ENTANGLED KALMAN FILTERS FOR COOPERATIVE ESTIMATION

Carlos Mosquera[*]

Dept. de Teoría de la Señal y Comunicaciones
Universidad de Vigo
36310 Vigo, Spain
E-mail: mosquera@gts.tsc.uvigo.es

Sudharman K. Jayaweera

Dept. of Electrical and Computer Engineering
University of New Mexico
Albuquerque, NM 87131, USA
E-mail: jayaweera@ece.unm.edu

ABSTRACT

In this paper we propose a distributed estimation scheme for tracking the state of a Gauss-Markov model by means of independent observations at sensors connected in a network. Our emphasis is on low communication demands to alleviate the burden on eventually battery-powered sensors, which will limit the achievable performance with respect to an ideal centralized Kalman filter with access to all sensors measurements. The cooperation is performed in a distributed way to guarantee scalability and robustness to failures, and it is designed to reduce the detrimental effects of the channel noise on the sensor exchanges.

1. INTRODUCTION

Deployment of sensors for monitoring, collaborative information processing and control has gained a good deal of attention and research in recent years. If the wireless sensor network can operate autonomously, that is, without a central repository or fusion center collecting and processing all measurements, important advantages become evident such as scalability and robustness against node failure. The coordinated action of the different sensors requires the local exchange of information to improve on their individual estimates. We will work with the following linear state-space model to describe the evolution of $x(k)$:

$$x(k+1) = F(k)x(k) + u(k) \qquad (1)$$

where process noise $u(k)$ is assumed to be white with covariance $E\{u^2(k)\} = Q_u(k)$ and independent of $x(k)$. In a centralized setting, a unique node collects a set of N noisy observations of $x(k)$ as

$$\boldsymbol{y}(k) = \boldsymbol{H}(k)x(k) + \boldsymbol{v}(k). \qquad (2)$$

Note that we follow the usual convention that vectors and matrices are denoted by bold letters. In addition, the $N \times 1$

column vectors of all ones and all zeros will be denoted respectively as $\mathbf{1}$ and $\mathbf{0}$. Superscript $(\cdot)^t$ denotes the transpose. The (i, j)-th element and the trace of the matrix $\boldsymbol{A}$ will be denoted by $[\boldsymbol{A}]_{ij}$ and $\mathrm{tr}\{\boldsymbol{A}\}$ respectively, whereas $\mathrm{diag}\{\boldsymbol{A}\}$ is a column vector collecting the main diagonal elements of the matrix $\boldsymbol{A}$. The vector $\boldsymbol{H}(k)$ in (2) is written as

$$\boldsymbol{H}(k) \doteq \begin{bmatrix} H_1(k) & H_2(k) & \cdots & H_N(k) \end{bmatrix}^t \qquad (3)$$

and the measurement noise $\boldsymbol{v}(k)$ is white and independent of $\boldsymbol{u}(k)$, with diagonal covariance matrix $\mathrm{E}\{\boldsymbol{v}(k)\boldsymbol{v}^t(k)\}$ denoted as $\boldsymbol{Q}_v(k)$:

$$\boldsymbol{Q}_v(k) \doteq \begin{pmatrix} Q_{v(1)}(k) & 0 & \cdots & 0 \\ 0 & Q_{v(2)}(k) & \ddots & 0 \\ \vdots & \ddots & \ddots & \vdots \\ 0 & \cdots & 0 & Q_{v(N)}(k) \end{pmatrix}. \qquad (4)$$

The recursive relation for the update of the linear estimation $\hat{x}(k)$ of $x(k)$ based on the observations $\boldsymbol{y}(0), \ldots, \boldsymbol{y}(k)$, is the Kalman filter, expressed in the *information form* by [1]

$$\begin{aligned} P^{-1}(k)\hat{x}(k) &= F^{-1}(k)(1 - K(k))P^{-1}(k-1)\hat{x}(k-1) \\ &+ \boldsymbol{H}^t(k)\boldsymbol{Q}_v^{-1}(k)\boldsymbol{y}(k) \end{aligned} \qquad (5)$$

where $P(k) = \mathrm{E}\{(x(k) - \hat{x}(k))^2\}$ is computed through the following updates:

$$\begin{aligned} R(k) &= Q_u^{-1}(k) + F^{-1}(k)P^{-1}(k-1)F^{-1}(k) &(6) \\ K(k) &= F^{-1}(k)P^{-1}(k-1)F^{-1}(k)R^{-1}(k) &(7) \\ P^{-1}(k) &= F^{-1}(k)P^{-1}(k-1)F^{-1}(k) &(8) \\ &+ \boldsymbol{H}^t(k)\boldsymbol{Q}_v^{-1}(k)\boldsymbol{H}(k) - K(k)R(k)K(k) \end{aligned}$$

for an initial uncertainty $\boldsymbol{P}(-1)$. Now, let us consider that instead of a central observer, we have a set of N sensors, each getting a value[1] $y_n(k)$ of the vector $\boldsymbol{y}(k)$ in (2):

$$y_n(k) = H_n(k)x(k) + v_n(k), n = 1, \ldots, N$$

---

[1]We keep the individual observations unidimensional for the sake of the exposition, although the results can be easily generalized to vector observations.

with $H_n(k)$ and $v_n(k)$ the nth elements of $\boldsymbol{H}(k)$ and $\boldsymbol{v}(k)$ respectively. From the above equations, each sensor would be able to emulate a central fusion center should it be able to compute $\boldsymbol{H}^t(k)\boldsymbol{Q}_v^{-1}(k)\boldsymbol{y}(k)$ and $\boldsymbol{H}^t(k)\boldsymbol{Q}_v^{-1}(k)\boldsymbol{H}(k)$. This was already observed in [2] and later used in [3] and [4]. For each Kalman update, an infinite number of consensus steps [5] must take place to agree on those quantities: this number is restricted to be one in [2], whereas in [6] state estimates are additionally exchanged once the observations have been shared and accounted for the estimates updates. The approach in [3] is more general, and can deal with noisy exchanges in the vein of [7]. The performance approaches that of the corresponding centralized counterpart for the noiseless case when the number of consensus stages goes to infinity, since in that case all sensors can agree on the aforementioned quantities $\boldsymbol{H}^t(k)\boldsymbol{Q}_v^{-1}(k)\boldsymbol{y}(k)$ and $\boldsymbol{H}^t(k)\boldsymbol{Q}_v^{-1}(k)\boldsymbol{H}(k)$. In practice, a low number of consensus stages could guarantee the performance of a centralized estimator, although the associated communication demands might not be acceptable in practical settings. In [8] the ideal exchange of estimations and observations is used to track a time-varying signal observed in noise, although no model is used for the dynamics of the signal. Yet another approach with low communication overhead is used in [9]: only the estimates at each Kalman update are exchanged; however, inter-nodes links are assumed to be ideal, and global statistical information from all other sensors must be known locally. On the other side, [10] analyzes the number of messages to exchange between succesive updates in a distributed Kalman filter with perfect communication and no model for the evolution of the signal. With the asymptotic error as target, optimum values for the exchange weights and Kalman gain are obtained for some limit cases, assuming identical sensors with the same quality.

In this paper we will study the formulation of a set of Kalman filters which are entangled so that their respective observations or state estimates are exchanged at each Kalman update. For a network topology as detailed in the following section, Section 3 will show how the individual Kalman filters running at each sensor must cooperate to improve their individual estimates. Section 4 will provide numerical illustrations before the final remarks.

2. NETWORK MODEL

We consider an oriented graph $G = (V, E)$, with N nodes (sensors) $v_n \in V$ and edges $e_{ij} \in E$ if there is a path from node v_i to node v_j. We assume that the graph is connected, so there is a sequence of edges to go from any node i to any other node j. The elements of the adjacency matrix $\mathcal{A}$ are defined as $[\mathcal{A}]_{ij} = 1$ if e_{ji} belongs to E, otherwise they are zero. The degree matrix $\boldsymbol{D}$ is a diagonal matrix such that $[\boldsymbol{D}]_{ii}$ is equal to the number of connections entering node i. With that, the Laplacian matrix $\boldsymbol{L}$ is defined as $\boldsymbol{L} = \boldsymbol{D} - \mathcal{A}$. In other words,

the elements $[\boldsymbol{L}]_{ij}$ of the Laplacian matrix $\boldsymbol{L}$ are defined as

$$[\boldsymbol{L}]_{ij} = \left\{ \begin{array}{ll} -1, & e_{ji} \in E \\ [\boldsymbol{D}]_{ii}, & i = j \end{array} \right.$$

The eigenvalues of $\boldsymbol{L}$ contain very significant information about the topology of the graph G. In fact, if they are ordered as $\lambda_1 \leq \lambda_2 \leq \cdots \leq \lambda_N$, we have that $\lambda_1 = 0$, and $\lambda_2 > 0$ for a connected graph. This second eigenvalue λ_2 is known as the algebraic connectivity of the graph, and its value plays a major role in the speed at which information can be diffused through the network [5]. The additive noise in the signal received at the j-th sensor from the i-th sensor is zero-mean with variance σ_w^2, and is independent for all channel realizations and different sensor links. The received noise values at the exchange associated with the kth iteration are collected in $\boldsymbol{W}(k)$:

$$\boldsymbol{W}(k) = \left(\begin{array}{cccc} 0 & w_{12}(k) & \cdots & w_{1N}(k) \\ w_{21}(k) & 0 & \cdots & w_{2N}(k) \\ \vdots & \vdots & \ddots & \vdots \\ w_{N1}(k) & w_{N2}(k) & \cdots & 0 \end{array} \right).$$

3. ENTANGLED KALMAN FILTERS

The upper bound performance for any distributed estimator of $x(k)$ in (1) corresponds to a centralized Kalman filter. This role might be played by a fusion center with access to all the measurements at each step or, on the other side, by the flooding of the measurements throughout the whole network. Since this latter option would require an enormous communication effort not scalable with the size of the network, other options have been explored in the literature as detailed in Section 1. As consensus implementations require fast exchanges to align the estimates between two consecutive Kalman updates, we will explore the case of one exchange between two Kalman updates. In such a case, an unless the network is fully connected and noiseless, it is not possible to compute $\boldsymbol{H}^t(k)\boldsymbol{Q}_v^{-1}(k)\boldsymbol{y}(k) = \sum_{n=1}^N H_n(k)y_n(k)/Q_{v(n)}(k)$ and $\boldsymbol{H}^t(k)\boldsymbol{Q}_v^{-1}(k)\boldsymbol{H}(k) = \sum_{n=1}^N H_n^2(k)/Q_{v(n)}(k)$ as required in (5) and (8) in a distributed form. However, the set of values that each sensor may receive from its neighbors will contribute to improve its isolated performance. The description of the extended capabilities of each sensor can follow the ideas in [3], applied to our particular setup and explicitly including the communication noise. Sensors must exchange $H_n(k)y_n(k)/Q_{v(n)}(k)$ and $H_n^2(k)/Q_{v(n)}(k)$, so the combined observations are written as

$$\bar{\boldsymbol{y}}(k) = \boldsymbol{A} \left(\begin{array}{c} H_1(k)y_1(k)/Q_{v(1)}(k) \\ \vdots \\ H_N(k)y_N(k)/Q_{v(N)}(k) \end{array} \right) + \mathrm{diag}\{\boldsymbol{A}\boldsymbol{W}(k)\}. \tag{9}$$

Note the last term in (9) due to the communication noise. In addition,

$$\bar{\boldsymbol{y}}(k) \doteq \left[\begin{array}{cccc} \bar{\boldsymbol{y}}_1(k) & \bar{\boldsymbol{y}}_2(k) & \cdots & \bar{\boldsymbol{y}}_N(k) \end{array}\right]^t \quad (10)$$

and the weighting matrix $\boldsymbol{A} = \boldsymbol{I} - \gamma \boldsymbol{L}$, where γ is a parameter to be chosen[2]. As a result the state-space model at each sensor is given by

$$\begin{align} x(k+1) &= F(k)x(k) + u(k) \quad (11) \\ \bar{y}_n(k) &= \bar{H}_n(k)x(k) + \bar{v}_n(k) \quad (12) \end{align}$$

for $n = 1, \ldots, N$, if we write $\bar{\boldsymbol{y}}(k) = \bar{\boldsymbol{H}}(k)x(k) + \bar{\boldsymbol{v}}(k)$. Each sensor can iterate the corresponding Kalman filter provided that the communication of $H_n^2(k)/Q_{v(n)}(k)$ is noiseless: these values may change slowly in practice, making robust communication easier. The covariance matrix of the vector noise containing the N components $\bar{v}_n(k)$ is now a function of the original observation noise statistics, the weighting matrix and the communication noise:

$$\boldsymbol{Q}_{\bar{v}}(k) = \boldsymbol{A} \begin{pmatrix} H_1^2(k)/Q_{v(1)}(k) & \cdots & 0 \\ \vdots & \ddots & \vdots \\ 0 & \cdots & H_N^2(k)/Q_{v(N)}(k) \end{pmatrix} \boldsymbol{A}^t$$
$$+ \gamma^2 \sigma_w^2 \boldsymbol{D} \quad (13)$$

assuming that the noise power σ_w^2 is the same for all links. In consequence, after each measurement the corresponding values are exchanged with neighbors to assist the Kalman update at each sensor. The signal-to-noise ratio (SNR) of the received values will determine their contribution. In fact, the weight γ should be chosen accordingly, so low quality exchanges would reduce the cooperation degree. Strictly speaking, the assigned weights to different links could be different, since as mentioned earlier, statistics are expected to vary slowly, so each sensor can anticipate the reliability degree of its neighbors contributions and weight them accordingly. We will illustrate this point in numerical results in Section 4.

Alternatively, we want to improve the performance of the isolated sensors by exchanging their estimates with their neighbors. Thus, information can diffuse further. Let us consider for example a low quality sensor with poor observations, only connected to another low quality sensor which gets signals from other sensors with higher quality measurements. In the above framework, the first low quality sensor would never benefit from those sensors which are more than one hop away. However, if states are exchanged instead of observations, overall better performance can be expected, specially in highly unbalanced networks. The downside of this approach is the required additional information which must be exchanged to weight properly all the contributions. In fact, the correlation among the different node estimates is needed

---

[2]More sophisticated weighthing coefficients can be used, see, e.g., [5].

for proper combination, which may be unfeasible to monitor in completely distributed structures. However, for the purpose of illustration, we detail next the analysis for a particular case together with some numerical results in the next section to evaluate the potential improvement. Thus, and after every Kalman update, all sensors exchange their estimates before the next measurement. We put all the N estimates at step k in the vectors $\bar{\boldsymbol{x}}(k)$ (before exchanging information) and $\hat{\boldsymbol{x}}(k)$ (after merging):

$$\begin{align} \bar{\boldsymbol{x}}(k) &\doteq \left[\begin{array}{cccc} \bar{x}_1(k) & \bar{x}_2(k) & \cdots & \bar{x}_N(k) \end{array}\right]^T \quad (14) \\ \hat{\boldsymbol{x}}(k) &\doteq \left[\begin{array}{cccc} \hat{x}_1(k) & \hat{x}_2(k) & \cdots & \hat{x}_N(k) \end{array}\right]^T . \end{align}$$

If we use the standard Kalman formulation, the update and the merging steps are given respectively by

$$\bar{x}_n(k) = K_1(k)\hat{x}_n(k-1) + K_2^{(n)}(k)y_n(k), n = 1, \ldots, N \quad (15)$$

where we are using the same weight $K_1(k)$ for all sensors, and

$$\hat{\boldsymbol{x}}(k) = \boldsymbol{A}\bar{\boldsymbol{x}}(k) + \text{diag}\{\boldsymbol{A}\boldsymbol{W}(k)\}. \quad (16)$$

Each sensor updates its estimate based on the relative contributions from all the other sensors, using the weight matrix $\boldsymbol{A}$. The error vectors before and after merging are given by

$$\begin{align} \bar{\boldsymbol{E}}(k) &\doteq \bar{\boldsymbol{x}}(k) - x(k)\boldsymbol{1} \quad (17) \\ \boldsymbol{E}(k) &\doteq \hat{\boldsymbol{x}}(k) - x(k)\boldsymbol{1} \end{align}$$

with the corresponding covariance matrices

$$\begin{align} \bar{\boldsymbol{P}}(k) &\doteq \text{E}\{\bar{\boldsymbol{E}}(k)\bar{\boldsymbol{E}}^t(k)\} \quad (18) \\ \boldsymbol{P}(k) &\doteq \text{E}\{\boldsymbol{E}(k)\boldsymbol{E}^t(k)\}. \quad (19) \end{align}$$

If we group all the sensors coefficients $K_2^{(n)}(k)$ as

$$\boldsymbol{K}_2(k) \doteq \begin{pmatrix} K_2^{(1)}(k) & 0 & \cdots & 0 \\ 0 & K_2^{(2)}(k) & \ddots & 0 \\ \vdots & \ddots & \ddots & \vdots \\ 0 & \cdots & 0 & K_2^{(N)}(k) \end{pmatrix}$$

and define a diagonal matrix $\boldsymbol{H}_d(k)$ out of $\boldsymbol{H}(k)$ in (2)

$$\boldsymbol{H}_d(k) \doteq \begin{pmatrix} H_1(k) & 0 & \cdots & 0 \\ 0 & H_2(k) & \ddots & 0 \\ \vdots & \ddots & \ddots & \vdots \\ 0 & \cdots & 0 & H_N(k) \end{pmatrix}$$

then the two steps at each iteration can be detailed as follows: *(i)* The error after the Kalman update and before the merging is written as

$$\begin{align} \bar{\boldsymbol{E}}(k) &= K_1(k)\boldsymbol{E}(k-1) \\ &+ (K_1(k)\boldsymbol{I} + \boldsymbol{K}_2(k)\boldsymbol{H}_d(k)F(k) - F(k)\boldsymbol{I}) \cdot x(k-1)\boldsymbol{1} \\ &+ (\boldsymbol{K}_2(k)\boldsymbol{H}_d(k) - \boldsymbol{I})\boldsymbol{1}u(k-1) + \boldsymbol{K}_2(k)\boldsymbol{v}(k). \quad (20) \end{align}$$

(ii) The error after the information merging is given by

$$\boldsymbol{E}(k) = \boldsymbol{A}\bar{\boldsymbol{E}}(k) + (\boldsymbol{A} - \boldsymbol{I})x(k)\mathbf{1} + \text{diag}\{\boldsymbol{A}\boldsymbol{W}(k)\}. \quad (21)$$

We have that $(\boldsymbol{A} - \boldsymbol{I})\mathbf{1} = 0$ for $\boldsymbol{A} = \boldsymbol{I} - \gamma\boldsymbol{L}$. On the other side,

$$K_1(k)\boldsymbol{I} + \boldsymbol{K}_2(k)\boldsymbol{H}_d(k)F(k) - F(k)\boldsymbol{I} \quad = \quad \mathbf{0}. \quad (22)$$

with

$$\boldsymbol{K}_2(k) = \left(1 - \frac{K_1(k)}{F(k)}\right)\boldsymbol{H}_d^{-1}(k). \quad (23)$$

With this, (20) and (21) can be rewritten as

$$\bar{\boldsymbol{E}}(k) = K_1(k)\boldsymbol{E}(k-1) + \\ (\boldsymbol{K}_2(k)\boldsymbol{H}_d(k) - \boldsymbol{I})\mathbf{1}u(k-1) + \boldsymbol{K}_2(k)\boldsymbol{v}(k), \quad (24)$$

and

$$\boldsymbol{E}(k) = \boldsymbol{A}\bar{\boldsymbol{E}}(k) + \text{diag}\{\boldsymbol{A}\boldsymbol{W}(k)\}, \quad (25)$$

respectively, from which the covariance matrices in (18) and (19) are shown next:

$$\bar{\boldsymbol{P}}(k) = K_1(k)\boldsymbol{P}(k-1)K_1(k) + Q_u(k) \\ \cdot (\boldsymbol{K}_2(k)\boldsymbol{H}_d(k) - \boldsymbol{I})\mathbf{1}\mathbf{1}^t(\boldsymbol{K}_2(k)\boldsymbol{H}_d(k) - \boldsymbol{I})^t \\ + \boldsymbol{K}_2(k)\boldsymbol{Q}_v(k)\boldsymbol{K}_2^t(k) \quad (26)$$

$$\boldsymbol{P}(k) = \boldsymbol{A}\bar{\boldsymbol{P}}(k)\boldsymbol{A}^t \\ + \text{E}\{\text{diag}\{\boldsymbol{A}\boldsymbol{W}(k)\}\text{diag}^t\{\boldsymbol{A}\boldsymbol{W}(k)\}\}. \quad (27)$$

The covariance error (27) after merging is rewritten as

$$\boldsymbol{P}(k) = \bar{\boldsymbol{P}}(k) - \gamma\bar{\boldsymbol{P}}(k)\boldsymbol{L}^t - \gamma\boldsymbol{L}\bar{\boldsymbol{P}}(k) \\ + \gamma^2\boldsymbol{L}\bar{\boldsymbol{P}}(k)\boldsymbol{L}^t + \gamma^2\sigma_w^2\boldsymbol{D} \quad (28)$$

or, as a function of $\boldsymbol{P}(k-1)$,

$$\boldsymbol{P}(k) = K_1^2(k)\boldsymbol{A}\boldsymbol{P}(k-1)\boldsymbol{A}^t + Q_u(k) \\ \cdot \boldsymbol{A}(\boldsymbol{K}_2(k)\boldsymbol{H}_d(k) - \boldsymbol{I})\mathbf{1}\mathbf{1}^t(\boldsymbol{K}_2(k)\boldsymbol{H}_d(k) - \boldsymbol{I})\boldsymbol{A}^t \\ + \boldsymbol{A}\boldsymbol{K}_2(k)\boldsymbol{Q}_v(k)\boldsymbol{K}_2(k)\boldsymbol{A}^t + \gamma^2\sigma_w^2\boldsymbol{D}. \quad (29)$$

In order to analyze the potential of improvement of the exchange of estimates, we use the optimum centralized value for $K_1(k)$, which can be proved to be the expression at the top of next page. This expression is for evaluation purposes, although it cannot be computed in a distributed form with one exchange per Kalman update. Next we show some numerical results to illustrate the effect of the links noise on the entangled Kalman filters based on (12) and their potential degradation with respect to an ideal implementation of (15) and (16).

4. NUMERICAL SIMULATIONS

Let us consider a network with $N = 20$ identical sensors randomly placed, with nodes communicating through symmetric noisy links as a function of their distance. The lowest five eigenvalues of the Laplacian matrix $\boldsymbol{L}$ are equal to $0, 1.7, 1.97, 2.1, 2.93$. We have computed the analytical average mean square error (mse) of all the N Kalman filters applied based on the model (12), for the stationary regime and the following fixed parameters: $F = 0.95, H_n = 1, n = 1, \ldots, N, Q_u = 0.5, \boldsymbol{Q}_v = 1\boldsymbol{I}$. The results are plotted in Fig. 1 for different channel SNRs and two different weighting criteria:

(i) The parameter γ is equal to $1/\max/[\boldsymbol{L}]_{nn}$ regardless of the SNR.

(ii) The parameter γ is chosen to maximize the observation SNR.

As expected, when the SNR gets very small, performance degrades significantly unless γ decreases accordingly. In the limit, $\gamma = 0$ and the performance becomes that of an isolated Kalman filter.
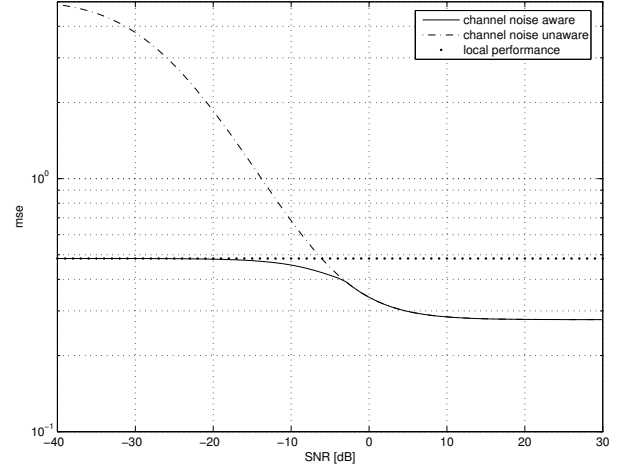

Fig. 1. Average stationary mean square error (mse) performance for a network of 20 sensors which exchange their observations.

On the other side, we wanted to analyze the potential degradation of the entangled Kalman filters applied on (12) with respect to the ideal exchange of estimates (16) using the centrally computed coefficient (30). For some cases the degradation is negligible. In some others, there is a marginal improvement by exchanging the estimates. Thus, Fig. 2 compares the evolution of the average mse for the previous network of 20 sensors for the exposed cases in the paper: exchange of observations (12) and exchange of estimates (16). The model values are $F = 0.99, H_n = 1, n = 1, \ldots, N, Q_u = 0.5, \boldsymbol{Q}_v =$

$$K_1(k) = \frac{F(k)\text{tr}\{\boldsymbol{A}\boldsymbol{H}_d^{-1}(k)\boldsymbol{Q}_v(k)\boldsymbol{H}_d^{-1}(k)\boldsymbol{A}^t\}}{\text{tr}\{F^2(k)\boldsymbol{A}\boldsymbol{P}(k-1)\boldsymbol{A}^t + Q_u(k)N + \boldsymbol{A}\boldsymbol{H}_d^{-1}(k)\boldsymbol{Q}_v(k)\boldsymbol{H}_d^{-1}(k)\boldsymbol{A}^t\}}. \tag{30}$$

$2\boldsymbol{I}$. The experimental results correspond to the averaging of 200 realizations. The exposed gain requires the knowledge of global statistics to compute the Kalman coefficients.
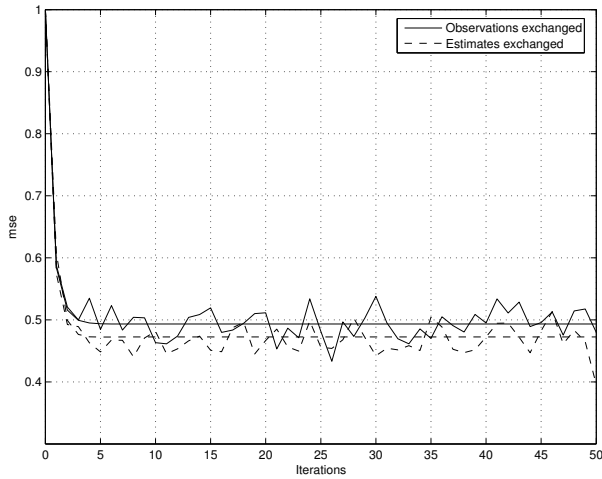

Fig. 2. Analytical and experimental mean square error (mse) performance for a network of 20 sensors. Observations and states estimates are exchanged through ideal links. $\gamma = 0.05$.

5. CONCLUDING REMARKS

We have shown how different sensors can follow the evolution of a common parameter by sharing their successive observations or estimates through noisy exchanges. The goal has not been to achieve the performance of a centralized Kalman filter with access to all the individual measurements, but rather to improve on the performance of individual sensors with a low communication burden, as imposed by practical limitations of sensor networks. The amount of information which must be exchanged between succesive measurements will determine the quality of the estimates. If only one set of exchanges is allowed between two consecutive Kalman updates, then the exchange of estimates makes it possible to propagate the information further than one hop, although it requires global statistical information for proper functioning. If more exchanges are allowed, then several rounds of consensus can be applied either to the observations or the states. It is even possible to devise hybrid strategies which exchange observations and states estimates to improve on the results exposed in this work.

6. REFERENCES

[1] Thomas Kailath, Ali H. Sayed, and Babak Hassibi, *Linear Estimation*, Prentice Hall, 2000.

[2] R. Olfati-Saber, "Distributed Kalman Filter with Embedded Consensus Filters," in *Proceedings of the 44th IEEE Conference on Decision and Control*, Sevilla, Spain, dec 2005, pp. 8179–8184.

[3] I.D. Schizas, G. B. Giannakis, A. Ribeiro, and S. I. Roumeliotis, "Consensus in Ad Hoc WSNs with Noisy Links - Part II: Distributed Estimation and Smoothing of Random Signals," *IEEE Transactions on Signal Processing*, vol. 56, no. 4, pp. 1650–1666, april 2008.

[4] S. Kirti and A. Scaglione, "Scalable Distributed Kalman Filtering Through Consensus," in *Proceedings of the IEEE International Conference on Acoustics, Speech and Signal Processing, ICASSP*, april 2008.

[5] Reza Olfati-Saber, J. Alex Fax, and Richard M. Murray, "Consensus and cooperation in networked multi-agent systems," *Proceedings of the IEEE*, vol. 95, no. 1, pp. 215–233, jan 2007.

[6] R. Olfati-Saber, "Distributed Kalman Filter for Sensor Networks," in *Proceedings of the 46th IEEE Conference on Decision and Control*, New Orleans, dec 2007, pp. 5492–5498.

[7] S. Barbarossa and G. Scutari, "Decentralized Maximum Likelihood Estimation for Sensor Networks Composed of Nonlinearly Coupled Dynamical Systems," *IEEE Transactions on Signal Processing*, vol. 55, no. 7, pp. 3456–3470, july 2007.

[8] A. Speranzon, C. Fischione, K.H. Johansson, and A. Sangiovanni-Vincentelli, "A distributed minimum variance estimator for sensor networks," *IEEE Journal on Selected Areas in Communications*, vol. 26, no. 4, pp. 609–621, may 2008.

[9] P. Alriksson and A. Rantzer, "Experimental evaluation of a distributed Kalman filter algorithm," in *Proceedings of the 46th IEEE Conference on Decision and Control*, 2007.

[10] L. Schenato R. Carli, A. Chiuso and S. Zampieri, "Distributed Kalman Filtering Based on Consensus Strategies," *IEEE Journal on Selected Areas in Communications*, vol. 26, no. 4, pp. 622–633, may 2008.