error variance at node i satisfies*

$$\begin{aligned} &\boldsymbol{E}[e_i(t) - \boldsymbol{E}[e_i(t)]]^2 \\ &\leq \frac{\sigma^2}{M_i + \left(\sum_{j\in N_i} M_j^{-1} + (M_i\psi_i)^{-1/2}\right)^{-1}}, \end{aligned} \tag{48}$$

where $t \in N_0\ \{0\}$.

Proof: Using the result in *Proposition* 4.3 we have that

$$Tr\Gamma_i(t-1) = \sum_{j\in N_i} \mathbf{E}[e_i(t-1) \quad - \quad \mathbf{E}[e_i(t-1)]]^2 \leq \sum_{j\in N_i} \frac{\sigma^2}{M_j}. \tag{49}$$

Thus

$$\begin{aligned} \lambda_{\max}(\Gamma_i(t-1)+\xi_i I) \quad &\leq \quad \sum_{j\in N_i} \frac{\sigma^2}{M_j} + \\ &+ \quad \max\left(0, \frac{\sigma^2}{\sqrt{M_i\psi_i}} - \lambda_{\min}(\Gamma_i(t-1))\right) \\ &\leq \quad \sum_{j\in N_i} \frac{\sigma^2}{M_j} + \frac{\sigma^2}{\sqrt{M_i\psi_i}}, \end{aligned} \tag{50}$$

where we used the bound on ξ determined in *Proposition* 4.2. Since

$$\mathbf{1}^T(\Gamma_i(t-1)+\xi_i I)^{-1} \geq \frac{1}{\lambda_{\max}(\Gamma_i(t-1)+\xi_i I)M_i} \tag{51}$$

we have that

$$\mathbf{E}[e_i(t) \quad - \quad \mathbf{E}[e_i(t)]]^2 \leq \frac{\sigma^2}{M_i + \sigma^2\mathbf{1}^T(\Gamma_i(t-1)+\xi_i I)^{-1}\mathbf{1}} \tag{52}$$

$$\leq \frac{\sigma^2}{M_i + \left(\sum_{j\in N_i} M_j^{-1} + (M_i\psi_i)^{-1/2}\right)^{-1}}. \tag{53}$$

which completes the proof. $\nabla\nabla\nabla$

Remark 4.3 *The parameter optimization scheme is achieved with the help of low pass FIR filter, which has filter weights for the quadratic cost function and constraints. These filter weights have two loops, one inner loop and one outer loop. The function of the inner loop is to calculate the state estimate, where as the function of outer loop is to calculate error covariance matrix. Then vectors $\kappa_i^T(t), \eta_i^T(t) \in R^{M_i}$ have been introduced corresponding to non-zero elements $K(t)$ and $H(t)$. After that the expected value $\boldsymbol{E}e_i(t)$ is calculated which is the value of estimation error. Then by using local optimization as P_2, the optimization problem is solved to minimize the variance of estimation error for a given positive definite matrix to achieve parameter optimization.*

4.4 Distributed computation of constraints

The choice of the constants $\psi_i, i = 1, \ldots, N$, in the local constraint of Problem 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* 4.2 we see that if ψ_i is large then the Lagrangian multiplier ξ_i is small, since it must lie in the in interval $\max [0, \sigma^2/\sqrt{M_i\psi_i} - \lambda_{\min}(\Gamma_i(t-1))]$. From *Proposition* 4.3 (and Corollary 4.1) 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* 4.1 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:

$$\max_{\psi_1,\ldots,\psi_N} \sum_{i=1}^{N} \psi_i \tag{54}$$

$$s.t. \qquad \psi_i + \sqrt{\psi} \cdot \sum_{j\in\Theta_i} \sqrt{\alpha_{i,j}^{(i)}\alpha_{i,j}^{(j)}\psi_j} \leq f^2(\Delta, \Upsilon)$$

$$\psi_i \geq 0,$$

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:

$$\psi_i + \sqrt{\psi} \sum_{j\in\Theta_i} \sqrt{\alpha_{i,j}^{(i)}\alpha_{i,j}^{(j)}\psi_j} = f^2(\Delta, \Upsilon) \quad i = 1, \ldots, N. \tag{55}$$

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 (55) 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 ([57] , pp. 181–191), we consider the iterative algorithm

$$\psi(t+1) = T(\psi(t)) = (T_1(\psi(t)), \ldots, T_N(\psi(t))) \tag{56}$$

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

$$T_i(\psi(t)) = \frac{1}{4}\Bigg[\sqrt{\left(\sum_{j\in\Theta_i} \sqrt{\alpha_{i,j}^{(i)}\alpha_{i,j}^{(j)}\psi_j(t)}\right)^2 + 4f^2(\Delta, \Upsilon)} - \sum_{j\in\Theta_i} \sqrt{\alpha_{i,j}^{(i)}\alpha_{i,j}^{(j)}\psi_j(t)}\Bigg]^2, \tag{57}$$

It is not difficult to show that $T_i(\psi)$ is a contractive function. The component solution method in [57] ensures that the fixed point solution at which the iteration converges is the solution of the nonlinear Equations (55). The computation of the iteration (56) can be done distributively. Note that node i does not need to know the thresholds $\psi_j, j \neq i$, 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.

4.5 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 4.1 *([30, p. 34]). A signal $s(t) : R \to R$ is said to be quasi-stationary if there exists a positive constant C and a function $R_s : R \to R$, such that s fulfills the following conditions*

(1) $Es(t) = m_s(t), |m_s(t)| \leq C$ *for all* t

(2) $Es(t)s(r) = R_s(t, r), |R_s(t, r)| \leq C$ *for all* t *and*

$$\lim_{N\to+\infty} \frac{1}{N} \sum_{t=1}^{N} R_s(t, t-\tau) = R_s(\tau) \tag{58}$$

for all τ.

It is easy to see that the time-varying linear system (6) is uniformly bounded-input bounded-output stable ([58], p. 509). If a quasi-stationary signal is the input of such system, then its output is also quasi-stationary [59]. 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 $E\epsilon_i = m_{\epsilon i}(t)$ and covariance $\Gamma_i(t)$ can be estimated from samples as:

$$\hat{m}_{\epsilon i}(t) = \frac{1}{t} \sum_{\tau=0}^{t} \hat{\varepsilon}_i(\tau) \tag{59}$$

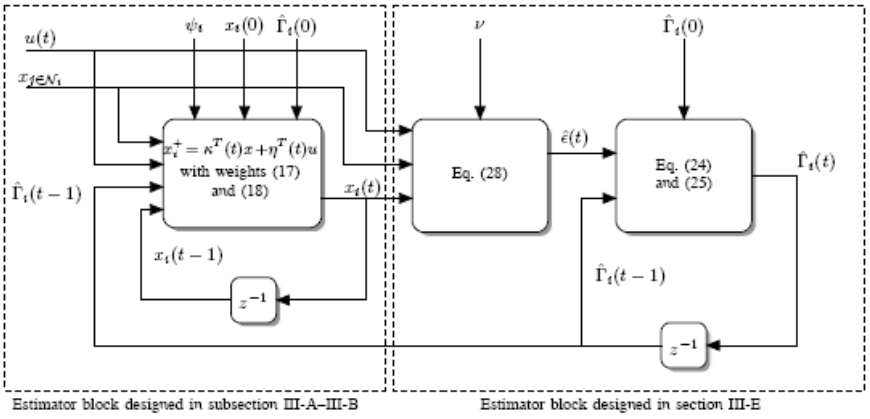
$$\hat{\Gamma}_i(\tau) = \frac{1}{\tau} \sum_{\tau=0}^{t} (\hat{\epsilon}_i(\tau) - \hat{m}_{\epsilon_i}(\tau))(\hat{\epsilon}_i(\tau) - \hat{m}_{\epsilon_i}(\tau))^T, \tag{60}$$

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 $x_{ij}(t)$ and measurements $u_{ij}(t), i_j \in N_i$, available. Let $x^{(i)}(t)$ and $u^{(i)}(t)$ denote the collection of all these variables. We can model this data set as

$$x^{(i)}(t) = d(t)\mathbf{1} + \beta(t) + w(t) \tag{61}$$

$$u^{(i)}(t) = d(t)\mathbf{1} + v(t) \tag{62}$$

where $\beta(t) \in R^{M_i}$ models the bias of the estimates and $w(t)$ is zero-mean Gaussian noise modeling 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 $\hat{\epsilon}_i(t) = x^{(i)}(t) - \hat{d}(t)1$, would have the advantage of being unbiased. However, its covariance is rather large since M_i is typically small. Thus, using only measurements



..

Figure 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 is an estimator block that estimates the error covariance matrix

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:

$$\begin{array}{ll} \min\limits_{\hat{d},\hat{\beta}} & \left\| \left[\begin{array}{c} x^i \\ u^i \end{array} \right] - A \left[\begin{array}{c} \hat{d} \\ \hat{\beta} \end{array} \right] \right\|^2 \\ s.t. & \| B \left[\begin{array}{cc} \hat{d} & \hat{\beta} \end{array} \right] \|^2 \leq \rho \end{array}$$

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

$$A = \left[\begin{array}{cc} 1 & I \\ 1 & 0 \end{array} \right], \quad B = \left[\begin{array}{cc} 0 & I \end{array} \right],$$

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 ([55], p. 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.

$$\min_{\hat{d},\hat{\beta}} \left\| \left[\begin{array}{c} x^i \\ u^i \end{array} \right] - A \left[\begin{array}{c} \hat{d} \\ \hat{\beta} \end{array} \right] \right\|^2 + \nu \left\| B \left[\begin{array}{c} \hat{d} \\ \hat{\beta} \end{array} \right] \right\|^2 \tag{63}$$

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

The solution of (63) is

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

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

Proposition 4.4 *If $\nu > 0$ then*

$$
\begin{aligned}
&[A^TA + \nu B^TB]^{-1} = \\
&= \frac{1}{M_i(1+2\nu)} \begin{bmatrix} 1+\nu & -\mathbf{1}^T \\ \bullet & \frac{M_i(1+2\nu)I+11^T}{1+\nu} \end{bmatrix}
\end{aligned} \tag{64}
$$

Proof: By Schur's complement we obtain

$$
\begin{aligned}
&[A^TA + \nu B^TB]^{-1} = \\
&\begin{bmatrix} \left(2M_i - \frac{M_i}{1+\nu}\right)^{-1} & \mathbf{1}^T(\mathbf{1}\mathbf{1}^T - 2M_i(1+\nu)I)^{-1} \\ \bullet & \left((1+\nu)I - \frac{11^T}{2M_i}\right)^{-1} \end{bmatrix}
\end{aligned}
$$

From [33], it follows that

$$
\left[(1+\nu)I - \frac{\mathbf{1}\mathbf{1}^T}{2M_i}\right]^{-1} = \frac{I}{1+\nu} + \frac{11^T}{M_i(1+2\nu)(1+\nu)}.
$$

It is easy from here to show that the resulting matrix is (64). $\nabla\nabla\nabla$

Since we are interested in estimating $\epsilon_i(t) = x(t) - d(t)1$ we observe that such an estimate is given by $\hat{\beta}$. From the solution of (63), we have

$$
\hat{\beta} = \frac{x^i}{1+\nu} - \frac{\nu 1^T x^i (1+\nu) 1^T u^i}{M_i(1+2\nu)(1+\nu)}\mathbf{1} \tag{65}
$$

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

$$
\nu = arg\min \frac{\|(A^TA + \nu B^TB)^{-1}A^T(x^i, u^i)^T\|}{Tr(A^TA + \nu B^TB)^{-1}}
$$

Typically the GCV approach is computationally expensive since the trace of the matrix $(A^TA + \nu B^TB)^{-1}$ 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

$$
\begin{aligned}
\nu &= arg\min \frac{\|(A^TA + \nu B^TB)^{-1}A^T(x^i, u^i)^T\|}{tr(A^TA + \nu B^TB)^{-1}} \\
&\leq arg\min \frac{\|(A^TA + \nu B^TB)^{-1}A^T\|}{tr(A^TA + \nu B^TB)^{-1}} \|(x^i, u^i)^T\|.
\end{aligned} \tag{66}
$$

a sub-optimal value of ν can be computed solving the right hand side of (66). Notice that the first term in the right hand side of (66) 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 (65) with the parameter ν computed from (66) we can then estimate the error mean and covariance matrix applying (59) and (60), respectively.

4.6 Sub-optimal approximation of estimation bias

The previous section explains the estimation of error covariance matrix, which in turn requires computation of two parameters naming, $\beta(t)$ and $\nu(t)$. The computation of these parameter causes the increase in the over-all computational complexity of the whole distributed estimation scheme. We propose an alleviation of computing $\beta(t)$ by utilizing the fact that in most of the real world data acquisition systems, the contaminating noise lies in frequency bands beyond the signal's spectrum. Since the data is low-pass filtered prior to estimation, we can assume that most of the noise power has been removed with the application of this filter, and thus (61) can be re-written as

$$x^{(i)}(t) = d(t)\mathbf{1} + \tilde{\beta}(t) \tag{67}$$

where

$$\tilde{\beta}(t) = \tilde{\beta}_i(t)\mathbf{1}$$

and

$$\tilde{\beta}_i(t) = u_{\nu_i'}(t) - u_i(t)$$

Though the accuracy of $\tilde{\beta}_i$ is less than $\hat{\beta}_i$, the reduced computational efforts justify this approximation in the practical estimation systems.

5 Implementation Structure and Numerical Implementation

This section presents the general implementation structure layout of the proposed estimation. Moreover, a figure illustrating the estimator structure and later on the algorithmic implementation is shown followed by some numerical results.

5.1 Layout for the proposed distributed estimation filter design

The general elementary implementation layout showing the connection between different steps can be described in Table 1 where the first step is involved towards initial conditions for the time delay, the second step defines the steps for the implementation for the FIR low pass filter. The third step shows the estimator stability conditions, which is followed by computational complexity completing the layout for the proposed distributed estimation filter design.

5.2 Estimator Structure and Implementation

Fig. 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 program 5.2. Initially, the distributed computation of the threshold is performed (lines 1-8): node i updates its threshold ψ_i until a given precision $\bar{\omega}$ is reached. In the computations of ψ_i, we chose $\alpha_{i,j}^{(i)} = |N_j \cap N_i|/(M_i - 1)$ and $\alpha_{i,j}^{(j)} = |N_j \cap N_i|/(M_j - 1)$. 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 3, and the bounds on the error variance in Section 4.3, ensure that estimates among nodes have similar performance.

Numerical results show that the while-loop (lines 4-8) converges after about 10-20 iterations. Line 9 and 10 calculate the time delay of node i and all of its neighbor nodes based on the knowledge of propagation speed c. Since there is delaying and advancing of the signal is involved, the farthest node is taken as the reference point and the corresponding time delay τ_i is taken as the starting time of the algorithm (line 11). The input signal at each node is then delayed or advanced, depending upon its geometrical location relative to the node i (line 12-16).These delay-adjusted signal are then filtered to remove the higher frequency noise (line 17-18). The estimators for the local mean estimation error and the local covariance matrix are then initialized (lines 19-20). The main loop of the estimator is lines 23-38. Lines 24-29 are related to the left subsystem of Fig. 1.

The optimal weights are computed using Equations (35) and (36) (lines 27-28). Notice that the optimal Lagrangian multiplier ξ_i is computed using the function bisection which takes as argument the interval $\max(0, \sigma^2\sqrt{M_i\psi_i} - \lambda_{\min}(\Gamma_i(t-1)))$ 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 Gershgorin's theorem can be used instead. The estimate of $d(t)$ is computed in line 29. Lines 30-38 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 4.5 and 4.6. Sample mean and covariance of the estimation error are updated in lines 36-37. These formulas correspond to recursive implementation of (59) and (60). As an option $\tilde{\beta}_i(t)$ can be computed using 68.

With regards to the inversions of the estimated error covariance matrix $\hat{\Gamma}_i$ in lines 27-28. 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.

5.3 Simulation Results

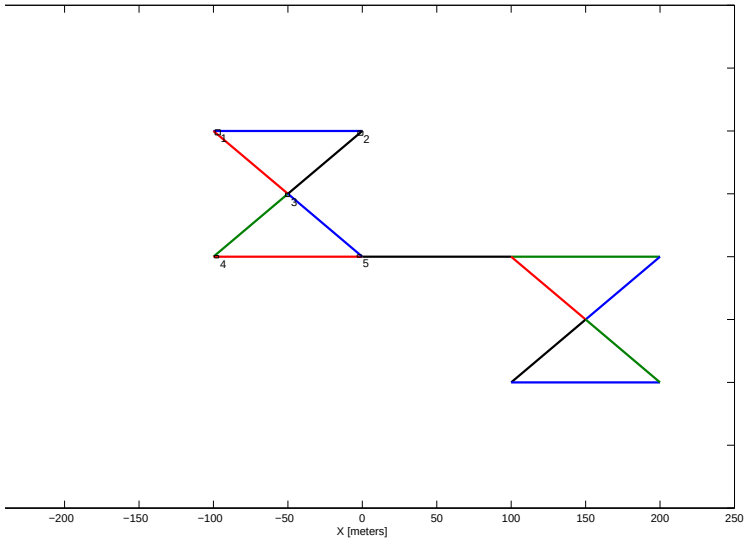
This section shows the simulation results of the proposed filter. Numerical simulations have been carried out in order to validate performance of the proposed distributed estimator. We have simulated a scenario of $N = 10$ nodes, with the structure of the sensor network shown in the Fig. 2, where each vertex is the location of a node. Fig. 3 shows the original data $d(t)$, and input data $u_{(\tau_i, v_i')}(t)$ at $u_{\tau_1, v_1'}(t)$ is presented from node 1 to 5, and node 6 to 10 in Fig. 4 and 5 respectively. Figs. 6 and 7 show the estimations x_1 to x_{10} for nodes 1 to 10. The data generation from the sensor networks goes through several steps as proposed in the algorithm which during simulation also involves signal generation from all the 10 nodes, FIR filtering, phase shift calculations and finally the estimation from node 1 to node 10. It has been that how the estimates have been improved by proposed filter algorithm weights, such that the variance of the estimation errors is minimized, thereby improving the estimation results. Moreover, in Fig. 8 and Fig. 9, it has been shown the comparison of the $d(t)$ with its estimate at a specific node 6 and 10, thus showing the effectivenss of the proposed filter.

6 Conclusion and Future Work

In this paper, an improved distributed minimum variance estimation algorithm has been developed by considering the propagation delay of the physical quantity to be measured for static SNs. The proposed filter

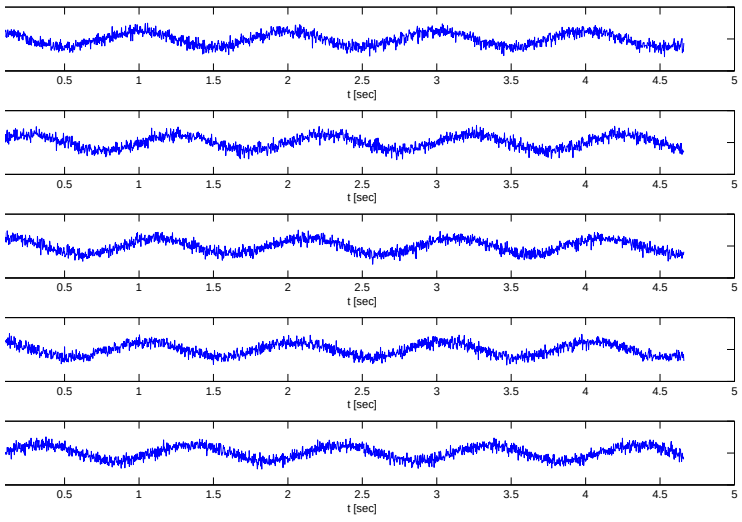
Program 5.2.0.1 Estimation algorithm for node i

STATE $t := 0$
$\psi_i(t-1) = 0$
$\psi_i(t) = 1/M_i$
$|\psi_i(t) - \psi_i(t-1)| \geq \bar{\omega} = 10^{-10}$
$\psi_i(t+1) = T_i(\psi(t))$
Collect thresholds from nodes in Θ_i
$t := t + 1$
ENDWHILE
STATE $\tau_i = \frac{c}{r_i}$
$\tau_j = \frac{c}{r_j}, j \in M_i$
$\tau_j = \frac{c}{r_j}, j \in M_i$
$t := max(\tau_i), i = 1, 2, 3, ..., N$
IF $r_j < r_i$
$u_{v'_j}(t) = u_{(\tau_j, v'_j)}(t - \tau_j)$
ELSE IF $r_i < r_j$
$u_{v'_j)}(t) = u_{(\tau_j, v'_j)}(t + \tau_j)$
END
STATE $u_i(t) = Filter(u_{v'_i)}(t))$
$u_i(t) = Filter(u_{v'_i)}(t))$
$u_j(t) = Filter(u_{v'_j)}(t)), j \in M_i$
$t := max(\tau_i), i = 1, 2, 3, ..., N$
$\hat{m}_{e_i}(t) := 0$
$\hat{\Gamma}_i(t) := \sigma^2 I$
$x_i(t) := u_i(t)$
WHILE FOREVER
STATE $M_i := |N_i|$
$t := t + 1$
$\xi_i = \text{bisection}(\max[0, \sigma^2/\sqrt{M_i\psi_i} - \lambda_{\min}(\Gamma_i(t-1))])$
$\kappa_i(t) := \frac{\sigma^2(\hat{\Gamma}_i(t-1)+\xi_i I)^{-1}1}{M_i+\sigma^2 1^T(\hat{\Gamma}_i(t-1)+\xi_i I)^{-1}1}$
$\eta_i(t) := \frac{1}{M_i+\sigma^2 1^T(\hat{\Gamma}_i(t-1)+\xi_i I)^{-1}1}$
$x_i(t) := \sum_{j\in N_i} \kappa_{i_j} x_j(t-1) + \sum_{j\in N_i} \eta_{i_j}(t) u_j(t)$
$\hat{\beta} := \frac{x^i}{1+\nu} - \frac{\nu 1^T x^i + (1+\nu) 1^T u^i}{M_i(1+2\nu)(1+\nu)} 1$
OR
$\tilde{\beta}_i(t) := u_{\nu'_i}(t) - u_i(t)$
$\hat{\epsilon}_i := \hat{\beta}$
OR
$\hat{\epsilon}_i := \tilde{\beta}$
$\hat{m}_{e_i}(t) := \frac{t-1}{t}\hat{m}_{e_i}(t-1) + \frac{1}{t}\hat{\epsilon}_i(t)$
$\hat{\Gamma}_i(t) := \frac{t-1}{t}\hat{\Gamma}_i(t-1) + \frac{1}{t}(\hat{\epsilon}_i(t) - \hat{m}_{e_i}(t))(\hat{\epsilon}_i(t) - \hat{m}_{e_i}(t))^T$
ENDWHILE



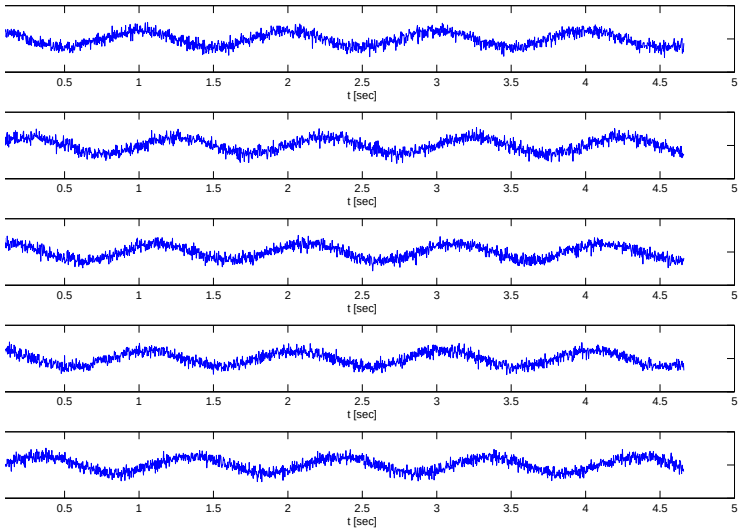
..

Figure 2: *Structure of Sensor network. Note that, for example, node 1 has node 2 and node 3 as neighbors, while node 3 has node 1, node 2, node 4, and node 5 as neighbors.*



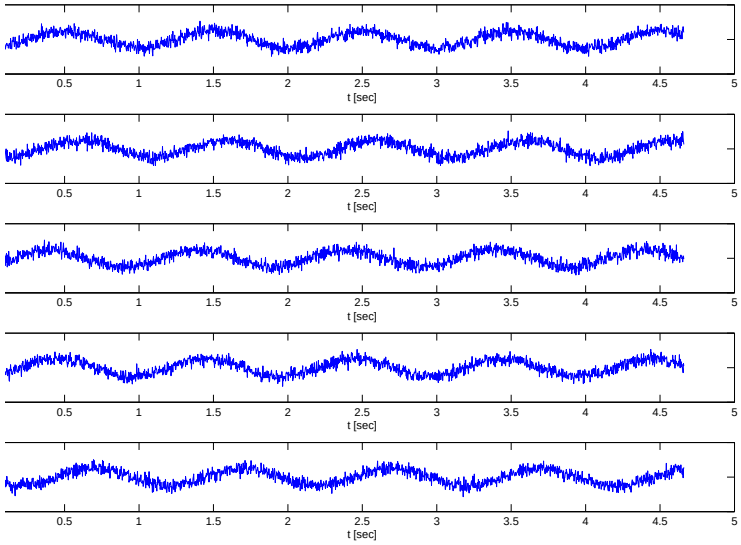
..

Figure 3: The original signal $d_(t)$



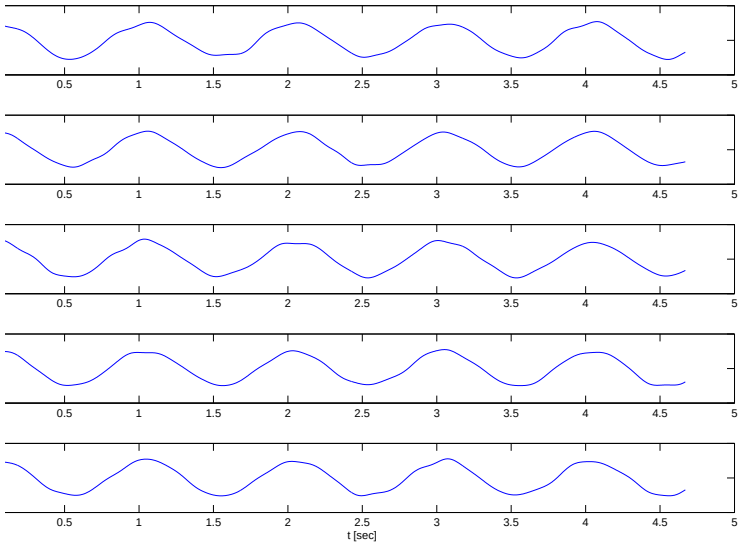
..

Figure 4: *The input $u_{\tau_i,v_i'}(t)$ at node i, for i=1,2,... 5*



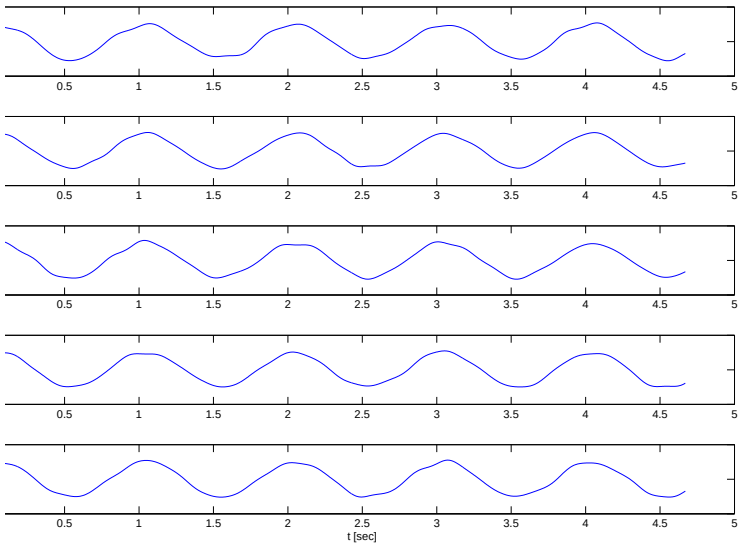
..

Figure 5: The input $u_{\tau_i,v_i'}(t)$ at node i, for i=6,7,... 10



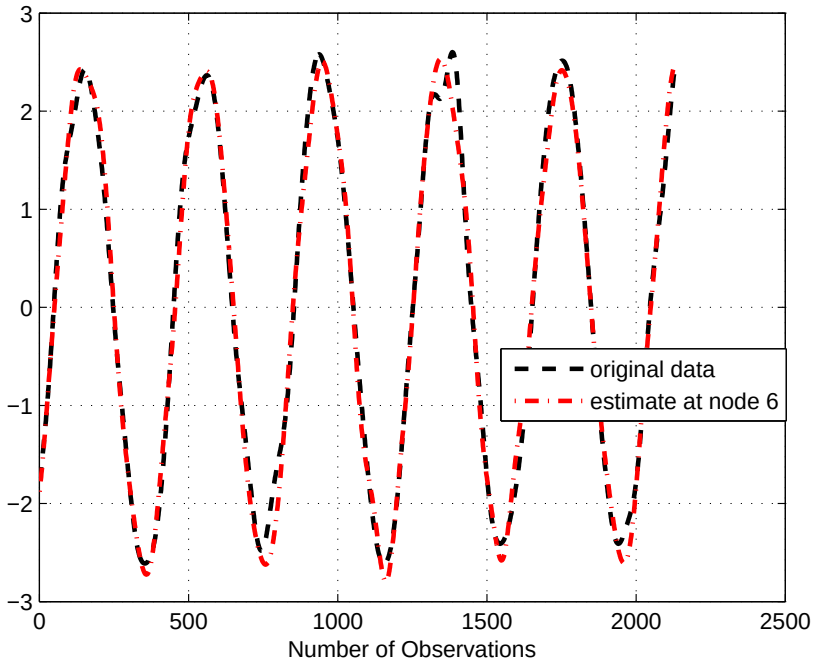
..

Figure 6: Estimates of $d(t)$ at node 1 to node 5



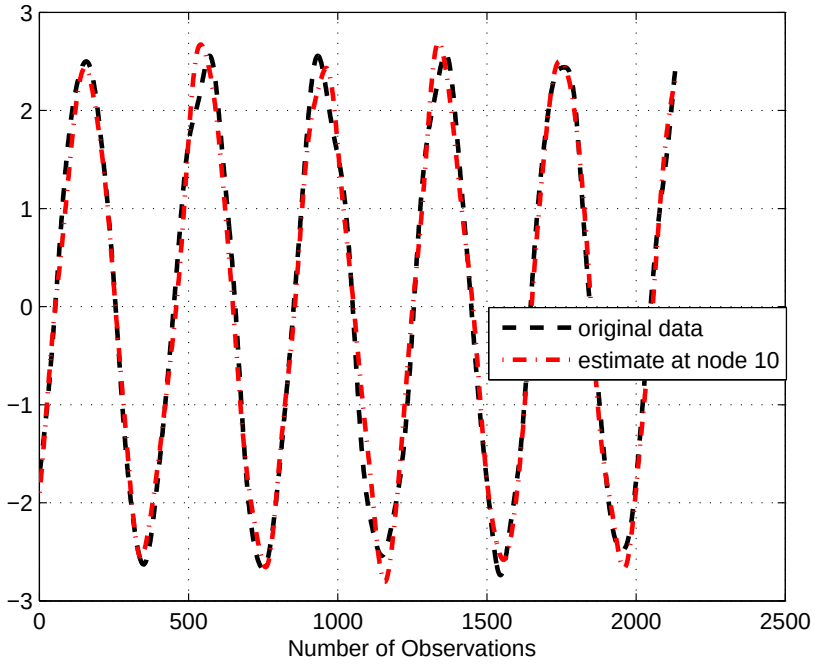
..

Figure 7: Estimates of $d(t)$ at node 6 to node 10



..

Figure 8: Comparison of $d(t)$ and estimated $d(t)$ at node 6



..

Figure 9: Comparison of $d(t)$ and estimated $d(t)$ at node 10

Table 1: LAYOUT FOR THE PROPOSED DISTRIBUTED ESTIMATION FILTER DESIGN

Step 1: Initial condition for time-delay

- Data Location: known

- Sensor nodes (Spatial ordinates): known

- Calculate time-delay b/w source data and any node in the network

$\tau_i = \frac{c}{r_i}$ (τ_i: *ith node time, c: propagation speed of light,* r_i: *distance b/w signal source & ith node which is known*)

$u_{(\tau_i, v_i')}(t) = d(t-\tau_i) + v_i'(t)$, (*ith node signal measurement for common scalar signal* $d(t)$ *with delay* τ_i)

$x(t) = K(t)x(t-1) + H(t)u(t)$ (*vector notation of neighboring estimates and measurements*)

- Convergence of centralized estimation error

Step 2: Low Pass FIR Filter

- Filter Weights: (*quadratic cost function and constraints*)

Inner loop: state estimate

Outer loop: error covariance estimate

- Introduce vectors $\kappa_i^T(t), \eta_i^T(t) \in R^{M_i}$ (*corresponding to non-zero elements* $K(t)$ *and* $H(t)$)

$\mathbf{E}e_i(t) = \kappa_i^T(t)\mathbf{E}\epsilon_i(t-1) - \kappa_i^T(t)\delta(t)1$, (*expected value of estimation error at node* i)

$\mathsf{P}_1 : \min_{\kappa_i(t),\eta_i(t)} \ \kappa_i^T(t)\Gamma_i(t-1)\kappa_i(t) + \sigma^2\eta_i^T(t)\eta_i(t) \ s.t. \quad (\kappa_i(t)+\eta_i(t))^T\mathbf{1} = 1$

and $\gamma_{\max}(K(t)) \leq f(\Delta, \Upsilon)$ (*optimization problem to minimize variance of estimation error*)

- Local optimization problem:

$\mathsf{P}_2 : \min_{\kappa_i(t),\eta_i(t)} \ \kappa_i(t)^T\Gamma_i(t-1)\kappa_i(t) + \sigma^2\eta_i(t)^T\eta_i(t) \ s.t. (\kappa_i(t)+\eta_i(t))^T 1 = 1 \ and \ \|\kappa_i\|^2 \leq \psi_i,$

(*For a given positive definite matrix* $\Gamma_i(t-1)$)

$\kappa_i(t) = \frac{\sigma^2(\Gamma_i(t-1)+\xi_i I)^{-1}\mathbf{1}}{\sigma^2\mathbf{1}^T(\Gamma_i(t-1)+\xi_i I)^{-1}\mathbf{1}+M_i}$ with $\xi_i \in [0, \max(0, \sigma^2/\sqrt{M_i\psi_i} - \lambda_{\min}(\Gamma_i(t-1)))]$

Step 3: Condition of Estimator Stability

- IF signal to track slow varying (*Each node estimate converge to the neighborhood*)

- Each node estimate has small variance and small bias

- Trade off: Amount of communication and resulting estimation quality

Step 4: Computational Complexity

- $x^{(i)}(t) = d(t)\mathbf{1} + \tilde{\beta}(t)$ (*where* $\tilde{\beta}(t)$=$\tilde{\beta}_i(t)\mathbf{1}$ *and* $\tilde{\beta}_i(t)$=$u_{\nu_i'}(t) - u_i(t)$ *& accuracy of* $\tilde{\beta}_i$ *is less than* $\hat{\beta}_i$)

- The reduced computational efforts justify this approximation in the practical estimation systems

developed has been established providing some improved features. It has incorporated an FIR filter, which aims at simplifying the signal estimation and alleviating the parameter optimization, thereby having a capability of tracking time-varying signal (environmental time-variant physical variables) without a pre-assigned coupling law imposed among the nodes. The proposed filter has also been designed considering a more general model of the filter structure, where the parameters were computed through distributed algorithms. The algorithm's filter weights were designed such that the variance of the estimation errors is minimized, thereby improving the filter design considerably and characterizing the performance limit of the filter. Simulation results have illustrated the effectiveness of the developed filter.

The research can be extended to distributed estimation with time varying delay and state delay considering the network control system.

Acknowledgments

The authors would like to thank the deanship for scientific research (DSR) at KFUPM for support through research group project **RG1105-1**.

References

[1] Saber, R. O., and Murray, R. M., 'Consensus problems in networks of agents with switching topology and time-delays', *IEEE Trans. Automat. Contr.*, vol. 49(9), pp. 1520–1533, September 2004.

[2] Carli, R., Fagnani, F., Speranzon, A., and Zampieri, S., 'Communication constraints in the average consensus problem', *Automatica*, vol. 44(3), pp. 671–684, March 2008.

[3] Xiao, L., Boyd, S.,and Lall, S., 'A scheme for robust distributed sensor fusion based on average consensus', *Proc. IEEE IPSN*, pp. 63–70, April 2005.

[4] Sinopoli, B., Sharp, C., Schenato, L., Schaffert, S., and Sastry, S., 'Distributed control applications within sensor networks', *Proc. the IEEE, Special Issue on Distributed Sensor Networks*, vol. 91(8), pp. 1235–1246, August 2003.

[5] Speranzon, A., Fischione, C., Johansson, K. H., and Sangiovanni, A., 'A distributed minimum variance estimator for sensor networks', *IEEE J. Selected Areas in Communications*, vol. 26(4), pp. 609–621, May 2008.

[6] Carli, R., Chiuso, A., Schenato, L., and Zampieri, S., 'Distributed Kalman filtering based on consensus strategies', *IEEE J. Selected Areas in Communications*, vol. 26(4), pp. 622–632, May 2008.

[7] Speranzon, A., Fischione, C., and Johansson, K. H., 'Distributed and collaborative estimation over wireless sensor networks', *Proc. IEEE Conference on Decision and Control*, pp. 1025–1030, December 2006.

[8] Estrin, D., Govindan, R., Eidemann, J., and Kumar, S., 'Next century challenges: Scalable coordination in sensor networks', *Proc. the ACM/IEEE International Conference on Mobile Computing and Networking*, pp. 263–270, 1999.

[9] Tanikawara, M., Ohba, K., Kubo, Y., and Sugimoto, S., 'The rice distribution and ML positioning using received signal powers in sensor networks', *International Journal of Innovative Computing, Information and Control*, vol.8(3(B)), pp. 2359–2374, 2012.

[10] Liu, Y., Pu, J., Shu, L., and Xiong, Z., 'A hierarchical congestion avoidance algorithm in wireless sensor networks', *International Journal of Innovative Computing, Information and Control*, vol.8(7(B)), pp. 5199–5210, 2012.

[11] Sun, H., Chen, C., and Li, P., 'A lightweight secure data aggregation protocol for wireless sensor networks', *International Journal of Innovative Computing, Information and Control*, vol.8(10(A)), pp.6503–6514, 2012.

[12] Gharavi, H., and Kumar, P. R., Eds., *Proc. IEEE: Special Issue on Sensor Networks and Applications*, vol. 91, pp. 1151–1153, August 2003.

[13] Spanos, D. P., Saber, R. O., and Murray, R. M., 'Approximate distributed Kalman filtering in sensor networks with quantifiable performance', *Proc. IEEE Conference on Decision and Control*, pp. 133–139, April 2005.

[14] Saber, R. O., and Shamma, J. S., 'Consensus filters for sensor networks and distributed sensor fusion', *Proc. IEEE Conference on Decision and Control*, pp. 6698–6703, December 2005.

[15] Gao, H., and Chen, T., 'Estimation for uncertain systems with limited communication capacity', *IEEE Transactions on Automatic Control*, vol. 52(11), November 2007.

[16] Mahmoud, M. S., and Shi, P., 'Asynchronous H_∞ filtering of discrete-time switched systems', *Signal Processing*, vol.92(10), pp. 2356–2364, 2012.

[17] Xiao, L., Boyd, S., and Kim, S. J., 'Distributed average consensus with least-mean-square deviation', *J. of Parallel and Distributed Computing*, vol. 67(1), pp. 33–46, 2007.

[18] Xiao, S. B. L., 'Fast linear iterations for distributed averaging', *System & Control Letter*, vol. 53, pp. 65–78, 2004.

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

[20] Hong, Y.-W., Cheow, L., and Scaglione, A., 'A simple method to reach detection consensus in massively distributed sensor networks', *Proc. IEEE Int. Symposium on Information Theory*, 2004.

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

[22] Barbarossa, S., and Scutari, G., 'Decentralized maximum likelihood estimation for sensor networks composed of nonlinearly coupled dynamical system', *IEEE Trans. Signal Processing*, 2007.

[23] Zhao, F., Liu, J., Guibas, L., and Reich, J., 'Collaborative signal and information processing: an information-directed approach', *Proc. IEEE*, vol. 91(8), 2003.

[24] Liu, J., Reich, J., and Zhao, F., 'Collaborative in-network processing for target tracking', *European Journal on Applied Signal Processing*, no. 4, pp. 378–391, 2003.

[25] Shang, Y., Ruml, W., Zhang, Y., and Fromherz, M., 'Localization from connectivity in sensor networks', *IEEE Trans. Parallel and Distrib. Sys.*, vol. 15(11), 2004.

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

[27] Luo, Z.-Q., Gatspar, M., Liu, J., and Swami, A., Eds., *IEEE Signal Processing Magazine: Special Issue on Distributed Signal Processing in Sensor Networks*, vol. 28(3), 2006.

[28] Saber, O., R., and Shamma, J., 'Consensus filters for sensor networks and distributed sensor fusion', In *Proc. 44th IEEE Conf. Decision Control*, Seville, Spain, pp. 6698–6703, Dec. 2005.

[29] Xiao, L., and Boyd, S., 'Fast linear iterations for distributed averaging', *Syst. Controls Lett.*, vol. 53(1), pp. 65–78, Apr. 2004.

[30] Carli, R., Chiuso, A., Schenato, L., and Zampieri, S., 'Distributed kalman filtering based on consensus strategies', Information Engineering Dept., Univ. of Padova, Padova, Italy, Tech. Rep., 2007.

[31] Xiao, L., and Boyd, S., 'Designing fast distributed iterations via semi-definite programming', presented at the *Workshop on Large Scale Nonlinear and Semidefinite Programming*, Waterloo, Canada, May 2004.

[32] Kar, S., and Moura, J. M. F., 'Sensor networks with random links: Topology design for distributed consensus', *IEEE Trans. Signal Process.*, vol. 56(7), pp. 3315–3326, Jul. 2008.

[33] Gu, D., 'Distributed EM algorithm for gaussian mixtures in sensor networks', *IEEE Transactions on Neural Networks*, vol. 19(7), pp. 1154–1166, July 2008.

[34] Saber, O., R., 'Distributed Kalman filter with embedded consensus filters', In: *Proc. 44th IEEE Conf. Decision Control*, Dec. 12–15, pp. 8179–8184, 2005.

[35] Mansouri, M., Snoussi, H., and Richard, C., 'Channel estimation and multiple target tracking in wireless sensor networks based on quantised proximity sensors', *IET Wireless Sensor Networks*, Vol. 1(1) , pp. 7–14, 2011.

[36] Tamee, K., Mitatha, S., and Yupapin, P. P., 'Distributed photon network sensors via a wavelength router', *IET Wireless Sensor Networks*, vol. 1(1), pp. 26–30, 2011.

[37] Paul, B., and Matin, M. A., 'Optimal geometrical sink location estimation for two-tiered wireless sensor networks', *IET Wireless Sensor Networks*, vol. 1(2), pp. 74–84, 2011.

[38] Liu, G., Xu, B., Zeng, M., and Chen, H., 'Distributed estimation over binary symmetric channels in wireless sensor networks', *IET Wireless Sensor Networks*, vol. 1(2), pp. 105–109, 2011.

[39] Abedi, A., 'Power-efficient-coded architecture for distributed wireless sensing', *IET Wireless Sensor Networks*, vol. 1(3), pp. 129–136, 2011.

[40] Farah, C., Schwaner, F., Abedi, A., and Worboys, M., 'Distributed homology algorithm to detect topological events via wireless sensor networks', *IET Wireless Sensor Networks*, vol. 1(3), pp. 151–160, 2011.

[41] Cui, X. -X., Fang, Z., Zhou, P. -C., and Liu, K., 'Compact distributed target-tracking algorithm with binary sensors', *IET Wireless Sensor Networks*, vol. 1(4), pp. 218–228, 2011.

[42] Gao, H., 'New results on stability of discrete-time systems with time-varying state delay', *IEEE Transactions on Automatic Control*, vol. 52(2), pp. 328–334, 2007.

[43] Gao, H., and Chen, T., 'H_∞ estimation for uncertain systems with limited communication capacity', *IEEE Transactions on Automatic Control*, vol. 52(11), pp. 2070–2084, November 2007.

[44] Gao, H., Chen, T., Lam, J., 'A new delay system approach to network-based control', em Automatica, vol. 44(1), pp. 39-52, January 2008.

[45] Li, H., Chen, B., Zhou, Q., Qian, W., 'Robust stability for uncertain delayed fuzzy hopfield neural networks with markovian jumping parameters', *Systems, Man, and Cybernetics, Part B: Cybernetics, IEEE Transactions on*, vol. 39(1), pp. 94–102, Feb. 2009.

[46] Luan, X., Shi, P., and Liu, F., 'Stabilization of networked control systems with random delays', *IEEE Trans on Industrial Electronics*, vol.58(9), pp. 4323–4330, 2011.

[47] Zhang, J., Xia, Y., Shi, P., and Mahmoud, M. S., 'New results on stability and stabilization of systems with interval time-varying delay', *IET Control Theory and Applications*, vol.5(3), pp. 429–436, 2011.

[48] Yang, R., Shi, P., and Liu, G., 'Filtering for discrete-time networked nonlinear systems with mixed random delays and packet dropouts, *IEEE Trans on Automatic Control*, vol.56, no.11, pp. 2655–2660, 2011.

[49] Yang, R., Shi, P., Liu, G., and Gao, H., 'Network-based feedback control for systems with mixed delays based on quantization and dropout compensation', *Automatica*, vol.47(12), pp. 2805–2809, 2011.

[50] Mahmoud, M. S., and Shi, P., 'Persistent bounded disturbance rejection for discrete-time delay systems, *Int. J. of Systems Science*, vol.42(6), pp. 921–930, 2011.

[51] Bertsekas, D. P., and Tsitsiklis, J. N., *Parallel and Distributed Computation: Numerical Methods*, Athena Scientific, 718 pages, 1997.

[52] Giridhar, A., and Kumar, P., 'Distributed clock synchronization over wireless networks: Algorithm and analysis', *Proc. 45th IEEE Conference on Decision and Control*, 2006.

[53] Alriksson, P., and Rantzer, A., 'Distributed Kalman filtering using weighted averaging', *Proc. 17th Int. Symposium on Mathematical Theory of Networks and Systems*, 2006.

[54] IEEE Standard for Information technology, Telecommunications and information exchange between systems, Local and metropolitan area networks, Specific requirements, 'Wireless Lan medium access control (MAC) and physical layer specification', *IEEE*, New York, NY, USA, IEEE std 802.11-1997 edition, Tech. Rep., 1997.

[55] Tomlin, C., 'STARMAC, Stanford testbed of autonomous rotor craft for multi-agent control', *Research project, Available: http://hybrid.stanford.edu/starmac/overview*.

[56] Ljung, L., and Walhberg, B., 'Asymptotic properties of the least-squares method for estimating transfer functions and disturbance spectra', *Advances in Applied Probability*, vol. 24(2), 1992.

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

[58] Boyd, S., and Vandenberghe, L., 'Convex optimization', *Cambridge University Press*, 2004.

[59] Ljung, L., 'System Identification: Theory for the User', 2nd Ed.. Prentice Hall, 1998.