Distributed finite-time observer for LTI systems: A kernel-based approach

Yunqing Chen, Jinman Yang, Peng Li and Boli Chen

***Abstract*— The problem of distributed state estimation of a linear-time-invariant (LTI) system is addressed in this paper. Given a directed communication network, the full state vector can be reconstructed at an appointed node under the assumption of an open Hamiltonian path. By a suitably designed coordinate transformation, the initial conditions of the subsystems can be estimated successively along the path. Thanks to the Volterra integral operator induced by non-asymptotic kernel functions, the estimation task at each agent can be achieved within a predefined finite time and transmitted to the next node. As such, the instant state vector can be reconstructed at the end node after a finite time interval. Such a scheme is prone to reduce the communication burden. Extensive numerical examples are conducted to verify the effectiveness of the proposed observer in both noise-free and noisy scenarios.**

I. Introduction

In recent years, increasing number of large-scale systems arouse the need for distributed control theory. Numerous examples can be seen in the field of power networks, environment monitoring, smart city, etc. The *divide et impera* structure of the distributed schemes brings about advantages in scalability and flexibility compared to the centralised methods. Distributed state estimation is one of the fundamental problems in the field of distributed control as it is significantly instrumental for state-based controller design [1], fault diagnosis [2], attack detection [3] and so on.

The goal of distributed state estimation is to reconstruct the full state vector of the system at a single node using local measurement and communication with its neighbouring nodes. Inspired by centralised frameworks, typical state estimation methods have been extended to deal with distributed systems, to name but a few, in [4], [5], [6], [7], [8]. A distributed Kalman filter-based observer is proposed in [9] and it was further applied to a consensus control problem in [10]. There exist a large variety of Luenberger observer-based distributed estimation schemes, see [11], [12], [13]. Particularly in [12], the Luenberger observer is developed to achieve distributed consensus at each node. With successive decomposition, the state convergence can be achieved by a suitable weight tunning rule. Similar Luenberger frameworks are also used in [13], in which the observability conditions are characterised in relation to the communication topology.A finite-time distributed observer is designed by exploiting the homogeneity theory, which ensures the decay of state estimation error in a finite time. An alternative solution method is designed in [14] that transfers the state estimation problem into a parameter estimation problem. The finite-time convergence is guaranteed by the dynamic regressor extension and mixing method. Recently, a fixed-time observer is designed in [15] based on a kernel-based estimation method by modifying the centralised framework in [16] and invoking multi-hop communication with minimised data exchange.

The aforementioned researches provide various solutions to the distributed problems. However, the adaptive and correction-based estimation methods rely on the quality of communication and local calculation ability. The convergence may suffer from delay and band-limited rate. Moreover, in some practical cases, full state reconstruction is not necessary to be achieved in all nodes. To save communication and computation resources, it is desirable to estimate the state at several required nodes. In this paper, a distributed observer is proposed to reconstruct the full state with instantaneous convergence at a distinguished node. The state space of the system is decomposed along the assumed open Hamiltonian path ending at the appointed node. Extending the centralised method in [17], estimators are designed for the observable subspace for each node to obtain their initial state with fast convergence. Collecting the information of the neighbouring nodes through the communication path, the appointed node is able to reconstruct the full state vector in finite time in conjunction with the state prediction. As a result, the requirements for communication can be relieved, making the observer more tolerant to possible delay and limited band rate which are unavoidable in cyber-physical systems [18], [19].

The paper is organised as follows: in the next section, the distributed state estimation problem is described and some preliminaries to achieve finite-time estimation are reviewed. In Section III, the distributed state estimation algorithm is demonstrated and its effectiveness is examined by numerical examples as shown in Section IV. Section V gives the conclusions and future research prospects.

II. Problem formulation and preliminaries

A. Finite-time distributed state estimation

Consider a linear time-invariant (LTI) system

$$x^{(1)}(t) = Ax(t), \qquad y(t) = Cx(t), \tag{1}$$

with $x(t) \in \mathbb{R}^n$, $y(t) \in \mathbb{R}^m$ the state vector and the output respectively, and $x^{(i)}(t)$ denotes the ith derivative of $x(t)$. $A \in \mathbb{R}^{n\times n}$ is the system matrix and $C \in \mathbb{R}^{m\times n}$ is the output matrix and the pair (C, A) is fully observable. The system is measured by N distributed sensors

$$y_i(t) = C_i x(t),\ i \in \{1, \dots, N\}, \tag{2}$$

Y. Chen, J. Yang and P. Li are with the School of Mechanical Engineering and Automation at Harbin Institute of Technology, Shenzhen, China (*chenyunqing2000@gmail.com;kiman2016@gmail.com;lipeng2020@hit.edu.cn*).

B. Chen is with the Dept. of Electronic and Electrical Engineering at University College London, UK (*boli.chen@ucl.ac.uk*).

Corresponding author: Peng Li. This work was supported in part by the Guangdong Basic and Applied Basic Research Foundation (2021A1515110262, 2022A1515011274) and in part by the Shenzhen Fundamental Research Project (JCYJ20210324120400003, JCYJ20220818102416036).

where $y(t) = [y_1(t)^\top, y_2(t)^\top, \ldots, y_N(t)^\top]^\top$ and $C = [C_1^\top, C_2^\top, \ldots, C_N^\top]^\top$. The pair (C_i, A) for each sensor is not fully observable and $C_i \neq 0, \forall i \in \{1, \ldots, N\}$, i.e. the ith sensor is not able to reconstruct the full state vector $x(t)$ using only its own measurement $y_i(t)$. On the other hand, each node corresponds to its own observable state variables $x_i(t), \forall i = 1, \ldots, N$. The set consisting of the observable state vectors $\{x_1(t), x_2(t), \ldots, x_N(t)\}$ subsumes the full state vector $x(t)$ recalling the fact that the overall system is observable.

The sensors are able to communicate with their neighbouring sensors, whose information flow is governed by a directed communication graph $\mathcal{G} = \{\mathcal{N}, \mathcal{E}, \mathcal{A}\}$, where $\mathcal{N} = \{1, 2, \cdots, N\}$ is the set sensors regarded as nodes in $\mathcal{G}$. $\mathcal{E} \subseteq \mathcal{N} \times \mathcal{N}$ is a set of edges, and $\mathcal{A} = [a_{ij}] \in \mathbb{R}^{N \times N}$ denotes the adjacency matrix. The element a_{ij} is the weight of the edge (i, j), and $a_{ij} = 1$ if and only if $(i, j) \in \mathcal{E}$ and $a_{ij} = 0$ otherwise. Specifically, $(i, j) \in \mathcal{E}$ means that the i-th node can send information to the j-th node, i.e., i is a neighbour of j. The set of neighbours of node j is described by $\mathcal{N}_j = \{i : (i, j) \in \mathcal{E}\}$.

The goal of this paper is to design a distributed state estimation algorithm so that the state estimates $\hat{x}(t)$ at a given node $i \in \mathcal{N}$ converges to the true state $x(t)$ in finite-time, i.e. $\tilde{x}(t) \triangleq \hat{x}(t) - x(t) = 0, \forall t \geq T_c$, where T_c is a finite convergence time.

Different from the goal in [15] to achieve state reconstruction at all nodes, the task in this paper is to reconstruct the full state at certain nodes, which is more common and economical in practical applications. As such, the requirement for a strong connection is weakened herein.

Assumption 1: Given the directed communication graph $\mathcal{G}$, for each node $i = 1, \ldots, N$, there exists an open Hamiltonian path that ends at node i and connects to all other nodes $j \in \mathcal{V} \backslash i$ without repetition.

B. Volterra operator and non-asymptotic kernel

In this work, the distributed state estimation is achieved by an observability decomposition and initial state estimation. Thanks to the adopted kernel-based estimation method, which is first proposed in [17], by suitably designed kernel functions, the initial conditions of the state variables can be retrieved in finite time. For readers' convenience and to make the paper sufficiently self-contained, some key facts instrumental to the proposed framework are recalled subsequently. The interested reader is referred to [17], [20] and [21] for a deeper insight on the algebra of Volterra integral operators.

Let $r(t) \in \mathbb{R}, \forall t \geq 0$ be an i-th order differentiable signal. Given a Hilbert-Schmidt ($\mathcal{HS}$) Kernel Function $K(\cdot, \cdot) : \mathbb{R} \times \mathbb{R} \to \mathbb{R}$, the Volterra integral operator induced by the kernel function is

$$[V_K r](t) \triangleq \int_0^t K(t, \tau) r(\tau) d\tau, \;\; t \in \mathbb{R}_{\geq 0} \,. \tag{3}$$

Consider an N-th order Bivariate Linear Non-asymptotic Kernel (BL-NK) function $K(t, \tau)$ in the shape of

$$K(t, \tau) = e^{-\omega(t-\tau)} \left(1 - e^{-\omega(t-\tau)}\right)^N , \tag{4}$$

with $\omega \in \mathbb{R}_{>0}$ being the tuning parameters. This type of kernel functions are characterised by two useful features:

- It is non-asymptotic up to the n-th order i.e. $K^{(i)}(t, t) = 0, \forall i \in \{1 \ldots n-1\}$, so that the Volterra image of the signal derivative can be expressed as

$$\begin{aligned} \left[V_K r^{(i)}\right](t) = &\sum_{j=0}^{i-1} (-1)^{i-j} r^{(j)}(0) K^{(i-j-1)}(t, 0) \\ &+ (-1)^i \left[V_{K^{(i)}} r\right](t), \end{aligned} \tag{5}$$

  in which effects of the lower-order derivatives $r(t), r^{(1)}(t), \ldots, r^{(i-1)}(t)$ are removed.
- The Volterra operators induced by this kernel can be implemented as an LTI system, processing the available I/O signals and producing the transformed signals as output. To be specific, letting $\xi(t) = [\mathcal{V}_{K^{(i)}} r](t)$, it holds that

$$\begin{aligned} \kappa^{(1)}(t) &= G\kappa(t) + Er(t) \\ \xi(t) &= \mathbf{1}_{N+1} \kappa(t) \end{aligned} \tag{6}$$

  with $\kappa(0) = 0 \in \mathbb{R}^{N+1}$ and where G is a diagonal, time-invariant and Hurwitz matrix, defined by $G = \mathrm{diag}(-\omega, -2\omega, \ldots, -(N+1)\omega)$, and $E = [\lambda_{i,1}, \lambda_{i,2}, \ldots, \lambda_{i,N+1}]^\top$, with constant elements $\lambda_{i,j} = (-1)^{j-1} \binom{N}{j-1} (j\omega)^i$, $j = 1, 2, \cdots, N+1$. $\mathbf{1}_{N+1}$ is an $N+1$ dimensional row vector of ones.

III. Distributed kernel-based finite-time observer

In this section, a distributed state estimation scheme is proposed deploying the kernel-based observer that guarantees finite-time convergence properties. Under Assumption 1 that the communication graph has an open Hamiltonian path $\mathcal{P}$ rooting at node i, by successive observability decompositions as in [7] along the Hamiltonian path, there exists a corresponding linear transformation $z_{\mathcal{P}}(t) = \mathcal{T}_{\mathcal{P}} x(t)$ and making the transformed system a downward stair form:

$$z_{\mathcal{P}}^{(1)}(t) = \bar{A}_{\mathcal{P}} z_{\mathcal{P}}(t), \qquad \bar{y}_{\mathcal{P}}(t) = \bar{C}_{\mathcal{P}} z_{\mathcal{P}}(t), \tag{7}$$

with

$$\begin{aligned} \bar{A}_{\mathcal{P}} &= \mathcal{T}_{\mathcal{P}} A \mathcal{T}_{\mathcal{P}}^{-1} \\ &= \begin{bmatrix} \bar{A}_{\mathcal{P},11} & 0 & 0 & \cdots & 0 \\ \bar{A}_{\mathcal{P},21} & \bar{A}_{\mathcal{P},22} & 0 & \cdots & 0 \\ \vdots & \vdots & \ddots & \ddots & 0 \\ \bar{A}_{\mathcal{P},(N-1)1} & \bar{A}_{\mathcal{P},(N-1)2} & \cdots & \bar{A}_{\mathcal{P},(N-1)(N-1)} & 0 \\ \bar{A}_{\mathcal{P},N1} & \bar{A}_{\mathcal{P},N2} & \cdots & \bar{A}_{\mathcal{P},N(N-1)} & \bar{A}_{\mathcal{P},NN} \end{bmatrix}, \end{aligned}$$

$$\begin{aligned} \bar{C}_{\mathcal{P}} &= C_{\mathcal{P}} \mathcal{T}_{\mathcal{P}} \\ &= \begin{bmatrix} \bar{C}_{\mathcal{P},11} & 0 & 0 & \cdots & 0 \\ \bar{C}_{\mathcal{P},21} & \bar{C}_{\mathcal{P},22} & 0 & \cdots & 0 \\ \vdots & \vdots & \ddots & \ddots & 0 \\ \bar{C}_{\mathcal{P},(N-1)1} & \bar{C}_{\mathcal{P},(N-1)2} & \cdots & \bar{C}_{\mathcal{P},(N-1)(N-1)} & 0 \\ \bar{C}_{\mathcal{P},N1} & \bar{C}_{\mathcal{P},N2} & \cdots & \bar{C}_{\mathcal{P},N(N-1)} & \bar{C}_{\mathcal{P},NN} \end{bmatrix}, \end{aligned}$$

where $(\bar{C}_{\mathcal{P},jj}, \bar{A}_{\mathcal{P},jj})$ are observable $\forall j = 1, \ldots, N$.

Let us partition the state vector and the output signal according to the size of $A_{\mathcal{P},jj}$ and $C_{\mathcal{P},jj}, \forall j = 1, \dots, N$ as follows

$$z_{\mathcal{P}}(t) = \left[z_{\mathcal{P},1}, z_{\mathcal{P},2}, \cdots, z_{\mathcal{P},N}\right]^\top,$$

$$\bar{y}_{\mathcal{P}}(t) = \left[\bar{y}_{\mathcal{P},1}, \bar{y}_{\mathcal{P},2}, \cdots, \bar{y}_{\mathcal{P},N}\right]^\top.$$

where $z_{\mathcal{P},j}(t) \in \mathbb{R}^{n_{\mathcal{P},j}}, \bar{y}_{\mathcal{P},j}(t) \in \mathbb{R}^{m_{\mathcal{P},j}}$ with $\sum_{j=1}^{N} n_{\mathcal{P},j} = n$ and $\sum_{j=1}^{N} m_{\mathcal{P},j} = m$. Note that the transformation $\mathcal{T}_{\mathcal{P}}$ is determined by the Hamiltonian path ending at node i. $\bar{y}_{\mathcal{P}}(t) \in \mathbb{R}^m$ is a permutation of $y(t)$ based on certain path. That is to say, the path $\mathcal{P}$ defines an index set $\{k_1, k_2, \dots, k_{N-1}, i\}$ that ordering the topological communication and it holds that $\bar{y}_{\mathcal{P},j}(t) = y_{k_j}(t), \forall j = 1, \dots, N-1$. The subsequent analysis is conducted with regard to the transformed system (7).

A. State estimation at the starting node

Considering the starting node of the chosen Hamiltonian path whose dynamics writes

$$z^{(1)}_{\mathcal{P},1}(t) = \bar{A}_{\mathcal{P},11} z_{\mathcal{P},1}(t), \quad \bar{y}_{\mathcal{P},1}(t) = \bar{C}_{\mathcal{P},11} z_{\mathcal{P},1}(t), \tag{8}$$

Recalling the fact that $z_{\mathcal{P},1}(t)$ is fully observable from the single measurement $\bar{y}_{\mathcal{P},1}(t)$, there exists another linear transformation that $\zeta_{\mathcal{P},1}(t) = T_{\mathcal{P},1} z_{\mathcal{P},1}(t)$ resulting in an observer canonical form with

$$\zeta^{(1)}_{\mathcal{P},1}(t) = \check{A}_{\mathcal{P},1}\zeta_{\mathcal{P},1}(t), \quad \bar{y}_{\mathcal{P},1}(t) = \check{C}_{\mathcal{P},1}\zeta_{\mathcal{P},1}(t), \tag{9}$$

with

$$\check{A}_{\mathcal{P},1} = T_{\mathcal{P},1}\bar{A}_{\mathcal{P},1}T^{-1}_{\mathcal{P},1} = \begin{bmatrix} a_{n_{\mathcal{P},1}-1,1} & 1 & 0 & \cdots & 0 \\ a_{n_{\mathcal{P},1}-2,1} & 0 & 1 & \cdots & 0 \\ \vdots & \vdots & \ddots & \ddots & 0 \\ a_{1,1} & 0 & \cdots & 0 & 1 \\ a_{0,1} & 0 & \cdots & 0 & 0 \end{bmatrix}$$

and $\check{C}_{\mathcal{P},1} = \begin{bmatrix}1 & 0 & \cdots & 0\end{bmatrix}$, in which $(-a_{0,1}, -a_{1,1}, \cdots, -a_{n_{\mathcal{P},1}-1,1})$ are the coefficients of the characteristic polynomial of matrix $\bar{A}_{\mathcal{P},11}$. For simplicity, we are assuming the system to have a single output. Indeed, the proposed method can be further extended to observers for multi-outputs by further observability decomposition [22].

In this context, system (9) admits the following I/O form

$$\bar{y}^{(n_{\mathcal{P},1})}_{\mathcal{P},1} = \sum_{p=0}^{n_{\mathcal{P},1}-1} a_{p,1}\bar{y}^{(p)}_{\mathcal{P},1} \tag{10}$$

Furthermore, the state variables of the realization (9) can be expressed in terms of the output derivatives as

$$\zeta_{\mathcal{P},1}(t) = \bar{y}^{(r)}_{\mathcal{P},1}(t) - \sum_{j=0}^{r-1} a_{n_{\mathcal{P},1}-r+p,1}\bar{y}^{(p)}_{\mathcal{P},1}(t). \tag{11}$$

where the convention $\sum_{p=0}^{k}\{\cdot\} = 0$, for $k < 0$ has been used. Applying the Volterra operator induced by an $n_{\mathcal{P},1}$th order non-asymptoic kernel function, one can obtain

$$\begin{aligned}&\sum_{p=0}^{n_{\mathcal{P},1}-1}(-1)^{n_{\mathcal{P},1}-p-1}\bar{y}^{(p)}_{\mathcal{P},1}(0)K^{(n_{\mathcal{P},1}-p-1)}(t,0)\\ &+(-1)^{n_{\mathcal{P},1}}\left[V_{K^{(n_{\mathcal{P},1})}}\bar{y}_{\mathcal{P},1}\right](t) = \sum_{q=0}^{n_{\mathcal{P},1}-1} a_{q,1}\Bigg((-1)^q\left[V_{K^{(q)}}\bar{y}_{\mathcal{P},1}\right](t)\\ &+\sum_{p=0}^{q-1}(-1)^{p+q-1}\bar{y}^{(p)}_{\mathcal{P},1}(0)K^{q-p-1}(t,0)\Bigg)\end{aligned} \tag{12}$$

After some algebra, (12) can be rearranged as

$$\mu_{\mathcal{P},1}(t) = \gamma_{\mathcal{P},1}(t)\zeta_{\mathcal{P},1}(0) \tag{13}$$

where

$$\begin{aligned}\mu_{\mathcal{P},1}(t) &\triangleq (-1)^{n_{\mathcal{P},1}-1}\left[V_{K^{(n_{\mathcal{P},1})}}\bar{y}_{\mathcal{P},1}\right](t)\\ &\quad+\sum_{p=0}^{n_{\mathcal{P},1}-1} a_{p,1}(-1)^p\left[V_{K^{(p)}}\bar{y}_{i,1}\right](t)\\ \gamma_{\mathcal{P},1}(t) &\triangleq \left[(-1)^{n_{\mathcal{P},1}-1}K^{(n_{\mathcal{P},1}-1)}(t,0), \cdots, K(t,0)\right].\end{aligned}$$

In order to solve for the initial state $\zeta_{\mathcal{P},1}(0)$, one can augment the scalar equation by the covariance filtering method, leading to

$$S_{\mathcal{P},1}(t) = \mathscr{L}^{-1}\{F(s)\gamma^\top_{\mathcal{P},1}\mu_{\mathcal{P},1}(s)\}(t), \tag{14}$$

$$R_{\mathcal{P},1}(t) = \mathscr{L}^{-1}\{F(s)\gamma^\top_{\mathcal{P},1}\gamma_{\mathcal{P},1}\}(t), \tag{15}$$

where $F(s) = \frac{1}{s+g}$ is a first-order low-pass filter with zero initial conditions with a positive g as the forgetting factor. $\mathcal{L}$ denotes the inverse Laplacian transformation. Being $F(s)$ a linear operator, it holds that

$$S_{\mathcal{P},1}(t) = R_{\mathcal{P},1}(t)\zeta_{\mathcal{P},1}(0). \tag{16}$$

Note that the vector $S_{\mathcal{P},1}(t)$ is composed of a linear combination of Volterra images with respect to $\bar{y}_{\mathcal{P},1}$, which can be calculated as the output the LTI system (6) induced by respective kernels $K(t,\tau)$. The matrix $R_{\mathcal{P},1}(t)$ is guaranteed to be non-singular $\forall t > 0$, thanks to the certain shape of the kernel function [17]. Thereby, the state observer is formulated as

$$\hat{\zeta}_{\mathcal{P},1}(0) = \begin{cases} R^{-1}_{\mathcal{P},1}(t)S_{\mathcal{P},1}(t), & t > \sigma_1, \\ 0, & \text{otherwise}, \end{cases} \tag{17}$$

where $\sigma_1 \in \mathbb{R}_{>0}$ is an small-valued threshold for estimator activation. In such case, the instantaneous state can be calculated by

$$\hat{\zeta}_{\mathcal{P},1}(t) = e^{\check{A}_{\mathcal{P},11}t}\hat{\zeta}_{\mathcal{P},1}(0), \forall t > \sigma_1. \tag{18}$$

Equivalently,

$$\hat{z}_{\mathcal{P},1}(t) = T^{-1}_{\mathcal{P},1}e^{\check{A}_{\mathcal{P},11}t}\zeta_{\mathcal{P},1}(0), \forall t > \sigma_1. \tag{19}$$

B. State estimation at the non-starting nodes

For nodes $j \geq 2$, their dynamics are described by

$$\begin{aligned} z_{\mathcal{P},j}^{(1)}(t) &= \bar{A}_{\mathcal{P},jj} z_{i,j}(t) + \sum_{k=1}^{j-1} \bar{A}_{\mathcal{P},kj} z_{i,k}(t), \\ \check{y}_{\mathcal{P},j}(t) &= \bar{C}_{\mathcal{P},jj} z_{\mathcal{P},j}(t). \end{aligned} \tag{20}$$

where $\check{y}_{\mathcal{P},j}(t) \triangleq \bar{y}_{\mathcal{P},j}(t) - \sum_{k=1}^{j-1} \bar{C}_{\mathcal{P},kj} z_{\mathcal{P},k}(t)$, with $z_{\mathcal{P},k}(t)$ received from the kth node through the Hamiltonian path. Accordingly, there exists a linear transform $\zeta_{\mathcal{P},j}(t) = T_{\mathcal{P},j} z_{\mathcal{P},j}(t)$ converting the system dynamics into an observer canonical form:

$$\begin{aligned} \zeta_{\mathcal{P},j}^{(1)}(t) &= \check{A}_{\mathcal{P},j} \zeta_{\mathcal{P},j}(t) + T_{\mathcal{P},j} \sum_{k=1}^{j-1} \bar{A}_{\mathcal{P},kj} z_{\mathcal{P},k}(t), \\ \check{y}_{\mathcal{P},j}(t) &= \check{C}_{\mathcal{P},j} z_{\mathcal{P},j}(t), \end{aligned} \tag{21}$$

where

$$\check{A}_{\mathcal{P},j} = T_{\mathcal{P},1} \bar{A}_{\mathcal{P},jj} T_{\mathcal{P},j}^{-1} = \begin{bmatrix} a_{n_{\mathcal{P},j}-1,j} & 1 & 0 & \cdots & 0 \\ a_{n_{\mathcal{P},j}-2,j} & 0 & 1 & \cdots & 0 \\ \vdots & \vdots & \ddots & \ddots & 0 \\ a_{1,j} & 0 & \cdots & 0 & 1 \\ a_{0,j} & 0 & \cdots & 0 & 0 \end{bmatrix},$$

and $\check{C}_{\mathcal{P},1} = \begin{bmatrix} 1 & 0 & \cdots & 0 \end{bmatrix}$. For simplicity, we stack the input into a vector with the dimension $\rho_{\mathcal{P},j} = \sum_{p=0}^{j-1} n_{\mathcal{P},p}$, as follows

$$\begin{aligned} u_{\mathcal{P},j}(t) &= [z_{\mathcal{P},1}(t)^\top, z_{\mathcal{P},2}(t)^\top, \ldots, z_{\mathcal{P},j-1}(t)^\top]^\top \\ &\triangleq [u_{\mathcal{P}j,1}(t), u_{\mathcal{P}j,2}(t), \ldots, u_{\mathcal{P}j,\rho_{\mathcal{P},j}}(t)], \end{aligned}$$

and define

$$\begin{aligned} \check{B}_{\mathcal{P},j} &\triangleq T_{\mathcal{P},j} \left[\bar{A}_{\mathcal{P},1j}, \bar{A}_{\mathcal{P},2j}, \ldots, \bar{A}_{\mathcal{P},(j-1)j} \right] \\ &\triangleq \begin{bmatrix} b_{\mathcal{P},11} & b_{\mathcal{P},12} & \cdots & b_{\mathcal{P},1(\rho_{\mathcal{P},j})} \\ b_{\mathcal{P}j,21} & b_{\mathcal{P}j,22} & \cdots & b_{\mathcal{P}j,2(\rho_{\mathcal{P},j})} \\ \vdots & \vdots & & \vdots \\ b_{\mathcal{P}j,n_j 1}(t) & b_{\mathcal{P}j,n_j 2}(t) & \cdots & b_{\mathcal{P}j,n_{\mathcal{P},j}(\rho_{\mathcal{P},j})} \end{bmatrix}. \end{aligned}$$

For the sake of further analysis, let us rewrite the system (21) in the I/O form

$$\check{y}_{\mathcal{P},j}^{(n_{\mathcal{P},j})}(t) = \sum_{p=0}^{n_{\mathcal{P},j}-1} a_{p,j} \check{y}_{\mathcal{P},j}^{(p)}(t) + \sum_{k=1}^{\rho_{\mathcal{P},j}} \sum_{q=0}^{n_{\mathcal{P},j}-1} b_{\mathcal{P}j,kq} u_{\mathcal{P}j,k}^{(q)}(t). \tag{22}$$

The rth state variable of system (21) can be expressed as

$$\begin{aligned} \zeta_{\mathcal{P},j,r}(t) &= \check{y}_{\mathcal{P},j}^{(r)}(t) - \sum_{p=0}^{r-1} a_{n_{\mathcal{P},j}-r+p,1} \check{y}_{\mathcal{P},j}^{(p)}(t) \\ &\quad - \sum_{k=1}^{\rho_{\mathcal{P},j}} \sum_{q=0}^{r-1} b_{\mathcal{P}j,k(n_{\mathcal{P},j}-r+q)} u_{\mathcal{P}j,k}^{(q)}(t). \end{aligned} \tag{23}$$

Applying the Volterra operator V_K to both sides of (22), one can obtain

$$\begin{aligned} &\sum_{p=0}^{n_{\mathcal{P},j}-1} (-1)^{n_{\mathcal{P},j}-p-1} \check{y}_{\mathcal{P},j}^{(p)}(0) K^{(n_{\mathcal{P},j}-p-1)}(t,0) \\ &+(-1)^{n_{\mathcal{P},j}} \left[V_{K^{(n_{i,j})}} \check{y}_{\mathcal{P},j} \right](t) = \sum_{q=0}^{n_{\mathcal{P},j}-1} a_{q,j} \Bigg((-1)^q [V_{K^{(q)}} \check{y}_{\mathcal{P},j}](t) \\ &+ \sum_{p=0}^{q-1} (-1)^{p+q-1} \check{y}_{\mathcal{P},j}^{(p)}(0) K^{q-p-1}(t,0) \Bigg) \\ &+ \sum_{k=1}^{\rho_{\mathcal{P},j}} \sum_{q=0}^{r-1} b_{\mathcal{P}j,k(n_{\mathcal{P},j}-r+q)} \Bigg((-1)^q [V_{K^{(q)}} u_{\mathcal{P}j,k}](t) \\ &+ \sum_{p=0}^{q-1} (-1)^{p+q-1} u_{\mathcal{P}j,k}^{(p)}(0) K^{q-p-1}(t,0) \Bigg). \end{aligned}$$

The above equation can be rearranged as

$$\mu_{\mathcal{P},j}(t) = \gamma_{\mathcal{P},j}(t) \zeta_{\mathcal{P},j}(0), \tag{24}$$

where

$$\begin{aligned} \mu_{\mathcal{P},j}(t) &\triangleq \sum_{k=1}^{\rho_{\mathcal{P},j}} \sum_{q=0}^{r-1} b_{\mathcal{P}j,kq} (-1)^q [V_{K^{(q)}} u_{\mathcal{P}j,k}](t) \\ &(-1)^{n_{\mathcal{P},j}-1} \left[V_{K^{(n_{\mathcal{P},j})}} \bar{y}_{\mathcal{P},j} \right](t) + \sum_{q=0}^{n_{i,j}-1} a_{q,j} (-1)^q \left[V_{K^{(q)}} \check{y}_{\mathcal{P},j} \right](t), \\ \gamma_{\mathcal{P},j}(t) &\triangleq \left[(-1)^{n_{\mathcal{P},j}-1} K^{(n_{\mathcal{P},j}-1)}(t,0), \cdots, K(t,0) \right]. \end{aligned}$$

Similarly to (25), by leveraging the covariance filtering technique, one can obtain

$$S_{\mathcal{P},j}(t) = \mathscr{L}^{-1}\{F(s) \gamma_{\mathcal{P},j}^\top \mu_{\mathcal{P},j}(s)\}(t) \tag{25}$$

$$R_{\mathcal{P},j}(t) = \mathscr{L}^{-1}\{F(s) \gamma_{\mathcal{P},j}^\top \gamma_{\mathcal{P},j}\}(t). \tag{26}$$

In consequence, the initial state of the non-starting nodes for $j \leq 2$ can be retrieved in order along the path

$$\hat{\zeta}_{\mathcal{P},j}(0) = \begin{cases} R_{\mathcal{P},j}(t)^{-1} S_{\mathcal{P},j}(t), & t > \sigma_j, \\ 0, & \text{otherwise}, \end{cases} \tag{27}$$

with the positive activation threshold $\sigma_j \in \mathbb{R}_{>0}$, verifying $\sigma_j \geq \sigma_{j-1}$. Then, the initial condition of system (20) can be recovered as

$$\hat{z}_{\mathcal{P},j}(0) = T_{\mathcal{P},j}^{-1} \hat{\zeta}_{\mathcal{P},j}(0), \tag{28}$$

By successively solving the observer equation (27) at each node following the path $\mathcal{P}$ till the Nth agent, it is possible to recover the full state at the Nth node with the initial conditions, $\forall t > \sigma_N$

$$\hat{z}_{\mathcal{P},N}(t) = e^{\bar{A}_{\mathcal{P}} t} [z_{\mathcal{P},1}(0)^\top, z_{\mathcal{P},2}(0)^\top, \ldots, z_{\mathcal{P},N}(0)^\top]^\top.$$

Consequently, the state vector of the original system is retrieved at the ith node as

$$\hat{x}(t) = \mathcal{T}_{\mathcal{P}}^{-1} \hat{z}_{\mathcal{P},N}(t). \tag{29}$$

It is worth noting that, the Volterra operator induced by the BL-NK is internally stable and the mapping system (6) is BIBO. Therefore, with the linear operations and algebraic calculations in (13),(17), (24), (27) and (29), if the measurement is perturbed by an additive noise, the state estimation error is guaranteed to be bounded with respect to bounded noise.

Remark 3.1: To recover the full state at a specific node, only an open Hamiltonian path is required. In the case of reconstruction in all nodes of the graph, a closed communication loop is needed.

Remark 3.2: The proposed method transfers the initial state $\hat{z}_{i,j}(0)$ among the nodes and reconstructs the instant value locally at a specific node. In such case, the effects of communication delay can be avoided compared to methods requiring continuous communication with instant state $\hat{z}_{i,j}(t)$. Moreover, the initial conditions are not necessarily to be transferred at each time step. Hence, the communication frequency can be reduced leading to a possible relief of the communication burden. Indeed, the open-loop prediction may cause error accumulation, especially in the presence of model uncertainties. Nonetheless, such issues can be mitigated by periodically resetting of the estimator.

IV. Numerical Examples

In this section, the performance of the proposed distributed observer is examined by a distributed system consisting of three nodes:

$$x^{(1)}(t) = Ax(t), \quad y(t) = Cx(t),$$

where $x(t) \in \mathbb{R}^6$ and $y(t) \in \mathbb{R}^3$ and

$$A = \begin{bmatrix} -1 & 0 & 0 & 0 & 0 & 0 \\ -1 & 1 & 1 & 0 & 0 & 0 \\ 1 & -2 & -1 & -1 & 1 & 1 \\ 0 & 0 & 0 & -1 & 0 & 0 \\ -8 & 1 & -1 & -1 & -2 & 0 \\ 4 & -0.5 & 0.5 & 0 & 0 & -4 \end{bmatrix}$$

$$C = \begin{bmatrix} 1 & 0 & 0 & 2 & 0 & 0 \\ 2 & 0 & 0 & 1 & 0 & 0 \\ 2 & 0 & 5 & 0 & 0 & 3 \end{bmatrix} = \begin{bmatrix} C_1 \\ C_2 \\ C_3 \end{bmatrix}.$$

It is readily seen that the pair (C, A) is fully observable, while none of the pair (C_i, A) for each sensor is fully observable. The communication network is given by the directed graph shown in Fig. 1. The initial condition of the overall

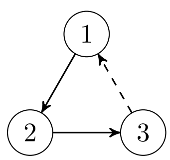

Fig. 1. Communication topology among nodes

system is given by $x(0) = \begin{bmatrix} 1 & 3 & -2 & -3 & -1 & 2 \end{bmatrix}^\top$. To verify the robustness of the observer, we also consider a uniformly distributed random noise $d_y(t)$ additively contaminating the sensors, i.e. $y_d(t) = y(t) + d_y(t)$ and the corresponding signal-noise ratio (SNR) is 35.3 .The goal is to estimate the full state vector $x(t)$ locally at node 3. There exists an open Hamiltonian path $\mathcal{P} : 1 \to 2 \to 3$. By successive observability decompositions, a linear transformation $z_{\mathcal{P}}(t) = \mathcal{T}_{\mathcal{P}} x(t)$ can be designed such that

$$\mathcal{T}_{\mathcal{P}} = \begin{bmatrix} -0.4472 & 0 & 0 & -0.8944 & 0 & 0 \\ -0.8944 & 0 & 0 & 0.4472 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & -1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 \end{bmatrix}.$$

Consequently, the global state vector $z(t) \in \mathbb{R}^6$ are partitioned as $z_{\mathcal{P}}(t) = \begin{bmatrix} z_{\mathcal{P},1}(t) & z_{\mathcal{P},2}(t) & z_{\mathcal{P},3}(t) \end{bmatrix}^\top$, where $z_{\mathcal{P},1}(t), z_{\mathcal{P},2}(t) \in \mathbb{R}$ and $z_{\mathcal{P},3}(t) \in \mathbb{R}^4$, corresponding to the partition of the system matrix.

Deploying the proposed state estimation scheme with the kernel parameters set to $N = 4$ and $\omega = 1$, the full state vector is reconstructed locally at Node 3, as shown in Fig. 2.

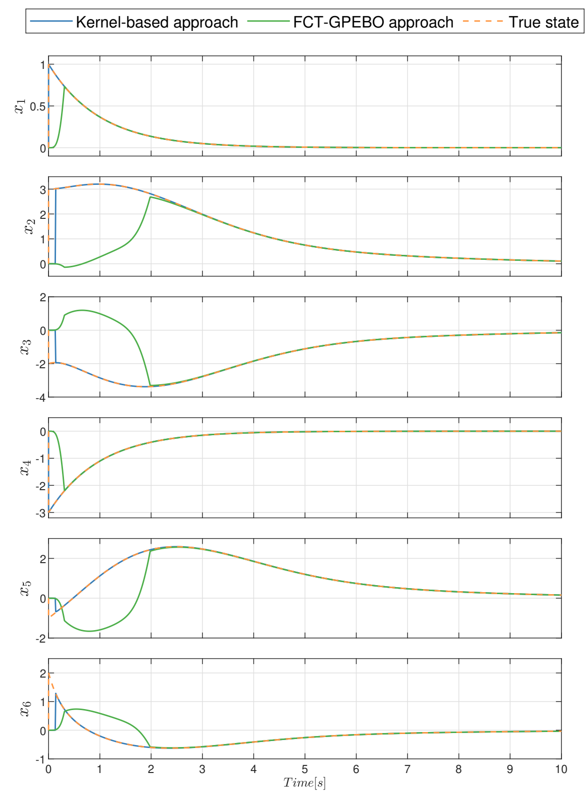

Fig. 2. Estimates of state x in noise-free scenario.

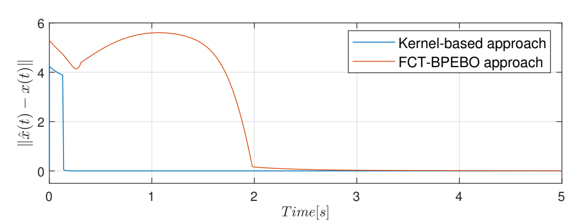

Fig. 3. State estimation error in the noise-free scenario.

The performance of the kernel-based estimation method is compared with the method proposed in [14]. It has been shown that both methods are able to reconstruct the entire state vector $x(t)$ at a given node in finite time. As compared in Fig. 3, the estimation error of both methods goes to zero while the kernel-based method shows faster convergence. To be specific, the convergence is reached immediately after the activation while the activation time can be chosen arbitrarily small.

In the noisy scenario, the estimation results of the two estimators are plotted in Fig. 4. The kernel-based method provides instantaneous convergence albeit with a slight overshoot at the beginning. Both methods show comparable noise immunity whereas the proposed method has a higher convergence rate.

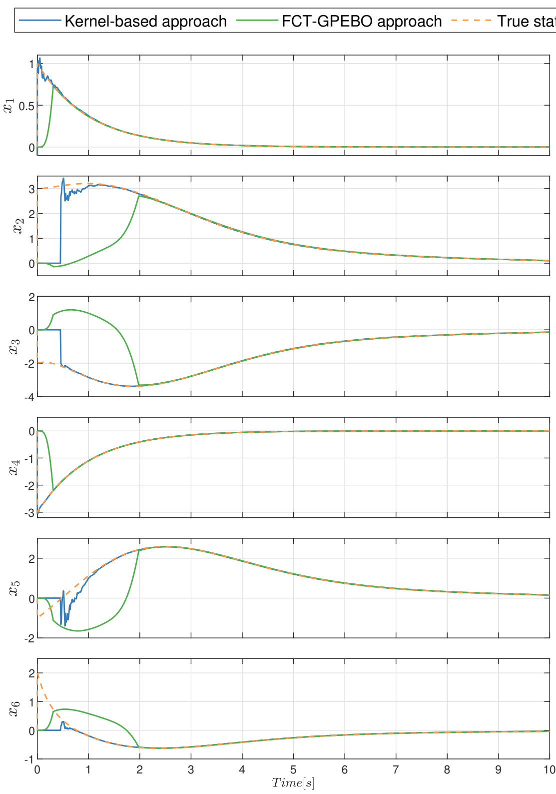

Fig. 4. Estimates of state x in the noisy scenario.

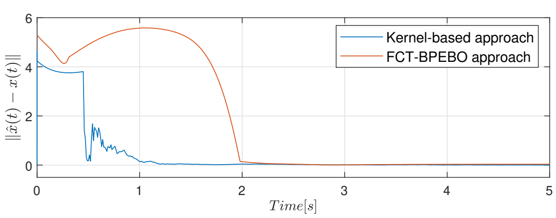

Fig. 5. State estimation error in the noisy scenario.

V. Conclusion

In this paper, a distributed framework is designed under the assumption of an open Hamiltonian path. By successive observability decomposition, the distributed network can be rearranged into a series of connected observable subsystems. Making use of the Volterra operator and the non-asymptotic kernel function, the initial condition of each subsystem can be estimated in order based on communication through the predefined path. Consequently, at the end of the Hamiltonian path which is the interested node, the full state vector can be reconstructed within a finite time interval. Moreover, the proposed method also offers benefits in terms of avoiding the effects of communication delay and limited band rate. Future research efforts can be paid to addressing time-varying graphs and resilient state estimation.

References

[1] B. Liu, T. Chu, L. Wang, and G. Xie, "Controllability of a leader–follower dynamic network with switching topology," *IEEE Transactions on Automatic Control*, vol. 53, no. 4, pp. 1009–1013, 2008.

[2] Y. Wu, Z. Wang, and Z. Huang, "Distributed fault detection for nonlinear multi-agent systems under fixed-time observer," *Journal of the Franklin Institute*, vol. 356, no. 13, pp. 7515–7532, 2019.

[3] A. J. Gallo, M. S. Turan, F. Boem, T. Parisini, and G. Ferrari-Trecate, "A distributed cyber-attack detection scheme with application to dc microgrids," *IEEE Transactions on Automatic Control*, vol. 65, no. 9, pp. 3800–3815, 2020.

[4] T. Kim, C. Lee, and H. Shim, "Completely decentralized design of distributed observer for linear systems," *IEEE Transactions on Automatic Control*, vol. 65, no. 11, pp. 4664–4678, 2019.

[5] F. C. Rego, Y. Pu, A. Alessandretti, A. P. Aguiar, A. M. Pascoal, and C. N. Jones, "A distributed luenberger observer for linear state feedback systems with quantized and rate-limited communications," *IEEE Transactions on Automatic Control*, vol. 66, no. 9, pp. 3922–3937, 2020.

[6] S. Park and N. C. Martins, "Design of distributed LTI observers for state omniscience," *IEEE Transactions on Automatic Control*, vol. 62, no. 2, pp. 561–576, 2016.

[7] A. Mitra and S. Sundaram, "Distributed observers for LTI systems," *IEEE Transactions on Automatic Control*, vol. 63, no. 11, pp. 3689–3704, 2018.

[8] W. Han, H. L. Trentelman, Z. Wang, and Y. Shen, "A simple approach to distributed observer design for linear systems," *IEEE Transactions on Automatic Control*, vol. 64, no. 1, pp. 329–336, 2018.

[9] R. Olfati-Saber, "Distributed kalman filtering for sensor networks," in *2007 46th IEEE Conference on Decision and Control*. IEEE, 2007, pp. 5492–5498.

[10] R. Carli, A. Chiuso, L. Schenato, and S. Zampieri, "Distributed kalman filtering based on consensus strategies," *IEEE Journal on Selected Areas in communications*, vol. 26, no. 4, p. 622, 2008.

[11] S. Park and N. C. Martins, "Design of distributed LTI observers for state omniscience," *IEEE Transactions on Automatic Control*, vol. 62, no. 2, pp. 561–576, 2017.

[12] Á. R. del Nozal, P. Millán, L. Orihuela, A. Seuret, and L. Zaccarian, "Distributed estimation based on multi-hop subspace decomposition," *Automatica*, vol. 99, pp. 213–220, 2019.

[13] V. Ugrinovskii, "Conditions for detectability in distributed consensus-based observer networks," *IEEE Transactions on Automatic Control*, vol. 58, no. 10, pp. 2659–2664, 2013.

[14] R. Ortega, E. Nuño, and A. Bobtsov, "Distributed observers for LTI systems with finite convergence time: A parameter-estimation-based approach," *IEEE Transactions on Automatic Control*, vol. 66, no. 10, pp. 4967–4974, 2020.

[15] P. Ge, P. Li, B. Chen, and F. Teng, "Fixed-time convergent distributed observer design of linear systems: A kernel-based approach," *IEEE Transactions on Automatic Control*, 2022.

[16] P. Li, F. Boem, G. Pin, and T. Parisini, "Kernel-based simultaneous parameter-state estimation for continuous-time systems," *IEEE Transactions on Automatic Control*, vol. 65, no. 7, pp. 3053–3059, 2019.

[17] B. Chen, P. Li, G. Pin, G. Fedele, and T. Parisini, "Finite-time estimation of multiple exponentially-damped sinusoidal signals: A kernel-based approach," *Automatica*, vol. 106, pp. 1–7, 2019.

[18] N. R. Chowdhury, J. Belikov, D. Baimel, and Y. Levron, "Observer-based detection and identification of sensor attacks in networked CPSs," *Automatica*, vol. 121, p. 109166, 2020.

[19] H. Silm, R. Ushirobira, D. Efimov, E. Fridman, J.-P. Richard, and W. Michiels, "Distributed observers with time-varying delays," *IEEE Transactions on Automatic Control*, vol. 66, no. 11, pp. 5354–5361, 2020.

[20] P. Li and B. Chen, "Parameter estimation for a sinusoidal signal with a time-varying amplitude," in *2022 European Control Conference (ECC)*. IEEE, 2022, pp. 1708–1713.

[21] P. Li, G. Pin, G. Fedele, and T. Parisini, "Deadbeat source localization from range-only measurements: A robust kernel-based approach," *IEEE Transactions on Control Systems Technology*, vol. 27, no. 3, pp. 923–933, 2018.

[22] J. C. Willems and J. W. Polderman, *Introduction to mathematical systems theory: a behavioral approach*. Springer Science & Business Media, 1998, vol. 26.