Distributing the Kalman Filter for Large-Scale Systems

Usman A. Khan, *Graduate Student Member, IEEE,* José M. F. Moura, *Fellow, IEEE*

Department of Electrical and Computer Engineering

Carnegie Mellon University

5000 Forbes Ave, Pittsburgh, PA 15213

{ukhan, moura}@ece.cmu.edu

Ph: (412)268-7103 Fax: (412)268-3890

Abstract

This paper derives a near optimal *distributed* Kalman filter to estimate a large-scale random field monitored by a network of N sensors. The field is described by a sparsely connected dynamical system of high-dimensionality, n. The main contributions of the paper are: (1) distribute the high-dimensional model into N *coupled* sensor-based reduced models of dimension $n_l \ll n$ using a graph-theoretic approach; (2) implement local Kalman filters on the reduced models; (3) fuse the observations and estimates that are common among the local Kalman filters using bipartite fusion graphs and consensus averaging algorithms; (4) invert the error covariances and information matrices of the local filters with a generalized distributed matrix Jacobi algorithm that we introduce and an $L-$banded matrix inversion theorem. These inversion algorithms compute the submatrices of interest with only local variables of dimension n_l, hence, performing the Riccati and Lyapunov equations with only local communication and local computation. The algorithm presented here achieves full distribution of the Kalman filter. Nowhere storage, communication, or computation of $n-$dimensional vectors and matrices is needed; only $n_l \ll n$ dimensional vectors and matrices are communicated or used in the computation at the sensors.

Index Terms

Large-scale systems, sparse matrices, distributed algorithms, matrix inversion, Kalman filtering, distributed estimation, iterative methods

This work was partially supported by the DARPA DSO Advanced Computing and Mathematics Program Integrated Sensing and Processing (ISP) Initiative under ARO grant # DAAD 19-02-1-0180, by NSF under grants # ECS-0225449 and # CNS-0428404, and by an IBM Faculty Award.

I. INTRODUCTION

Direct implementation of the Kalman filter [1], [2], for sparse large-scale dynamical systems monitored by sensor networks is unfeasible because of both intensive computation and inordinate communication requirements that challenge the limited resources at the sensors. To account for the processing, communication, and power constraints at the sensors, the Kalman filter implemented at each sensor, l, should only involve computations and communications with local quantities, i.e., vectors and matrices of low dimensions, $n_l \ll n$, where n is the dimension of the state vector. This paper presents a fully distributed Kalman filter for sparse large-scale dynamical systems with only local communication and reduced-order computation at each sensor—no sensor computes, communicates, or stores any $n-$dimensional quantity.

Efforts to decentralize the Kalman filter date back to at least [3], [4]. The work in [3], [4], decentralizes the Kalman filter equations by replicating at each sensor an nth order version of the dynamical system. These $n-$dimensional Kalman filters are then fused over all-to-all communication networks to achieve the global performance. A decentralized linear-quadratic-Gaussian control problem for an all-to-all sensor communication network is addressed in [5], with, again, a complete state-space replication at each sensor. Current research on distributed Kalman filters has focused mainly on the communication aspects at the sensors. Decentralized Kalman filters with local communication over arbitrary communication networks is discussed in [6], using consensus protocols in [7]. References [8], [9] incorporate packet losses, intermittent observations, and communication delays in the decentralized implementation of the Kalman filter. Although these solutions work well for low-dimensional dynamical systems, e.g., target tracking applications considered in [4], [8], [9], [10], they implement a replication of the $n-$dimensional Kalman filter at each sensor, hence communicating and inverting $n \times n$ matrices, which, in general, is an $O(n^3)$ operation. In contrast, in the problems we consider, the state dimension, n, is very large, for example, in the range of 10^2 to 10^9, and so the nth order replication of the global dynamics in the local Kalman filters is not an option.

Early attempts to reduce the computational burden at the sensors were made in [11], [12]. They use the Information filter [13], format of the Kalman filter, which iterates the information matrices (inverse of the error covariance matrices) at each time step. To reduce the computational load, these references decouple the state dynamics and hence force the model matrix to be block diagonal; they do not take into account the structure of the large-scale systems of interest to us. Ignoring the coupling between different reduced models can lead to highly inaccurate results. In [11], a fully connected communication network is required adding undesirable long distance communication links between the sensors.

We are motivated by problems where the large-scale systems, although sparse, cannot be decoupled, and where, due to the sensor constraints or the high-dimensionality of the application, the communication and computation should both be local. Examples of sparse large-scale systems include networks of loosely connected dynamical systems, spatially distributed random fields, weather systems, or earthquake monitoring systems, [14], [15], [16], [17], [18], [19], [20], [21]. We exploit the local structure on the system dynamics, motivated by the fact that, in practice, the correlations between the states farther apart in the state vector often decay rapidly. This occurs in a large

set of interesting applications, e.g.: system dynamics with banded model matrices arise naturally when the spatially correlated random fields are discretized by partial differential or difference equations (PDEs) [22]; in images, the field at a pixel depends usually on neighboring pixel values [23], [24]; power grid models exhibit commonly banded structures, [25], [26]. Systems with local or sparse structure (including banded matrices as a special case) are not decomposed into decoupled components; rather, they usually correspond to the system state being spatially Markovian. Even if the system local structure is not apparent, several algorithms exist in the literature, specially in the field of parallelizing iterative solvers for multiprocessor systems, to convert unstructured sparse matrices into sparse structured matrices. These algorithms apply matrix bandwidth reduction methods such as Reverse Cuthill Mckee (RCM) and Approximate Minimum Degree (AMD) reorderings [27], [28], [29]. So, the approach presented in the paper can apply to sparse, not necessarily banded, dynamical systems.

We exploit the inherent sparse/localized structure in the large-scale system in order to distribute the $n-$dimensional global dynamical system into several, possibly many, *coupled* n_lth$-$dimensional reduced-order local dynamical systems, one at each sensor, l, where $n_l \ll n$. Distributing the model is not the whole story needed to distribute the Kalman filter. The local Kalman filters corresponding to the local reduced order models have states of dimension n_l, but are coupled. This leads to the need of fusing observations of common states by different sensors, which we achieve by bipartite fusion graphs and average consensus techniques, and fusing correlated estimates of the same states, for which we use a variant of Durbin's formula [30]. Still, the major problem is to propagate the $n_l \times n_l$ error covariance matrices (or the corresponding $n_l \times n_l$ blocks in the information matrices) of the local state estimation errors. This requires inversion of the full $n \times n$ covariance or information matrices, a global $O\left(n^3\right)$ computation rather than a local $O\left(n_l^3\right)$ computation. To localize and distribute these inversions, we introduce a simplifying assumption, namely, we assume that the estimation error in the Kalman filter is a Gauss-Markov field of order L. This implies that the information matrices are $L-$banded, [31], [32]. We use this assumption to develop a *distributed* generalization of the Jacobi algorithm, [33], and use an $L-$banded matrix inversion theorem, [32], [34], to *fully localize* the inversion of the covariance and information matrices and, as a consequence, fully distribute the Kalman filter. Our generalized distributed matrix Jacobi algorithm solves matrix inversion (n coupled linear systems of n equations each, i.e., solves $AX = B$ for X, where A, X, and B are all $n \times n$ matrices) with only local communication and reduced computations, $O(n_l^3)$. This contrasts with and extends the distributed Jacobi algorithm in [35], which essentially solves $Ax = b$ for x, where A is a matrix, x and b are vectors.

In summary, the paper distributes the Kalman filter avoiding at every sensor the storage, computation, or communication of vectors or matrices of dimension n. Our distributed implementation has only local communication and local computation with vectors and matrices of reduced dimensions, $n_l \ll n$. We achieve this by: (1) distributing the global field into n_l-dimensional sensor-based reduced-order models through digraphs and cut-point sets; (2) fusing the observations and the estimates of the states common across more than one reduced-order model through bipartite fusion graphs and consensus averaging algorithms; and (3) by avoiding the direct inversion of $n-$dimensional covariance and information matrices. This is achieved by a generalized distributed matrix Jacobi (GDMJ) algorithm that we introduce and an $L-$banded matrix inversion theorem, [32]. These algorithms compute

the n_l-dimensional principal blocks in the inverses of the $n-$dimensional covariance and information matrices without the need to store or invert these $n-$dimensional matrices.

We describe the rest of the paper. Section II covers the background topics on state-space representation of a discrete-time dynamical system, centralized Information filters, and centralized $L-$banded Information filters. Section III covers the model distribution step detailing the formulation of reduced sensor-based models using system digraphs and cut-point sets. We introduce the local Information filters in section III-B along with the necessary notation. Section IV gives the observation fusion and the estimate fusion steps of the local Information filters, and section V presents the generalized distributed matrix Jacobi (GDMJ) algorithm. The filter step of the local Information filters is provided in section VI, and the prediction step of the local Information filters is provided in section VII. We conclude the paper with results in section VIII and conclusions in section IX. Appendix I discusses the $L-$banded inversion theorem, [32].

II. Background

In this section, we present the global state-space model of a sparse and localized large-scale dynamical system, see subsection II-A. This large-scale system is monitored by a sensor network and hence the observations of the field are distributed across the sensors. State estimation is achieved using the Information filter. To fix notation, subsection II-B presents the centralized version of the Information filter.

A. Global Model

The object of study is a large-scale discrete-time and discrete-space dynamical system that is sparse and localized, e.g., discrete-time random fields. We choose a scanning order so that the values of the field at the spatial lattice nodes at time k are collected in the column vector, $\mathbf{x}_k$. This is a standard procedure, e.g., in image processing the vector, $\mathbf{x}_k$, stacks the grey levels at every pixel in the image, or, when the large-scale system under study is described by partial difference equations, where the space indexing variables are then discretized by a lattice.

The state-space representation of the system is given by

$$\mathbf{x}_{k+1} = \mathbf{F}\mathbf{x}_k + \mathbf{G}\mathbf{u}_k, \qquad k \geq 0 \tag{1}$$

where k is the discrete time index, $\mathbf{x}_k \in \mathbb{R}^n$ is the state vector, $\mathbf{x}_0 \in \mathbb{R}^n$ are the initial state conditions, $\mathbf{F} \in \mathbb{R}^{n\times n}$ is the model matrix, $\mathbf{u}_k \in \mathbb{R}^j$ is the state noise vector and $\mathbf{G} \in \mathbb{R}^{n\times j}$ is the state noise matrix. When the system is sparse, the model matrix, $\mathbf{F}$, captures this localized and sparse structure of the system dynamics. Without loss of generality, we can add external inputs in (1), which are known for all time k. Specifically, we may add a term $\mathbf{B}\psi_k$, where $\psi_k \in \mathbb{R}^m$ is the external known input vector, and $\mathbf{B} \in \mathbb{R}^{n\times m}$ is the input matrix. The results below can be easily adjusted to account for this term, so we ignore it in the sequel.

This random field (1) is monitored by a network of N sensors. Observations at sensor l and time k are given by

$$\mathbf{y}_k^{(l)} = \mathbf{H}_l\mathbf{x}_k + \mathbf{w}_k^{(l)}, \tag{2}$$

where $\mathbf{H}_l \in \mathbb{R}^{p_l \times n}$ is the local observation matrix for sensor l, p_l is the number of observations made by sensor l, and $\mathbf{w}^{(l)} \in \mathbb{R}^{p_l}$ is the local observation noise.

We stack the observations at all N sensors in the sensor network to get the global observation model as follows. Let p be the total number of observations, adding the observations at all the sensors. Let the global observation vector, $\mathbf{y}_k \in \mathbb{R}^{p\times 1}$, the global observation matrix, $\mathbf{H} \in \mathbb{R}^{p\times n}$, and the global observation noise vector, $\mathbf{w}_k \in \mathbb{R}^{p\times 1}$, be

$$\mathbf{y}_k = \begin{bmatrix} \mathbf{y}_k^{(1)} \\ \vdots \\ \mathbf{y}_k^{(N)} \end{bmatrix}, \qquad \mathbf{H} = \begin{bmatrix} \mathbf{H}_1 \\ \vdots \\ \mathbf{H}_N \end{bmatrix}, \qquad \mathbf{w}_k = \begin{bmatrix} \mathbf{w}_k^{(1)} \\ \vdots \\ \mathbf{w}^{(N)} \end{bmatrix}. \tag{3}$$

Then the global observation model is given by

$$\mathbf{y}_k = \mathbf{H}\mathbf{x}_k + \mathbf{w}_k. \tag{4}$$

We adopt standard assumptions on the statistical characteristics of the noise. The state noise sequence, $\{\mathbf{u}_k\}_{k\geq 0}$, and the observation noise sequence, $\{\mathbf{w}_k\}_{k\geq 0}$, are both zero mean Gaussian white noise sequences with

$$E[\mathbf{u}_k\mathbf{u}_j^H] = \mathbf{Q}\delta_{ij} \text{ and } E[\mathbf{w}_k\mathbf{w}_j^H] = \mathbf{R}\delta_{ij}, \tag{5}$$

where 'H' denotes the Hermitian, the Kronecker delta $\delta_{ij} = 1$, if and only if $i = j$, and zero otherwise. The noise sequences are independent of the initial condition, $\mathbf{x}_0$, which is a zero mean Gaussian random vector with

$$E[\mathbf{x}_0\mathbf{x}_0^H] = \mathbf{S}_0. \tag{6}$$

Since the observation noise vectors at different sensors are independent, we can partition the global observation noise covariance matrix, $\mathbf{R}$, with $\mathbf{R}_l \in \mathbb{R}^{p_l\times p_l}$ being the local observation noise covariance matrix at sensor l, as

$$\mathbf{R} = \text{blockdiag}[\mathbf{R}_1, \ldots, \mathbf{R}_N]. \tag{7}$$

Equations (1) and (4) complete what we refer to as the *global model* of the system. We next provide the centralized Information filter for the global model.

B. Centralized Information Filter

We introduce the following notation for the centralized Information filter (CIF) for the global model (1) and (4). Let $\mathbf{S}_{k|k}$ and $\mathbf{S}_{k|k-1}$ be the filtered estimate error covariance matrix and the predicted estimate error covariance matrix, respectively, and their inverses be the information matrices, $\mathbf{Z}_{k|k}$ and $\mathbf{Z}_{k|k-1}$, respectively. Let $\widehat{\mathbf{x}}_{k|k}$ and $\widehat{\mathbf{x}}_{k|k-1}$ be the filtered estimate and the predicted estimate of the state vector, $\mathbf{x}_k$, based on all measurements up to time k and time $k-1$, respectively. We have the following relations.

$$\mathbf{S}_{k|k} = \mathbf{Z}_{k|k}^{-1} \tag{8}$$

$$\mathbf{S}_{k|k-1} = \mathbf{Z}_{k|k-1}^{-1} \tag{9}$$

$$\widehat{\mathbf{z}}_{k|k-1} = \mathbf{Z}_{k|k-1}\widehat{\mathbf{x}}_{k|k-1} \tag{10}$$

$$\widehat{\mathbf{z}}_{k|k} = \mathbf{Z}_{k|k}\widehat{\mathbf{x}}_{k|k} \tag{11}$$

Define the $n-$dimensional global observation variables as

$$\mathbf{i}_k = \mathbf{H}^T\mathbf{R}^{-1}\mathbf{y}_k, \tag{12}$$

$$\mathbf{I}_k = \mathbf{H}^T\mathbf{R}^{-1}\mathbf{H}, \tag{13}$$

and the $n-$dimensional local observation variables at sensor l as

$$\mathbf{i}_{l,k} = \mathbf{H}_l^T\mathbf{R}_l^{-1}\mathbf{y}_k^{(l)}, \tag{14}$$

$$\mathbf{I}_{l,k} = \mathbf{H}_l^T\mathbf{R}_l^{-1}\mathbf{H}_l. \tag{15}$$

When the observations are distributed among the sensors (39), the CIF can be implemented by collecting all the sensor observations at a central location. Another way of implementing the CIF is to fuse the distributed observations by realizing that the global observation variables in (12)−(13), can be written as the sum of the local observation variables (14)−(15), see [4], [6], [12]. Using (39), (3) and (7), in (12)−(13) we have,

$$\begin{aligned}\mathbf{i}_k &= \mathbf{H}^T\mathbf{R}^{-1}\mathbf{y}_k \\ &= \mathbf{H}_1^T\mathbf{R}_1^{-1}\mathbf{y}_k^{(1)} + \cdots + \mathbf{H}_N^T\mathbf{R}_N^{-1}\mathbf{y}_k^{(N)} \\ &= \sum_{l=1}^{N}\mathbf{H}_l^T\mathbf{R}_l^{-1}\mathbf{y}_k^{(l)} \\ &= \sum_{l=1}^{N}\mathbf{i}_{l,k}. \end{aligned} \tag{16}$$

Similarly,

$$\mathbf{I}_k = \sum_{l=1}^{N}\mathbf{I}_{l,k}. \tag{17}$$

The *filter step* of the CIF is given for $k \geq 0$ as

$$\mathbf{Z}_{k|k} = \mathbf{Z}_{k|k-1} + \sum_{l=1}^{N}\mathbf{I}_{l,k}, \tag{18a}$$

$$\widehat{\mathbf{z}}_{k|k} = \widehat{\mathbf{z}}_{k|k-1} + \sum_{l=1}^{N}\mathbf{i}_{l,k}. \tag{18b}$$

The *prediction step* of the CIF is given for $k \geq 1$ as

$$\mathbf{Z}_{k|k-1} = \mathbf{S}_{k|k-1}^{-1} = (\mathbf{F}\mathbf{Z}_{k-1|k-1}^{-1}\mathbf{F}^T + \mathbf{G}\mathbf{Q}\mathbf{G}^T)^{-1}, \qquad \mathbf{Z}_{0|-1} = \mathbf{S}_0^{-1}, \tag{19a}$$

$$\widehat{\mathbf{z}}_{k|k-1} = \mathbf{Z}_{k|k-1}\left(\mathbf{F}\mathbf{Z}_{k-1|k-1}^{-1}\widehat{\mathbf{z}}_{k-1|k-1}\right), \qquad \widehat{\mathbf{z}}_{0|-1} = \mathbf{0}. \tag{19b}$$

The CIF needs: (i) the knowledge of all the observations, $\mathbf{y}_k$, at a central location to implement, e.g., (16), a non-trivial communication task when the number of sensors, N, is large; and (ii) global filter computations, e.g., (19), an infeasible challenge when the number of states, n, is large. Further, the CIF has the disadvantages of large latency and a single point of failure.

C. Centralized $L-$Banded Information filters

To avoid the $O(n^3)$ computations of the global quantities in (19), e.g., the inversion, $\mathbf{Z}_{k-1|k-1}^{-1}$, we develop an approximation assuming that the information matrices, $\mathbf{Z}_{k|k}$ and $\mathbf{Z}_{k|k-1}$, are $L-$banded. We refer to the CIF with this approximation as the centralized $L-$banded Information filter (CLBIF). Simulations in [34], show that, under reasonable conditions, CLBIF gives virtually indistinguishable results when compared to the CIF.

This $L-$banded approximation of the information matrices is equivalent to approximating the Gaussian error processes,

$$\epsilon_{k|k} = \mathbf{x}_k - \widehat{\mathbf{x}}_{k|k}, \tag{20}$$

$$\epsilon_{k|k-1} = \mathbf{x}_k - \widehat{\mathbf{x}}_{k|k-1}, \tag{21}$$

to be Gauss-Markovian of Lth order [31]. Reference [32], shows that this approximation is optimal in Kullback-Leibler or maximum entropy sense. With this approximation, and using the algorithms provided in [32], the $O(n^3)$ matrix inversions in equations (19), can be achieved using $O(n^2)$ computation complexity.

The CLBIF with the $L-$banded approximations on the information matrices, $\mathbf{Z}_{k|k}$ and $\mathbf{Z}_{k|k-1}$, is given by the *filter step* in (18a)$-$(18b) and the *prediction step* in (19a)$-$(19b), but the $L-$banded information matrix, $\mathbf{Z}_{k-1|k-1}$, is inverted using a forward-backward recursion algorithm [32], with $O(n^2)$ computation complexity. The inversion of the prediction error covariance matrix, $\mathbf{S}_{k|k-1}$, to get the $L-$banded information matrix, $\mathbf{Z}_{k|k-1}$, is achieved with $O(n^2)$ computation complexity, using the $L-$banded inversion theorem provided in appendix I, see [32], for details. Although, this $L-$banded approximation reduces the computation to $O(n^2)$, it is still is a centralized algorithm as it deals with the $n-$dimensional state. To distribute the CLBIF, we start by distributing the global model (1)$-$(4) in the following section.

III. Distributing the Model

Although CLBIF reduces the computational complexity of the CIF from $O(n^3)$ to $O(n^2)$, it still involves operations with vectors and matrices of the dimensions, n, of the state vector. These operations are practically infeasible for the large-scale systems of interest here, since n is very large. Instead of implementing CLBIF based on the global model, we implement local Information filters at each sensor based on sensor-based reduced models. To obtain the reduced models, we distribute the global model into sensor-based models, i.e., each reduced sensor-based model captures the local dynamics of the large-scale system at that sensor. This is done by exploiting the sparse and localized structure of the model matrix, $\mathbf{F}$.

A. Distributing the Model: Reduced Models at Each Sensor

In this subsection, we show how to distribute the global model (1) and (4), in order to get the reduced order models at the sensors; parts of this procedure are also presented in [36]. We illustrate the procedure with the help

of a simple example. Consider a five dimensional system with the global dynamical model

$$\begin{aligned}\mathbf{x}_{k+1} &= \begin{bmatrix} f_{11} & f_{12} & 0 & 0 & 0 \\ f_{21} & f_{22} & 0 & f_{24} & 0 \\ f_{31} & 0 & f_{33} & 0 & 0 \\ 0 & 0 & f_{43} & 0 & f_{45} \\ 0 & 0 & 0 & f_{54} & f_{55} \end{bmatrix} \mathbf{x}_k + \begin{bmatrix} 0 & 0 \\ 0 & 0 \\ 0 & g_{32} \\ 0 & 0 \\ g_{51} & 0 \end{bmatrix} \mathbf{u}_k \\ &= \mathbf{F}\mathbf{x}_k + \mathbf{G}\mathbf{u}_k. \end{aligned} \tag{22}$$

The system has two external noise sources $\mathbf{u}_k = [u_{1k}, u_{2k}]^T$. We monitor this system with $N = 3$ sensors, having local observation vectors, $\mathbf{y}_k^{(l)}$, at each sensor l. The global observation vector, $\mathbf{y}_k$, stacks the local observation vectors, $\mathbf{y}_k^{(l)}$, corresponding to each sensor. The global observation model is

$$\begin{aligned}\mathbf{y}_k = \begin{bmatrix} y_k^{(1)} \\ y_k^{(2)} \\ y_k^{(3)} \end{bmatrix} &= \begin{bmatrix} 1 & 1 & 1 & 0 & 0 \\ 0 & 1 & 1 & 1 & 0 \\ 0 & 0 & 0 & 1 & 1 \end{bmatrix} \mathbf{x}_k + \begin{bmatrix} w_k^{(1)} \\ w_k^{(2)} \\ w_k^{(3)} \end{bmatrix} \\ &= \mathbf{H}\mathbf{x}_k + \mathbf{w}_k, \end{aligned} \tag{23}$$

where $\mathbf{H} = [\mathbf{H}_1^T, \mathbf{H}_2^T, \mathbf{H}_3^T]^T$. Equations (22) and (23) comprise the *global model* of our example system that we distribute by formulating reduced order models.

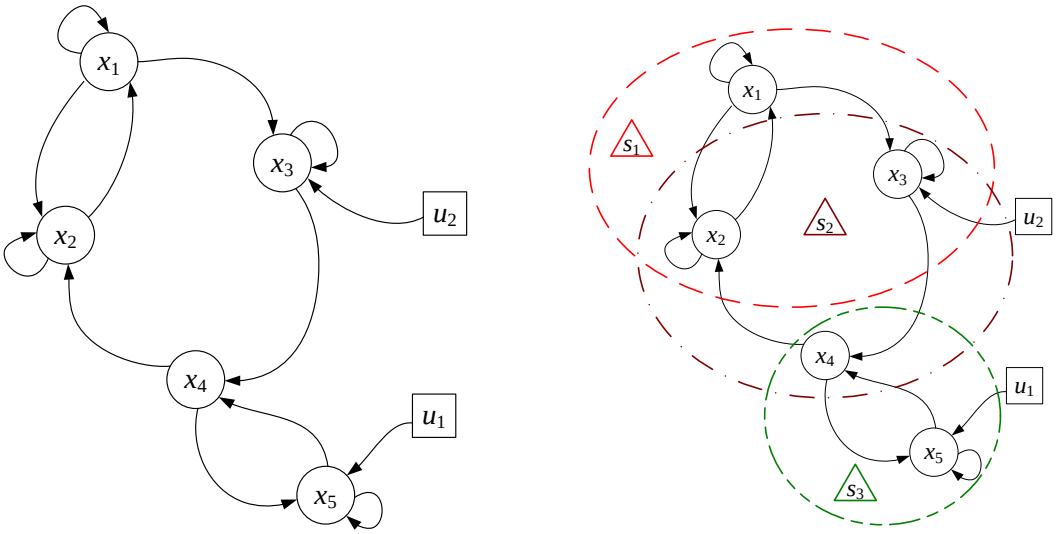
Subsection 1 provides a graphical representation of the global model using the concept of *systems digraphs*, and subsection 2 distributes this global model into reduced models by using the concept of *cut-point sets* on the system digraph.

1) Graphical Representation using System Digraphs: A system digraph helps us in visualizing the dynamical interdependence of the system. A system digraph, [37], or the state-dependency graph, $\mathcal{J} = [V, E]$, is a directed graphical representation of the system, where V is the vertex set that contains the state set, $X = \{x_i\}_{i=1,\ldots,n}$, and the noise input set, $U = \{u_i\}_{i=1,\ldots,j}$, of the system, and E is the edge matrix or the interconnection matrix of the system. The interconnection matrix, E, is the binary representation (having a 1 for each non-zero entry) of the model matrix, $\mathbf{F}$, and the state noise matrix, $\mathbf{G}$, concatenated together, $E = [\overline{\mathbf{F}}|\overline{\mathbf{G}}]$, where the overbar indicates the binary representation. The interconnection matrix, E, for the system in (22) is,

$$E = \begin{bmatrix} 1 & 1 & 0 & 0 & 0 & 0 & 0 \\ 1 & 1 & 0 & 1 & 0 & 0 & 0 \\ 1 & 0 & 1 & 0 & 0 & 0 & 1 \\ 0 & 0 & 1 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 & 1 & 1 & 0 \end{bmatrix}. \tag{24}$$

The system digraph is shown in figure 1(a).

2) Reduced Models from Cut-sets: We have $N = 3$ sensors monitoring the system (figure 1(a)) through the observation model (23). We associate to each sensor l a *cut-point set*, $V^{(l)}$, where $V^{(l)} \subseteq X$. If a sensor observes a linear combination of several states, the cut-point set at that particular sensor, $V^{(l)}$, includes all these states; see [38],



(a) Digraph representation of the 5 dimensional system, (22)−(23). The circles represent the states, $\mathbf{x}$, of the system and the squares represent the input noise sources, $\mathbf{u}$.

(b) The cut-point sets associated to the 3 sensors ($\triangle$) are shown by the dashed circles.

Fig. 1. System Digraph and cut-point sets.

for an alternative definition of the cut-point sets. The cut-point sets provide a natural procedure for selecting the local states involved in the local dynamics at each sensor. From (23), we have the following cut-point sets,

$$V^{(1)} = \{x_1, x_2, x_3\}, \tag{25}$$

$$V^{(2)} = \{x_2, x_3, x_4\}, \tag{26}$$

$$V^{(3)} = \{x_4, x_5\}. \tag{27}$$

The cut-point sets corresponding to the three sensors are shown in figure 1(b). For simplicity of the presentation, we chose here that each state variable is observed by at least one sensor. If this is not true, we can easily account for this by extending the cut-point sets, $V^{(l)}$, to $\overline{V}^{(l)}$, such that

$$\bigcup_{l=1}^{N} \overline{V}^{(l)} = X. \tag{28}$$

The local states at sensor l, i.e., the components of the local state vector, $\mathbf{x}_k^{(l)}$, are the elements in its associated cut-point set, $V^{(l)}$. The directed edges coming into a cut-point set are the inputs required by the reduced model. In the context of our running illustration (22)−(23), we see that the local state vector for sensor 1 is, $\mathbf{x}_k^{(1)} = [x_{1,k}, x_{2,k}, x_{3,k}]^T$, and the inputs to the local model consist of a subset of the state set, X, (at sensor s_1, $x_{4,k}$ is the input coming from sensor s_2) and a subset of the noise input set, U, ($u_{2,k}$ at sensor s_1).

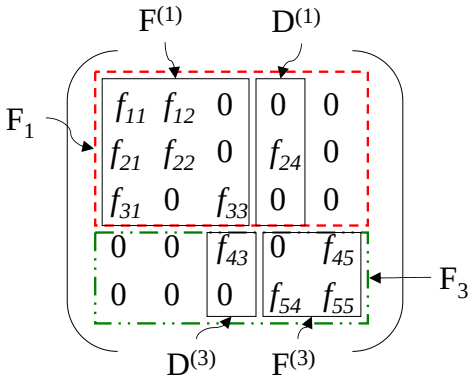


Fig. 2. Partitioning of the global model matrix, $\mathbf{F}$, into local model matrices, $\mathbf{F}^{(l)}$, and the local internal input matrices, $\mathbf{D}^{(l)}$, shown for sensor 1 and sensor 3, from the example system, (22)−(23).

For the local model at sensor l, we collect the states being required as input in a local internal (we use the word *internal* to distinguish from the externally applied input, ψ_k, which might be a part of (1)) input vector, $\mathbf{d}_k^{(l)}$, and the noise sources being required as input in a local noise input vector, $\mathbf{u}_k^{(l)}$. We can collect the entries corresponding to the local state vector, $\mathbf{x}_k^{(l)}$, from the model matrix, $\mathbf{F}$, in a local model matrix, $\mathbf{F}^{(l)}$, the entries corresponding to the local internal input vector, $\mathbf{d}_k^{(l)}$, from the model matrix, $\mathbf{F}$, in a local internal input matrix, $\mathbf{D}^{(l)}$, and the entries corresponding to the local noise input vector, $\mathbf{u}_k^{(l)}$, from the state noise matrix, $\mathbf{G}$, in a local state noise matrix, $\mathbf{G}^{(l)}$. Figure 2 shows this partitioning for sensors s_1 and s_3. We have the following local models for (22).

$$\begin{aligned}\mathbf{x}_{k+1}^{(1)} &= \begin{bmatrix} f_{11} & f_{12} & 0 \\ f_{21} & f_{22} & 0 \\ f_{31} & 0 & f_{33} \end{bmatrix} \mathbf{x}_k^{(1)} + \begin{bmatrix} 0 \\ f_{24} \\ 0 \end{bmatrix} x_{4,k} + \begin{bmatrix} 0 \\ 0 \\ g_{32} \end{bmatrix} u_{2,k}, \\ &= \mathbf{F}^{(1)}\mathbf{x}_k^{(1)} + \mathbf{D}^{(1)}\mathbf{d}_k^{(1)} + \mathbf{G}^{(1)}\mathbf{u}_k^{(1)} \end{aligned} \tag{29}$$

$$\begin{aligned}\mathbf{x}_{k+1}^{(2)} &= \begin{bmatrix} f_{22} & 0 & f_{42} \\ 0 & f_{33} & 0 \\ 0 & f_{43} & 0 \end{bmatrix} \mathbf{x}_k^{(2)} + \begin{bmatrix} f_{21} & 0 \\ f_{31} & 0 \\ 0 & f_{45} \end{bmatrix} \begin{bmatrix} x_{1,k} \\ x_{5,k} \end{bmatrix} + \begin{bmatrix} 0 \\ g_{32} \\ 0 \end{bmatrix} u_{2,k}, \\ &= \mathbf{F}^{(2)}\mathbf{x}_k^{(2)} + \mathbf{D}^{(2)}\mathbf{d}_k^{(2)} + \mathbf{G}^{(2)}\mathbf{u}_k^{(2)} \end{aligned} \tag{30}$$

$$\begin{aligned}\mathbf{x}_{k+1}^{(3)} &= \begin{bmatrix} 0 & f_{45} \\ f_{54} & f_{55} \end{bmatrix} \mathbf{x}_k^{(3)} + \begin{bmatrix} f_{43} \\ 0 \end{bmatrix} x_{3,k} + \begin{bmatrix} 0 \\ g_{51} \end{bmatrix} u_{1,k}, \\ &= \mathbf{F}^{(3)}\mathbf{x}_k^{(3)} + \mathbf{D}^{(3)}\mathbf{d}_k^{(3)} + \mathbf{G}^{(3)}\mathbf{u}_k^{(3)} \end{aligned} \tag{31}$$

We capture the above extraction of the local states by the cut-point sets, with the following procedure. Let the total number of states in the cut-point set at sensor l, $V^{(l)}$, be n_l. Let $\mathbf{T}_l$ be an $n_l \times n$ selection matrix, such that

it selects n_l states in the cut-point set, $V^{(l)}$, from the entire state vector, $\mathbf{x}_k$, according to the following relation,

$$\mathbf{x}_k^{(l)} = \mathbf{T}_l \mathbf{x}_k. \tag{32}$$

For sensor 1, the selection matrix, $\mathbf{T}_1$ is

$$\mathbf{T}_1 = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 \end{bmatrix}. \tag{33}$$

We establish a *reduced* local observation matrix, $\mathbf{H}^{(l)}$, by retaining the terms corresponding to the local state vector, $\mathbf{x}_k^{(l)}$, from the local observation matrix, $\mathbf{H}_l$. We may write

$$\mathbf{H}^{(l)} = \mathbf{H}_l \mathbf{T}_l^{\#}, \tag{34}$$

where '$\#$' denotes the pseudo-inverse of the matrix. In the context of the running illustration, the reduced local observation matrix $\mathbf{H}^{(1)} = [1, 1, 1]$ is obtained from the local observation matrix $\mathbf{H}_1 = [1, 1, 1, 0, 0]$. Note that $\mathbf{H}_l$ picks the states from the global state vector, $\mathbf{x}_k$, whereas $\mathbf{H}^{(l)}$ picks the states from the local state vector, $\mathbf{x}_k^{(l)}$. The reduced local observation models are given by

$$\mathbf{y}_k^{(l)} = \mathbf{H}^{(l)} \mathbf{x}_k^{(l)} + \mathbf{w}_k^{(l)}. \tag{35}$$

We now make some additional comments. For simplicity of the explanation, we refer to our running example, (22)$-$(23). We note that the reduced models at the sensors overlap, as shown by overlapping cut-point sets in figure 1(b). Due to this overlap, observations and estimates corresponding to the shared states are available at multiple sensors. For this reason, the observations and estimates corresponding to the overlapped states must be fused. For example, sensor 1 and sensor 2 share the state variables $x_{2,k}$ and $x_{3,k}$ in their local models, and, hence, share the observations and estimates corresponding to these states, which should be fused. We further note that the reduced model (29) at sensor 1 is coupled to the reduced model (30) at sensor 2 through the state $x_{4,k}$. The state $x_{4,k}$ at sensor 1 does not appear in the local state vector, i.e., $x_{4,k} \notin \mathbf{x}_k^{(1)}$. But it is still required as an internal input at sensor 1 to preserve the global dynamics. Hence, sensor 2 will communicate the state $x_{4,k}$, which appears in its local state vector, i.e., $x_{4,k} \in \mathbf{x}_k^{(2)}$, to sensor 1. Since the value of the state itself is unknown, sensor 2 communicates its estimate, $\widehat{x}_{4,k|k}^{(2)}$, to sensor 1. This allows sensor 1 to complete its local model and preserve global dynamics, thus, taking into account the coupling between the local reduced-order models. This process is repeated at all sensors. Hence, the local internal input vector, $\mathbf{d}_k^{(l)}$, is replaced by its estimate, $\widehat{\mathbf{d}}_{k|k}^{(l)}$. From the above discussion, we derive the reduced model at an arbitrary sensor l to be

$$\mathbf{x}_{k+1}^{(l)} \quad = \quad \mathbf{F}^{(l)} \mathbf{x}_k^{(l)} + \mathbf{D}^{(l)} \widehat{\mathbf{d}}_{k|k}^{(l)} + \mathbf{G}^{(l)} \mathbf{u}_k^{(l)}. \tag{36}$$

B. Local Information Filters

To estimate the global state vector, $\mathbf{x}_k$, in a distributed fashion, we implement local Information filters (LIFs) at each sensor l, which are based on the sensor-based reduced models (36) and (35). For complete distribution,

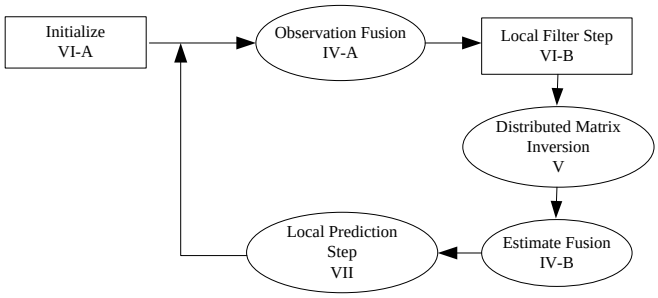


Fig. 3. Block Diagram for the LIFs: The steps involved in the LIF implementation are shown. The ovals represent the steps that require local communication.

nowhere at any sensor should we compute, store, or communicate any global variable, an $n-$dimensional matrix (or a vector) involved in CLBIF. Each LIF computes local objects (matrices and vectors of dimension n_l), which are then updated (if required) by exchanging information among the neighbors using only local communication. Some of the update procedures are iterative. If the iterative algorithms converge, the local variables over the entire sensor network possess the global knowledge in the CLBIF. Of course, in most applications nowhere in the network is there the need for this global knowledge — so, the global knowledge is known only in distributed fashion. The LIFs asymptotically guarantee the performance of the CLBIF.

The LIFs consist of initial conditions, a local filter step (including observation fusion and distributed matrix inversion) and a local prediction step (including estimate fusion), see figure 3. The LIFs are presented in the next four sections. Section IV discusses the observation fusion (section IV-A) and estimate fusion (section IV-B), which are required due to the overlapping state vectors in the reduced-order models. Section V discusses a distributed matrix inversion technique, which we call the generalized distributed matrix Jacobi algorithm, required in the local predictions step (section VII). Section VI contains the initial conditions and the local filter step. The local prediction step is then presented in section VII.

To proceed with the next sections, we provide notation in the following subsection.

1) Notation: The superscript (l) refers to a local reduced-order variable ($n_l \times 1$ vector or $n_l \times n_l$ matrix) at sensor l. We define the reduced observation vector, $\mathbf{i}_k^{(l)}$, and the reduced observation matrix, $\mathbf{I}_k^{(l)}$, as

$$\mathbf{i}_k^{(l)} = (\mathbf{H}^{(l)})^T \mathbf{R}_l^{-1} \mathbf{y}_k^{(l)}, \tag{37}$$

$$\mathbf{I}_k^{(l)} = (\mathbf{H}^{(l)})^T \mathbf{R}_l^{-1} \mathbf{H}^{(l)}. \tag{38}$$

The local error covariance matrices, $\mathbf{S}_{k|k}^{(l)}$ and $\mathbf{S}_{k|k-1}^{(l)}$, are the overlapping diagonal submatrices of the global error covariance matrices, $\mathbf{S}_{k|k}$ and $\mathbf{S}_{k|k-1}$. Let $\mathbf{Z}_{k|k}^{(l)}$ and $\mathbf{Z}_{k|k-1}^{(l)}$ be the local information matrices. These local information matrices are overlapping diagonal submatrices of the global $L-$banded information matrices, $\mathbf{Z}_{k|k}$ and $\mathbf{Z}_{k|k-1}$. These local matrices overlap because the reduced sensor-based models (36) have overlapping state vectors, $\mathbf{x}_k^{(l)}$. Global information matrices and global error covariance matrices are related by (8) and (9). Figure 4 captures the relationship between the local error covariance matrices and the local information matrices, when we

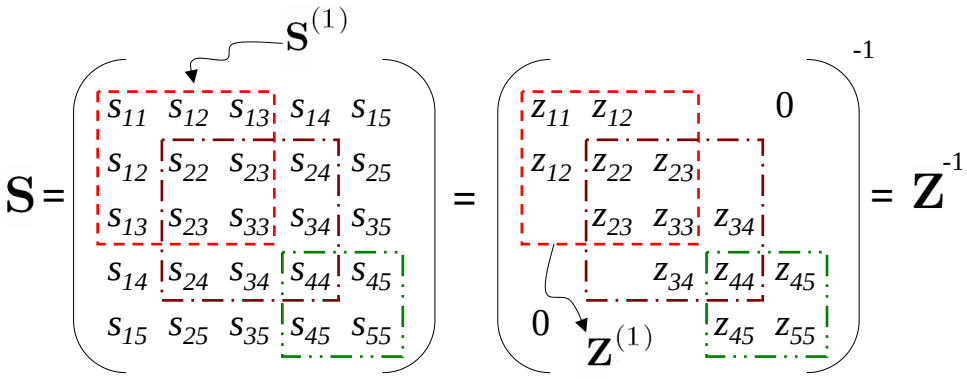


Fig. 4. Relationship between the global error covariance matrices, $\mathbf{S}$ and its inverse, the global information matrices, $\mathbf{Z}$, with $L = 1-$banded approximation on $\mathbf{Z}$. The figure also shows how the local matrices, $\mathbf{S}^{(l)}$ and $\mathbf{Z}^{(l)}$, constitute their global counterparts. Since this relation holds for both the estimation and prediction matrices, the figure removes the subscripts.

implement LIFs on our example system (22)−(23) with reduced models (29)−(31). We remove the subscripts from the variables, $\mathbf{S}$ and $\mathbf{Z}$, since figure 4 holds for both estimation and prediction matrices.

We note here that the local information matrices correspond to the local state vectors, e.g., in the context of (22), $\mathbf{x}_k^{(1)}$ has a local information matrix, $\mathbf{Z}_{k|k}^{(1)}$, spanning the portion of the global information matrix, $\mathbf{Z}_{k|k}$, that corresponds to the state variables $\{x_{1,k}, x_{2,k}, x_{3,k}\}$ in $\mathbf{x}_k^{(1)}$, as shown in figure 4. A necessary condition for the LIFs to achieve the performance of the CLBIF is that the $L-$band of the global information matrices, $\mathbf{Z}_{k|k}$ and $\mathbf{Z}_{k|k-1}$, is preserved among the local information matrices, $\mathbf{Z}_{k|k}^{(l)}$ and $\mathbf{Z}_{k|k-1}^{(l)}$, i.e., with a slight abuse of notation the following should hold, see figure 4,

$$\bigcup_{l=1}^{N} \mathbf{Z}_{k|k}^{(l)} \supseteq \text{L-band}\left[\mathbf{Z}_{k|k}\right]. \tag{39}$$

Hence, if (39) is not satisfied, we again extend the cut-point sets at each sensor, as shown in (28), so that each local model has at least that many state variables in the local state vector that preserve the $L-$band of the global information matrices and (39) holds.

IV. Overlapping Reduced Models

After the model distribution step introduced in section III, the reduced models among the sensors may share states, as shown by the overlapped cut-point sets in figure 1(b). Since the sensors sharing the states have independent observations of the shared states, observations corresponding to the shared states need to be fused. Since each sensor implements a separate LIF, the estimates corresponding to the shared states also need to be fused across the LIFs containing those states. We present observation fusion in subsection IV-A and estimate fusion in subsection IV-B, with the help of bipartite fusion graphs, [39].

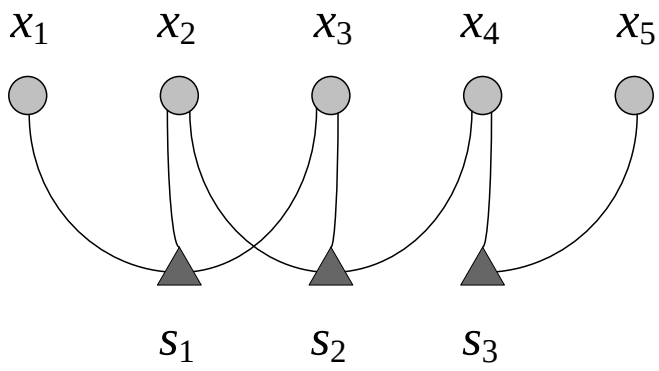


Fig. 5. Bipartite Fusion graph, $\mathcal{B}$, is shown for the example system, (22)−(23).

A. Observation Fusion

Since the local filter step involves local observations, $\mathbf{y}_k^{(l)}$, and sensors having common states in their reduced models have different observations corresponding to these common states, we require observation fusion. Equations (16) and (17) show that the observation fusion is equivalent to adding the corresponding $n-$dimensional local observation variables, (14)−(15). In CLBIF, we implement this fusion directly because each local observation variable in (14)−(15) corresponds to the full $n-$dimensional state vector, $\mathbf{x}_k$. Since the n_l-dimensional *reduced* observation variables, (37)−(38), correspond to different local state vectors, $\mathbf{x}_k^{(l)}$, they cannot be added directly.

To achieve the observation fusion we introduce the following undirected bipartite fusion graph, $\mathcal{B}$. Let $S_N = \{s_1, \ldots, s_N\}$ be the set of sensors and X the set of states. The vertex set of the bipartite fusion graph, $\mathcal{B}$, is $S_N \bigcup X$. We now define the edge set, $E_{\mathcal{B}}$, of the fusion graph, $\mathcal{B}$. The sensor s_i is connected to the state variable x_j, if s_i observes the state variable x_j. In other words, we have an edge between sensor s_i and state variable x_j, if the local observation matrix, $\mathbf{H}_i$, at sensor s_i, contains a non-zero entry in its jth column. Figure 5 shows the bipartite graph for the example system in (22)−(23). Note that there are no edges within the state set, X, or within the sensor set, S_N, in the bipartite fusion graph, $\mathcal{B}$.

We now further provide some notation on the communication topology. Let $\mathcal{G}$ denote the sensor communication graph that determines how the sensors communicate among themselves. Let $\mathcal{K}(l)$ be the subgraph of $\mathcal{G}$, that contains the sensor l and the sensors directly connected to the sensor l. For the jth state, x_j, let $\mathcal{G}_j$ be the induced subgraph of the sensor communication graph, $\mathcal{G}$, such that it contains the sensors having the state x_j in their reduced models. Note here the difference between the state-dependency graph, $\mathcal{J}$, that captures the dynamic behavior of the system given by model matrices, $\mathbf{F}$ and $\mathbf{G}$, the sensor communication graph, $\mathcal{G}$, that captures how the sensors communicate with each other, and the bipartite fusion graph, $\mathcal{B}$, defined above.

The set of vertices in the graphs $\mathcal{G}_j$ come directly form the bipartite fusion graph, $\mathcal{B}$. For example, from figure 5, we see that $\mathcal{G}_1$ contains s_1 as a single vertex, $\mathcal{G}_2$ contains s_1, s_2 as vertices, and so on. The interconnection matrix of the sub-graph $\mathcal{G}_j$ comes from the sensor communication graph, $\mathcal{G}$. States having more than one sensor connected to

them in the bipartite fusion graph, $\mathcal{B}$, are the states for which fusion is required, since we have multiple observations for that state. Furthermore, figure 5 also gives the vertices in the associated subgraphs, $\mathcal{G}_j$, over which the fusion is to be carried out.

With the help of the above discussion, we establish the fusion of the reduced observation variables, (37)−(38). The reduced model at each sensor involves n_l state variables, and each element in the $n_l \times 1$ reduced observation vector, $\mathbf{i}_k^{(l)}$, corresponds to one of these states, i.e., each entry in $\mathbf{i}_k^{(l)}$ has some information about its corresponding state variable. Let the entries of the $n_l \times 1$ reduced observation vector, $\mathbf{i}_k^{(l)}$, at sensor l, be subscripted by the n_l state variables modeled at sensor l. In the context of the example given by system (22)−(23), we have

$$\mathbf{i}_k^{(1)} = \begin{bmatrix} i_{k,x_1}^{(1)} \\ i_{k,x_2}^{(1)} \\ i_{k,x_3}^{(1)} \end{bmatrix}, \qquad \mathbf{i}_k^{(2)} = \begin{bmatrix} i_{k,x_2}^{(2)} \\ i_{k,x_3}^{(2)} \\ i_{k,x_4}^{(2)} \end{bmatrix}, \qquad \mathbf{i}_k^{(3)} = \begin{bmatrix} i_{k,x_4}^{(3)} \\ i_{k,x_5}^{(3)} \end{bmatrix}. \tag{40}$$

For each state x_j, the observation fusion is carried out on the sensors attached to this state in the bipartite fusion graph, $\mathcal{B}$. The fused observation vectors denoted by $\mathbf{i}_{f,k}^{(l)}$ are given by

$$\mathbf{i}_{f,k}^{(1)} = \begin{bmatrix} i_{k,x_1}^{(1)} \\ i_{k,x_2}^{(1)} + i_{k,x_2}^{(2)} \\ i_{k,x_3}^{(1)} + i_{k,x_3}^{(2)} \end{bmatrix}, \qquad \mathbf{i}_{f,k}^{(2)} = \begin{bmatrix} i_{k,x_2}^{(2)} + i_{k,x_2}^{(1)} \\ i_{k,x_3}^{(2)} + i_{k,x_3}^{(1)} \\ i_{k,x_4}^{(2)} + i_{k,x_4}^{(3)} \end{bmatrix}, \qquad \mathbf{i}_{f,k}^{(3)} = \begin{bmatrix} i_{k,x_4}^{(3)} + i_{k,x_4}^{(2)} \\ i_{k,x_5}^{(3)} \end{bmatrix}. \tag{41}$$

Generalizing to the arbitrary sensor l, we may write the entry, $i_{f,k,x_j}^{(l)}$, corresponding to x_j in the fused observation vector, $\mathbf{i}_{f,k}^{(l)}$ as

$$i_{f,k,x_j}^{(l)} = \sum_{s \in \mathcal{G}_j} i_{k,x_j}^{(s)}, \tag{42}$$

where $i_{k,x_j}^{(s)}$ is the entry corresponding to x_j in the reduced observation vector at sensor s, $\mathbf{i}_k^{(s)}$.

Since the communication network on $\mathcal{G}_j$ may not be all-to-all, an iterative weighted averaging algorithm [40], is used to compute the fusion in (42) over arbitrarily connected communication networks with only local communication. A similar procedure on the pairs of state variables and their associated subgraphs can be implemented to fuse the reduced observation matrices, $\mathbf{I}_k^{(l)}$.

B. Estimate Fusion

For each state x_j, the LIF at each sensor $s \in \mathcal{G}_j$ provides an estimate of this state, $\widehat{x}_{j,k|k}^{(s)}$. If $\mathcal{G}_j$ contains more than one sensor, there are multiple correlated estimates of the same state, which should be fused in order to obtain an estimate with smaller variance, a problem considered by Durbin [30]; its vector extensions can be found in [41], [42], [43]. At each sensor $s \in \mathcal{G}_j$, let $\pi_j^{(s)}$ be the variance of the jth state estimate, $\widehat{x}_{j,k|k}^{(s)}$, where $\pi_j^{(s)}$ is a diagonal element in the local estimation error covariance matrix, $\mathbf{S}_{k|k}^{(s)}$, at sensor s. We fuse the estimates using the parallel fusion of estimates formula, which for scalar quantities can be derived by using Lagrange multipliers and is given by

$$\widehat{x}_{j,k|k} = \left(\sum_{s \in \mathcal{G}_j} \left(\pi_j^{(s)} \right)^{-1} \right)^{-1} \left(\sum_{s \in \mathcal{G}_j} \left(\pi_j^{(s)} \right)^{-1} \widehat{x}_{j,k|k}^{(s)} \right). \tag{43}$$

Each of the sums in (43) is carried out using the weighted averaging algorithm [40], and then combined to compute (43) at each sensor $s \in \mathcal{G}_j$.

V. Distributed Matrix Inversion with Local Communication

As we will show in the local prediction step (section VII), a crucial step in distributing the CLBIF computations is to compute the local prediction information matrix, $\mathbf{Z}_{k+1|k}^{(l)}$, from the local estimation information matrix, $\mathbf{Z}_{k|k}^{(l)}$, since this computation involves the local error covariance matrices, $\mathbf{S}_{k|k}^{(l)}$.

Consider the example model (22)$-$(23), when we employ LIFs on the distributed models (29)$-$(31). The local estimation information matrices, $\mathbf{Z}_{k|k}^{(1)}$, $\mathbf{Z}_{k|k}^{(2)}$, and $\mathbf{Z}_{k|k}^{(3)}$, correspond to the overlapping diagonal submatrices of the global 5×5 estimation information matrix, $\mathbf{Z}_{k|k}$, see figure 4, with $L = 1-$banded assumption on $\mathbf{Z}_{k|k}$. These *overlapping* submatrices are denoted by a superscript (l) for the lth sensor, $\mathbf{Z}_{k|k}^{(l)}$ and $\mathbf{S}_{k|k}^{(l)}$, as shown in figure 4. It will be shown (see section VII-A) that the local prediction information matrix, $\mathbf{Z}_{k+1|k}^{(l)}$, is a function of the local error covariance matrices, $\mathbf{S}_{k|k}^{(l)}$, and hence we need to compute $\mathbf{S}_{k|k}^{(l)}$ from the local filter information matrices, $\mathbf{Z}_{k|k}^{(l)}$, which we get from the local filter step, see section VI. As can be seen from figure 4 and (8), for these local submatrices,

$$\mathbf{S}_{k|k}^{(l)} \neq \left(\mathbf{Z}_{k|k}^{(l)}\right)^{-1}, \tag{44}$$

since the inverse relation (8) is on their global counterparts, $\mathbf{S}_{k|k}$ and $\mathbf{Z}_{k|k}$.

Collecting all the local information matrices, $\mathbf{Z}_{k|k}^{(l)}$, at each sensor and then carrying out an $n \times n$ matrix inversion is *not* a practical solution for large-scale systems (where n may be large), because of the large communication overhead and $O(n^3)$ computational cost. Using the $L-$banded structure on the global estimation information matrix, $\mathbf{Z}_{k|k}$, we present below a generalized distributed matrix Jacobi (GDMJ) algorithm[1]. For notational convenience, we disregard the time indices in the following development.

We present the centralized Jacobi algorithm for solving a linear system of n equations and its generalization to solve matrix inversion in section V-A. We then present the GDMJ algorithm with the help of an example in section V-B.

A. Centralized Jacobi Algorithm

The centralized Jacobi algorithm for vectors [33], solves a linear system of n equations, iteratively by successive substitution. It can be easily extended to get the centralized Jacobi algorithm for matrices that solves

$$\mathbf{ZS} = \mathbf{T}, \tag{45}$$

[1]Forward-Backward Recursion Algorithm: For an $L-$banded matrix, $\mathbf{Z}_{k|k}$, a forward-backward recursion algorithm to invert $\mathbf{Z}_{k|k}$ using only the submatrices in its $L-$band is provided in [32]. This algorithm computes the local error covariance matrices, $\mathbf{S}_{k|k}^{(l)}$, from the local information matrices, $\mathbf{Z}_{k|k}^{(l)}$. The problem in implementing this in a multisensor environment is that it is sequential: involving a forward recursion to start from the first sensor and reach the last sensor, and a backward recursion that proceeds in the opposite direction. Since, the iterations involve serial communication of the local matrices among all the sensors, the associated latency is impractical, besides requiring an inordinate amount of communication.

for the unknown matrix, $\mathbf{S}$, where the matrices $\mathbf{Z}$ and $\mathbf{T}$ are known. Let $\mathbf{M} = \text{diag}(\mathbf{Z})$, then

$$\mathbf{S}_{t+1} = \mathbf{M^{-1}}(\mathbf{M} - \mathbf{Z})\mathbf{S}_t + \mathbf{M^{-1}}\mathbf{T}, \tag{46}$$

converges to $\mathbf{S}$, giving us the centralized Jacobi algorithm for matrices, solving n coupled linear systems of equations (45). Putting $\mathbf{T} = \mathbf{I}$, we can solve for

$$\mathbf{ZS} = \mathbf{I} \Rightarrow \mathbf{S} = \mathbf{Z}^{-1}. \tag{47}$$

Hence, if $\mathbf{Z}$ is known, (48) converges to its inverse, $\mathbf{Z}^{-1}$,

$$\mathbf{S}_{t+1} = \mathbf{P}\mathbf{S}_t + \mathbf{M^{-1}}, \tag{48}$$

where

$$\mathbf{P} = \mathbf{M}^{-1}(\mathbf{M} - \mathbf{Z}). \tag{49}$$

Equation (48) is a centralized algorithm as it requires the complete $n \times n$ matrices involved. This requires global communication and an nth order computation at each iteration of the algorithm. We distribute below the centralized Jacobi algorithm given by (48) with the help of a GDMJ algorithm, with only local communication and reduced-order computations.

B. Generalized Distributed Matrix Jacobi (GDMJ) Algorithm

We present here the GDMJ algorithm with the help of an example in the context of the $5-$dimensional system, (22)$-$(23), we presented in section III-A. This is done for illustration purposes. The general case is easy to grasp from this example. With the sensor-based reduced models (29)$-$(31) and $L = 1$ banded approximation, the local estimation information matrices are shown in figure 4. We mention here that the matrix $\mathbf{P}$ in (49), being a product of a diagonal matrix and an $L-$banded matrix, is always $L-$banded. The submatrices, $\mathbf{P}^{(l)}$, in $\mathbf{P}$, at sensor l can be computed by

$$\mathbf{P}^{(l)} = \left(\mathbf{M}^{(l)}\right)^{-1}\left(\mathbf{M}^{(l)} - \mathbf{Z}^{(l)}\right), \tag{50}$$

where $\mathbf{M}^{(l)} = \text{diag}\left(\mathbf{Z}^{(l)}\right)$. Note that equation (50), does not require any communication and can be computed at each sensor directly from the local information matrix, $\mathbf{Z}^{(l)}$. This is because the matrix $\mathbf{M}$ is diagonal, and its local submatrix, $\mathbf{M}^{(l)}$ in (50), is the exact inverse of $\mathbf{Z}^{(l)}$.

Using the $L-$bandedness of the global matrix, $\mathbf{P}$ in (48), we can partition the computations in (48). These partitions are coupled due to the bandedness of $\mathbf{P}$. This is shown for sensor 3 as follows. Let p_{ij} denote the $ij-$th element of the 5×5, $L = 1-$banded matrix, $\mathbf{P}$. At sensor 3, the computation of each element of the 2×2 local error covariance matrix (see figure 4), $\mathbf{S}^{(3)}$, can be written from (48) as

$$s_{44,t+1} = p_{43}s_{34,t} + p_{44}s_{44,t} + p_{45}s_{45,t} \tag{51}$$

$$s_{55,t+1} = p_{54}s_{45,t} + p_{55}s_{55,t} \tag{52}$$

$$s_{45,t+1} = p_{43}s_{35,t} + p_{44}s_{45,t} + p_{45}s_{55,t} \tag{53}$$

Note that we are not iterating on s_{54}, since $\mathbf{S}$ is symmetric and $s_{54} = s_{45}$. All the elements involved in (51)$-$(53), except p_{43}, $s_{34,t}$, and $s_{35,t}$, are the elements of the local matrices, $\mathbf{P}^{(3)}$ and $\mathbf{S}_t^{(3)}$, at sensor 3. The elements, p_{43} and $s_{34,t}$, in (51), are the elements of the local matrices, $\mathbf{P}^{(2)}$ and $\mathbf{S}_t^{(2)}$, at sensor 2, respectively. Sensor 2, therefore, communicates these elements to sensor 3.

The element $s_{35,t}$ does not lie in the $L-$band of $\mathbf{S}$, and, hence, does not belong to any local error covariance matrix, $\mathbf{S}^{(l)}$, see figure 4. On the other hand, this element is required, since it is involved in the iterations on $s_{45,t+1}$ in (53). Since s_{35} lies in the non $L-$band, iterating on it requires further elements in the non $L-$band of $\mathbf{S}$, and we end up spanning the entire matrix, $\mathbf{S}$. This can be seen by writing out the equation for $s_{35,t}$, from (48),

$$s_{35,t+1} = p_{32}s_{25,t} + p_{33}s_{35,t} + p_{34}s_{45,t}. \tag{54}$$

The computation in (54), involves $s_{25,t}$, iterating on which, in turn, requires another off $L-$band element, $s_{15,t}$, and so on. Computing the elements outside the $L-$band, thus, requires iterating on all the elements in a single row of $\mathbf{S}$, at the sensor corresponding to that row. Hence, a single iteration of the algorithm, although distributed and requiring only local communication, sweeps the entire rows in $\mathbf{S}$, which is not feasible when n is large.

To address this problem, we assume that $\mathbf{S}_t$ is the inverse of an $L-$banded matrix and use the results in [34], to compute the off $L-$band element of $\mathbf{S}_t$ from its in band elements. Instead of iterating on s_{35} as in (54), since s_{35} lies in the non $L-$band of $\mathbf{S}$, it can be computed from the elements in the $L-$band[2],

$$s_{35,t} = s_{34,t}s_{44,t}^{-1}s_{45,t}, \tag{55}$$

where $s_{34,t}$ is already communicated to sensor 3 by sensor 2. Combining (51)$-$(53) with (55) and appropriate communication from sensor 2 finishes the iterations at sensor 3 for the local error covariance matrix, $\mathbf{S}^{(3)}$, with local computation and local communication.

We refer to the equations (51)$-$(53) and (55), along with appropriate communication from neighboring sensors, the generalized distributed matrix Jacobi (GDMJ) algorithm. The GDMJ, presented here with the help of the above example, can be easily extended to any arbitrary sensor l monitoring an $n-$dimensional system. The only step to take care of is going from $L = 1-$band to $L > 1-$band, since (55) holds only for $L = 1$. The appropriate formulae to replace (55), when $L > 1$, are provided in [34].

1) Convergence: Using (55) for the non $L-$banded elements, instead of the iterations of the exact Jacobi algorithm (48), has the following implications. When $\mathbf{Z}$ is $L-$banded, the iterations in (48) converge to the inverse of an $L-$banded matrix. The Jacobi algorithm in (48) does not ensure that at each iteration $\mathbf{S}_t$ is the inverse of an $L-$banded matrix, but only that $\mathbf{S}_t$ is converging to $\mathbf{S}$, which is the inverse of an $L-$banded matrix. Using (55) in the GDMJ algorithm, we require that, at each iteration, the matrix that results from the iteration, $\mathbf{S}_t$ be the

[2]If $\mathbf{S}$ is the inverse of an $L-$banded matrix, then the submatrices that do not lie in the $L-$band of $\mathbf{S}$, can be computed from the submatrices in the $L-$band of $\mathbf{S}$, [34]. So, to compute the inverse, $\mathbf{S} = \mathbf{Z}^{-1}$, of an $L-$banded matrix, $\mathbf{Z}$, we just compute the submatrices that lie in the $L-$band of its inverse, $\mathbf{S}$; the remaining submatrices will be derived from these.

inverse of an $L-$banded matrix, which is not true. Hence, the convergence of GDMJ algorithm is not implied by (48). We show the convergence of GDMJ algorithm, numerically, as follows.

We first discuss the convergence of the Jacobi algorithm, (48). Let $\mathbf{S}$ be the stationary point of the iterations in (48), and let $\tilde{\mathbf{S}}_{t+1}$ denote the iterations of (48). The error process, $\tilde{\mathbf{E}}_{t+1}$, for the Jacobi algorithm (48), can be written as

$$\begin{aligned}\tilde{\mathbf{E}}_{t+1} &= \tilde{\mathbf{S}}_{t+1} - \mathbf{S} \\ &= \mathbf{M}^{-1}\left(\mathbf{M}-\mathbf{Z}\right)\left(\tilde{\mathbf{S}}_t - \mathbf{S}\right) \\ &= \mathbf{P}\tilde{\mathbf{E}}_t, \end{aligned} \tag{56}$$

which dies out if the spectral radius, $||\mathbf{P}||$, of the matrix, $\mathbf{P}$, i.e., the absolute value of its maximum eigenvalue is less than 1.

The convergence of the GDMJ algorithm is shown numerically through figure 6. Let $\widehat{\mathbf{S}}_t$ denote the iterations of the GDMJ algorithm. The error process, $\widehat{\mathbf{E}}_{t+1}$, of the GDMJ algorithm is given by

$$\widehat{\mathbf{E}}_{t+1} = \widehat{\mathbf{S}}_{t+1} - \mathbf{S}. \tag{57}$$

Figure 6 plots the difference, $\triangle\mathbf{E}_{t+1}$, between the error process of the exact Jacobi algorithm, (56), and the GDMJ algorithm, (57),

$$\triangle\mathbf{E}_{t+1} = \widehat{\mathbf{E}}_{t+1} - \tilde{\mathbf{E}}_{t+1} \tag{58}$$

and since $\tilde{\mathbf{S}}_{t+1}$ converges, $\widehat{\mathbf{S}}_{t+1}$ also converges if the difference in the error processes, $\triangle\mathbf{E}_{t+1}$, goes to 0. We plot $\triangle\mathbf{E}_{t+1}$ for different spectral radii, $||\mathbf{P}||$, of $\mathbf{P}$ in figure 6.

VI. Local Information Filters: Initial Conditions and Local Filter Step

The initial conditions and the local filter step of the LIFs are presented in the next subsections.

A. Initial Conditions

The initial condition on the local predictor is

$$\widehat{\mathbf{z}}_{0|-1}^{(l)} = \mathbf{0}. \tag{59}$$

The relationship (19a) on the global initial conditions implies that

$$\mathbf{Z}_{0|-1}^{(l)} \neq \left(\mathbf{S}_0^{(l)}\right)^{-1}. \tag{60}$$

To obtain the initial condition on the prediction information matrix from the local matrices, $\mathbf{S}_0^{(l)}$, in $\mathbf{S}_0$, introduced in (6), we use the $L-$banded inversion theorem [32], provided in appendix I. This step may require a local communication step further elaborated in section VII-A.

$$\mathbf{Z}_{0|-1}^{(l)} \xleftarrow{\quad L-\text{Banded Inversion Theorem}\quad} \mathbf{S}_0^{(l)} \tag{61}$$

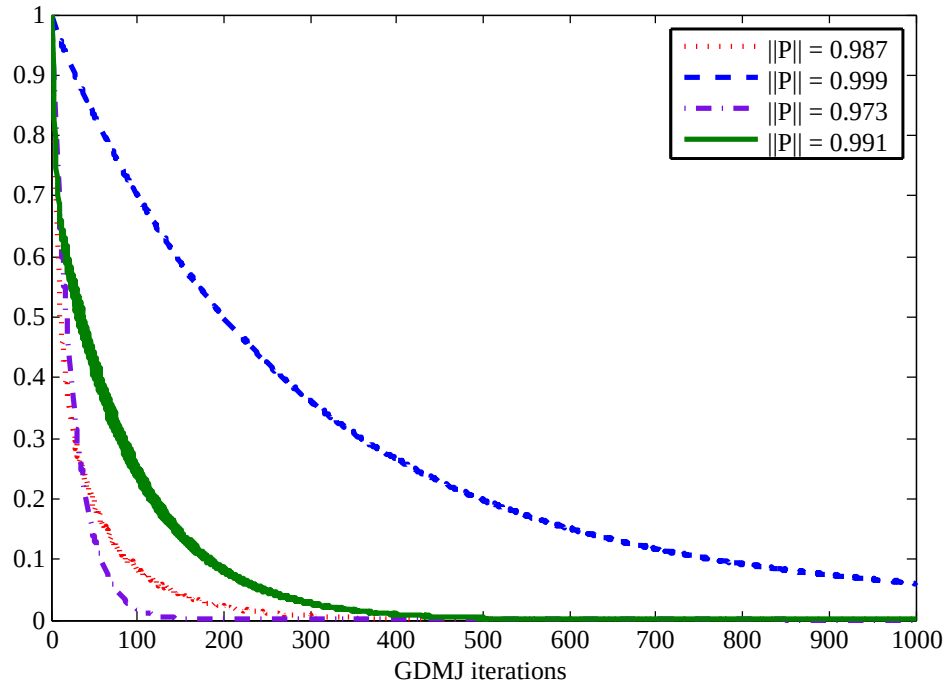


Fig. 6. Trace of $\triangle \mathbf{E}_{t+1}$ in (58) plotted against the number of iterations of the GDMJ algorithm for several different spectral radii of the matrix $\mathbf{P}$.

B. Local Filter Step

In this section, we present the local filter step of the LIFs. The local filter step is given by

$$\mathbf{Z}_{k|k}^{(l)} = \mathbf{Z}_{k|k-1}^{(l)} + \mathbf{I}_{f,k}^{(l)}, \tag{62a}$$

$$\widehat{\mathbf{z}}_{k|k}^{(l)} = \widehat{\mathbf{z}}_{k|k-1}^{(l)} + \mathbf{i}_{f,k}^{(l)}, \tag{62b}$$

where $\mathbf{I}_{f,k}^{(l)}$ and $\mathbf{i}_{f,k}^{(l)}$ denote the fused observation variables. Fusion of the observations is presented in section IV-A. The distribution of the addition operation, '$+$', in (18) is straightforward in (62). Recall that the observation fusion, (42), is carried out using the iterative weighted averaging algorithm. The convergence of this iterative algorithm is guaranteed under certain conditions, e.g., connected sensor communication sub-graph, $\mathcal{G}_j$, see [40], for further details. Hence, with the required assumptions on the sub-graph, $\mathcal{G}_j$, the observation fusion algorithm, (42), converges, and hence,

$$\mathbf{i}_{f,k}^{(l)} \subseteq \mathbf{i}_k \text{ and } \mathbf{I}_{f,k}^{(l)} \subseteq \mathbf{I}_k. \tag{63}$$

This, in turn, implies that the local filter step of the LIFs converges to the global filter step, (18), of the CLBIF. Furthermore, if the local variables involved in (62) are combined over the entire sensor network as shown in figure 4, we will get their corresponding global CLBIF variables at each time index k.

Once the local filter step is completed, the GDMJ algorithm is employed on the local information matrices, $\mathbf{Z}_{k|k}^{(l)}$ obtained from (62a), to convert them into the local error covariance matrices, $\mathbf{S}_{k|k}^{(l)}$.

Finally, to convert the estimates in the information domain, $\widehat{\mathbf{z}}_{k|k}^{(l)}$, to the estimates in the Kalman filter domain, $\widehat{\mathbf{x}}_{k|k}^{(l)}$, we specialize the GDMJ algorithm to matrix-vector product. (The estimates in the Kalman filter domain, $\widehat{\mathbf{x}}_{k|k}^{(l)}$, are required in the estimate fusion step, (43), and as local internal inputs, $\widehat{\mathbf{d}}_{k|k}^{(l)}$ in (36), to the local reduced-order models.) This special case is to solve a linear system of n equations, (11), the variables of which are distributed across the sensor network. Since the GDMJ algorithm solves n coupled linear systems of n equations, solution of (11) can be easily extracted from the GDMJ algorithm.

VII. Local Information Filters: Local Prediction Step

This section presents the distribution of the global prediction step, (19), into the local prediction step at each LIF. This section requires the results of the GDMJ algorithm for the $L-$banded matrices, introduced in section V.

A. Computing the local prediction information matrix, $\mathbf{Z}_{k|k-1}^{(l)}$

Because of the coupled local dynamics of the reduced sensor-based models, each sensor may require that some of the estimated states be communicated as internal inputs, $\widehat{\mathbf{d}}_{k|k}^{(l)}$, to its LIF, as shown in (36). These states are the directed edges into each cut-point set in figure 1(b). Hence, the error associated to a local estimation procedure is also influenced by the error associated to the neighboring estimation procedure, from where the internal inputs are being communicated. This dependence is true for all sensors and is reflected in the local prediction error covariance matrix, $\mathbf{S}_{k|k-1}^{(l)}$, as it is a function of the global estimation error covariance matrix, $\mathbf{S}_{k-1|k-1}$. Equation (64) follows from (19a) after expanding (19a) for each diagonal submatrix, $\mathbf{S}_{k|k-1}^{(l)}$, in $\mathbf{S}_{k|k}$.

$$\mathbf{S}_{k|k-1}^{(l)} = \mathbf{F}_l \mathbf{S}_{k-1|k-1} \mathbf{F}_l^T + \mathbf{G}^{(l)} \mathbf{Q}^{(l)} \mathbf{G}^{(l)T}, \tag{64}$$

where

$$\mathbf{F}_l = \mathbf{T}_l \mathbf{F}. \tag{65}$$

The matrix $\mathbf{T}_l$ is introduced in (32). The assumptions on the model matrix, $\mathbf{F}$, being sparse and localized help us in computing (64), as shown below.

The matrix, $\mathbf{F}_l$, is an $n_l \times n$ matrix, which relates the state vector, $\mathbf{x}_k$, to the local state vector, $\mathbf{x}_k^{(l)}$. Figure 2 shows that the matrix, $\mathbf{F}_l$, is further divided into $\mathbf{F}^{(l)}$ and $\mathbf{D}^{(l)}$. With this sub-division of $\mathbf{F}_l$, the first term on the right hand side of (64), $\mathbf{F}_l \mathbf{S}_{k-1|k-1} \mathbf{F}_l^T$, can be expanded, so that (64) is given by

$$\begin{aligned} \mathbf{S}_{k|k-1}^{(l)} &= \mathbf{F}^{(l)} \mathbf{S}_{k-1|k-1}^{(l)} \mathbf{F}^{(l)T} + \mathbf{F}^{(l)} \mathbf{S}_{k-1|k-1}^{x^{(l)}d^{(l)}} \mathbf{D}^{(l)T} \\ &+ \left(\mathbf{F}^{(l)} \mathbf{S}_{k-1|k-1}^{x^{(l)}d^{(l)}} \mathbf{D}^{(l)T} \right)^T + \mathbf{D}^{(l)} \mathbf{S}_{k-1|k-1}^{d^{(l)}d^{(l)}} \mathbf{D}^{(l)T} + \mathbf{G}^{(l)} \mathbf{Q}^{(l)} \mathbf{G}^{(l)T}, \end{aligned} \tag{66}$$

where

$\mathbf{S}_{k-1|k-1}^{(l)}$ is the local error covariance matrix, which is available from (62a) and the GDMJ algorithm at sensor l;

$\mathbf{S}_{k-1|k-1}^{d^{(l)}d^{(l)}}$ is the local error covariance matrix, which is available from (62a) and the GDMJ algorithm at the sensors having the states, $\mathbf{d}_k^{(l)}$, in their reduced models;

$\mathbf{S}_{k-1|k-1}^{x^{(l)}d^{(l)}}$ is the error cross correlation between the local state vector, $\mathbf{x}_k^{(l)}$, and the local internal input vector, $\mathbf{d}_k^{(l)}$.

The non $L-$banded entries in this matrix can be computed from the equation (55), see [34]. Since the model matrix, $\mathbf{F}$, is sparse, we do not need the entire error covariance matrix, $\mathbf{S}_{k-1|k-1}$, only certain of its submatrices. Since the model matrix, $\mathbf{F}$, is localized, long-distance communication is not required, and the submatrices are available at the neighboring sensors.

We illustrate this for the example system (22)$-$(23). We show a sample computation for the local prediction error covariance matrix at sensor 3, $\mathbf{S}^{(3)}_{k|k-1}$. Let s_{ij} denote the ijth element in the global error covariance matrix, $\mathbf{S}_{k-1|k-1}$, see figure 4.

$$\begin{aligned}
\mathbf{S}^{(3)}_{k|k-1} &= \begin{bmatrix} 0 & 0 & f_{43} & 0 & f_{45} \\ 0 & 0 & 0 & f_{54} & f_{55} \end{bmatrix} \begin{bmatrix} s_{11} & s_{12} & s_{13} & s_{14} & s_{15} \\ s_{12} & s_{22} & s_{23} & s_{24} & s_{25} \\ s_{13} & s_{23} & s_{33} & s_{34} & s_{35} \\ s_{14} & s_{24} & s_{34} & s_{44} & s_{45} \\ s_{15} & s_{25} & s_{35} & s_{45} & s_{55} \end{bmatrix} \begin{bmatrix} 0 & 0 \\ 0 & 0 \\ f_{43} & 0 \\ 0 & f_{54} \\ f_{45} & f_{55} \end{bmatrix} + \mathbf{G}^{(3)}\mathbf{Q}^{(3)}\mathbf{G}^{(3)T}, \\
&= \mathbf{F}^{(3)} \begin{bmatrix} s_{44} & s_{45} \\ s_{45} & s_{55} \end{bmatrix} \mathbf{F}^{(3)T} + \mathbf{F}^{(3)} \begin{bmatrix} s_{34} \\ s_{35} \end{bmatrix} \mathbf{D}^{(3)T} \\
&+ \left(\mathbf{F}^{(3)} \begin{bmatrix} s_{34} \\ s_{35} \end{bmatrix} \mathbf{D}^{(3)T} \right)^T + \mathbf{D}^{(3)} \left[s_{33} \right] \mathbf{D}^{(3)T} + \mathbf{G}^{(3)}\mathbf{Q}^{(3)}\mathbf{G}^{(3)T}. \qquad (67)
\end{aligned}$$

From (31), at sensor 3 the local state vector has the states $x_{4,k}$ and $x_{5,k}$ and the internal input vector consists of the state $x_{3,k}$. Matching the terms in the equation (67), with (66), we note that:

$\mathbf{S}^{(3)}_{k-1|k-1}$ is the local error covariance of the local state vector, $\mathbf{x}^{(3)}_k$, which is available from (62a) and the GDMJ algorithm at sensor 3;

$\mathbf{S}^{d^{(3)}d^{(3)}}_{k-1|k-1}$ is the error covariance of the local internal input vector input, $\mathbf{d}^{(3)}_k = x_{3,k}$, which is available from (62a) and the GDMJ algorithm at sensor 2. Hence, only local communication is required from sensor 2 to sensor 3;

$\mathbf{S}^{x^{(3)}d^{(3)}}_{k-1|k-1}$ is the error cross correlation between the local states $x_{4,k}$ and $x_{5,k}$ and the local internal input $x_{3,k}$. The term s_{34} lies in the $L-$band of $\mathbf{S}_{k-1|k-1}$ and is available at sensor 2, see figure 4. Finally, the term s_{35}, which lies in the non $L-$band of $\mathbf{S}_{k-1|k-1}$, is computed at sensor 3 from (55).

Once we have calculated the local prediction error covariance matrix, $\mathbf{S}^{(l)}_{k|k-1}$, we can compute the local prediction information matrix, $\mathbf{Z}^{(l)}_{k|k-1}$, using the $L-$banded Inversion Theorem [32], in appendix I. The local matrices $\mathbf{Z}^{(l)}_{k|k-1}$ and $\mathbf{S}^{(l)}_{k|k-1}$ are not related by inversion, but they are diagonal blocks of the matrices that are inverses of each other, see figure 4. We compute $\mathbf{Z}^{(l)}_{k|k-1}$ as

$$\mathbf{Z}^{(l)}_{k|k-1} \xleftarrow{\qquad L-\text{Banded Inversion Theorem} \qquad} \mathbf{S}^{(l)}_{k|k-1}. \qquad (68)$$

As evident from (73) in appendix I, to calculate the local prediction information matrix, $\mathbf{Z}^{(l)}_{k|k-1}$, we only need the $\mathbf{S}^{(l)}_{k|k-1}$ from sensor 'l' and some additional neighboring sensors. Hence $\mathbf{Z}^{(l)}_{k|k-1}$ is again computed with only local communication and n_lth order computation.

B. Computing the local predictor, $\widehat{\mathbf{z}}_{k|k-1}^{(l)}$

We illustrate the computation of the local predictor, $\widehat{\mathbf{z}}_{k|k-1}^{(1)}$, for the five dimensional system, (22)$-$(23), with $L=1$. The local predictor, $\widehat{\mathbf{z}}_{k|k-1}^{(1)}$, at sensor 1 follows from the global predictor, (19b), and is given by

$$\widehat{\mathbf{z}}_{k|k-1}^{(1)} = \mathbf{Z}_{k|k-1}^{(1)}\left(\mathbf{F}^{(1)}\widehat{\mathbf{x}}_{k-1|k-1}^{(1)} + \mathbf{D}^{(1)}\widehat{\mathbf{d}}_{k-1|k-1}^{(1)}\right) + \begin{bmatrix} 0 \\ 0 \\ z_{34}\left(f_{43}\widehat{x}_{3,k-1|k-1}^{(2)} + f_{45}\widehat{x}_{5,k-1|k-1}^{(3)}\right) \end{bmatrix}, \tag{69}$$

where z_{34} is the only term arising due to the $L=1-$banded (tridiagonal) assumption on the prediction information matrix, $\mathbf{Z}_{k|k-1}$. Note that $f_{43}\widehat{x}_{3,k-1|k-1}^{(2)} + f_{45}\widehat{x}_{5,k-1|k-1}^{(3)}$ is a result of $\mathbf{f}_4\widehat{\mathbf{x}}_{k-1|k-1}$, where $\mathbf{f}_4$ is the fourth row of the model matrix, $\mathbf{F}$. A model matrix with a localized and sparse structure ensures that $\mathbf{f}_4\widehat{\mathbf{x}}_{k-1|k-1}$ is computed from a small subset of the estimated state vector, $\widehat{\mathbf{x}}_{k-1|k-1}^{(\mathcal{Q})}$, communicated by a subset $\mathcal{Q} \subseteq \mathcal{K}(l)$ of the neighboring sensors, which are modeling these states in their reduced models.

Generalizing, the local predictor in the information domain, $\widehat{\mathbf{z}}_{k|k-1}^{(l)}$, is given by

$$\widehat{\mathbf{z}}_{k|k-1}^{(l)} \;=\; \mathbf{Z}_{k|k-1}^{(l)}\left(\mathbf{F}^{(l)}\widehat{\mathbf{x}}_{k-1|k-1}^{(l)} + \mathbf{D}^{(l)}\widehat{\mathbf{d}}_{k-1|k-1}^{(l)}\right) + f_1\left(\mathbf{Z}_{k|k-1}^{(\mathcal{V})}, \mathbf{F}^{(\mathcal{V})}, \widehat{\mathbf{x}}_{k-1|k-1}^{(\mathcal{Q})}\right) \tag{70}$$

for some $\mathcal{V}, \mathcal{Q} \subseteq \mathcal{K}(l)$, where $f_1(\cdot)$ is a linear function and depends on L.

VIII. Results

A. Summary of the LIFs

We summarize the distributed local Information filters. The initial conditions are given by (59) and (61). Observation fusion is carried out using (42). The fused observation variables, $\mathbf{i}_{f,k}^{(l)}$ and $\mathbf{I}_{f,k}^{(l)}$, are then employed in the local filter step, (62a) and (62b), to obtain the local information matrix and the local estimator, $\widehat{\mathbf{Z}}_{k|k}^{(l)}$ and $\mathbf{z}_{k|k}^{(l)}$, respectively. We then implement the GDMJ algorithm (51)$-$(53) and (55) to compute the local error covariance matrix, $\mathbf{S}_{k|k}^{(l)}$, from the local information matrix, $\mathbf{Z}_{k|k}^{(l)}$. The GDMJ algorithm is again employed to compute the local estimates in the Kalman filter domain, $\widehat{\mathbf{x}}_{k|k}^{(l)}$, from the local estimator, $\widehat{\mathbf{z}}_{k|k}^{(l)}$, as a special case. Estimate fusion, (43), is then carried out on the local estimates in the Kalman filter domain, $\widehat{\mathbf{x}}_{k|k}^{(l)}$. Finally the prediction step is completed by computing the local prediction error covariance matrix, $\widehat{\mathbf{S}}_{k|k-1}^{(l)}$, from (66), the local prediction information matrix, $\widehat{\mathbf{Z}}_{k|k-1}^{(l)}$, using the $L-$banded inversion theorem in appendix I, and, the local predictor, $\widehat{\mathbf{z}}_{k|k-1}^{(l)}$, from (70).

B. Simulations

We simulate the $5-$dimensional system of (22)$-$(23). Figure 7 plots the sum of the squared errors, SSE, averaged over $1,000$ Monte-Carlo trials, for several Information filters

$$\text{SSE} = \text{tr}\left[E\left(\mathbf{x}_k - \widehat{\mathbf{x}}_{k|k}\right)\left(\mathbf{x}_k - \widehat{\mathbf{x}}_{k|k}\right)^T\right] \tag{71}$$

versus the number of iterations, $k = 1, \ldots, 500$. The results for the CIF are shown in red/dashed, for the CLBIF with $L=1-$banded approximation are shown in green/solid, for the CLBIF with $L=2-$banded approximation are shown in grey/dash–dot, and for the LIFs with $L=1-$banded approximation are shown in black/dotted (this

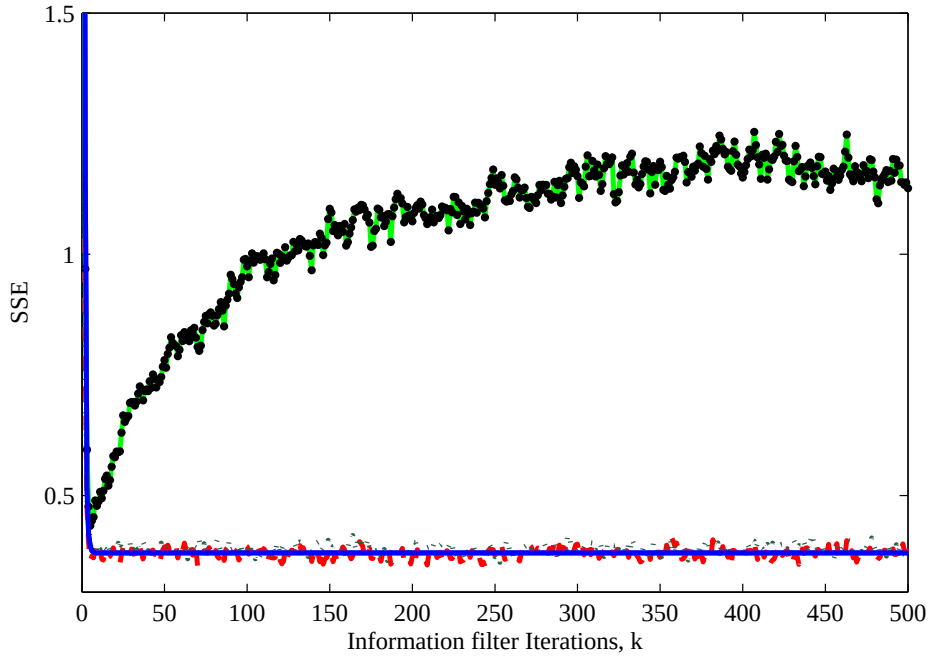


Fig. 7. Simulation Results: The sum of squared errors is plotted against the number of iterations of the several Information filters. The LIFs with $L = 1$ (black/dotted) tracks the CLBIF with $L = 1$ (green/solid), both of them having exact overlap. The CLBIF with $L = 2-$banded approximation (grey/dash-dot) is virtually indistinguishable from the CIF (red/dashed).

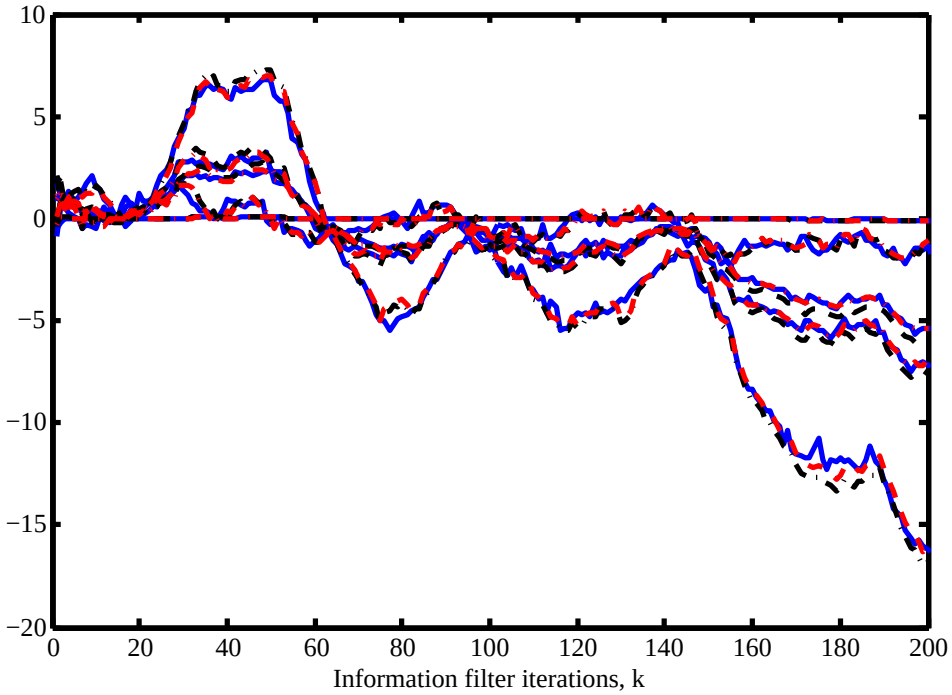


Fig. 8. Simulation Results: The original states are shown in blue/solid, with their estimates using the CIF shown in red/dashed and their estimates with the LIFs with $L = 1$ shown in black/dash-dot.

curve is exactly on top of the CLBIF with $L = 1$ curve in green/solid.) We further plot the trace of the solution of the Riccati equation, inverse of (18a), in blue/solid. These results show that, for the $L = 1-$banded approximation, there is a noticeable steady state error (top curve in figure 7, notice the zoomed up vertical axis.) This steady state error is considerably reduced when we increase the band from $L = 1$ to $L = 2$.

Several remarks are in order. First of all, the LIFs (black/dotted) with $L = 1$ track the results of the CLBIF with $L = 1$ (green/solid), see figure 7. This suggests that the iterative algorithms, GDMJ algorithm and weighted averaging algorithms, have converged and GDMJ does converge to the centralized Jacobi algorithm as shown in figure 6. Secondly, if we compare the performance of the CLBIF ($L = 1$ in green/solid and $L = 2$ in grey/dash-dot) with the performance of the CIF in red/dashed, we verify that as $L \uparrow$, the performance is virtually indistinguishable from that of the CIF, as pointed out in [34]; this is in agreement with the fact that the approximation is optimal in Kullback-Leibler sense, as shown in [32]. Also notice that if we increase the number of Monte Carlo simulations the variations in the CIF (red/dashed) reduce, and the filters eventually follow the theoretical value of tr$[\mathbf{S}_{k|k}]$, in blue/solid.

Figure 8 shows the estimates for a single trial over 200 iterations, with the original states in blue/solid, its estimates with the CIF in red/dashed and its estimates with the LIFs in black/dash-dot. The agreement between the two estimates suggests the robustness of the $L-$banded approximation. Hence, we can see a near optimal performance of the $L = 1-$banded LIFs. We further note that the discrepancy is removed almost completely when we approximate the information matrices to be penta-diagonal, $L = 2$, instead of being tridiagonal, $L = 1$.

We next discuss the computational advantage of the LIFs over some of the existing methods.

C. Complexity

We regard the multiplication of two $n \times n$ matrices as an $O(n^3)$ operation, inversion of an $n \times n$ matrix also as an $O(n^3)$ operation and multiplication of an $n \times n$ matrix and an $n \times 1$ vector as an $O(n^2)$ operation. For all of the following, we assume N sensors monitoring the global system, (1).

1) Centralized Information Filter, CIF: This is the case where each sensor sends its local observation vector to a centralized location or a fusion center, where the global observation vector is then put together. The fusion center then implements the CIF, with an $O(n^3)$ computation complexity for each time k, so we have the complexity as $O(n^3 k)$, with inordinate communication requirements (back and forth communication between the sensors and the central location).

2) Information Filters with Replicated Models at each sensor and Observation Fusion: In this scheme, the local observation vectors are not communicated to the central location, but are fused over an all-to-all communication network, [4], or an arbitrary network, [6]. Recall that the arbitrary communication network requires an iterative algorithm.

Computational complexity at each sensor is $O(n^3 k)$ in [4]. In [6], let the complexity of the iterative algorithm be $O(f(n))$ and let t be the number of iterations required by the iterative algorithm to converge. Each sensor implements a CIF after fusing the observations. For each time index, k, each sensor requires $O(n^3)$ operations plus

the operations required for the iterative algorithms, which are $O(f(n)t)$; so the total computation complexity is $O((n^3 + f(n)t)k)$ at each sensor. Communication requirements are global in [4], and local in [6].

3) Distributed Kalman Filters: Local Information Filters, LIFs: The distributed Kalman filter presented in this paper has three iterative algorithms. In all the other steps the computation complexity is dominated by $O(n_l^3)$ operations, where $n_l \ll n$. Let t_o be the iterations required by the weighted averaging algorithm, where at each step of the iterative algorithm the computations are dominated by an $O(n_l^2)$ operations. Let t_{J_1} be the iterations required by the GDMJ algorithm for vectors, where at each step of the iterative algorithm the computations are dominated by an $O(n_l^2)$ operations. Let t_{J_2} be the iterations required by the GDMJ algorithm, where at each step of the iterative algorithm the computations are dominated by an $O(n_l^3)$ operations. The total computation complexity will be $O((n_l^3 + n_l^2 t_o + n_l^2 t_{J_1} + n_l^3 t_{J_2})k)$. Let $t = \max(t_o, t_{J_1}, t_{J_2})$, then the computation complexity is bounded by $O(n_l^3 tk)$ at each sensor for the LIFs, which is much smaller than the computational cost of the solutions in VIII-C.1 and VIII-C.2. In this example, typical values for t_{J_1}, t_{J_2} and t_o (we used local degree weights for observation fusion in [40], with a sensor communication network $1 \leftrightarrow 2 \leftrightarrow 3$, other techniques e.g., semi-definite programming [40], can be used to decrease t_o significantly) are $6, 7$ and 9 respectively. The convergence rate of the iterative algorithms can be increased by optimizing the sensor communication graph, $\mathcal{G}$, [44]. The communication requirement in the LIFs is always local.

IX. Conclusions

In conclusion, we presented a *distributed* implementation of the Kalman filter for sparse large-scale systems monitored by sensor networks. In our solution, the communication, computing, and storage is local and distributed across the sensor network, no single sensor processes $n-$dimensional vectors or matrices, where n, usually a large number, is the dimension of the state vector representing the random field. We achieve this by solving three linked problems: (1) *Distributing* the global state variable model representing the random field into sensor based reduced-order models. These reduced-order models are obtained using a graph-theoretic model distribution technique that decomposes the random field model into coupled low-order models; (2) *Fusing*, through distributed averaging, multiple sensor observations and multiple estimates of the state variables that are common across reduced-order models; and (3) *Inverting* full $n-$dimensional matrices, with local communication only, by deriving an iterative generalized distributed matrix Jacobi (GDMJ) algorithm. The GDMJ algorithm only requires matrices and vectors of the order n_l of the reduced state vectors. The GDMJ algorithm preserves the coupling among the local Kalman filters. Our solution contrasts with existing Kalman filter solutions for sensor networks that either replicate an $n-$dimensional Kalman filter at each sensor or reduce the model dimension at the expense of decoupling the field dynamics into lower-dimensional models. The former are infeasible for large-scale systems and the latter are highly sub-optimal.

Simulations show that the distributed implementation with local Kalman filters implemented at each sensor converges to the global Kalman filter as the number of bands in the information matrices is increased.

APPENDIX I
$L-$BANDED INVERSION THEOREM

Let $\mathbf{Z} = \mathbf{S}^{-1}$ be $L-$banded. We apply the algorithm, given in [32], to obtain $\mathbf{Z}$ from the submatrices in the $L-$band of $\mathbf{S}$, as . We use the following notation, in (72), to partition matrix addition and subtraction in terms of its constituent submatrices. Also $\mathbf{S}_j^i$ represents the principal submatrix of $\mathbf{S}$ spanning rows i through j, and columns i through j.

$$\begin{bmatrix} a_1 & a_2 & 0 \\ a_3 & x+y+z & a_4 \\ 0 & a_5 & a_6 \end{bmatrix} = \left[\begin{bmatrix} a_1 & a_2 \\ a_3 & x \end{bmatrix} + \boxed{y} + \begin{bmatrix} z & a_4 \\ a_5 & a_5 \end{bmatrix} \right] \tag{72}$$

The inverse of $\mathbf{S}$, when $\mathbf{Z} = \mathbf{S}^{-1}$ is $L-$banded, is given by (73), taken from [32], in terms of the $L-$banded submatrices of $\mathbf{S}$. Note that, to compute a principal submatrix in $\mathbf{Z}$, we do not need the entire $\mathbf{S}$, or even all the $L-$banded submatrices in $\mathbf{S}$. Instead, we only require three neighboring submatrices in the $L-$band of $\mathbf{S}$. For proofs and further details, the interested reader can refer to [32].

$$\mathbf{Z} = \begin{bmatrix} \boxed{\mathbf{S}_{L+1}^1}^{-1} - \boxed{\mathbf{S}_{L+1}^2}^{-1} + \boxed{\mathbf{S}_{L+2}^2}^{-1} - \cdots & & \mathbf{0} \\ & \ddots & \\ \mathbf{0} & & \cdots + \boxed{\mathbf{S}_{N-1}^{N-L-1}}^{-1} - \boxed{\mathbf{S}_{N-1}^{N-L}}^{-1} + \boxed{\mathbf{S}_N^{N-L}}^{-1} \end{bmatrix} \tag{73}$$

REFERENCES

[1] R. Kalman, "A new approach to linear filtering and prediction problems," *Transactions of the ASME - Journal of Basic Engineering*, vol. 82, no. 2, pp. 35–45, 1960.

[2] R. Kalman and R.Bucy, "New results in linear filtering and prediction theory," *ASME Journal of Basic Engineering*, vol. 83, pp. 95–108, 1961.

[3] H. Hashemipour, S. Roy, and A. Laub, "Decentralized structures for parallel Kalman filtering," *IEEE Transactions on Automatic Control*, vol. 33, no. 1, pp. 88–94, Jan. 1988.

[4] B. Rao and H. Durrant-Whyte, "Fully decentralized algorithm for multisensor Kalman filtering," *IEE Proceedings-Control Theory and Applications*, vol. 138, pp. 413–420, Sep. 1991.

[5] J. Speyer, "Computation and transmission requirements for a decentralized linear-quadratic-Gaussian control problem," *IEEE Transactions on Automatic Control*, vol. 24, no. 2, pp. 266–269, Feb. 1979.

[6] R. Olfati-Saber, "Distributed Kalman filters with embedded consensus filters," in *44th IEEE Conference on Decision and Control*, Seville, Spain, Dec. 2005, pp. 8179 – 8184.

[7] R. Olfati-Saber and J. Shamma, "Consensus filters for sensor networks and distributed sensor fusion," in *44th IEEE Conference on Decision and Control*, Seville, Spain, Dec. 2005, pp. 6698 – 6703.

[8] M. Alanyali and V. Saligrama, "Distributed tracking in multi-hop networks with communication delays and packet losses," in *13th IEEE Workshop on Statistical Signal Processing*, Bordeaux, France, Jul. 2005, pp. 1190–1195.

[9] V. Saligrama and D. Castanon, "Reliable distributed estimation with intermittent communications," in *45th IEEE Conference on Decision and Control*, San Diego, CA, Dec. 2006, pp. 6763–6768.

[10] T. Chung, V. Gupta, J. Burdick, and R. Murray, "On a decentralized active sensing strategy using mobile sensor platforms in a network," in *43rd IEEE Conference on Decision and Control*, Paradise Island, Bahamas, Dec. 2004, vol. 2, pp. 1914–1919.

[11] T. Berg and H. Durrant-Whyte, "Model distribution in decentralized multi-sensor data fusion," Tech. Rep., University of Oxford, 1990.

[12] A. Mutambara, *Decentralized estimation and control for multisensor systems*, CRC Press, Boca Raton, FL, 1998.

[13] B. Anderson and J. Moore, *Optimal filtering*, Prentice Hall, Englewood Cliffs, NJ, 1979.

[14] R. Nash Jr. and S. Jordan, "Statistical geodesy: An engineering perspective," *Proceedings of the IEEE*, vol. 66, no. 5, pp. 532–550, May 1978.

[15] W. Ledsham and D. Staelin, "An extended Kalman-Bucy filter for atmospheric temperature profile retrieval with a passive microwave sounder," *Journal of Applied Meteorology*, vol. 17, pp. 1023–1033, Jul. 1978.

[16] R. Brammer, "Estimation of the ocean geoid near the Blake Escarpment using Geos-3 satellite altimetry data," *Journal of Geophysical Research*, vol. 84, no. B8, pp. 3843–3852, Jul. 1979.

[17] J. Galantowicz, D. Entekhabi, and E. Njoku, "Tests of sequential data assimilation for retrieving profile soil moisture and temperature from observed L-band radiobrightness," *IEEE Transactions on Geoscience and Remote Sensing*, vol. 37, no. 4, pp. 1860–1870, Jul. 1999.

[18] M. Buehner and P. Malanotte-Rizzoli, "Reduced-rank Kalman filters applied to an idealized model of the wind-driven ocean circulation," *Journal of Geophysical Research*, vol. 108, no. C6, 2003.

[19] M. Buehner, P. Malanotte-Rizzoli, A. Busalacchi, and T. Inui, "Estimation of the tropical Atlantic circulation from altimetry data using a reduced-rank stationary Kalman filter," *Interhemispheric water exchanges in the Atlantic Ocean, Elsevier Oceanographic Series*, vol. 68, no. 9, pp. 49–92, 2003.

[20] J. Apt, L. Lave, S. Talukdar, G. Morgan, and M. Ilic, "Electrical blackouts: A systemic problem," *Issues in Science and Technology*, vol. 40, no. 4, pp. 55–61, 2004.

[21] M. Ilic, E. Allen, J. Chapman, C. King, J. Lang, and E. Litvinov, "Preventing future blackouts by means of enhanced electric power systems control: From complexity to order," *Proceedings of the IEEE*, vol. 93, no. 11, pp. 1920–1941, Nov. 2005.

[22] A. Asif and J. Moura, "Fast recursive reconstruction of large time-varying multidimensional fields," in *IEEE International Conference on Acoustics, Speech, and Signal Processing*, Munich, Germany, Apr. 1997, vol. 4, pp. 3037–3040.

[23] J. Brailean and A. Katsaggelos, "Simultaneous recursive displacement estimation and restoration of noisy-blurred image sequences," *IEEE Transactions on Image Processing*, vol. 4, no. 9, pp. 12361251, Sep. 1995.

[24] F. Khellah, P. Fieguth, M. Murray, and M. Allen, "Statistical processing of large image sequences," *IEEE Transactions on Image Processing*, vol. 14, no. 1, pp. 80–93, Jan. 2005.

[25] A. Bergen and D. Hill, "A structure preserving model for power system stability analysis," *IEEE Transactions on Power Apparatus and Systems*, vol. PAS-100, no. 1, pp. 25–35, Jan. 1981.

[26] D. Hill, "On the equilibria of power systems with nonlinear loads," *IEEE Transactions on Circuits and Systems*, vol. 36, no. 11, pp. 1458–1463, Nov. 1989.

[27] A. George and J. Liu, *Computer solution of large sparse positive definite systems*, Prentice Hall, Englewood Cliffs, NJ, 1981.

[28] J. Liu., "Modification of the minimum-degree algorithm by multiple elimination," *ACM Transactions on Mathematical Software*, vol. 11, no. 2, pp. 141–153, 1985.

[29] P. Amestoy, T. Davis, and I. Duff, "An approximate minimum degree ordering algorithm," *SIAM Journal on Matrix Analysis and Applications*, vol. 17, no. 4, pp. 886–905, 1996.

[30] J. Durbin, "Efficient estimation of parameters in moving-average models," *Biometrika*, vol. 45, no. 3/4, pp. 306–316, Dec. 1959.

[31] N. Balram and J. Moura, "Noncausal Gauss Markov random fields: Parameter structure and estimation," *IEEE Transactions on Information Theory*, vol. 39, no. 4, pp. 1333–1355, Jul. 1993.

[32] A. Kavcic and J. Moura, "Matrices with banded inverses: Inversion algorithms and factorization of Gauss-Markov processes," *IEEE Transactions on Information Theory*, vol. 46, no. 4, pp. 1495–1509, Jul. 2000.

[33] G. Strang, *Introduction to applied mathematics*, Wellesley Cambridge Press, Wellesley, MA, 1986.

[34] A. Asif and J. Moura, "Inversion of block matrices with L-block banded inverse," *IEEE Transactions on Signal Processing*, vol. 53, no. 2, pp. 630–642, Feb. 2005.

[35] V. Delouille, R. Neelamani, and R. Baraniuk, "Robust distributed estimation using the embedded subgraphs algorithm," *IEEE Transactions on Signal Processing*, vol. 54, no. 8, pp. 2998–3010, Aug. 2006.

[36] Usman A. Khan and José M. F. Moura, "Model distribution for distributed Kalman filters: A graph theoretic approach," Submitted, 5 pages, Jun. 2007.

[37] Dragoslav Šiljak, *Decentralized control of complex systems*, Academic Press Inc., Boston, MA, 1991.

[38] H. Zhang, J. Moura, and B. Krogh, "Estimation in sensor networks: A graph approach," in *4th International Symposium on Information Processing in Sensor Networks*, Los Angeles, CA, Apr. 2005, pp. 203–209.

[39] Usman A. Khan and José M. F. Moura, "Distributed Kalman filters in sensor networks: Bipartite fusion graphs," in *15th IEEE Workshop on Statistical Signal Processing*, Madison, WI, Aug. 26-29 2007.

[40] L. Xiao and S. Boyd, "Fast linear iterations for distributed averaging," *Systems and Controls Letters*, vol. 53, no. 1, pp. 65–78, Apr. 2004.

[41] F. Schweppe, *Uncertain dynamic systems*, Prentice Hall, Englewood Cliffs, N.J., 1973.

[42] J. Wall Jr., A. Willsky, and N. Sandell Jr., "On the fixed-interval smoothing problem," *Stochastics*, vol. 5, no. 1-2, pp. 1–42, 1981.

[43] A. Willsky, M. Bello, D. Castanon, B. Levy, and G. Verghese, "Combining and updating of local estimates and regional maps along sets of one dimensional tracks," *IEEE Transactions on Automatic Control*, vol. 24, no. 4, pp. 799–813, 1982.

[44] S. Aldosari and J. Moura, "Distributed detection in sensor networks: Connectivity graph and small world networks," in *39th Asilomar Conference on Signals, Systems, and Computers*, Pacific Grove, CA, Nov. 2005, pp. 230–234.