Hindawi Publishing Corporation
Discrete Dynamics in Nature and Society
Volume 2015, Article ID 576341, 9 pages
http://dx.doi.org/10.1155/2015/576341

Research Article

Consensus Information Filtering for Large-Scale Systems with Application to Heat Conduction Process

Liguo Zhang[1,2] and Ying Lyu[3]

[1] *School of Electronic and Control Engineering, Beijing University of Technology, Beijing 100124, China*
[2] *Key Laboratory of Computational Intelligence and Intelligent Systems, Beijing 100124, China*
[3] *School of Science, Communication University of China, Beijing 100024, China*

Correspondence should be addressed to Liguo Zhang; zhangliguo@bjut.edu.cn

Received 20 October 2015; Revised 9 November 2015; Accepted 17 November 2015

Academic Editor: Deqing Huang

For large-scale distributed systems, the evolution of system dynamics is dominated by current states and boundary conditions simultaneously. This work describes a distributed consensus filtering for a class of large-scale distributed systems with unknown boundary conditions, which are monitored by a set of sensors. Because of the difference of spatial positions among the sensor network, only the single state variables or both the states and outside input jointly could be estimates with Kalman information filtering, respectively. On diffusion processing, we fuse the common state estimations of the local information filters using consensus averaging algorithms and algebraic graph theory. Stability and performance analysis is provided for this distributed filtering algorithm. Finally, we consider an application of distributed estimation to a heat conduction process. The performance of the proposed distributed algorithm is compared to the centralized Kalman filtering.

1. Introduction

Large-scale systems have been studied extensively in the cutting-edge technology-based social sectors, ranging from ocean, atmospheric, and hydrological sciences [1, 2] and oil reservoir simulations [3] to intelligent transportation systems [4–6]. The large-scale systems are usually formulated with a set of partial differential equations to express their distributed, multiparameter coupled dynamical behaviors. In recent years, more attention has been given to cyber-physical systems, in which the physical plants usually are described with large-scale systems [1, 4, 7].

State estimate of large-scale systems is a challenging task, since the evolution of the large-scale systems usually codominated by the current system states and the outside boundary conditions. The troublesome problem is the boundary conditions are often unknown or inaccessible in advance in some applications. Two kinds of approaches have been developed in recent years to solve this problem. One method assumes that the boundary conditions are random walking variables and then estimates these variables as extended system states [8]. Another method treats these boundary conditions as outside system input and estimates system states and unknown input simultaneously based on the iterating two-step or multistep Kalman filtering [9, 10].

In this work, we present a distributed simultaneous state and input estimate method for a class of large-scale systems with unknown boundary conditions which are monitored by a sensor network. Based on the spatial positions of sensor nodes, the large-scale system is firstly decomposed into low-dimensional subsystems. If the sensor nodes are located at the boundary grids, information filtering for simultaneous state and input estimate is developed. Then consensus strategy fuses state estimations which are common among the local information filters using the consensus averaging algorithms [11].

The aims of this paper are (1) to decompose the large-scale distributed parameter system into spatial subsystems with reduced observation models; locations of the sensor group throughout the spatial domain significantly affect the outcome and quality of state estimation; we model sensors placed in different locations with different output

operators, (2) to develop the information filtering algorithm for simultaneous state and boundary condition estimation for local sensor groups and design consensus estimator for every sensor so that the overall state can be estimated via local exchange of messages among neighboring nodes, and (3) to present a stability analysis distributed Kalman consensus filtering algorithm.

The subsequent sections are organized as follows. Some background on centralized and decentralized observation models is provided in Section 2. State-space partition of large-scale system is formulated in Section 3. In Section 4, we present the information filtering for simultaneous state and input estimate. Distributed Kalman consensus filters for large-scale systems with unknown boundary conditions are introduced in Section 5.1. Stability analysis is provided in Section 5.2. Simulation results and performance comparisons for a 1D heat conduction process are presented in Section 6. Conclusions and future work are discussed in Section 7.

2. Problem Statement

2.1. Spatially Distributed Processes. Consider the spatially distributed physical process represented by the following convection-diffusion equation:

$$\frac{\partial u}{\partial t}(t,\xi) - e\frac{\partial^2 u}{\partial \xi^2}(t,\xi) + c(t,\xi)\frac{\partial u}{\partial \xi}(t,\xi) = f(t,\xi), \quad (1)$$

where $t \in [0,\infty)$, $0 < \xi < L$, and L is the length of one spatial dimension. $f(t,\xi)$ is the given outside environment, e is the small diffusion coefficient, and $c(t,\xi)$ is the convection coefficient.

Assume that the initial conditions of (1) are given by

$$u(0,\xi) = u_0(\xi), \quad 0 < \xi < L, \quad (2)$$

where $u_0(\xi)$ is a given function. And the boundary conditions are imposed on both edges as

$$\begin{aligned} u(t,0) &= G_0(t), \\ u(t,L) &= G_L(t), \\ t &\in [0,\infty), \end{aligned} \quad (3)$$

where $G_0(t)$, $G_L(t)$ are outside input functions.

When the initial and boundary conditions (2)-(3) are compatible, (1) is well posed in the sense that it has a unique solution $u(t,\xi)$.

2.2. Spatiotemporal Discretization. The above PDEs could be solved numerically on a spatial uniformed mesh grid with the finite-difference discretization method. The central approximations of the first- and the second-order derivative are, respectively,

$$\frac{\partial u}{\partial \xi}(t,jh) \approx \frac{u_{j+1} - u_{j-1}}{2}, \quad (4)$$

$$\frac{\partial^2 u}{\partial \xi^2}(t,jh) \approx \frac{u_{j+1} - 2u_j + u_{j-1}}{h^2}, \quad (5)$$

where u_j is the value of the random field $u(t,jh)$ at the jth location of the spatial grid and h is the length of the spatial grid.

Temporal discretization approximates the time derivative of PDEs (1) by using the forward Euler method. For those boundary grids whose outer grids are missing, the temporally discretized boundary conditions $G_0(t)$, $G_L(t)$ are represented as discrete variable $d(k)$, where $k = 1, 2, \ldots$, is the discrete-time instant.

In general, the spatiotemporal discretization of the PDEs (1) is formulated by the following discrete-time state-space model:

$$x(k+1) = Ax(k) + Bd(k) + w(k), \quad k \geq 0, \quad (6)$$

where the state vector $x = (x_1, \ldots, x_n)^T \in R^n$ is the collection of the sorted variables $\{u_j\}$, by, for example, using the lexicographic ordering. Unknown input $d = (d_1, \ldots, d_m)^T \in R^m$ is the discretized boundary condition, and $w(k) \in R^n$ is the model approximation error assumed to be mutually uncorrelated, zero-mean, white random signals with known covariance matric $Q = E[w(k)w(k)^T]$. $A \in R^{n\times n}$, $B \in R^{n\times m}$ are system state matrix and boundary input matrix, respectively.

Remark 1. For the system matrix B of the discrete-time model (6), nonzero element $b_{ij} \in B$ means that the state x_i representing the value of the spatial grid is affected directly by the outside boundary input d_j; otherwise, zero element of B means that x_i represents the inside grid.

2.3. Distributed Observation Model. We assume that system (6) is monitored by a network of N sensors, in which the boundary information is not accessible directly. Then the local observations at sensor i at time k are

$$z^{(i)}(k) = H^{(i)}x(k) + v^{(i)}(k), \quad (7)$$

where $H^{(i)} \in R^{p_i\times n}$ is the local observation matrix, p_i is the number of simultaneous observations made by sensor i at time k, and $v^{(i)}(k) \in R^{p_i}$ is the local observation noise assumed to be mutually uncorrelated, zero-mean, white random signals with known covariance matric $R^{(i)} = E[v^{(i)}(k)v^{(i)}(k)^T]$.

Remark 2. The local measurement $z^{(i)}(k)$ may be temperature or some other physical attribute at local sensor i. The observation matrix $H^{(i)}$ is particular to the spatial position (corresponding to the nonzero element in $H^{(i)}$) of the local sensor i, in the uniformed mesh grid.

We collect all observations in the network to get the global observation model. Let p be the total number of observations at all sensors. Let the global observation vector, $z(k) \in R^p$,

the global observation matrix, $H \in R^{p\times n}$, and the global observation noise vector, $v(k) \in R^p$, be

$$z(k) = \left[z^{(1)}(k) \ \cdots \ z^{(N)}(k)\right]^T,$$
$$H = \left[H^{(1)} \ \cdots \ H^{(N)}\right]^T, \tag{8}$$
$$v(k) = \left[v^{(1)}(k) \ \cdots \ v^{(N)}(k)\right]^T.$$

Then the global observation model is given by

$$z(k) = Hx(k) + v(k). \tag{9}$$

Since the observation noises at the different sensors are independent, we can combine the local observation noise covariance matrices at each sensor into the global observation noise covariance matrix, R, as

$$R = \text{blockdiag}\left[R^{(1)}, \ldots, R^{(N)}\right]. \tag{10}$$

For large-scale systems with unknown boundary conditions, centralized and distributed filtering for simultaneous input and state estimation are discussed, respectively. No prior information about the unknown boundary input, $d(k)$, in the system model (7) is available or accessible in advance.

3. Partitioned Large-Scale Systems

In this section, we partition the large-scale system into subsystems based on the different spatial positions of the local sensors. Each sensor has the capability to generate its own estimate using locally available measurements.

Case 1. The local sensor i is placed at the boundary grids of the spatiotemporal discretization model (6), $i \in N_b$. $N_b \subseteq N$ is the corresponding index set.

In this case, $\text{Rank}(H^{(i)}B) = m_i \le \text{Rank}(B)$, $m_i \ne 0$. We select submatrices $B^{(i)}$ containing the elements taken from the full-order matrix B with indices belonging to $n\times I_i^d$, where the index set $I_i^d = \{j_1^i, \ldots, j_{m_i}^i\}$, $j_1^i, \ldots, j_{m_i}^i \in \{1, \ldots, m\}$, $j_1^i < \cdots < j_{m_i}^i$, and $m_i \le m$, such that

$$\text{Rank}\left(H^{(i)}B^{(i)}\right) = \text{Rank}\left(B^{(i)}\right) = \text{Rank}\left(d_k^{(i)}\right) = m_i, \tag{11}$$

where $d_k^{(i)}$ is an m_i-dimensional vector composed of the components of d_k selected by I_i^d.

Then, the local subsystem models available to the ith sensor can be defined as

$$x(k+1) = Ax(k) + B^{(i)}d^{(i)}(k) + w(k), \tag{12}$$

$$z^{(i)}(k) = H^{(i)}x(k) + v^{(i)}(k), \tag{13}$$

$i \in N_b$. In order to estimate all boundary inputs d_k, we further assume that $\bigcup I_i^d = \{1, \ldots, m\}$.

Case 2. The local sensor i is placed at the inside grids of the spatiotemporal discretization model (6); $i \in N - N_b$.

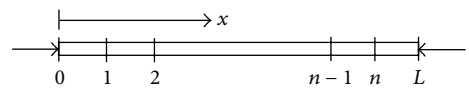

Figure 1: One-dimensional heat conduction model with unknown boundary conditions.

In this case, the elements of matrix B corresponding to the inside grids are zeroes; that is, $H^{(i)}B = 0$. Then, we define the local submatrices as $B^{(i)} = 0$ and extract the local subsystem models as

$$x(k+1) = Ax(k) + w(k),$$
$$z^{(i)}(k) = H^{(i)}x(k) + v^{(i)}(k), \quad i \in N - N_b. \tag{14}$$

Example 3. Consider the one-dimensional distributed parameter heat conduction model on a rod. Spatial discretization results in the grid shown in Figure 1. A number of sensors are located on the specified grid points and each sensor measures the specified temperature. Unknown boundary conditions on the end of the rod are $d(k) = \left[d^L(k) \ \ d^R(k)\right]^T$, and the boundary system matrix is

$$B = \begin{bmatrix} 1 & 0 & \cdots & 0 \\ 0 & 0 & \cdots & 1 \end{bmatrix}^T. \tag{15}$$

When the sensor is located at the grid point 1, the local observation matrix should be $H^{(1)} = \left[1 \ \ 0 \ \ \cdots \ \ 0\right]$. Then, $B^{(1)}$ is obtained as a submatrix including the first column of the full-order boundary matrix B; that is,

$$B^{(1)} = \left[1 \ \ 0 \ \ \cdots \ \ 0\right]^T. \tag{16}$$

We define the input as $d^{(1)}(k) = d^L(k)$.

When the sensor location is at the grid point 2, or the other inside grids, the local observation matrix should be $H^{(2)} = [0 \ \ 1 \ \ \cdots \ \ 0]$. In this case, $H^{(2)}B = 0$, and we define $B^{(2)} = 0$. The subsystem models are not including outside input.

4. Information Filtering for Simultaneous Input and State Estimation

In this subsection, we extend the classical Kalman information filtering for simultaneous input and state estimation. Recursive equations of information filtering for subsystems (14) (omit index i) can be written as

$$\hat{x}(k) = \overline{x}(k) + P(k)\left(y(k) - U\overline{x}(k)\right),$$
$$U = H^T R^{-1} H,$$
$$P(k) = \left(\overline{P}(k)^{-1} + U\right)^{-1}, \tag{17}$$
$$\overline{P}(k) = AP(k-1)A^T,$$
$$\overline{x}(k) = A\hat{x}(k-1),$$

with weighted sensor data $y(k) = H^T R^{-1} z(k)$ and information matrix U. The error covariance matrices are defined as $P(k) = E[\eta(k)\eta(k)^T]$, $\overline{P}(k) = E[\overline{\eta}(k)\overline{\eta}(k)^T]$, where $\eta(k) = x(k) - \hat{x}(k)$, $\overline{\eta}(k) = x(k) - \overline{x}(k)$, with $P(0) = P_0$.

For subsystems (12) with unknown input, two-step optimal filtering for simultaneous input and state estimation has been developed in [9, 10]. The main purpose is to estimate input from the innovation by least-squares estimation. Recursive state and input filters are as follows:

$$\tilde{x}(k) = A\hat{x}(k-1), \tag{18}$$

$$\hat{d}(k-1) = M(k)(z(k) - H\tilde{x}(k)), \tag{19}$$

$$\overline{x}(k) = \tilde{x}(k) + B\hat{d}(k-1), \tag{20}$$

$$\hat{x}(k) = \overline{x}(k) + K(k)(z(k) - H\overline{x}(k)), \tag{21}$$

where $\hat{d}$ represents input estimate and $\overline{x}(k)$ and $\tilde{x}(k)$ represent the state prediction with and without the input information, respectively.

The optimal gain matrices are obtained as

$$M(k) = \left(D^T \tilde{R}(k)^{-1} D\right)^{-1} D^T \tilde{R}(k)^{-1}, \tag{22}$$

$$K(k) = \left(\overline{P}(k) H^T + S(k)\right) \check{R}(k)^{-1}, \tag{23}$$

where $D = HB$, $S(k) = -BM(k)R$, $\tilde{R}(k) = H(AP(k)A^T + Q)H^T + R$, and $\check{R}(k) = H\overline{P}(k)H^T + R + HS(k) + S(k)^T H^T$.

The error covariance matrices are updated according to

$$\begin{aligned} P(k) &= \overline{P}(k) - K(k)\left(\overline{P}(k) H^T + S(k)\right)^T, \\ \overline{P}(k) &= \widetilde{A}(k) P(k-1) \widetilde{A}(k)^T + \widetilde{Q}(k), \end{aligned} \tag{24}$$

where $\widetilde{A}(k) = (I - BM(k)H)A$, I denotes the identity matrix, and $\widetilde{Q}(k) = BM(k)RM(k)^T B^T$.

Recall from (23) that the Kalman gain $K(k)$ can be written as

$$\begin{aligned} K(k) &= \left(\overline{P}(k) H^T + S(k)\right) \check{R}(k)^{-1} \\ &= \overline{P}(k)\left(H + S(k)^T \overline{P}(k)^{-1}\right)^T \left(H\overline{P}(k) H^T + R \right. \\ &\quad \left. + HS(k) + S(k)^T H^T\right)^{-1} \\ &= \overline{P}(k)\left(H + S(k)^T \overline{P}(k)^{-1}\right)^T \left(\left(H + S(k)^T \overline{P}(k)^{-1}\right)\right. \\ &\quad \cdot \overline{P}(k)\left(H + S(k)^T \overline{P}(k)^{-1}\right)^T + R - S(k)^T \overline{P}(k)^{-1} \\ &\quad \left. \cdot S(k)\right)^{-1}. \end{aligned} \tag{25}$$

Defining

$$\begin{aligned} \overline{H}(k) &= H + S(k)^T \overline{P}(k)^{-1}, \\ \overline{R}(k) &= R - S(k)^T \overline{P}(k)^{-1} S(k) \end{aligned} \tag{26}$$

the equation of the Kalman gain matrix can be equivalently expressed as

$$K(k) = \overline{P}(k)\overline{H}(k)^T \left(\overline{H}(k)\overline{P}(k)\overline{H}(k)^T + \overline{R}(k)\right)^{-1} \tag{27}$$

and the error covariance matrix $P(k)$ can be expressed as

$$\begin{aligned} P(k) &= \overline{P}(k) - K(k)\left(\overline{P}(k) H^T + S(k)\right)^T \\ &= \overline{P}(k) - K(k)\overline{H}(k)\overline{P}(k) \\ &= \overline{P}(k) - \overline{P}(k)\overline{H}(k)^T \left(\overline{H}(k)\overline{P}(k)\overline{H}(k)^T + \overline{R}(k)\right)^{-1} \\ &\quad \cdot \overline{H}(k)\overline{P}(k). \end{aligned} \tag{28}$$

Defining $F(k) = I - K(k)\overline{H}(k)$, we further have

$$\begin{aligned} P(k) &= \left(I - K(k)\overline{H}(k)\right)\overline{P}(k)\left(I - K(k)\overline{H}(k)^T\right) \\ &\quad + K(k)\overline{R}(k)K(k)^T \\ &= F(k)\overline{P}(k)F(k)^T + K(k)\overline{R}(k)K(k)^T, \end{aligned} \tag{29}$$

$$K(k) = P(k)\overline{H}(k)^T \overline{R}(k)^{-1}. \tag{30}$$

Derivation details of (29) and (30) are omitted here for conciseness.

Using the matrix inversion lemma,

$$\left(A + BD^{-1}C\right)^{-1} = A^{-1} - A^{-1}B\left(D + CA^{-1}B\right)^{-1} CA^{-1}. \tag{31}$$

Equation (29) can be written as

$$P(k) = \left(\overline{P}(k)^{-1} + \overline{H}(k)^T \overline{R}(k)^{-1} \overline{H}(k)\right)^{-1}. \tag{32}$$

Substituting the definition of $\overline{H}(k)$ into the state equation (21) gives

$$\hat{x}(k) = \overline{x}(k) + K(k)(z(k) - H\overline{x}(k)) \tag{33}$$

$$\begin{aligned} &= \overline{x}(k) + K(k) \\ &\quad \cdot \left(z(k) + S(k)^T \overline{P}(k)^{-1} \overline{x}(k) - \overline{H}(k)\overline{x}(k)\right). \end{aligned} \tag{34}$$

Defining the weighted sensor data $y(k)$ and the information matrix $U(k)$ as

$$\begin{aligned} y(k) &= \overline{H}(k)^T \overline{R}(k)^{-1}\left(z(k) + S(k)^T \overline{P}(k)^{-1} \overline{x}(k)\right), \\ U(k) &= \overline{H}(k)^T \overline{R}(k)^{-1} \overline{H}(k) \end{aligned} \tag{35}$$

the information filtering for simultaneous input and state estimation can be restated as

$$\begin{aligned} \tilde{x}(k) &= A\hat{x}(k-1), \\ \hat{d}(k-1) &= M(k)(z(k) - H\tilde{x}(k)), \\ \overline{x}(k) &= \tilde{x}(k) + B\hat{d}(k-1), \\ \hat{x}(k) &= \overline{x}(k) + P(k)\left(y(k) - U(k)\overline{x}(k)\right), \\ P(k) &= \left(\overline{P}(k)^{-1} + U(k)\right)^{-1}. \end{aligned} \tag{36}$$

5. Consensus Based Information Filtering

5.1. Consensus Based Filtering Algorithm. A distributed sensor network consisting of N sensor nodes is adopted to monitor a large-scale system with unknown boundary conditions. The nodes are connected to each other in some network topology. The set of neighbors of node i including node i itself is denoted by the set N_i.

Consensus information filtering for system (6) can be expressed as

$$\hat{x}^{(i)} = \overline{x}^{(i)} + P^{(i)}\left(y^{(i)} - U^{(i)}\overline{x}^{(i)}\right) + C^{(i)}\sum_{j\in N_i}\left(\overline{x}^{(j)} - \overline{x}^{(i)}\right), \tag{37}$$

where $P^{(i)}$, $U^{(i)}$, and $y^{(i)}$ are calculated from (17) or (36) according to the sensor node i lying in the outside grids (Case 1) of the large-scale system or inside grids (Case 2), respectively.

The choice of consensus gain $C^{(i)}$ is related to the estimation variance of local sensor i; that is,

$$C^{(i)} = \epsilon\frac{\overline{P}^{(i)}}{1 + \left\|\overline{P}^{(i)}\right\|}, \tag{38}$$

where $\|\cdot\|$ is the Frobenius norm of a matrix and $\epsilon > 0$ is a relatively small constant.

The local boundary condition estimate

$$\hat{d}^{(i)} = M^{(i)}\left(z^{(i)} - H^{(i)}\hat{x}^{(i)}\right), \tag{39}$$

for $i \in N_b$.

In construction of subsystems, we require $\bigcup I_i^d = \{1,\ldots,m\}$, which means that the global boundary conditions are covered by a set of local estimates. Therefore, we can reconstruct the estimate of boundary conditions by choosing the corresponding components from the local estimate:

$$\hat{d}(j) = \hat{d}^{(i)}(j), \tag{40}$$

$j \in I_i^d$, where $\hat{d}(j)$ is the jth component of the estimate vector $\hat{d}$.

5.2. Convergence Analysis. Stability of estimation is often formulated in terms of stabilizability and detectability conditions. However, these conditions are not sufficient to guarantee stability in the more complicated simultaneous state and input estimations. Following the results in [12], we present stability analysis for distributed estimation of large-scale systems with unknown boundary conditions.

Lemma 4. *For the information filter (36), $F(k) = I - P(k)U(k) = P(k)\overline{P}^{-1}(k)$, and $P(k+1) = F(k)G(k)F^T(k)$ with*

$$G(k) = \widetilde{A}(k-1)P(k-1)\widetilde{A}^T(k-1) + \widetilde{Q}(k-1) + \overline{P}(k-1)U(k)\overline{P}(k-1). \tag{41}$$

Moreover, if $U(k)$ is positive definite, then $G(k)$ is positive definite as well.

Lemma 5. *Suppose that information matrix $U(k)$ is positive definite. Then, the error dynamics of the discrete-time information filter (36) is globally asymptotically stable with a Lyapunov function $V(\eta(k)) = \eta(k)^T P^{-1}(k)\eta(k)$.*

Proofs of Lemmas 4 and 5 are deduced from (30), (32) and with $P(k)(\overline{P}(k) + U(k)) = I$.

Theorem 6. *Consider the distributed filter (37) with consensus gain $C^{(i)}(k) = \gamma F^{(i)}(k)G^{(i)}(k)$ and $F^{(i)}(k) = I - P^{(i)}(k)U^{(i)}(k)$:*

$$G^{(i)}(k) = \widetilde{A}^{(i)}(k-1)P^{(i)}(k-1)\widetilde{A}^{(i)T}(k-1) + \widetilde{Q}^{(i)}(k-1) + \overline{P}^{(i)}(k)U^{(i)}(k)\overline{P}^{(i)}(k). \tag{42}$$

Suppose that the information matrix $U^{(i)}(k)$ is positive definite for all i. Then, the error dynamics of the distributed Kalman consensus filter (37) is globally asymptotically stable for a sufficient small $\gamma > 0$. Furthermore, all estimators asymptotically reach a consensus on state estimates; that is, $\hat{x}^{(1)}(k) = \hat{x}^{(2)}(k) = \cdots = \hat{x}^{(N)}(k) = x(k)$.

Proof. The error dynamics (without noise) of the consensus filter can be written as

$$\begin{aligned}\eta^{(i)}(k+1) &= F^{(i)}(k)\widetilde{A}^{(i)}(k)\eta^{(i)}(k) \\ &\quad + C^{(i)}(k)\widetilde{A}^{(i)}(k)\sum_{j\in N_i}\left(\eta^{(j)}(k) - \eta^{(i)}(k)\right).\end{aligned} \tag{43}$$

Using the Lyapunov function in Lemma 5 and calculating the change

$$\delta V(\eta(k)) = V(\eta(k+1)) - V(\eta(k)), \tag{44}$$

we have

$$\begin{aligned}\delta V &= \sum_i \eta^{(i)T}(k+1)\left(P^{(i)}(k)\right)^{-1}\eta^{(i)}(k+1) \\ &\quad - \eta^{(i)T}(k)\left(P^{(i)}(k)\right)^{-1}\eta^{(i)}(k) \\ &= \sum_i \left(F^{(i)}(k)\widetilde{A}^{(i)}(k)\eta^{(i)}(k) + C^{(i)}(k)\mu^{(i)}(k)\right)^T \\ &\quad \cdot \left(P^{(i)}(k+1)\right)^{-1}\left(F^{(i)}(k)\widetilde{A}^{(i)}(k)\eta^{(i)}(k) + C^{(i)}(k)\right. \\ &\quad \left.\cdot\,\mu^{(i)}(k)\right) - \eta^{(i)T}(k)\left(P^{(i)}(k)\right)^{-1}\eta^{(i)}(k) \\ &= \sum_i \eta^{(i)T}(k) \\ &\quad \cdot \left[\widetilde{A}^{(i)T}(k)F^{(i)T}(k)\left(P^{(i)}(k+1)\right)^{-1}F^{(i)}(k)\widetilde{A}^{(i)}(k)\right.\end{aligned}$$

$$-\left(P^{(i)}(k)\right)^{-1}\Big]\eta^{(i)}(k)+2\sum_{i}\overline{\eta}^{(i)T}(k+1)F^{(i)T}(k)$$
$$\cdot\left(P^{(i)}(k+1)\right)^{-1}C^{(i)}(k)\mu^{(i)}(k)+\sum_{i}\mu^{(i)T}(k)$$
$$\cdot C^{(i)T}(k)\left(P^{(i)}(k+1)\right)^{-1}C^{(i)}(k)\mu^{(i)}(k) \tag{45}$$

with

$$\mu^{(i)}(k)=\widetilde{A}^{(i)}(k)\sum_{j\in N_i}\left(\eta^{(j)}(k)-\eta^{(i)}(k)\right). \tag{46}$$

The first term in (45) is negative semidefinite from Lemma 4, and the second term can be designed negative semidefinite using the quadrtic property of graph Laplacians [12, 13], by setting the consensus gain:

$$C^{(i)}(k)=\gamma F^{(i)}(k)G^{(i)}(k). \tag{47}$$

The last term in the expression of δV is positive semidefinite, while we could choose γ sufficiently small so that the sum of the first and third terms is negative semidefinite. The following proof is analogous to the Theorem 2 in [14]. □

6. Example: A Heat Conduction Process

Consider a rod with length L and cross section radius r; see Figure 1. The density, heat capacity, and thermal conductivity of the material are denoted by ρ, C_p, and κ, respectively. Using the energy balance equation, we get the following partial differential equation:

$$\frac{\partial T}{\partial t}=\frac{1}{\rho C_p}\left[\kappa\frac{\partial^2 T}{\partial l^2}+\frac{\lambda P_e}{A_T}\left(T_e-T\right)\right], \tag{48}$$

where T is the temperature of the rod, T_e the temperature of the environment, λ the heat transfer coefficient of the surface of the rod, l the spatial coordinate of the length, $P_e=2\pi r$ the perimeter of the rod, and $A_T=\pi r^2$ the area of the longitudinal section.

For the distributed parameter system (48), we assume that the initial condition is $T(l,0)=T_0$, and the boundary conditions $T(0,t)$, $T(L,t)$ are unknown inputs, respectively.

Employing the central approximation (4), (48) becomes the ordinary differential equation

$$\frac{dT_i}{dt}=C_xT_{i-1}-\left(2C_x+C_{\text{Peh}}\right)T_i+C_xT_{i+1}+C_{\text{Peh}}T_e \tag{49}$$

with i being the node index corresponding to the grid point index increasing from the left to the right and

$$C_x=\frac{\kappa}{\rho C_p h^2},\qquad C_{\text{Peh}}=\frac{\lambda P_e}{\rho C_p A_T}, \tag{50}$$

where h is the distance between each grid.

Table 1: Model and simulation parameters.

Parameters	Values	Units
ρ	8700	kgm^{-3}
κ	400	$\text{Wm}^{-1}\,\text{K}^{-1}$
C_p	385	$\text{Jkg}^{-1}\,\text{K}^{-1}$
λ	10	$\text{Wm}^{-1}\,\text{K}^{-1}$
T_e	0	K
T_b	298.15	K
L	4	m
n	10	–
T_0	50	K
x_0	100	K
Q	0.2	–
R	0.2	–

Using the temporal Euler approximation (time increment assumed to be less than $1/2C_x$ for convergence), the state equation is discretized to get the discrete-time linear equation

$$x(k+1)=Ax(k)+Bd(k)+w(k), \tag{51}$$

with state vector $x=\left[T_1\ \cdots\ T_n\right]^T$, unknown boundary conditions $d(k)=\left[T^L(k)\ \ T^R(k)\right]^T$, and model approximation error $w(k)$ with the covariance matric Q.

In this simulation, the boundary conditions are taken as

$$T^L(k)=T^R(k)=\frac{T_b}{4}\sin(k)\zeta(k), \tag{52}$$

where $\zeta(k)$ is a normal Gaussian variable. Suppose that no information about $T^L(k)$, $T^R(k)$ is available for use.

The model and simulation parameters are shown in Table 1. Three local sensor groups are located separately at the left, central, and right grids of the rod. The simulation final time is 80 seconds and the consensus gains are $C^{(i)}=0.2$. All sensor groups are fully connected.

We get the following three subsystem observation matrices as

$$H^{(1)}=\begin{bmatrix}1&0&\cdots&0&0\\0&1&\cdots&0&0\end{bmatrix},$$
$$B^{(1)}=\begin{bmatrix}1&0&\cdots&0\end{bmatrix}^T,$$
$$H^{(2)}=\begin{bmatrix}\cdots&1&0&0&\cdots\\\cdots&0&1&0&\cdots\\\cdots&0&0&1&\cdots\end{bmatrix},$$
$$B^{(2)}=0,$$
$$H^{(3)}=\begin{bmatrix}0&0&\cdots&1&0\\0&0&\cdots&0&1\end{bmatrix},$$
$$B^{(3)}=\begin{bmatrix}0&0&\cdots&1\end{bmatrix}^T. \tag{53}$$

The simulations are performed in MATLAB 7.0 environment using an Intel 2.40 GHz processor with 512 MB of

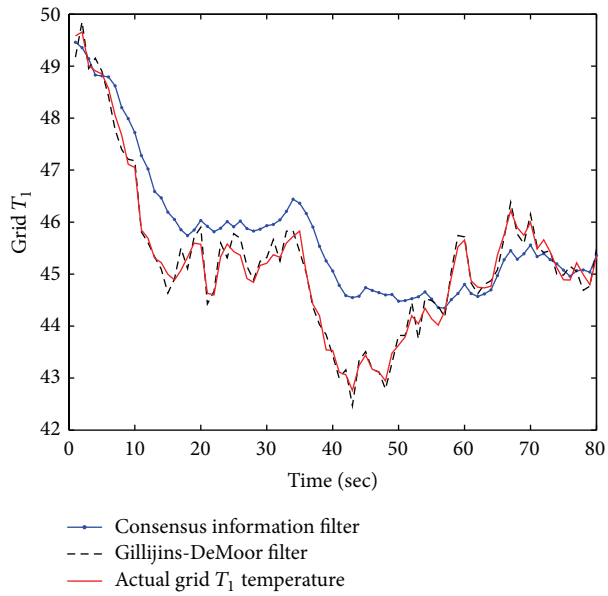

Figure 2: Temperature values of the leftmost grid.

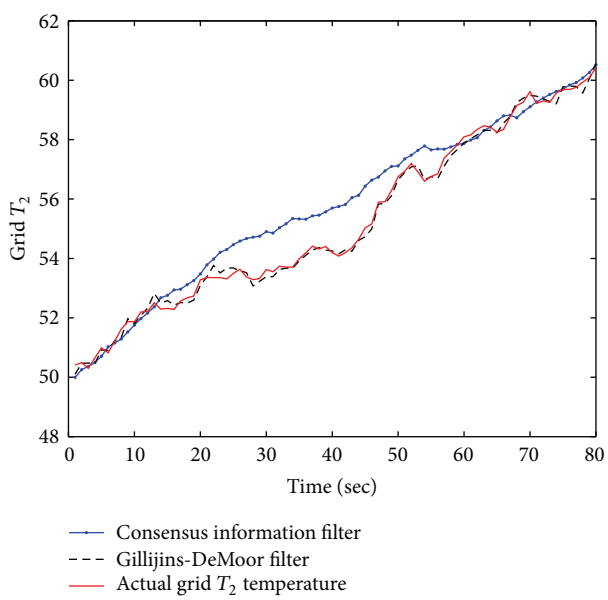

Figure 3: Temperature values of the second grid.

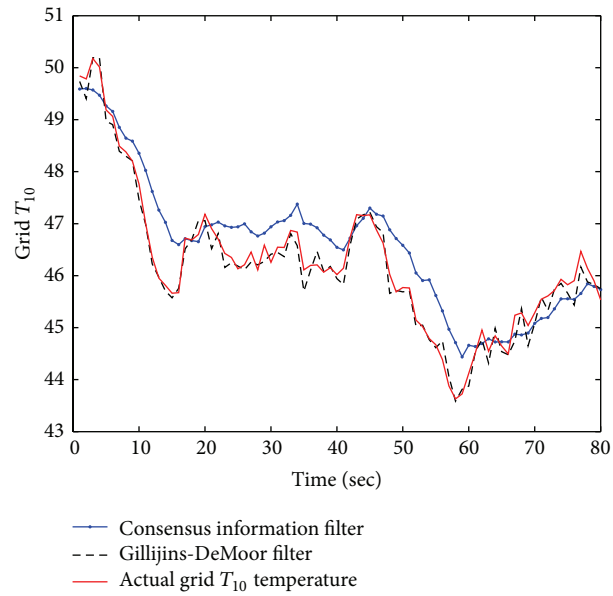

Figure 4: Temperature values of the rightmost grid.

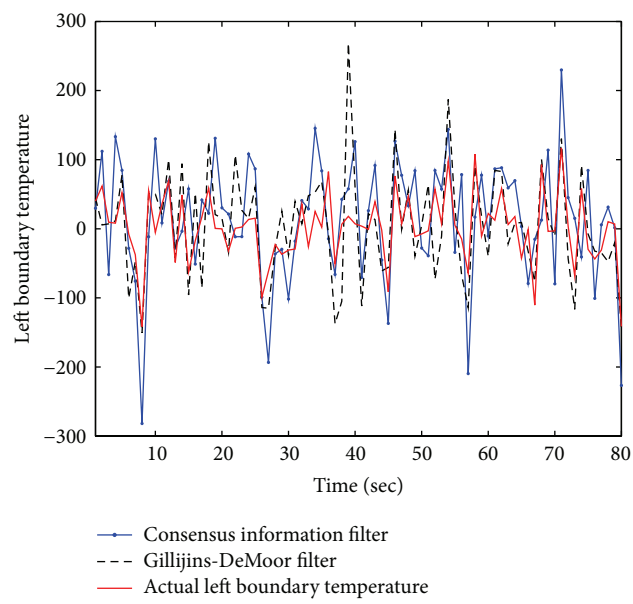

Figure 5: The left boundary conditions.

RAM and under Microsoft Windows XP operating systems. The simulations have been repeated 50 times, and averagely the distributed information filtering takes about 1.07 seconds while the simulation time of the optimal Gillijins-De Moor algorithm requires about 2.60 seconds.

We compare the estimation error between the distributed information filtering and the Gillijins-De Moor algorithm. The state estimation results for grids 1, 2, and 10 are shown in Figures 2–4, and left and right boundary condition estimations are shown in Figures 5-6. It may be seen that the distributed filtering provides a relatively satisfied estimation for both states and boundary inputs. However, compared with the Gillijins-De Moor algorithm, the developed filtering slightly reduces the accuracy of estimation within about 8 percent. An obvious improvement (with simple model magnitude) is the fact that the algorithm speed is faster than using the Gillijins-De Moor algorithm.

Following Olfati-Saber and Shamma [15] on the appropriate measure of the disagreement of the estimates, we use the following disagreement measure:

$$\|\delta(k)\|^2 = \sum_{i=1}^{N} \left(\delta^{(i)}(k)\right)^2, \tag{54}$$

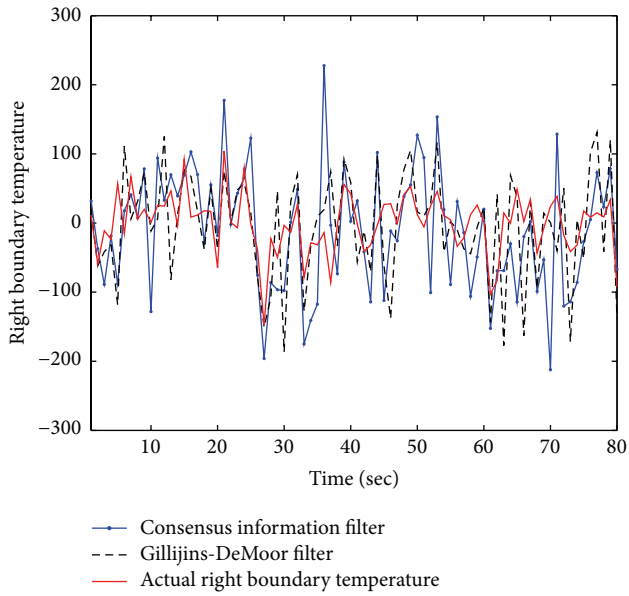

Figure 6: The right boundary conditions.

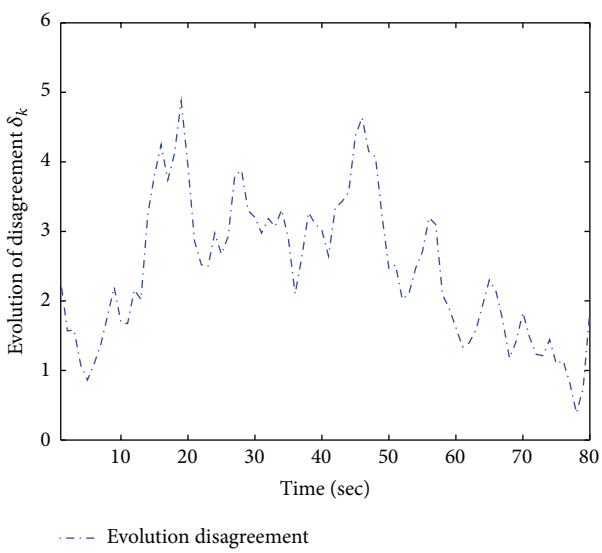

Figure 7: Evolution of consensus disagreement $\delta(k)$.

where

$$\delta^{(i)}(k) = \hat{x}^{(i)}(k) - \mu(k),$$
$$\mu(k) = \frac{1}{N}\sum_{i}\hat{x}^{(i)}(k). \tag{55}$$

The evolution of the disagreement term is depicted in Figure 7, which shows that a relatively high disagreement at the beginning is bounded asymptotically.

7. Conclusions

The concept of consensus filters is applied to distributed parameter systems with unknown boundary conditions. The local sensor groups have different reduced observation models according to their spatial locations, and we decompose the whole large-scale distributed parameter systems into different local subsystems. Local information filtering algorithms are developed to estimate the state and the boundary condition simultaneously and extended to the distributed Kalman consensus filter through a penalty term that enforces consensus. A 1-dimensional heat conduction process model with two known boundary conditions is used to illustrate the spatially distributed consensus filter with three sensor groups.

Future work will consider applications of the distributed consensus filtering estimation to some concrete distributed parameter systems, such as the networks of traffic flows with changing traffic demands, and hydrological or atmospheric transportation models.

Conflict of Interests

The authors declare that there is no conflict of interests regarding the publication of this paper.

Acknowledgment

This work was partially supported by the National Science Foundation of China (NSFC) under Grant no. 61374076.

References

[1] E. Fiorelli, N. E. Leonard, P. Bhatta, D. A. Paley, R. Bachmayer, and D. M. Fratantoni, "Multi-AUV control and adaptive sampling in Monterey Bay," *IEEE Journal of Oceanic Engineering*, vol. 31, no. 4, pp. 935–948, 2006.

[2] S.-J. Lin and R. B. Rood, "An explicit flux-form Semi-Lagrangian shallow-water model on the sphere," *Quarterly Journal of the Royal Meteorological Society*, vol. 123, no. 544, pp. 2477–2498, 1997.

[3] D. S. Oliver, Y. Zhang, H. A. Phale, and Y. Chen, "Distributed parameter and state estimation in petroleum reservoirs," *Computers & Fluids*, vol. 46, no. 1, pp. 70–77, 2011.

[4] O.-P. Tossavainen, J. Percelay, M. Stacey, J. P. Kaipio, and A. Bayen, "State estimation and modeling error approach for 2-D shallow water equations and Lagrangian measurements," *Water Resources Research*, vol. 47, no. 10, Article ID W10510, 2011.

[5] M. J. Lighthill and G. B. Whitham, "On kinematic waves, II: a theory of traffic flow on long crowded roads," *Proceedings of the Royal Society of London A*, vol. 229, pp. 317–345, 1955.

[6] M. Papageorgiou, J.-M. Blosseville, and H. Hadj-Salem, "Modelling and real-time control of traffic flow on the southern part of Boulevard Peripherique in Paris. Part I. Modelling," *Transportation Research Part A*, vol. 24, no. 5, pp. 345–359, 1990.

[7] J. C. Herrera and A. M. Bayen, "Incorporation of Lagrangian measurements in freeway traffic state estimation," *Transportation Research Part B: Methodological*, vol. 44, no. 4, pp. 460–481, 2010.

[8] Y. Wang and M. Papageorgiou, "Real-time freeway traffic state estimation based on extended Kalman filter: a general approach," *Transportation Research Part B: Methodological*, vol. 39, no. 2, pp. 141–167, 2005.

[9] S. Gillijns and B. De Moor, "Unbiased minimum-variance input and state estimation for linear discrete-time systems," *Automatica*, vol. 43, no. 1, pp. 111–116, 2007.

[10] S. Gillijns and B. De Moor, "Unbiased minimum-variance input and state estimation for linear discrete-time systems with direct feedthrough," *Automatica*, vol. 43, no. 5, pp. 934–937, 2007.

[11] Y. Lu, L. Zhang, and X. Mao, "Distributed information consensus filters for simultaneous input and state estimation," *Circuits, Systems, and Signal Processing*, vol. 32, no. 2, pp. 877–888, 2013.

[12] W. Ren and R. W. Beard, "Consensus seeking in multiagent systems under dynamically changing interaction topologies," *IEEE Transactions on Automatic Control*, vol. 50, no. 5, pp. 655–661, 2005.

[13] R. Olfati-Saber, "Flocking for multi-agent dynamic systems: algorithms and theory," *IEEE Transactions on Automatic Control*, vol. 51, no. 3, pp. 401–420, 2006.

[14] R. Olfati-Saber, "Kalman-consensus filter: optimality, stability, and performance," in *Proceedings of the 48th IEEE Conference on Decision and Control, Held Jointly with the 28th Chinese Control Conference (CDC/CCC '09)*, pp. 7036–7042, Shanghai, China, December 2009.

[15] R. Olfati-Saber and J. S. Shamma, "Consensus filters for sensor networks and distributed sensor fusion," in *Proceedings of the 44th IEEE Conference on Decision and Control, and the European Control Conference*, pp. 6698–6703, December 2005.