

A Linear Distributed Filter Inspired by the Markovian Jump Linear System Filtering Problem

Ion Matei ^a, John S. Baras ^b

^a*Institute for Research in Electronics and Applied Physics, University of Maryland, College Park 20742*

^b*Institute for Systems Research and Department of Electrical and Computer Engineering, University of Maryland, College Park 20742*

Abstract

In this paper we introduce a consensus-based distributed filter, executed by a sensor network, inspired by the Markovian jump linear system filtering theory. We show that the optimal filtering gains of the Markovian jump linear system can be used as an approximate solution of the optimal distributed filtering problem. This parallel allows us to interpret each filtering gain corresponding to a mode of operation of the Markovian jump linear system as a filtering gain corresponding to a sensor in the network. The approximate solution can be implemented distributively and guarantees a quantifiable level of performance.

Key words: distributed filtering, Markovian jump systems, state estimation

1 Introduction

A fundamental problem in sensor networks is developing distributed algorithms for estimating the state of a process of interest. The goal of each sensor is to compute accurate state estimates of the process in a distributed manner, that is, using only local information. The distributed filtering (estimation) problem received a lot of attention during the past thirty years. Major contributions were made by Borkar, Varaiya and Teneketzis [1,9], who addressed the distributed estimation problem of a random variable by a group of sensors. The recent technological advances in mobile sensor networks have re-ignited the interest in the distributed estimation problem. Most papers focusing on distributed estimation propose different mechanisms for combining the standard Kalman filter with a consensus filter in order to ensure that the estimates asymptotically converge to the same value [2,5–7]. More recent contributions on the design of distributed decentralized estimators can be found in [8].

* This material is based upon work supported in part by the US Air Force Office of Scientific Research MURI award FA9550-09-1-0538, in part by the Defence Advanced Research Projects Agency (DARPA) under award number 013641-001 for the Multi-Scale Systems Center (MuSyC) through the FRCP of SRC and DARPA.

*Corresponding author J. S. Baras. Tel. +1 301 405 6606. Fax +1 301 314 8486.

Email addresses: imatei@umd.edu (Ion Matei),
baras@umd.edu (John S. Baras).

In this paper we argue that the optimal filtering gains of a particular Markovian jump linear system (MJLS) can be used as an approximate solution of the optimal consensus-based distributed state estimation scheme. Namely, we show that each (scaled) filtering gain corresponding to a mode of operation of an (appropriately defined) MJLS can be used as a local filtering gain corresponding to a sensor in the network. In general, heuristic, sub-optimal schemes are used to compute state estimates distributively, for which the performance of these schemes is difficult to quantify. Although we do not compute the exact filtering performance under our sub-optimal solution, we are able to *guarantee* a certain level of performance which can be *evaluated*. Partial results of this paper were presented in [4].

Paper structure: In Section 2 we describe the models used in the MJLS and the distributed filtering problems. In Section 3 we introduce the optimal linear filtering gains of an appropriately defined MJLS, while in Section 4 we show why these filtering gains can be used as an approximate solution for the optimal distributed filtering problem. We conclude with a numerical example in Section 5.

Remark 1 Given a positive integer N , a set of vectors $\{x_i\}_{i=1}^N$, a set of matrices $\{A_i\}_{i=1}^N$ and a set of non-negative scalars $\{p_i\}_{i=1}^N$ summing up to one, the following holds

$$\left(\sum_{i=1}^N p_i A_i x_i \right) \left(\sum_{i=1}^N p_i A_i x_i \right)' \leq \sum_{i=1}^N p_i A_i x_i x_i' A_i'.$$

2 Problem formulation

In this section, we first describe the distributed estimation model, followed by the description of a particular MJLS.

2.1 Distributed estimation model

We consider a discrete-time, linear stochastic process, given by

$$x(k+1) = Ax(k) + w(k), \quad x(0) = x_0, \quad (1)$$

where $x(k) \in \mathbb{R}^n$ is the state vector and $w(k) \in \mathbb{R}^n$ is a driving noise, assumed Gaussian with zero mean and covariance matrix Σ_w . The initial condition x_0 is assumed to be Gaussian with mean μ_0 and covariance matrix Σ_0 . The state of the process is observed by a network of N sensors indexed by i , whose sensing models are given by

$$y_i(k) = C_i x(k) + v_i(k), \quad i = 1 \dots N, \quad (2)$$

where $y_i(k) \in \mathbb{R}^{r_i}$ is the observation made by sensor i and $v_i(k) \in \mathbb{R}^{r_i}$ is the measurement noise, assumed Gaussian with zero mean and covariance matrix Σ_{v_i} . We assume that the matrices $\{\Sigma_{v_i}\}_{i=1}^N$ and Σ_w are positive definite and that the initial state x_0 , the noises $v_i(k)$ and $w(k)$ are independent for all $k \geq 0$.

The set of sensors form a communication network whose topology is modeled by a directed graph that describes the information exchanged among agents. The goal of the agents is to distributively (using only local information) compute estimates of the state of the process (1). Let $\hat{x}_i(k)$ denote the state estimate computed by sensor i at time k . The sensors update their estimates in two steps. In the first step, an intermediate estimate, denoted by $\varphi_i(k)$, is produced using a Luenberger observer filter

$$\varphi_i(k) = A\hat{x}_i(k) + L_i(k)(y_i(k) - C_i\hat{x}_i(k)), \quad i = 1 \dots N, \quad (3)$$

where $L_i(k)$ is the *filter gain*.

In the second step, the new state estimate of sensor i is generated by a convex combination between $\varphi_i(k)$ and all other intermediate estimates within its communication neighborhood, i.e.,

$$\hat{x}_i(k+1) = \sum_{j=1}^N p_{ij} \varphi_j(k), \quad i = 1 \dots N, \quad (4)$$

where p_{ij} are the non-negative entries of a stochastic matrix (rows sum up to one) $P = (p_{ij})$, whose structure is induced by the communication topology (i.e., $p_{ij} = 0$ if no link from j to i exists). Combining (3) and (4) we obtain the dynamic equations for the *consensus based distributed filter*

$$\hat{x}_i(k+1) = \sum_{j=1}^N p_{ij} \left[A\hat{x}_j(k) + L_j(k)(y_j(k) - C_j\hat{x}_j(k)) \right], \quad (5)$$

for $i = 1 \dots N$.

Assumption 2 For reasons we will make clear later in the paper, we assume that the matrix $P = (p_{ij})$ is doubly stochastic.

2.2 Markovian jump linear system model

We define the following MJLS

$$\begin{aligned} \xi(k+1) &= \tilde{A}_{\theta(k)} \xi(k) + \tilde{B}_{\theta(k)} \tilde{w}(k) \\ z(k) &= \tilde{C}_{\theta(k)} \xi(k) + \tilde{D}_{\theta(k)} \tilde{v}(k), \quad \xi(0) = \xi_0, \end{aligned} \quad (6)$$

where $\xi(k)$ is the state, $z(k)$ is the output, $\theta(k) \in \{1, \dots, N\}$ is a Markov chain with probability transition matrix P' , $\tilde{w}(k)$ and $\tilde{v}(k)$ are independent Gaussian random variables with zero mean and identity covariance matrices. Additionally, ξ_0 is a Gaussian noise with mean μ_0 and covariance matrix Σ_0 . We denote by $\pi_i(k)$ the probability distribution of $\theta(k)$ ($\Pr(\theta(k) = i) = \pi_i(k)$) and we assume that $\pi_i(0) > 0$, for all i . We have that $\tilde{A}_{\theta(k)} \in \{\tilde{A}_i\}_{i=1}^N$, $\tilde{B}_{\theta(k)} \in \{\tilde{B}_i\}_{i=1}^N$, $\tilde{C}_{\theta(k)} \in \{\tilde{C}_i\}_{i=1}^N$ and $\tilde{D}_{\theta(k)} \in \{\tilde{D}_i\}_{i=1}^N$, where the index i refers to the state i of $\theta(k)$. We set

$$\tilde{A}_i = A, \tilde{B}_i = \frac{\sqrt{\pi_i(0)}}{\sqrt{\pi_i(k)}} \Sigma_w^{1/2}, \tilde{C}_i = \frac{1}{\sqrt{\pi_i(0)}} C_i, \tilde{D}_i = \frac{1}{\sqrt{\pi_i(k)}} \Sigma_{v_i}^{1/2}, \quad (7)$$

for all i . Note that since P is assumed doubly stochastic and $\pi_i(0) > 0$, we have that $\pi_i(k) > 0$ for all $i, k \geq 0$. In addition, ξ_0 , $\theta(k)$, $\tilde{w}(k)$ and $\tilde{v}(k)$ are assumed independent for all $k \geq 0$. The random process $\theta(k)$ is also called *mode*.

3 Markovian jump linear system filtering

In this section we introduce an optimal linear filter for the state estimation of MJLSs. Assuming that the mode is directly observed, a linear filter for the state estimation is given by

$$\hat{\xi}(k+1) = \tilde{A}_{\theta(k)} \hat{\xi}(k) + M_{\theta(k)}(z(k) - \tilde{C}_{\theta(k)} \hat{\xi}(k)), \quad (8)$$

where we assume that the filter gain $M_{\theta(k)}$ depends only on the current mode. The dynamics of the estimation error $e(k) \triangleq \xi(k) - \hat{\xi}(k)$ is given by

$$e(k+1) = (\tilde{A}_{\theta(k)} - M_{\theta(k)} \tilde{C}_{\theta(k)}) e(k) + \tilde{B}_{\theta(k)} \tilde{w}(k) - M_{\theta(k)} \tilde{D}_{\theta(k)} \tilde{v}(k). \quad (9)$$

Let $\Gamma(k)$ denote the covariance matrix of $e(k)$, i.e., $\Gamma(k) \triangleq E[e(k)e(k)']$. We define also the covariance matrix of $e(k)$, when the system is in mode i , i.e. $\Gamma_i(k) \triangleq E[e(k)e(k)' \mathbb{1}_{\{\theta(k)=i\}}]$, where $\mathbb{1}_{\{\theta(k)=i\}}$ is the indicator function. Using (7), the dynamic equations of the matrices $\Gamma_i(k)$ are given by

$$\begin{aligned} \Gamma_i(k+1) &= \sum_{j=1}^N p_{ij} \left(A - \frac{1}{\sqrt{\pi_j(0)}} M_j(k) C_j \right) \Gamma_j(k) \\ &\quad \cdot \left(A - \frac{1}{\sqrt{\pi_j(0)}} M_j(k) C_j \right)' + \sum_{j=1}^N p_{ij} \left(M_j(k) \Sigma_{v_j} M_j(k)' + \pi_j(0) \Sigma_w \right), \end{aligned} \quad (10)$$

with $\Gamma_i(0) = \pi_i(0) \Sigma_0$.

Remark 3 To be consistent with the distributed estimation model, in the above expression we need the summation to be over p_{ij} (and not p_{ji}). This was obtained by imposing P' to be the transition probability matrix of $\theta(k)$, which explains the need for Assumption 2.

The optimal filtering gains are obtained as a solution of the following minimization problem

$$\mathbf{M}^*(K) = \arg \min_{\mathbf{M}(K)} \mathcal{J}_K(\mathbf{M}(K)), \quad (11)$$

where $\mathcal{J}_K(\mathbf{M}(K))$ is the finite horizon filtering cost

$$\mathcal{J}_K(\mathbf{M}(K)) = \sum_{k=0}^K \text{tr}(\Gamma(k)) = \sum_{k=0}^K \sum_{i=1}^N \text{tr}(\Gamma_i(k)),$$

and $\mathbf{M}(K) \triangleq \{M_i(k), k = 0 \dots K-1\}_{i=1}^N$ is the set of filtering gains corresponding to the operating modes of the system over the finite horizon.

Proposition 4 The optimal solution of (11) is given by

$$M_i^*(k) = \frac{1}{\sqrt{\pi_i(0)}} A \Gamma_i^*(k) C_i' \left(\Sigma_{v_i} + \frac{1}{\pi_i(0)} C_i \Gamma_i^*(k) C_i' \right)^{-1}, \quad (12)$$

for $i = 1 \dots N$, where $\Gamma_i^*(k)$ satisfies

$$\begin{aligned} \Gamma_i^*(k+1) = & \sum_{j=1}^N p_{ij} \left[A \Gamma_j^*(k) A' + \pi_j(0) \Sigma_w - \frac{1}{\sqrt{\pi_j(0)}} A \Gamma_j^*(k) C_j' \right. \\ & \cdot \left(\Sigma_{v_j} + \frac{1}{\pi_j(0)} C_j \Gamma_j^*(k) C_j' \right)^{-1} \frac{1}{\sqrt{\pi_j(0)}} C_j \Gamma_j^*(k) A' \Big], \end{aligned} \quad (13)$$

with $\Gamma_i^*(0) = \pi_i(0) \Sigma_0$.

PROOF. Follows from Theorems 5.3, 5.4 and 5.5 of [3].

Let us now define a scaled version of the matrices $\Gamma_i(k)$, namely $Q_i(k) \triangleq \frac{1}{\pi_i(0)} \Gamma_i(k)$. These matrices will appear in the next section as “approximations” of the covariance matrices of the estimation errors resulting from the distributed filter (5). It can be easily checked that $Q_i(k)$ respects the following dynamic equation

$$\begin{aligned} Q_i(k+1) = & \sum_{j=1}^N p_{ij} \left(A - L_j(k) C_j \right) Q_j(k) \cdot \\ & \cdot \left(A - L_j(k) C_j \right)' + \sum_{j=1}^N p_{ij} L_j(k) \Sigma_{v_j} L_j(k)' + \Sigma_w, \end{aligned} \quad (14)$$

with $Q_i(0) = \Sigma_0$ and where $L_i(k) = \frac{1}{\sqrt{\pi_i(0)}} M_i(k)$. In the following Corollary, we introduce the optimal solution of the optimization problem

$$\min_{\mathbf{L}(K)} \bar{J}_K(\mathbf{L}(K)), \quad (15)$$

where $\bar{J}_K(\mathbf{L}(K)) = \sum_{k=0}^K \sum_{i=1}^N \text{tr}(Q_i(k))$ and $\mathbf{L}(K) \triangleq \{L_i(k), k = 0 \dots K-1\}_{i=1}^N$, which follows immediately from Proposition 4.

Corollary 5 The optimal solution of the optimization problem (15) is given by

$$L_i^*(k) = A Q_i^*(k) C_i' \left(\Sigma_{v_i} + C_i Q_i^*(k) C_i' \right)^{-1}, \quad (16)$$

for $i = 1 \dots N$, where $Q_i^*(k)$ satisfies

$$\begin{aligned} Q_i^*(k+1) = & \sum_{j=1}^N p_{ij} \left[A Q_j^*(k) A' + \Sigma_w - A Q_j^*(k) C_j' \right. \\ & \cdot \left(\Sigma_{v_j} + C_j Q_j^*(k) C_j' \right)^{-1} C_j Q_j^*(k) A' \Big], \end{aligned} \quad (17)$$

with $Q_i^*(0) = \Sigma_0$.

4 Sub-optimal distributed consensus-based linear filtering

In this section we introduce the distributed filtering problem and show why the filtering gains derived in the previous section, corresponding to the optimal linear filter of a MJLS, can be used as an approximate solution of the optimal distributed filtering problem. Let $\epsilon_i(k)$ denote the estimation error of sensor i , i.e., $\epsilon_i(k) \triangleq x(k) - \hat{x}_i(k)$. The covariance matrix of the estimation error of sensor i is denoted by

$$\Sigma_i(k) \triangleq E[\epsilon_i(k) \epsilon_i(k)'], \quad \Sigma_i(0) = \Sigma_0. \quad (18)$$

From (5), the estimation errors evolve according to

$$\epsilon_i(k+1) = \sum_{j=1}^N p_{ij} \left[(A - L_j(k) C_j) \epsilon_j(k) + w(k) - L_j(k) v_j(k) \right]. \quad (19)$$

We introduce the following optimization problem

$$\min_{\mathbf{L}(K)} J_K(\mathbf{L}(K)), \quad (20)$$

where $J_K(\mathbf{L}(K))$ is the finite horizon filtering cost function

$$J_K(\mathbf{L}(K)) = \sum_{k=0}^K \sum_{i=1}^N E[\|\epsilon_i(k)\|^2] = \sum_{k=0}^K \sum_{i=1}^N \text{tr}(\Sigma_i(k)), \quad (21)$$

where by $\mathbf{L}(K)$ we understand the set of matrices $\mathbf{L}(K) \triangleq \{L_i(k), k = 0 \dots K-1\}_{i=1}^N$.

The problem of obtaining the optimal filtering gains of the above cost is intractable, very much in the same spirit of the (still open) decentralized control problem. Inspired by the MJLS filtering theory, in what follows we show how we can obtain an approximate solution of (20). The advantage of this approximate solution is that it can be computed in a

distributed manner and that grants a level of performance that can be quantified.

The approximate solution of (20) is based on the next result.

Lemma 6 Assume that the distributed filtering scheme and the MJLS state estimation scheme use that same filtering gains. Then the following inequality holds

$$\Sigma_i(k) \leq Q_i(k), \forall i, k, \quad (22)$$

where $\Sigma_i(k)$ was defined in (18) and $Q_i(k)$ satisfies (14).

PROOF. Let $L_i(k)$ be the filtering gains. Using (19), the matrix $\Sigma_i(k+1)$ can be explicitly written as

$$\begin{aligned} \Sigma_i(k+1) = E \Big[& \left(\sum_{j=1}^N p_{ij} (A - L_j(k)C_j) \epsilon_j(k) + w(k) - \right. \\ & \left. - \sum_{j=1}^N p_{ij} L_j(k) v_j(k) \right) \left(\sum_{j=1}^N p_{ij} (A - L_j(k)C_j) \epsilon_j(k) + w(k) - \right. \\ & \left. - \sum_{j=1}^N p_{ij} L_j(k) v_j(k) \right)' \Big]. \end{aligned}$$

Using the fact that the noises $w(k)$ and $v_i(k)$ have zero mean, and they are independent with respect to themselves and x_0 , for every time instant, we can further write

$$\begin{aligned} \Sigma_i(k+1) = E \Big[& \left(\sum_{j=1}^N p_{ij} (A - L_j(k)C_j) \epsilon_j(k) \right) \left(\sum_{j=1}^N p_{ij} (A - L_j(k)C_j) \epsilon_j(k) \right)' \Big] \\ & + E \Big[\left(\sum_{j=1}^N p_{ij} L_j(k) v_j(k) \right) \left(\sum_{j=1}^N p_{ij} L_j(k) v_j(k) \right)' \Big] + \Sigma_w. \end{aligned}$$

By Remark 1, it follows that

$$\begin{aligned} E \Big[& \left(\sum_{j=1}^N p_{ij} (A - L_j(k)C_j) \epsilon_j(k) \right) \left(\sum_{j=1}^N p_{ij} (A - L_j(k)C_j) \epsilon_j(k) \right)' \Big] \\ & \leq \sum_{j=1}^N p_{ij} (A - L_j(k)C_j) \Sigma_j(k) (A - L_j(k)C_j)' \end{aligned}$$

and

$$\begin{aligned} E \Big[& \left(\sum_{j=1}^N p_{ij} L_j(k) v_j(k) \right) \left(\sum_{j=1}^N p_{ij} L_j(k) v_j(k) \right)' \Big] \leq \\ & \leq \sum_{j=1}^N p_{ij} L_j(k) \Sigma_{v_j} L_j(k)', \quad i = 1 \dots N. \end{aligned}$$

From the previous two expressions, we obtain that

$$\begin{aligned} \Sigma_i(k+1) \leq & \sum_{j=1}^N p_{ij} (A - L_j(k)C_j) \Sigma_j(k) (A - L_j(k)C_j)' + \\ & + \sum_{j=1}^N p_{ij} L_j(k) \Sigma_{v_j} L_j(k) + \Sigma_w. \end{aligned}$$

We prove (22) by induction. Assume that $\Sigma_i(k) \leq Q_i(k)$ for all $i = 1 \dots N$. Then

$$\begin{aligned} (A - L_i(k)C_i) \Sigma_i(k) (A - L_i(k)C_i)' & \leq \\ \leq (A - L_i(k)C_i) Q_i(k) (A - L_i(k)C_i)', \end{aligned}$$

and

$$L_i(k) \Sigma_i(k) L_i(k)' \leq L_i(k) Q_i(k) L_i(k)', \quad i = 1 \dots N.$$

and therefore

$$\Sigma_i(k+1) \leq Q_i(k+1), \quad i = 1 \dots N.$$

The next Corollary follows immediately from Lemma 6 and shows that $\bar{J}_K(\mathbf{L}(K))$ is an upper bound on the filtering cost of the distributed filtering problem.

Corollary 7 The following inequality holds

$$J_K(\mathbf{L}(K)) \leq \bar{J}_K(\mathbf{L}(K)), \quad (23)$$

for any set of matrices $\mathbf{L}(K) = \{L_i(k), k = 0 \dots K-1\}_{i=1}^N$.

Remark 8 Corollary 7 leads us to a rigorous approach to deriving a sub-optimal scheme for the distributed filtering problem. More precisely, instead of minimizing the cost $J_K(\mathbf{L}(K))$, we minimize its upper bound, namely $\bar{J}_K(\mathbf{L}(K))$. The minimizer of $\bar{J}_K(\mathbf{L}(K))$ was introduced in Corollary 5 (equations (16)-(17)). This sub-optimal solution has the advantage that it can be implemented in a distributed manner (i.e., each agents requires information only from neighbors) and guarantees a filtering cost no larger than

$$\bar{J}_K(\mathbf{L}^*(K)) = \sum_{k=0}^K \sum_{i=1}^N Q_i^*(k),$$

where $L_i^*(k)$ and $Q_i^*(k)$ satisfy (16) and (17), respectively.

We can also formulate an infinite horizon optimization cost for the distributed filtering problem. The sub-optimal solution inspired by the MJLS theory has the same expression as in formulas (16)-(17) and the question we can ask is under what conditions the dynamics of $Q_i^*(k)$ is stable. Sufficient conditions under which there exists a unique set of matrices $\{Q_i^\infty\}_{i=1}^N$ such that $\lim_{k \rightarrow \infty} Q_i^*(k) = Q_i^\infty$, for $i = 1 \dots N$, are introduced in Appendix A of [3].

5 Numerical example

In this section we present a numerical example where a sensor network (Figure 1) estimates the state of a stochastic linear process. We compare the estimation errors under three estimation strategies: centralized filtering (i.e., a central entity receives information from all sensors), collaborative filtering described in the previous sections and non-collaborative filtering (i.e., each sensor computes an estimate based only on its own measurements). The parameters of the stochastic process are given by

$$A = \begin{pmatrix} 0.9996 & -0.03 \\ 0.03 & 0.9996 \end{pmatrix}, \quad \Sigma_w = \begin{pmatrix} 0.1 & 0 \\ 0 & 0.1 \end{pmatrix}, \quad \mu_0 = 10, \Sigma_0 = 0_{2 \times 2}.$$

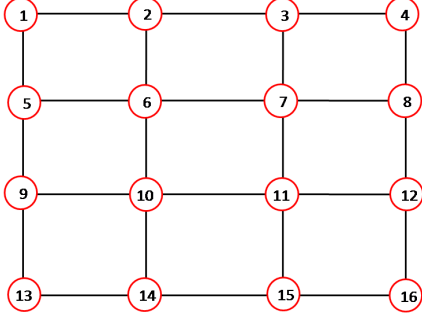


Fig. 1. Sensor network

The parameters of the sensing models are as follows: $C_i = [1 \ 0]$, $\sigma_{v_i}^2 = 0.01$ for $i \in \{1, \dots, 8\}$, and $C_i = [0 \ 1]$, $\sigma_{v_i}^2 = 1$ for $i \in \{9, \dots, 16\}$. In other words we assume that the first half of the sensors are very accurate compared to the second half of the sensors. In the collaborative filtering scenario, the consensus matrix P is chosen as $P = I - \frac{1}{N}L$, where $N = 16$ and L is the Laplacian of the (undirected) graph in Figure 1.

Let $\hat{x}^c(k)$, $\hat{x}_i(k)$ and $\hat{x}_i^{nc}(k)$ be the state estimates obtained using the centralized filter, collaborative filter and the non-collaborative filter, respectively.

In the centralized case the filter dynamics is given by

$$\hat{x}^c(k+1) = A\hat{x}^c(k) + L^c(k)(y(k) - C\hat{x}^c(k))$$

where $C' = [C'_1, \dots, C'_N]$, $L^c(k)$ is the filtering gain in the centralized case, $y(k) = Cx(k) + v(k)$, and $v(k) = (v_i(k))$. In the non-collaborative case, the dynamics of the filters corresponding to each of the sensors are given by

$$\hat{x}_i^{nc}(k+1) = A\hat{x}_i^{nc}(k) + L_i^{nc}(k)(y_i(k) - C_i\hat{x}_i^{nc}(k)),$$

where L_i^{nc} are the filtering gains and $y_i(k)$ satisfy (2).

The estimation errors in the three considered cases are denoted by $\epsilon^c(k)$, $\epsilon_i(k)$ and $\epsilon_i^{nc}(k)$, respectively. In the case of the centralized and non-collaborative cases, the optimal filters are computed by minimizing a quadratic cost in terms of the estimation errors similar to (21), and using standard optimal filtering techniques for linear systems.

In the following numerical simulations, the quantities $E[\|\epsilon^c(k)\|^2]$ and $E[\|\epsilon_i^{nc}(k)\|^2]$ are computed exactly, since we can derive the update equations of the covariances matrices of the estimation errors. The quantities $E[\|\epsilon_i(k)\|^2]$ (resulting from our algorithm) are approximated by averaging over 2000 realizations, since computing the covariance errors is intractable due to cross-correlations.

In Figure 2 we plot the variances of the estimation errors for the three filtering cases enumerated above. We represent the mean values of the norm of the estimation error in the case of the centralized filter ($E[\|\epsilon^c(k)\|^2]$). In addition, we represent the mean values of the norm of the estimation errors for

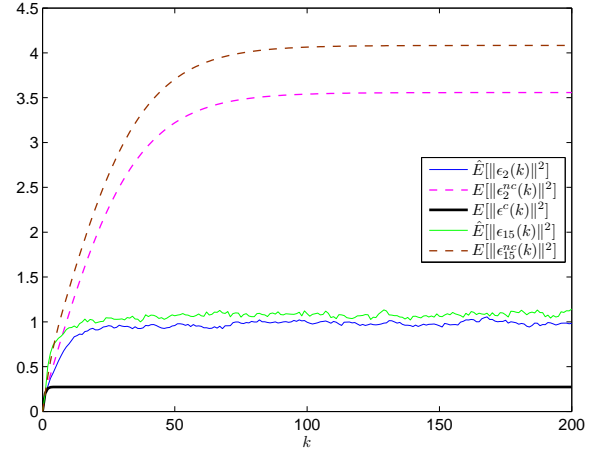


Fig. 2. Estimation errors in the centralized, collaborative and non-collaborative filtering scenarios

the accurate sensor 2 and for the less accurate sensor 15, in the case of our filtering algorithm and in the case of the non-collaborative filtering algorithm ($E[\|\epsilon_2(k)\|^2]$, $E[\|\epsilon_{15}(k)\|^2]$, $E[\|\epsilon_2^{nc}(k)\|^2]$ and $E[\|\epsilon_{15}^{nc}(k)\|^2]$). In the case of our algorithm, we added a hat symbol on the expectation operator to signify approximation. As expected, the collaborative filter scores less than the centralized filter but better than the localized, non-collaborative filter. We also notice that the accurate sensor 2 computes a better estimate than sensor 15.

References

- [1] V. Borkar and P. Varaya. Asymptotic agreement in distributed estimation. *IEEE Trans. Autom. Control*, AC-27(3):650–655, Jun. 1982.
- [2] R. Carli, A. Chiuso, L. Schenato, and S. Zampieri. Distributed kalman filtering based on consensus strategies. *IEEE Journal on Selected Area in Communication*, 26(4):622–633, May 2008.
- [3] O.L.V. Costa, M.D. Fragoso, and R.P. Marques. *Discrete-Time Markov Jump Linear Systems*. Springer-Verlag, London, 2005.
- [4] I. Matei and J. Baras. Consensus-based linear filtering. *Proceedings of the 49th IEEE Conference on Decision and Control*, pages 7009–70014, Dec 2010.
- [5] R. Olfati-Saber. Distributed Kalman filter with embedded consensus filters. *Proceedings of the 44th IEEE Conference on Decision and Control*, pages 8179 – 8184, Dec. 2005.
- [6] R. Olfati-Saber. Distributed Kalman filtering for sensor networks. *Proceedings of the 46th IEEE Conference on Decision and Control*, pages 5492–5498, Dec. 2007.
- [7] A. Speranzon, C. Fischione, K. H. Johansson, and A. Sangiovanni-Vincentelli. A distributed minimum variance estimator for sensor networks. *IEEE Journal on Select Areas in Communication*, 26(4):609–621, May 2008.
- [8] M.V. Subbotin and R.S. Smith. Design of distributed decentralized estimators for formations with fixed and stochastic communication topologies. *Automatica*, 45:2491–2501, 2009.
- [9] D. Teneketzis and P. Varaiya. Consensus in distributed estimation. *Advances in Statistical Signal Processing*, pages 361–386, Jan. 1988.