Consensus-Based Distributed Nonlinear Filtering With Kernel Mean Embedding

LIPING GUO
Chinese Academy of Sciences, Beijing, China

JIMIN WANG, Member, IEEE
University of Science and Technology Beijing, Beijing, China
Ministry of Education, Beijing, China

YANLONG ZHAO, Senior Member, IEEE
Chinese Academy of Sciences, Beijing, China
University of Chinese Academy of Sciences, Beijing, China

JI-FENG ZHANG, Fellow, IEEE
Zhongyuan University of Technology, Zheng Zhou, China
Chinese Academy of Sciences, Beijing, China

Received 9 April 2024; revised 21 September 2024; accepted 2 December 2024. Date of publication 9 December 2024; date of current version 14 April 2025.

DOI. No. 10.1109/TAES.2024.3513280

Refereeing of this contribution was handled by H. Oh.

This work was supported in part by the National Natural Science Foundation of China under Grant 62025306, Grant 62433020, Grant 62203045, and Grant T2293770, in part by the CAS Project for Young Scientists in Basic Research under Grant YSBR-008, in part by the CAS Special Research Assistant Project under Grant E4559305, in part by the China Postdoctoral Science Foundation under Grant 2024M763479, and in part by the Postdoctoral Fellowship Program of CPSF under Grant GZB20230813.

Authors' addresses: Liping Guo is with the Key Laboratory of Systems and Control, Institute of Systems Science, Academy of Mathematics and Systems Science, Chinese Academy of Sciences, Beijing 100190, China, E-mail: (lipguo@outlook.com); Jimin Wang is with the School of Automation and Electrical Engineering, University of Science and Technology Beijing, Beijing 100083, China, and also with the Key Laboratory of Knowledge Automation for Industrial Processes, Ministry of Education, Beijing 100083, China, E-mail: (jimwang@ustb.edu.cn); Yanlong Zhao is with the Key Laboratory of Systems and Control, Institute of Systems Science, Academy of Mathematics and Systems Science, Chinese Academy of Sciences, Beijing 100190, China, and also with the School of Mathematical Sciences, University of Chinese Academy of Sciences, Beijing 100049, China, E-mail: (ylzhao@amss.ac.cn); Ji-Feng Zhang is with the School of Automation and Electrical Engineering, Zhongyuan University of Technology, Zheng Zhou 450007, China, and also with the Key Laboratory of Systems and Control, Institute of Systems Science, Academy of Mathematics and Systems Science, Chinese Academy of Sciences, Beijing 100190, China, E-mail: (jif@iss.ac.cn). *(Corresponding author: Jimin Wang.)*

0018-9251 © 2024 IEEE

This article proposes a consensus-based distributed nonlinear filter with kernel mean embedding (KME) to fill the gap of kernel-based filters for distributed sensor networks. Specifically, to approximate the posterior distribution, the system state is embedded into a higher dimensional reproducing kernel Hilbert space (RKHS), and then the nonlinear measurement function is linearly represented. As a result, an update rule for the KME of posterior distribution is established in the RKHS. To demonstrate that the proposed distributed filter can achieve centralized estimation accuracy, a centralized filter is first developed by extending the standard Kalman filter in the state space to the RKHS. Then, the proposed distributed filter is proved to be equivalent to the centralized one. Two examples highlight the effectiveness of the developed filters in target tracking scenarios, including a nearly constantly moving target and a turning target, respectively, with range, bearing, and range-rate measurements.

I. INTRODUCTION

THE nonlinear filtering problem is critically important in many fields, such as target tracking [1], [2], [3], [4], [5], [6], [7], [8], navigation [9], and detection [10]. In practice, the measurements are usually collected by some sensors in a network structure. Since conventional centralized filtering can achieve high estimation accuracy, it is often regarded as an evaluation benchmark for filtering accuracy. However, centralized filtering requires each sensor to send its measurement to the fusion center promptly, which is frequently burdened with a computationally demanding task and susceptible to processing unit failures [11]. In contrast, distributed filtering allows each sensor to implement filtering independently using its measurements and to reach an agreement by communicating with its neighbors. This strategy is generally more practical, robust, resilient, and computationally efficient [12], [13], [14], [15], [16]. Consequently, research on the distributed nonlinear filtering problem warrants attention.

Bayesian filtering offers a unified recursive framework for nonlinear filtering problem, but the posterior probability density function cannot be expressed analytically due to the system's nonlinearity. As a result, many approximation schemes have been developed, focusing either on the estimated quantities, such as moments, or the probability density function. By approximating the first two moments of the posterior distribution, the distributed Kalman-type filters gain significant popularity and have been widely applied in many practical scenarios. For example, by employing Taylor expansion to linearize the nonlinear functions, the distributed Kalman filter can be directly applied to nonlinear systems [17], [18], [19], [20]. Still, it may diverge for problems of high nonlinearity [21]. Hence, to solve this problem, the distributed unscented Kalman filter [22], the distributed cubature Kalman filter [23], and their extensions have been proposed [24]. Although these Kalman-type filters are efficient and convenient to implement, they cannot obtain the global posterior probability density function for further statistical inference.

By approximating the posterior probability density function, the celebrated particle filters adopt the Monte Carlo technique to draw a set of particles for fitting the probability density functions. Under different sampling

Authorized licensed use limited to: Yunnan University. Downloaded on April 14,2025 at 13:45:04 UTC from IEEE Xplore. Restrictions apply.

strategies, the distributed particle filters have been developed and widely applied [25], [26], [27], [28], [29], [30]. Although the particle filters are quite valuable for distribution approximation, they often suffer from a heavy computational burden. Therefore, to efficiently approximate the posterior probability density function, the kernel method is widely recognized as a powerful tool for distribution approximation [31], [32], [33], [34], [35], [36]. Besides, by introducing a reproducing kernel Hilbert space (RKHS), the kernel method represents the posterior probability density function as a point in the RKHS where standard mathematical operations can be performed. It should be noted that there exist many kernel-based filters for single-sensor setups using either the kernel Bayes' rule [37] or the kernel Kalman rule [38], which can be found in [32], [35], and [39]. As far as we know, however, for distributed sensor networks, research on the kernel-based filter is still lacking.

This article develops an effective kernel-based nonlinear filter for a distributed sensor network, where "effective" reflects the centralized estimation accuracy. Among various kernel methods, this study adopts the kernel mean embedding (KME) method, the most commonly used and recognized scheme owing to its effectiveness. When approximating the posterior distribution, the nonlinearity of the measurement function poses a challenge. Therefore, we linearly approximate the nonlinear measurement function in the RKHS, which is still nonlinear in the state space. This strategy represents the nonlinear function as a linear operator in a higher dimensional space. Then, we provide the KME of posterior distribution in the RKHS, from which the posterior distribution's sample can also be obtained. For a distributed sensor network, the nodes update the KME of posterior distribution by exchanging information with each other until they reach a consensus. When the finite consensus is reached, the nodes reach an agreement in the same form as the result of a centralized filter. This study is unique in extending kernel-based filtering to distributed setting while ensuring centralized estimation accuracy simultaneously. The proposed filters are tested in various target-tracking scenarios.

The major contributions of this article are as follows.

1) We develop a novel consensus-based distributed nonlinear filter by approximating the KME of posterior distribution, filling the kernel-based filter gap for distributed sensor networks. In each time step, the nodes reach an agreement with finite consensus steps, specifically, less than the number of nodes.
2) We propose a novel kernel-based centralized nonlinear filter, which extends the centralized Kalman filter to the RKHS. Compared with the existing kernel-based filters, the proposed filter significantly improves the filtering efficiency, which is highly valuable for real-time applications.
3) We prove that the proposed two filters are equivalent, which indicates that the distributed filter can reach the centralized estimation accuracy while maintaining the distributed setting. We further demonstrate the filters' superiority compared with existing approaches through target tracking examples.

Discussions: The major contributions of this article compared with our published related work, i.e., [7], [8], [40], and [41], are discussed here. First,the works in [7], [8], [40], and [41] address nonlinear systems for single-sensor setup, while this article addresses nonlinear systems for multisensor setup. Specifically, we have developed nonlinear filters for both distributed and centralized sensor networks. Second, the work in [7] focuses on moment approximation, and the works in [8], [40], and [41] focus on Gaussian approximation of posterior distribution by divergence minimization of some kind, while this article focuses on posterior distribution approximation by KME, so that the posterior approximation is not limited to a particular form, such as the Gaussian distribution. Last but not least, if the sensor network consists of only one sensor (which degenerates to the single-sensor setup), then the proposed centralized algorithm in this article results in a nonlinear filter for single-sensor setup, which provides a more effective posterior approximation than those in [7], [8], [40], and [41] for problem with high nonlinearity.

The rest of this article is organized as follows. Section II introduces some preliminaries and formulates the problem investigated. Section III presents the developed distributed nonlinear filter, and Section IV introduces the proposed centralized nonlinear filter. Section V analyzes the performance of both filters, and Section VI presents two target tracking examples. Finally, Section VII concludes this article, and then some proofs and derivations are provided in the Appendix.

Notations: The scalars, vectors, and matrices are denoted by lowercase, bold lowercase, and lightface capital letters, respectively. For simplicity and convenience, scalar 0, zero vector, and zero matrix are all denoted by 0. $\mathbb{N}^+$ stands for the set of all positive integers. The sets of all p-dimensional real vectors and all $p \times q$ real matrices are denoted by $\mathbb{R}^p$ and $\mathbb{R}^{p\times q}$, respectively. For a vector $\mathbf{a}$, diag($\mathbf{a}$) represents a diagonal matrix with principal diagonal consisting of $\mathbf{a}$, $\|\mathbf{a}\|_q$ denotes ℓ_q norm of $\mathbf{a}$. When $q = 2$, it is the Euclidean norm abbreviated as $\|\mathbf{a}\|$. All vectors in this article are column vectors. For a matrix A, its Frobenius norm is denoted by $\|A\|$. I is an identity matrix with an appropriate dimension. block-diag$(A_1, \ldots, A_p)$ represents a block diagonal matrix with a main diagonal block consisting of matrices A_i $(i = 1, \ldots, p)$. The notation "$\otimes$" represents the Kronecker product operation between matrices (vectors). For a Hilbert space $\mathcal{H}$, its inner product is denoted by $\langle\cdot, \cdot\rangle_{\mathcal{H}}$; its norm induced by the inner product is defined by $\|\cdot\|_{\mathcal{H}} = \sqrt{\langle\cdot, \cdot\rangle_{\mathcal{H}}}$; and $\Phi^* : \mathcal{H} \to \mathbb{R}^p$ represents the adjoint operator of an operator $\Phi : \mathbb{R}^p \to \mathcal{H}$. $\mathcal{E}[\cdot]$ is the mathematical expectation. $\mathcal{N}(\bar{\mathbf{x}}, P)$ denotes the Gaussian distribution with mean $\bar{\mathbf{x}}$ and covariance matrix P.

Authorized licensed use limited to: Yunnan University. Downloaded on April 14,2025 at 13:45:04 UTC from IEEE Xplore. Restrictions apply.

II. PRELIMINARIES AND PROBLEM FORMULATION

A. Distribution Embedding

An RKHS, denoted by $\mathcal{H}$, on Ω is a Hilbert space of functions $g : \Omega \to \mathbb{R}$ equipped with a kernel function $k : \Omega \times \Omega \to \mathbb{R}$ satisfying the following reproducing property:

$$\langle g, k(\cdot, \mathbf{x})\rangle_{\mathcal{H}} = g(\mathbf{x}) \quad \forall g \in \mathcal{H}$$

and thus

$$\langle k(\cdot, \mathbf{x}), k(\cdot, \mathbf{x}')\rangle_{\mathcal{H}} = k(\mathbf{x}', \mathbf{x}) = k(\mathbf{x}, \mathbf{x}').$$

The kernel function can also be treated as an implicit feature map $\phi : \Omega \to \mathcal{H}$ satisfying

$$\langle \phi(\mathbf{x}), \phi(\mathbf{x}')\rangle_{\mathcal{H}} = k(\mathbf{x}', \mathbf{x}).$$

Some commonly used kernel functions on $\mathbb{R}^p$ are listed below for later use:

$$k(\mathbf{x}, \mathbf{x}') = \left(\mathbf{x}^{\mathrm{T}}\mathbf{x}' + c\right)^d, \text{ (polynomial kernel)}$$

$$k(\mathbf{x}, \mathbf{x}') = \exp\left(-\frac{\|\mathbf{x} - \mathbf{x}'\|^2}{\sigma}\right), \text{ (Gaussian kernel)} \quad (1)$$

$$k(\mathbf{x}, \mathbf{x}') = \exp\left(-\frac{\|\mathbf{x} - \mathbf{x}'\|_1}{\sigma}\right), \text{ (Laplace kernel)} \quad (2)$$

where $c, \sigma > 0$ and $d \in \mathbb{N}^+$.

The KME of a distribution with probability density function $p(\mathbf{x})$ is an element in the RKHS defined by

$$\mu_{\mathbf{x}} := \mathcal{E}[\phi(\mathbf{x})] = \int_{\Omega} \phi(\mathbf{x})p(\mathbf{x})\mathrm{d}\mathbf{x}$$

and has the following property:

$$\langle \mu_{\mathbf{x}}, g\rangle_{\mathcal{H}} = \mathcal{E}[g(\mathbf{x})] \quad \forall g \in \mathcal{H}.$$

Given a weighted sample $\{(\mathbf{x}_l, w_l)\}_{l=1}^m$ of $p(\mathbf{x})$, the KME can be approximated as

$$\widehat{\mu}_{\mathbf{x}} = \sum_{l=1}^{m} w_l \phi(\mathbf{x}_l).$$

For two separable Hilbert spaces $\mathcal{F}$ and $\mathcal{G}$ with $\{e_i\}_{i\in I}$ and $\{f_j\}_{j\in J}$ being their orthonormal bases, respectively, where the index sets I and J are either finite or countable, the Hilbert–Schmidt norm of a compact linear operator $L : \mathcal{G} \to \mathcal{F}$ is defined as

$$\|L\|_{\mathrm{HS}} = \sum_{j\in J} \|Lf_j\|_{\mathcal{F}}^2 = \sum_{i\in I}\sum_{j\in J} |\langle Lf_j, e_i\rangle_{\mathcal{F}}|^2. \quad (3)$$

We say that the operator L is Hilbert–Schmidt when (3) is finite. For more details, the reader is referred to [34].

B. Problem Formulation

We consider the sensor network with communication topology modeled by an undirected graph $\mathcal{G} = (\mathcal{V}, \mathcal{E}, A)$, where $\mathcal{V} = \{1, 2, \ldots, n\}$, $\mathcal{E}$, and A stand for the node set comprising n sensors, an edge set consisting of communication links between the sensors, and a weighted adjacency matrix, respectively. For $i, j \in \mathcal{V}, A = [a_{ij}]$ is assumed to be double stochastic and satisfies $a_{ii} > 0, a_{ij} \geq 0, \sum_{j\in\mathcal{V}} a_{ij} = 1$, and $\sum_{i\in\mathcal{V}} a_{ij} = 1$. The neighboring sensors of Node i form a set $\mathcal{N}_i = \{j \in \mathcal{V} | a_{ij} > 0\}$ and $i \in \mathcal{N}_i$, from which Node i receives information.

ASSUMPTION 1 $\mathcal{G}$ is undirected and connected.

Assumption 1 is standard and commonly used (see, e.g., [17], [18], and [19]). If Assumption 1 satisfies, then for any pair of nodes (i, j), there is a path made up of edges between them.

For each node $i \in \mathcal{V}$, the following nonlinear dynamic system is considered:

$$\begin{aligned} \mathbf{x}_k &= f_{k-1}(\mathbf{x}_{k-1}, \boldsymbol{\omega}_k) \\ \mathbf{y}_{i,k} &= h_{i,k}(\mathbf{x}_k) + \mathbf{v}_{i,k} \end{aligned} \quad (4)$$

where $\mathbf{x}_k \in \mathbb{R}^{n_x}$ and $\mathbf{y}_{i,k} \in \mathbb{R}^{n_{y_i}}$ are the state and the node i's measurement, respectively. Besides, $k = 1, 2, \ldots$ is the time index, $f_k : \mathbb{R}^{n_x} \times \mathbb{R}^{n_\omega} \to \mathbb{R}^{n_x}$ and $h_{i,k} : \mathbb{R}^{n_x} \to \mathbb{R}^{n_{y_i}}$ are known vector-valued transition and measurement functions and typically nonlinear, and the process noise $\{\boldsymbol{\omega}_k\}$ and measurement noise $\{\mathbf{v}_{i,k}\}$ are mutually independent zero-mean white noise sequences whose joint covariance matrix is given by

$$\mathcal{E}\left\{\begin{bmatrix}\boldsymbol{\omega}_k \\ \mathbf{v}_{i,k}\end{bmatrix}\begin{bmatrix}\boldsymbol{\omega}_{k'}^{\mathrm{T}} & \mathbf{v}_{j,k'}^{\mathrm{T}}\end{bmatrix}\right\} = \begin{bmatrix}Q_k & 0 \\ 0 & R_{i,k}\delta_{i,j}\end{bmatrix}\delta_{k,k'}$$

where $Q_k \succeq 0$, $R_{i,k} \succ 0$, and $\delta_{i,j}$ is the Kronecker delta function. The initial state with mean $\bar{\mathbf{x}}_0$ and covariance matrix P_0 is independent of the noise. Each node cannot acquire the other nodes' measurement functions and noise information in the distributed setting. In addition, each node utilizes its measurement data and communicates with its neighbors to approximate the state posterior distribution as accurately as possible and to reach a consensus with the other nodes.

It is well known that the optimal solution to such a filtering problem for the sensor network with each node under system (4) is given by the centralized Bayesian filtering, e.g., [11], consisting of prediction and update steps

$$\begin{aligned} p(\mathbf{x}_k|\mathbb{Y}_{k-1}) &= \int_{\mathbb{R}^{n_x}} p(\mathbf{x}_k|\mathbf{x}_{k-1})p(\mathbf{x}_{k-1}|\mathbb{Y}_{k-1})\mathrm{d}\mathbf{x}_{k-1} \\ p(\mathbf{x}_k|\mathbb{Y}_k) &= c \cdot p(\mathbf{y}_k|\mathbf{x}_k)p(\mathbf{x}_k|\mathbb{Y}_{k-1}) \end{aligned}$$

where $\mathbb{Y}_k = \{\mathbf{y}_1, \ldots, \mathbf{y}_k\}$, $\mathbf{y}_k = [\mathbf{y}_{1,k}^{\mathrm{T}}, \ldots, \mathbf{y}_{n,k}^{\mathrm{T}}]^{\mathrm{T}}$, and c is a normalization constant. The centralized Bayesian filtering is elegant in theory but challenging to implement for the following reasons.

1) The closed-form expressions for the prediction and posterior probability density functions, i.e., $p(\mathbf{x}_k|\mathbb{Y}_{k-1})$ and $p(\mathbf{x}_k|\mathbb{Y}_k)$, are generally difficult or even impossible to obtain due to the nonlinearity of the transition and measurement functions.
2) The centralized setting is often burdened with a computationally demanding task and susceptible to processing unit failures. In addition, it requires each

Authorized licensed use limited to: Yunnan University. Downloaded on April 14,2025 at 13:45:04 UTC from IEEE Xplore. Restrictions apply.

sensor to send its measurement to the fusion center promptly.

Spurred by these deficiencies, this article develops a nonlinear filter that achieves the following two goals simultaneously.

1) *Distributed setting:* It should be designed for a distributed sensor network.
2) *Centralized accuracy:* It should have centralized estimation accuracy, i.e., it should effectively approximate the centralized Bayesian filtering.

Next, we resort to the kernel method (see, e.g., [31], [32], [33], [34], [35], and [36]) to approximate the posterior distribution and achieve the above two goals with the help of KME.

III. PROPOSED DISTRIBUTED NONLINEAR FILTERING

This section provides the distributed filtering comprising prediction and update steps, where the KMEs of prediction and posterior distributions are given.

A. Prediction

For each node $i \in \mathcal{V}$, we denote the sample of the posterior distribution at time $k-1$ by

$$X_{i,k-1|k-1} = \left[\mathbf{x}_{i,1}^{-}, \ldots, \mathbf{x}_{i,m}^{-}\right].$$

The corresponding weight vector is

$$\mathbf{w}_{i,k-1} = \left[w_{i,k-1,1}, \ldots, w_{i,k-1,m}\right]^{\mathrm{T}} \tag{5}$$

where $\mathbf{1}^{\mathrm{T}}\mathbf{w}_{i,k-1} = 1$, and each $w_{i,k-1,l} \geq \epsilon$ for $l = 1, \ldots, m$ with ϵ being a small positive real number. Then, the sample of the prediction distribution, denoted by $X_{i,k|k-1}$, is generated as follows:

$$\begin{aligned} X_{i,k|k-1} &= \left[\mathbf{x}_{i,1}^{+}, \ldots, \mathbf{x}_{i,m}^{+}\right] \\ \mathbf{x}_{i,l}^{+} &\sim p(\mathbf{x}_k|\mathbf{x}_{i,l}^{-}),\ l = 1, \ldots, m \end{aligned} \tag{6}$$

where $p(\mathbf{x}_k|\mathbf{x}_{i,l}^{-})$ is the probability density function of the random vector $f_{k-1}(\mathbf{x}_{i,l}^{-}, \boldsymbol{\omega}_k)$. Specifically, by generating m sample points from the distribution of process noise $\boldsymbol{\omega}_k$ and propagating them through nonlinear functions $f_{k-1}(\mathbf{x}_{i,l}^{-}, \cdot) : \mathbb{R}^{n_\omega} \to \mathbb{R}^{n_x}$ for $l = 1, \ldots, m$, we obtain $\{\mathbf{x}_{i,1}^{+}, \ldots, \mathbf{x}_{i,m}^{+}\}$. The sampling methods can be any of the following.

1) Deterministic sampling, such as the quasi-Monte Carlo method [42] and the importance Gaussian quadrature [43].
2) Stochastic sampling, such as the Monte Carlo method [44], where all nodes use the same random number generator with the same initial random number seed.

The commonality of the above two sampling methods lies in their fixed sample points for a given distribution. In other words, if all nodes have the same posterior sample $X_{i,k-1|k-1}$, then they also have the same predicted sample $X_{i,k|k-1}$. Note that the weight vector corresponding to $X_{i,k|k-1}$ is also $\mathbf{w}_{i,k-1}$ given by (5), which indicates that the prediction step updates the sample points' positions to fit the prediction distribution while keeping their weights unchanged.

To obtain the KME of prediction distribution, denoted by $\mu_{i,k|k-1}$, we consider embedding the state into an RKHS $\mathcal{H}$ of functions with a feature map ϕ (time-invariant) defined by a positive-definite kernel function $k : \mathbb{R}^{n_x} \times \mathbb{R}^{n_x} \to \mathbb{R}$, i.e.,

$$\phi(\mathbf{x}) = k(\cdot, \mathbf{x}).$$

Then, an estimate for the KME of prediction distribution is given by

$$\mu_{i,k|k-1} = \sum_{l=1}^{m} w_{i,k-1,l}\phi(\mathbf{x}_{i,l}^{+}) = \Phi\mathbf{w}_{i,k-1} \tag{7}$$

where

$$\Phi = \left[\phi(\mathbf{x}_{i,1}^{+}), \ldots, \phi(\mathbf{x}_{i,m}^{+})\right].$$

B. Update

The main difficulty in the update step is dealing with nonlinear measurement functions. For each node $i \in \mathcal{V}$, we care about the nonlinearity of $h_{i,k}(\cdot)$ only in the range of the state. Thus, we consider representing the measurement functions as inner products in the RKHS [36]. Specifically, $h_{i,k}(\cdot)$ can be represented in the following form:

$$\begin{aligned} h_{i,k}(\cdot) &= \left[h_{i,k}^{(1)}(\cdot), \ldots, h_{i,k}^{(n_{y_i})}(\cdot)\right]^{\mathrm{T}} \\ &\triangleq \left[\left\langle\lambda_{i,k}^{(1)}, \phi(\cdot)\right\rangle_{\mathcal{H}}, \ldots, \left\langle\lambda_{i,k}^{(n_{y_i})}, \phi(\cdot)\right\rangle_{\mathcal{H}}\right]^{\mathrm{T}} \\ &\triangleq \Lambda_{i,k}^{*}\phi(\cdot) \end{aligned}$$

where

$$\Lambda_{i,k} = \left[\lambda_{i,k}^{(1)}, \ldots, \lambda_{i,k}^{(n_{y_i})}\right] \tag{8}$$

is a Hilbert–Schmidt linear operator mapping from $\mathbb{R}^{n_{y_i}}$ to $\mathcal{H}$ with $\lambda_{i,k}^{(1)}, \ldots, \lambda_{i,k}^{(n_{y_i})} \in \mathcal{H}$ to be determined.

REMARK 1 The nonlinear function $h_{i,k}(\cdot)$ can be approximated up to arbitrary accuracy by some large RKHSs (see [45, Sect. 4.6]). For example, if $k(\cdot, \cdot)$ is a Gaussian kernel, then for all $\varepsilon > 0$, $q \in [1, \infty)$, there exists $\widetilde{\Lambda}_{i,k}$ such that $\widetilde{h}_{i,k}(\cdot) = \widetilde{\Lambda}_{i,k}^{*}\phi(\cdot)$ satisfying

$$\begin{aligned} &\left(\int_{\mathbb{R}^{n_x}} \left\|h_{i,k}(\mathbf{x}_k) - \widetilde{h}_{i,k}(\mathbf{x}_k)\right\|_q^q \cdot p(\mathbf{x}_k|\mathbb{Y}_{k-1})\mathrm{d}\mathbf{x}_k\right)^{\frac{1}{q}} < \varepsilon \\ &\Leftrightarrow \int_{\mathbb{R}^{n_x}} \left\|h_{i,k}(\mathbf{x}_k) - \widetilde{h}_{i,k}(\mathbf{x}_k)\right\|_q^q \cdot p(\mathbf{x}_k|\mathbb{Y}_{k-1})\mathrm{d}\mathbf{x}_k < \delta \end{aligned} \tag{9}$$

where $\delta \geq \varepsilon^q > 0$.

Let

$$Y_{i,k} = \left[h_{i,k}(\mathbf{x}_{i,1}^{+}), \ldots, h_{i,k}(\mathbf{x}_{i,m}^{+})\right]. \tag{10}$$

Authorized licensed use limited to: Yunnan University. Downloaded on April 14,2025 at 13:45:04 UTC from IEEE Xplore. Restrictions apply.

Then, an estimate of $\Lambda_{i,k}$ can be determined by solving the following minimization problem:

$$\widehat{\Lambda}_{i,k} = \arg\min_{\Lambda} \sum_{l=1}^{m} w_{i,k-1,l} \left\| h_{i,k}(\mathbf{x}_{i,l}^{+}) - \Lambda^{*}\phi(\mathbf{x}_{i,l}^{+}) \right\|^2. \tag{11}$$

The objective function of (11) is essentially an estimate of the second integral in (9) by taking $q = 2$. Note that (11) is a convex optimization problem whose optimal solution can be analytically obtained.

THEOREM 1 The solution to (11) is given by

$$\widehat{\Lambda}_{i,k} = \Phi K^{-1} Y_{i,k}^{\mathrm{T}}$$

where K is the Gram matrix (see [45, p. 117]), given by

$$K = \begin{bmatrix} k(\mathbf{x}_{i,1}^{+}, \mathbf{x}_{i,1}^{+}) & \dots & k(\mathbf{x}_{i,1}^{+}, \mathbf{x}_{i,m}^{+}) \\ \vdots & \ddots & \vdots \\ k(\mathbf{x}_{i,m}^{+}, \mathbf{x}_{i,1}^{+}) & \dots & k(\mathbf{x}_{i,m}^{+}, \mathbf{x}_{i,m}^{+}) \end{bmatrix}.$$

PROOF See Appendix A. □

REMARK 2 Theorem 1 indicates that the minimizer of (11) is independent of the weights $w_{i,k-1,1}, \dots, w_{i,k-1,m}$. This is because $h_{i,k}(\cdot)$ is fitted exactly by $\widehat{\Lambda}_{i,k}^{*}\phi(\cdot)$ in the evaluated points $\mathbf{x}_{i,1}^{+}, \dots, \mathbf{x}_{i,m}^{+}$.

REMARK 3 If the kernel function $k(\cdot,\cdot)$ is strictly positive definite, then the Gram matrix K is positive definite. In practice, K^{-1} can be replaced by $(K + \sigma I)^{-1}$ with $\sigma > 0$ being a small real number for numerical stability.

Based on Theorem 1, the measurement equation can be linearly converted into

$$\mathbf{y}_{i,k} = \widehat{\Lambda}_{i,k}^{*}\phi(\mathbf{x}_k) + \mathbf{v}_{i,k}.$$

It is worth emphasizing that $\mathbf{y}_{i,k}$ is linear with respect to $\phi(\mathbf{x}_k)$ since $\Lambda_{i,k}$ is a linear operator, but still nonlinear with respect to $\mathbf{x}_k$ since ϕ is a nonlinear map. This means that the filtering problem is simplified significantly in the RKHS while the original nonlinearity is still maintained. To obtain the KME of posterior distribution, denoted by $\mu_{i,k|k}$, we design the update rule as follows:

$$\begin{aligned} \mu_{i,k|k} &= \mu_{i,k|k-1} + \Phi W_{i,k-1}^{\frac{1}{2}} \Gamma_{i,k}^{-1} \boldsymbol{\xi}_{i,k} \\ W_{i,k-1} &= \mathrm{diag}(\mathbf{w}_{i,k-1}) - \mathbf{w}_{i,k-1}\mathbf{w}_{i,k-1}^{\mathrm{T}} \end{aligned} \tag{12}$$

where the matrix $\Gamma_{i,k} \in \mathbb{R}^{m\times m}$ and the vector $\boldsymbol{\xi}_{i,k} \in \mathbb{R}^{m}$ are to be determined by the following consensus step.

REMARK 4 W_i is positive semidefinite. Specifically, for all $\mathbf{z} = [z_1, \dots, z_m]^{\mathrm{T}} \in \mathbb{R}^m$, we have

$$\begin{aligned} \mathbf{z}^{\mathrm{T}} W_{i,k-1}\mathbf{z} &= \mathbf{z}^{\mathrm{T}}\mathrm{diag}(\mathbf{w}_{i,k-1})\mathbf{z} - \mathbf{z}^{\mathrm{T}}\mathbf{w}_{i,k-1}\mathbf{w}_{i,k-1}^{\mathrm{T}}\mathbf{z} \\ &= \sum_{l=1}^{m} w_{i,k-1,l} z_l^2 - \left(\sum_{l=1}^{m} w_{i,k-1,l} z_l \right)^2 \\ &\geq 0 \end{aligned}$$

where the inequality holds based on Jensen's inequality (see [46, p. 246]).

Algorithm 1: Distributed Consensus Algorithm for $\Gamma_{i,k}$ and $\boldsymbol{\xi}_{i,k}$.

Input: $\Gamma_{j,k}^{(0)}$ and $\boldsymbol{\xi}_{j,k}^{(0)}$, for all $j \in \mathcal{N}_i$

1: Set $r = 0$.
2: Set $\Gamma_{i,k}^{*} = 0$, $\boldsymbol{\xi}_{i,k}^{*} = 0$.
3: **repeat**
4: Implement

$$\Gamma_{i,k}^{(r+1)} = \Gamma_{i,k}^{(r)} + \sum_{j\in\mathcal{N}_i} a_{ij}\left(\Gamma_{j,k}^{(r)} - \Gamma_{i,k}^{(r)}\right), \tag{13}$$

$$\boldsymbol{\xi}_{i,k}^{(r+1)} = \boldsymbol{\xi}_{i,k}^{(r)} + \sum_{j\in\mathcal{N}_i} a_{ij}\left(\boldsymbol{\xi}_{j,k}^{(r)} - \boldsymbol{\xi}_{i,k}^{(r)}\right),$$

$$\Gamma_{i,k}^{*} = \Gamma_{i,k}^{*} + s_{d+1-r}\Gamma_{i,k}^{(r)},$$

$$\boldsymbol{\xi}_{i,k}^{*} = \boldsymbol{\xi}_{i,k}^{*} + s_{d+1-r}\boldsymbol{\xi}_{i,k}^{(r)}. \tag{14}$$

5: Set $r = r + 1$.
6: **until** $r = d$.
7: Let

$$\Gamma_{i,k}^{*} = (\mathbf{1}^{\mathrm{T}}\mathbf{s})^{-1}\Gamma_{i,k}^{*},$$

$$\boldsymbol{\xi}_{i,k}^{*} = (\mathbf{1}^{\mathrm{T}}\mathbf{s})^{-1}\boldsymbol{\xi}_{i,k}^{*}.$$

Output: $\Gamma_{i,k}^{*}$, $\boldsymbol{\xi}_{i,k}^{*}$

To reach a consensus within finite number of steps, we resort to the concept of the minimal polynomial of a matrix [47]. Suppose that the minimal polynomial of the weighted adjacency matrix A has degree $d+1$ ($d+1 \leq n$) and is given as

$$q(t) = t^{d+1} + \alpha_d t^d + \cdots + \alpha_1 t + \alpha_0$$

where $\alpha_0, \alpha_1, \dots, \alpha_d$ are constants. Let

$$\begin{aligned} \mathbf{s} &= [s_1, s_2, \dots, s_{d+1}]^{\mathrm{T}} \\ &= \left[1, 1+\alpha_d, 1+\alpha_{d-1}+\alpha_d, \dots, 1+\sum_{j=1}^{d}\alpha_j\right]^{\mathrm{T}}. \end{aligned}$$

Then, the distributed consensus algorithm for $\Gamma_{i,k}$ and $\boldsymbol{\xi}_{i,k}$ is summarized in Algorithm 1, where the initialization is designed as follows:

$$\begin{aligned} \Gamma_{i,k}^{(0)} &= nW_{i,k-1}^{\frac{1}{2}} Y_{i,k}^{\mathrm{T}} R_{i,k}^{-1} Y_{i,k} W_{i,k-1}^{\frac{1}{2}} + I \\ \boldsymbol{\xi}_{i,k}^{(0)} &= nW_{i,k-1}^{\frac{1}{2}} Y_{i,k}^{\mathrm{T}} R_{i,k}^{-1} \left(\mathbf{y}_{i,k} - \widehat{\Lambda}_{i,k}^{*}\mu_{i,k|k-1}\right). \end{aligned}$$

It is known from [47] that if Assumption 1 holds, then for all $i \in \mathcal{V}$, the following two equations hold:

$$\begin{aligned} \Gamma_{i,k}^{*} &= \frac{1}{n}\sum_{j\in\mathcal{V}} \Gamma_{j,k}^{(0)} \\ &= \frac{1}{n}\sum_{j\in\mathcal{V}} \left(nW_{j,k-1}^{\frac{1}{2}} Y_{j,k}^{\mathrm{T}} R_{j,k}^{-1} Y_{j,k} W_{j,k-1}^{\frac{1}{2}} + I\right) \\ \boldsymbol{\xi}_{i,k}^{*} &= \frac{1}{n}\sum_{j\in\mathcal{V}} \boldsymbol{\xi}_{j,k}^{(0)} \end{aligned} \tag{15}$$

Authorized licensed use limited to: Yunnan University. Downloaded on April 14,2025 at 13:45:04 UTC from IEEE Xplore. Restrictions apply.

$$= \frac{1}{n}\sum_{j\in\mathcal{V}} nW_{j,k-1}^{\frac{1}{2}} Y_{j,k}^{\mathrm{T}} R_{j,k}^{-1}\left(\mathbf{y}_{j,k} - \widehat{\Lambda}_{j,k}^{*}\mu_{j,k|k-1}\right). \quad (16)$$

Note that the right-hand sides of (15) and (16) are independent of index i. Therefore, we have

$$\Gamma_{1,k}^{*} = \Gamma_{2,k}^{*} = \cdots = \Gamma_{n,k}^{*}$$
$$\boldsymbol{\xi}_{1,k}^{*} = \boldsymbol{\xi}_{2,k}^{*} = \cdots = \boldsymbol{\xi}_{n,k}^{*}$$

which indicates that all the nodes have reached an agreement. Moreover, from Step 6 in Algorithm 1, the consensus step is d with $d+1 \leq n$. As a result, all the nodes can reach a consensus with no more than $n-1$ consensus steps.

REMARK 5 The computational complexity of Algorithm 1 is $\mathcal{O}((n-1)m^2 n_i)$, where n_i is the number of neighbors of Node i.

C. Proposed Distributed Filtering Algorithm

To obtain the state estimate $\mathbf{x}_{i,k|k}$, and its error covariance matrix $P_{i,k|k}$, we first provide the sample for posterior distribution from its KME given by (12). Note that (12) can be rewritten as follows:

$$\mu_{i,k|k} = \Phi\left(\mathbf{w}_{i,k-1} + W_{i,k-1}^{\frac{1}{2}}\Gamma_{i,k}^{-1}\boldsymbol{\xi}_{i,k}\right).$$

Let

$$\widetilde{\mathbf{w}}_{i,k} = \mathbf{w}_{i,k-1} + W_{i,k-1}^{\frac{1}{2}}\Gamma_{i,k}^{-1}\boldsymbol{\xi}_{i,k}. \quad (17)$$

Then, we show that $\mu_{i,k|k}$ is actually a weighted combination of each column of Φ.

THEOREM 2 $\widetilde{\mathbf{w}}_{i,k}$ is a weight vector satisfying $\mathbf{1}^{\mathrm{T}}\widetilde{\mathbf{w}}_{i,k} = 1$.

PROOF See Appendix B. □

Next, we obtain the sample of the posterior distribution

$$X_{i,k|k} = \left[\mathbf{x}_{i,1}^{+}, \ldots, \mathbf{x}_{i,m}^{+}\right]$$

with the corresponding weight vector $\widetilde{\mathbf{w}}_{i,k}$ given by (17). Since the elements of $\widetilde{\mathbf{w}}_{i,k}$ may be negative due to the randomness of $\mathbf{y}_{i,k}$, we normalize the weight vector $\widetilde{\mathbf{w}}_{i,k}$ by minimizing the empirical maximum mean discrepancy

$$\begin{aligned}\min_{\mathbf{w}_{i,k}} \ & \left\|\Phi\widetilde{\mathbf{w}}_{i,k} - \Phi\mathbf{w}_{i,k}\right\|_{\mathcal{H}}^{2} \\ \text{s.t. } & \mathbf{1}^{\mathrm{T}}\mathbf{w}_{i,k} = 1 \\ & w_{i,k,l} \geq \epsilon,\ l = 1, \ldots, m \end{aligned} \quad (18)$$

where $\mathbf{w}_{i,k} = [w_{i,k,1}, \ldots, w_{i,k,m}]^{\mathrm{T}}$, and ϵ is a small positive real number. Based on the fact that

$$\begin{aligned}\left\|\Phi\widetilde{\mathbf{w}}_{i,k} - \Phi\mathbf{w}_{i,k}\right\|_{\mathcal{H}}^{2} &= (\mathbf{w}_{i,k} - \widetilde{\mathbf{w}}_{i,k})^{\mathrm{T}}\Phi^{*}\Phi(\mathbf{w}_{i,k} - \widetilde{\mathbf{w}}_{i,k}) \\ &= (\mathbf{w}_{i,k} - \widetilde{\mathbf{w}}_{i,k})^{\mathrm{T}}K(\mathbf{w}_{i,k} - \widetilde{\mathbf{w}}_{i,k})\end{aligned}$$

we know that (18) is a convex quadratic programming and can be solved efficiently by many classical numerical methods, such as the interior-point methods [48].

REMARK 6 In contrast to the L_2 distance between two kernel density estimates, the maximum mean discrepancy between two embeddings has more power against local departures from the null hypothesis for a two-sample test [33], [34].

Algorithm 2: Distributed Nonlinear Filtering Algorithm.

Input: $X_{i,k-1,k-1}$, $\mathbf{w}_{i,k-1}$
I. Prediction:
1: Calculate $X_{i,k|k-1}$ using (6).
II. Update:
2: Set $X_{i,k|k} = X_{i,k|k-1}$.
3: Implement Algorithm 1 to get $\Gamma_{i,k}^{*}$, $\boldsymbol{\xi}_{i,k}^{*}$.
4: Calculate $\widetilde{\mathbf{w}}_{i,k}$ using (17).
5: Calculate $\mathbf{w}_{i,k}$ by solving (18).
6: Calculate $\mathbf{x}_{i,k|k}$ and $P_{i,k|k}$ using (19) and (20).
Output: $X_{i,k|k}$, $\mathbf{w}_{i,k}$, $\mathbf{x}_{i,k|k}$, $P_{i,k|k}$

Then, we consider the so-called preimage problem, i.e., recover the state estimation in the state space [36]. At time k, based on the sample of the posterior distribution, we obtain the concerned moment approximations. In the minimum mean-square-error sense, the optimal state estimate and its error covariance matrix are given by the mean and covariance matrix of the posterior distribution, respectively, which are calculated as follows:

$$\mathbf{x}_{i,k|k} = \sum_{l=1}^{m} w_{i,k,l}\mathbf{x}_{i,l}^{+} \quad (19)$$

$$P_{i,k|k} = \sum_{l=1}^{m} w_{i,k,l}(\mathbf{x}_{i,l}^{+} - \mathbf{x}_{i,k|k})(\mathbf{x}_{i,l}^{+} - \mathbf{x}_{i,k|k})^{\mathrm{T}}. \quad (20)$$

For $k = 1, 2, \ldots,$ each node $i \in \mathcal{V}$ implements the proposed distributed filtering algorithm, which is summarized in Algorithm 2.

REMARK 7 Generally, the preimage problem is challenging, as it is difficult to get the distribution sample from its KME directly. Benefiting from the update rule we designed in (12), the corresponding preimage problem becomes rather simple.

REMARK 8 The update step changes the weights of the sample points of the prediction distribution while keeping the positions unchanged. This is similar to the celebrated particle filter, where the positions of the particles are changed to fit the prediction distribution in the prediction step, and the weights are updated to fit the posterior distribution in the update step.

D. Discussions

Some discussions about the estimators' consistency, the network's communication, and the algorithm's implementation are as follows.

1) Consistency: In the prediction and update steps, we use m sample points to calculate the KME of prediction distribution by (7) and the KME of posterior distribution by (12), respectively. It can be shown that the estimators (7) and (12) are both consistent. Specifically, similarly to the analysis in [32], the estimation errors converge to zero in

Authorized licensed use limited to: Yunnan University. Downloaded on April 14,2025 at 13:45:04 UTC from IEEE Xplore. Restrictions apply.

the RKHS norm (i.e., the maximum mean discrepancy) at an overall rate of $\mathcal{O}(m^{-\frac{1}{2}})$.

2) Communication interference: Note that the proposed distributed algorithm considers ideal communication links, where "ideal" means no disturbed noise. If they are disturbed by a noise sequence at time k, denoted by $\epsilon_{1,k}, \ldots, \epsilon_{d,k}$, then each element of $\Gamma^*_{i,k}$ and $\boldsymbol{\xi}^*_{i,k}$ in Algorithm 1 will be disturbed by $(\mathbf{1}^{\mathrm{T}}\mathbf{s})^{-1}\sum_{j=0}^{d}\sum_{i=1}^{j}A^i\epsilon_{j-i,k}s_{d+1-j}$. Specifically, let $\gamma_{j,k}$ be an element of $\Gamma_{j,k}$ (or $\boldsymbol{\xi}_{i,k}$) for $j = 1, \ldots, n$, and $\boldsymbol{\gamma}_k = [\gamma_{1,k}, \ldots, \gamma_{n,k}]^{\mathrm{T}}$. Then, we have $\boldsymbol{\gamma}_k^{(r)} = A\boldsymbol{\gamma}_k^{(r-1)}$ for $r = 1, 2, \ldots, d$. If the communication links are noisy by $\epsilon_{r,k}$ at the rth consensus step, then we can obtain $\widetilde{\boldsymbol{\gamma}}_k^{(r)} = \boldsymbol{\gamma}_k^{(r)} + \sum_{i=1}^{r}A^i\epsilon_{r-i,k}$. Furthermore, we have

$$\begin{aligned}
\widetilde{\boldsymbol{\gamma}}_k^* &= (\mathbf{1}^{\mathrm{T}}\mathbf{s})^{-1}\sum_{j=0}^{d}\widetilde{\boldsymbol{\gamma}}_k^{(j)}s_{d+1-j} \\
&= (\mathbf{1}^{\mathrm{T}}\mathbf{s})^{-1}\sum_{j=0}^{d}\left(\boldsymbol{\gamma}_k^{(j)} + \sum_{i=1}^{j}A^i\epsilon_{j-i,k}\right)s_{d+1-j} \\
&= (\mathbf{1}^{\mathrm{T}}\mathbf{s})^{-1}\Bigg(\sum_{j=0}^{d}\boldsymbol{\gamma}_k^{(j)}s_{d+1-j} \\
&\quad + \sum_{j=0}^{d}\sum_{i=1}^{j}A^i\epsilon_{j-i,k}s_{d+1-j}\Bigg) \\
&= \boldsymbol{\gamma}_k^* + (\mathbf{1}^{\mathrm{T}}\mathbf{s})^{-1}\sum_{j=0}^{d}\sum_{i=1}^{j}A^i\epsilon_{j-i,k}s_{d+1-j}.
\end{aligned}$$

It is important to save the communication cost in a distributed filtering algorithm. From Algorithm 1, we know that at each time k, for $i \in \mathcal{V}$, $j \in \mathcal{N}_i$, an $m \times m$ matrix $\Gamma_{j,k}$ and an m-dimensional vector $\boldsymbol{\xi}_{j,k}$ should be transmitted from the jth sensor to the ith sensor. If the sample size m is large, so is the communication cost. In contrast, transmitting an $n_{y_i} \times m$ matrix $Y_{j,k}$, an $n_{y_i} \times n_{y_i}$ matrix $R_{j,k}$, and an n_{y_i}-dimensional vector $\mathbf{y}_{j,k}$ is more efficient.

3) Communication delay: When the communication delay exists, we consider two different cases.

Case 1: The delay is shorter than the sensor's sampling time. Inspired by the studies in [49] and [50], we can equip each sensor with a buffer that records data from neighboring nodes and their corresponding timestamps. Then, the $(r+1)$th consensus step of Algorithm 1 is carried out only after $\{\Gamma_{j,k}^{(r)}\}_{j\in\mathcal{N}_i}$ and $\{\boldsymbol{\xi}_{j,k}^{(r)}\}_{j\in\mathcal{N}_i}$ have been collected. As a result, such a buffering scheme allows the proposed distributed filtering to function normally under this case.

Case 2: The delay exceeds the sensor's sampling time. Under this case, the buffering scheme for the sensor can still be implemented. However, the proposed distributed filtering is no longer real time as the data collection process introduces a delay. To achieve real-time filtering, we may design a tailored algorithm that first processes the nondelayed data to obtain a filtering result. Once the delayed data arrive, the filtering result could be updated using the delayed data to further improve filtering accuracy. This case could be a potential topic for future research.

4) Implementation: When implementing Algorithm 2, the weights of the sample of the posterior distribution may be assigned to a few points and thus make many points useless. Technically, this phenomenon can be avoided by repeatedly drawing randomly from a discrete approximation of the posterior probability density function

$$p(\mathbf{x}_k|\mathbb{Y}_k) \approx \sum_{l=1}^{m} w_{i,k,l}\delta(\mathbf{x}_k - \boldsymbol{\beta}_l)$$

where $\delta(\cdot)$ is the Dirac delta function. This process creates a uniformly weighted, independent, identically distributed sample. Although the weights still accumulate on the points with large probabilities, in the subsequent prediction step, the same posterior sample point will correspond to different predicted sample points resulting from the process noise.

For the proposed distributed filtering algorithm, one of our goals, i.e., "distributed setting," is achieved. In the following two sections, it is demonstrated that another goal, "centralized accuracy," is also achieved. Specifically, we propose a centralized filtering algorithm superior to some widely used approaches and then prove that the proposed distributed filtering algorithm is equivalent to the centralized one.

IV. PROPOSED CENTRALIZED NONLINEAR FILTERING

This section introduces the centralized filtering comprising prediction and update steps, where the KMEs of the prediction and posterior distributions are obtained, respectively.

1) *Prediction:* Denote the sample of the posterior distribution at time $k-1$ by

$$X_{k-1|k-1} = \left[\mathbf{x}_1^-, \ldots, \mathbf{x}_m^-\right]$$

with corresponding weight vector

$$\mathbf{w}_{k-1} = \left[w_{k-1,1}, \ldots, w_{k-1,m}\right]^{\mathrm{T}} \tag{21}$$

where $\mathbf{1}^{\mathrm{T}}\mathbf{w}_{k-1} = 1$, and each $w_{k-1,l} \geq 0$ for $l = 1, \ldots, m$. Then, the sample of the prediction distribution, denoted by $X_{k|k-1}$, is generated as follows:

$$\begin{aligned}
X_{k|k-1} &= \left[\mathbf{x}_1^+, \ldots, \mathbf{x}_m^+\right] \\
\mathbf{x}_l^+ &\sim p(\mathbf{x}_k|\mathbf{x}_l^-),\ l = 1, \ldots, m
\end{aligned} \tag{22}$$

where $p(\mathbf{x}_k|\mathbf{x}_l^-)$ is the probability density function of the random vector $f_{k-1}(\mathbf{x}_l^-, \boldsymbol{\omega}_k)$. The corresponding weight vector is given by (21). By embedding the state into the RKHS $\mathcal{H}$ with the feature map ϕ, an estimate for the KME of prediction distribution, denoted by $\mu_{k|k-1}$, is given as

$$\mu_{k|k-1} = \sum_{l=1}^{m} w_{k-1,l}\phi(\mathbf{x}_l^+) \tag{23}$$

Authorized licensed use limited to: Yunnan University. Downloaded on April 14,2025 at 13:45:04 UTC from IEEE Xplore. Restrictions apply.

with its error covariance operator given as

$$P^{\phi}_{k|k-1} = \sum_{l=1}^{m} w_{k-1,l} \left(\phi(\mathbf{x}^+_l) - \mu_{k|k-1}\right)\left(\phi(\mathbf{x}^+_l) - \mu_{k|k-1}\right)^{\mathrm{T}}. \tag{24}$$

In fact, $P^{\phi}_{k|k-1}$ is a Hilbert–Schmidt linear operator mapping from $\mathcal{H}$ to itself.

2) *Update:* At time k, each node $i \in \mathcal{V}$ transmits its measurement data $\mathbf{y}_{i,k}$ to the fusion center. Let

$$\mathbf{y}_k = \begin{bmatrix} \mathbf{y}_{1,k} \\ \vdots \\ \mathbf{y}_{n,k} \end{bmatrix}, \; h_k(\cdot) = \begin{bmatrix} h_{1,k}(\cdot) \\ \vdots \\ h_{n,k}(\cdot) \end{bmatrix}, \; \mathbf{v}_k = \begin{bmatrix} \mathbf{v}_{1,k} \\ \vdots \\ \mathbf{v}_{n,k} \end{bmatrix}$$

where $\mathbf{y}_k \in \mathbb{R}^{n_y}$ is an augmented measurement vector, $n_y = \sum_{i=1}^{n} n_{y_i}$, $h_k : \mathbb{R}^{n_x} \to \mathbb{R}^{n_y}$ is an augmented measurement function, and augmented noise $\mathbf{v}_k \in \mathbb{R}^{n_y}$ has a covariance matrix

$$R_k = \text{block-diag}(R_{1,k}, \ldots, R_{n,k}). \tag{25}$$

Then, the measurement equation at the fusion center is given by

$$\mathbf{y}_k = h_k(\mathbf{x}_k) + \mathbf{v}_k$$

which can be similarly converted into

$$\mathbf{y}_k = \Lambda^*_k \phi(\mathbf{x}_k) + \mathbf{v}_k$$

where $\Lambda_k = [\lambda^{(1)}_k, \ldots, \lambda^{(n_y)}_k]$ is a Hilbert–Schmidt linear operator from $\mathbb{R}^{n_y}$ to $\mathcal{H}$ and needs to be determined. Based on the sample points $\mathbf{x}^+_1, \ldots, \mathbf{x}^+_m$, an estimate of Λ_k is determined by the following minimization problem:

$$\widehat{\Lambda}_k = \arg\min_{\Lambda} \sum_{l=1}^{m} w_{k-1,l} \left\| h_k(\mathbf{x}^+_l) - \Lambda^* \phi(\mathbf{x}^+_l) \right\|^2. \tag{26}$$

PROPOSITION 1 The solution to (26) is

$$\widehat{\Lambda}_k = \Phi K^{-1} Y_k^{\mathrm{T}} \tag{27}$$

where

$$Y_k = \left[h_k(\mathbf{x}^+_1), \ldots, h_k(\mathbf{x}^+_m)\right]$$

and K is the Gram matrix (see [45, p. 117]) given by

$$K = \begin{bmatrix} k(\mathbf{x}^+_1, \mathbf{x}^+_1) & \ldots & k(\mathbf{x}^+_1, \mathbf{x}^+_m) \\ \vdots & \ddots & \vdots \\ k(\mathbf{x}^+_m, \mathbf{x}^+_1) & \ldots & k(\mathbf{x}^+_m, \mathbf{x}^+_m) \end{bmatrix}.$$

PROOF It is omitted since it is similar to that of Theorem 1. □

To obtain the KME of posterior distribution, denoted by $\mu_{k|k}$, we design the following update rule:

$$\mu_{k|k} = \mu_{k|k-1} + G_k(\mathbf{y}_k - \widehat{\Lambda}^*_k \mu_{k|k-1}) \tag{28}$$

with gain operator $G_k : \mathbb{R}^{n_y} \to \mathcal{H}$ given by

$$G_k = P^{\phi}_{k|k-1} \widehat{\Lambda}_k (\widehat{\Lambda}^*_k P^{\phi}_{k|k-1} \widehat{\Lambda}_k + R_k)^{-1}. \tag{29}$$

Algorithm 3: Centralized Nonlinear Filtering Algorithm.

Input: $X_{k-1|k-1}$, $\mathbf{w}_{k-1}$

I. Prediction:

1: Calculate $X_{k|k-1}$ using (22).

II. Update:

2: Set $X_{k|k} = X_{k|k-1}$.

3: Calculate $\widetilde{\mathbf{w}}_k$ using (30).

4: Calculate $\mathbf{w}_k$ by solving (31).

5: Calculate

$$\mathbf{x}_{k|k} = X_{k|k-1}\mathbf{w}_k,$$
$$P_{k|k} = X_{k|k-1}\left(\text{diag}(\mathbf{w}_k) - \mathbf{w}_k\mathbf{w}_k^{\mathrm{T}}\right)X_{k|k-1}^{\mathrm{T}}.$$

Output: $X_{k|k}$, $\mathbf{w}_k$, $\mathbf{x}_{k|k}$, $P_{k|k}$

The above update rule is essentially a direct extension of the standard Kalman filtering in the Euclidean space to the RKHS. We can also obtain the sample of the posterior distribution from (28). Let

$$\Phi = \left[\phi(\mathbf{x}^+_1), \ldots, \phi(\mathbf{x}^+_m)\right]$$
$$W_{k-1} = \text{diag}(\mathbf{w}_{k-1}) - \mathbf{w}_{k-1}\mathbf{w}^{\mathrm{T}}_{k-1}.$$

Then, (28) can be rewritten as follows:

$$\mu_{k|k} = \Phi \widetilde{\mathbf{w}}_k$$

where

$$\widetilde{\mathbf{w}}_k = \mathbf{w}_{k-1} + W_{k-1}Y_k^{\mathrm{T}}\left(Y_k W_{k-1} Y_k^{\mathrm{T}} + R_k\right)^{-1} \cdot (\mathbf{y}_k - Y_k \mathbf{w}_{k-1}) \tag{30}$$

and satisfies $\mathbf{1}^{\mathrm{T}}\widetilde{\mathbf{w}}_k = 1$ (because $\mathbf{1}^{\mathrm{T}}\mathbf{w}_{k-1} = 0$ and $\mathbf{1}^{\mathrm{T}}W_{k-1} = 0$). The derivation of (30) can be found in Appendix C. Therefore, the sample of the posterior distribution, denoted by $X_{k|k}$, is given by

$$X_{k|k} = \left[\mathbf{x}^+_1, \ldots, \mathbf{x}^+_m\right]$$

with the corresponding weight vector given by (30). Similarly, we normalize the weight vector by minimizing the empirical maximum mean discrepancy

$$\min_{\mathbf{w}_k} (\mathbf{w}_k - \widetilde{\mathbf{w}}_k)^{\mathrm{T}} K (\mathbf{w}_k - \widetilde{\mathbf{w}}_k)$$
$$\text{s.t. } \mathbf{1}^{\mathrm{T}}\mathbf{w}_k = 1$$
$$w_{k,l} \geq \epsilon, \; l = 1, \ldots, m \tag{31}$$

where $\mathbf{w}_k = [w_{k,1}, \ldots, w_{k,m}]^{\mathrm{T}}$, and ϵ is a small positive real number. The proposed centralized filtering algorithm is summarized in Algorithm 3.

Generally, a "good" nonlinear filter should degenerate to the optimal filter under the linear case, i.e., under the linear dynamic system:

$$\mathbf{x}_k = F_{k-1}\mathbf{x}_{k-1} + \boldsymbol{\omega}_k$$
$$\mathbf{y}_{i,k} = H_{i,k}\mathbf{x}_k + \mathbf{v}_{i,k}$$

where $F_{k-1} \in \mathbb{R}^{n_x \times n_x}$ and $H_{i,k} \in \mathbb{R}^{n_{y_i} \times n_x}$ are known matrices.

PROPOSITION 2 The proposed centralized filter degenerates to the centralized Kalman filter.

Authorized licensed use limited to: Yunnan University. Downloaded on April 14,2025 at 13:45:04 UTC from IEEE Xplore. Restrictions apply.

PROOF By taking ϕ as the identity map, (28) with (29) reduces to the standard Kalman filter. □

Proposition 2 indicates that the proposed centralized filter extends the centralized Kalman filter to nonlinear dynamic systems. If $\{\boldsymbol{\omega}_k\}$ and $\{\mathbf{v}_{i,k}\}$ are mutually independent Gaussian white noise sequences, then the centralized Kalman filter is also the optimal filter regarding the minimum mean square error. It is worth emphasizing that the proposed centralized filter is also effective for nonlinear dynamic systems with a single sensor. This means that the covariance matrix of the measurement noise is not limited to a block diagonal form as in (25). Compared with some KME-based filters using either the kernel Kalman rule [38], [39] or the kernel Bayes' rule [32], [35], [37], the proposed centralized filter significantly reduces the computational burden in the update step of filtering. Specifically, when computing the KME of posterior distribution, the sizes of the matrices in the matrix inverse operations involved in the kernel Kalman rule and the kernel Bayes' rule depend on the number of sample points. However, in the proposed centralized filter, the matrix size in the matrix inverse operation, i.e., the size of $(Y_k W_{k-1} Y_k^{\mathrm{T}} + R_k)$ in (30), is independent of the number of the sample points. Such an improvement is precious for filtering, especially for real-time applications, such as target tracking.

V. PERFORMANCE ANALYSIS

This section demonstrates that the proposed distributed filter has centralized estimation accuracy by proving that it is equivalent to the proposed centralized filter. The following lemma is required for this proof.

LEMMA 1 Let $\widehat{\Lambda}_k$ and $\widehat{\Lambda}_{i,k}$ be given by (27) and (8), respectively. Then, we have

$$\widehat{\Lambda}_k = \left[\widehat{\Lambda}_{1,k}, \ldots, \widehat{\Lambda}_{n,k}\right].$$

PROOF See Appendix D. □

Based on the above lemma, we have the following theorem.

THEOREM 3 The proposed distributed filter is equivalent to the proposed centralized filter.

PROOF See Appendix E. □

Theorem 3 indicates that the proposed distributed filter has the high accuracy of the centralized filtering while ensuring the distributed setting, which accomplishes both our goals, i.e., "distributed setting" and "centralized accuracy," simultaneously.

From Theorem 3 and Proposition 2, we have the following corollary.

COROLLARY 1 The proposed distributed filter degenerates to the centralized Kalman filter under the linear case.

VI. EXAMPLES

This section provides two different target tracking scenarios to demonstrate the effectiveness of the proposed distributed and centralized filters, as they are equivalent. Our filters are compared with the Kalman-type distributed nonlinear filter proposed in [23] and the celebrated particle filter [51]. The proposed filters adopt the classic Gaussian and Laplace kernels given by (1) and (2) for comparison.

The filters' estimation accuracy is measured by the root-mean-square error (RMSE) and the average Euclidean error (AEE) of state estimate. As analyzed in [52], AEE is generally better than RMSE as a performance measure. All filtering algorithms run in Octave on an Intel Core i7 2.60 GHz computer.

A. Target of Nearly Constant Velocity

1. Bearing-Only Measurement

Similarly to the works in [1], [53], and [54], we consider a target moving with a nearly constant velocity. The state evolution equation is

$$\mathbf{x}_k = F_{k-1}\mathbf{x}_{k-1} + G_{k-1}\boldsymbol{\omega}_{k-1} \tag{32}$$

where $\mathbf{x}_k = [x_k, \dot{x}_k, y_k, \dot{y}_k]^{\mathrm{T}}$, $[x_k, y_k]^{\mathrm{T}}$ and $[\dot{x}_k, \dot{y}_k]^{\mathrm{T}}$ are the position and velocity of the target, respectively, and

$$F_{k-1} = \text{block-diag}(F, F),\ G_{k-1} = \text{block-diag}(G, G) \tag{33}$$

$$F = \begin{bmatrix} 1 & \Delta t \\ 0 & 1 \end{bmatrix},\ G = \begin{bmatrix} \Delta t^2/2 \\ \Delta t \end{bmatrix} \tag{34}$$

with sampling period $\Delta t = 1$ s, and the process noise $\boldsymbol{\omega}_{k-1}$ follows the Gaussian distribution $\mathcal{N}(0, Q_{k-1})$ with $Q_{k-1} = 10^2 GG^T (\mathrm{m}^2/\mathrm{s}^2)^2$. The initial state is generated from the Gaussian distribution $\mathcal{N}(\bar{\mathbf{x}}_0, P_0)$ with $\bar{\mathbf{x}}_0 = [-500\,\mathrm{m}, 18\,\mathrm{m/s}, 500\,\mathrm{m}, -12\,\mathrm{m/s}]^{\mathrm{T}}$ and

$$P_0 = \text{diag}([100\,\mathrm{m}^2, 10\,\mathrm{m}^2/\mathrm{s}^2, 100\,\mathrm{m}^2, 10\,\mathrm{m}^2/\mathrm{s}^2]^{\mathrm{T}}). \tag{35}$$

The distributed sensor network under consideration comprises ten sensors. For each node $i = 1, 2, \ldots, 10$, the measurement is a bearing

$$\theta_{i,k} = \arctan\left(\frac{y_k - b_i}{x_k - a_i}\right) + v_{i,k}$$

where $[a_i\mathrm{m}, b_i\mathrm{m}]^{\mathrm{T}}$ is the position of Node i, and is generated randomly from the square area with center $[0\mathrm{m}, 0\mathrm{m}]^{\mathrm{T}}$ and side length $d = 5000$ m. The measurement noise $v_{i,k}$ is distributed from the Gaussian distribution $\mathcal{N}(0, \sigma_{i,k})$ with $\sigma_{i,k} = 0.01\ \mathrm{rad}^2$. We set 500 particles in the particle filter and take $\sigma = 1$ in the Gaussian and Laplace kernels and 30 sample points in the proposed filter.

Fig. 1 illustrates the communication topology of the sensor network, where the solid dots represent the sensors and the solid lines are their links. We can see that Fig. 1 satisfies Assumption 1. Besides, Fig. 1 also depicts the real target trajectory and the estimated trajectories of the compared filters over 50 time-steps. It is observed that the proposed filter with Laplace kernel shows the best tracking performance among all the competitors.

Table I reports the averaged RMSEs and AEEs of the position and velocity estimates over 100 Monte Carlo runs and 50 time-steps. Compared with the method in [23], the proposed filters with Gaussian and Laplace kernels present

Authorized licensed use limited to: Yunnan University. Downloaded on April 14,2025 at 13:45:04 UTC from IEEE Xplore. Restrictions apply.

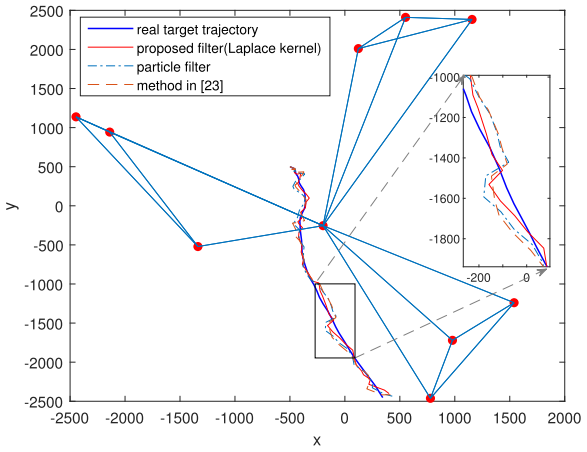


Fig. 1. Communication topology, real target trajectory, and estimated trajectories (Example A.1).

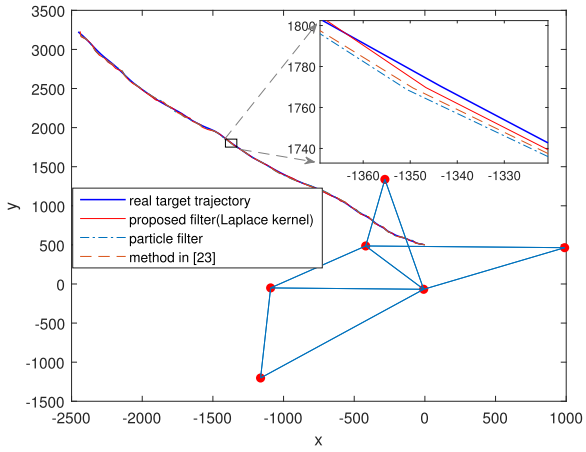


Fig. 2. Communication topology, real target trajectory, and estimated trajectories (Example A.2).

TABLE I
Comparison Results of Averaged RMSEs and AEEs of Position Estimates (m) and Velocity Estimates (m/s) (Example A.1)

error \ method / Measure	Method in [23]	Particle filter	Proposed filter (Gaussian kernel)	Proposed filter (Laplace kernel)
Position RMSEs	522.30	696.95	532.75	402.17
Velocity RMSEs	86.37	63.46	46.64	60.46
Position AEEs	141.34	239.71	130.03	115.90
Velocity AEEs	35.36	37.81	21.92	28.29

TABLE II
Comparison Results of Averaged RMSEs and AEEs of Position Estimates (m) and Velocity Estimates (m/s) (Example A.2)

error \ method / Measure	Method in [23]	Particle filter	Proposed filter (Gaussian kernel)	Proposed filter (Laplace kernel)
Position RMSEs	19.10	9.67	8.53	7.01
Velocity RMSEs	10.19	10.03	8.53	8.39
Position AEEs	7.59	7.24	6.51	5.82
Velocity AEEs	8.66	8.77	6.51	6.02

an improvement of −2% and 23% in position RMSE, 46% and 30% in velocity RMSE, 8% and 18% in position AEE, and 38% and 20% in velocity AEE, respectively. Compared with the particle filter, the corresponding results improve by 24% and 42% in position RMSE, 27% and 5% in velocity RMSE, 46% and 52% in position AEE, and 42% and 25% in velocity AEE, respectively. Hence, the proposed filters achieve more accurate position and velocity estimates than those in [23] and the particle filter in the RMSE and AEE metrics. More specifically, the proposed filters with Gaussian and Laplace kernels have the best velocity and position estimates, respectively. Interestingly, compared with the work in [23], the proposed filter using the Gaussian kernel has a relatively large position RMSE (second row, fourth column) but a smaller position AEE (fourth row, fourth column). This could be caused by an estimate with a relatively large deviation at a certain time since the RMSE is susceptible to an individual point with a large deviation. Thus, the proposed filter using a Gaussian kernel produces good state estimates most of the time except for some single moments. The numerical results reported in Table I provide the corresponding theoretical supports. While the method in [23] derives moment estimates only, the proposed filter provides posterior density estimates and thus results in a significantly improved filtering performance.

2. Range and Bearing Measurements

The target's motion is also described by (32)–(34) with a sampling period $\Delta t = 0.5$ s. Besides, the initial state is drawn from a Gaussian distribution $\mathcal{N}(\bar{\mathbf{x}}_0, P_0)$ with $\bar{\mathbf{x}}_0 = [0\text{m}, -18\text{m/s}, 500\text{m}, 12\text{m/s}]^{\mathrm{T}}$ and P_0 given by (35).

We consider the sensor network presented in Fig. 2, which comprises six sensors, where the ith node ($i = 1, 2, \ldots, 6$), located at $[a_i\text{m}, b_i\text{m}]^{\mathrm{T}}$, is randomly distributed in the square area of length $d = 3000$ m, centered at $[0\text{m}, 0\text{m}]^{\mathrm{T}}$. At time k, Node i observes the range $r_{i,k}$ and bearing $\theta_{i,k}$ of the target, and the measurement equation is as follows:

$$\begin{bmatrix} r_{i,k} \\ \theta_{i,k} \end{bmatrix} = \begin{bmatrix} \sqrt{(x_k - a_i)^2 + (y_k - b_i)^2} \\ \arctan\left(\frac{y_k - b_i}{x_k - a_i}\right) \end{bmatrix} + \mathbf{v}_{i,k}$$

where the measurement noise $\{\mathbf{v}_{i,k}\}$ is the zero-mean Gaussian white noise sequence with covariance matrix $R_{i,k} = \text{diag}([100\text{m}^2, 0.01\text{rad}^2]^{\mathrm{T}})$. We set 500 particles in the particle filter and take $\sigma = 2$ in the Gaussian and Laplace kernels and 20 sample points in the proposed filter.

The tracking performance of the proposed filter with Laplace kernel is also depicted in Fig. 2, where the target moving at a nearly constant velocity is observed. The proposed filter performs satisfactorily as it shows better tracking performance than the other two competitors. Table II reports the averaged RMSE and AEE of the position and velocity estimates over 100 time-steps and 100 Monte Carlo runs for all competitor methods. Compared with the work in [23], the proposed filters using Gaussian and Laplace kernels demonstrate improved performance of 55% and

Authorized licensed use limited to: Yunnan University. Downloaded on April 14,2025 at 13:45:04 UTC from IEEE Xplore. Restrictions apply.

63% in position RMSE, 16% and 18% in velocity RMSE, 14% and 23% in position AEE, and 25% and 30% in velocity AEE, respectively. Compared with the particle filter, our filters afford an improved performance by 12% and 28% in position RMSE, 15% and 16% in velocity RMSE, 10% and 20% in position AEE, and 26% and 31% in velocity AEE, respectively. Similar to Example A.1, the proposed filters using Gaussian and Laplace kernels have higher filtering accuracy than that in [23] and the particle filter. In contrast, they show a broadly similar performance in this example. This indicates that in this scenario, choosing the kernel function may not be the decisive factor affecting the filtering performance of the proposed filter.

B. Aircraft Coordinated Turn

Similar to the work in [55], we consider the following dynamic system:

$$\mathbf{x}_k = \begin{bmatrix} 1 & \frac{\sin(\omega_{k-1}\Delta t)}{\omega_{k-1}} & 0 & \frac{\cos(\omega_{k-1}\Delta t)-1}{\omega_{k-1}} & 0 \\ 0 & \cos(\omega_{k-1}\Delta t) & 0 & -\sin(\omega_{k-1}\Delta t) & 0 \\ 0 & \frac{1-\cos(\omega_{k-1}\Delta t)}{\omega_{k-1}} & 1 & \frac{\sin(\omega_{k-1}\Delta t)}{\omega_{k-1}} & 0 \\ 0 & \sin(\omega_{k-1}\Delta t) & 0 & \cos(\omega_{k-1}\Delta t) & 0 \\ 0 & 0 & 0 & 0 & 1 \end{bmatrix} \mathbf{x}_{k-1} + \boldsymbol{\omega}_{k-1}$$

where $\mathbf{x}_k = [x_k, \dot{x}_k, y_k, \dot{y}_k, \omega_k]^{\mathrm{T}}$ with ω_k being an unknown turn rate at time k, $\Delta t = 0.2$ is the sampling period, and $\{\boldsymbol{\omega}_k\}$ is the Gaussian white noise sequence with zero mean and covariance matrix

$$Q_{k-1} = q \begin{bmatrix} \frac{\Delta t^3}{3} & \frac{\Delta t^2}{2} & 0 & 0 & 0 \\ \frac{\Delta t^2}{2} & \Delta t & 0 & 0 & 0 \\ 0 & 0 & \frac{\Delta t^3}{3} & \frac{\Delta t^2}{2} & 0 \\ 0 & 0 & \frac{\Delta t^2}{2} & \Delta t & 0 \\ 0 & 0 & 0 & 0 & 1.75 \times 10^{-3}\Delta t \end{bmatrix}$$

where $q = 1\,\mathrm{m}^2/\mathrm{s}^3$ is a scalar with respect to noise intensity. The initial state of the aircraft is distributed from $\mathcal{N}(\bar{\mathbf{x}}_0, P_0)$ with $\bar{\mathbf{x}}_0 = [5000\,\mathrm{m}, 180\,\mathrm{m/s}, 5000\,\mathrm{m}, 180\,\mathrm{m/s}, 0.01\mathrm{rad}]$ and $P_0 = \mathrm{diag}([1000\,\mathrm{m}^2, 100\,\mathrm{m}^2/\mathrm{s}^2, 1000\,\mathrm{m}^2, 100\,\mathrm{m}^2/\mathrm{s}^2, 0.001\,\mathrm{rad}^2]^{\mathrm{T}})$.

The measurements of range $r_{i,k}$, bearing $\theta_{i,k}$, and range-rate $\dot{r}_{i,k}$ are acquired using eight sensors, as depicted in Fig. 3, where the side length is $d = 5000$ m. For each node $i = 1, \ldots, 8$, the measurement equation is

$$\begin{bmatrix} r_{i,k} \\ \theta_{i,k} \\ \dot{r}_{i,k} \end{bmatrix} = \begin{bmatrix} \sqrt{(x_k - a_i)^2 + (y_k - b_i)^2} \\ \arctan\left(\frac{y_k - b_i}{x_k - a_i}\right) \\ \frac{(x_k - a_i)\dot{x}_k + (y_k - b_i)\dot{y}_k}{\sqrt{(x_k - a_i)^2 + (y_k - b_i)^2}} \end{bmatrix} + \mathbf{v}_{i,k}$$

with $\mathbf{v}_{i,k} \sim \mathcal{N}(0, \mathrm{diag}([10000\mathrm{m}^2, 0.01\mathrm{rad}^2, 10\mathrm{m}^2/\mathrm{s}^2]^{\mathrm{T}}))$.

Fig. 3 also illustrates the tracking performance for the method in [23], the particle filter with 1000 particles, and the proposed filter using Laplace kernel with 60 sample points and $\sigma = 0.01$. The results highlight that the proposed filter has a good tracking performance. Besides, Table III reports the averaged RMSEs and AEEs of the position, velocity, and turn rate estimates over 100 time-steps and 50 Monte Carlo runs. In this example, the proposed filter with Laplace kernel demonstrates the best position, velocity, and turn rate estimation performance among all competitor filters. Compared with the work in [23], the proposed filters with Gaussian and Laplace kernels attain an improved performance by 30% and 36% in position RMSE, 21% and 30% in velocity RMSE, 27% and 27% in position AEE, 15% and 25% in velocity AEE, 16% and 19% in turn rate RMSE, and 4% and 7% in turn rate AEE, respectively. Compared with the particle filter, the proposed filters improve performance by 44% and 49% in position RMSE, 28% and 35% in velocity RMSE, 38% and 38% in position AEE, and 22% and 31% in velocity AEE, 17% and 20% in turn rate RMSE, and 5% and 8% in turn rate AEE, respectively. This suggests an appealing performance improvement of the proposed filter with Laplace kernel over [23] and the particle filter.

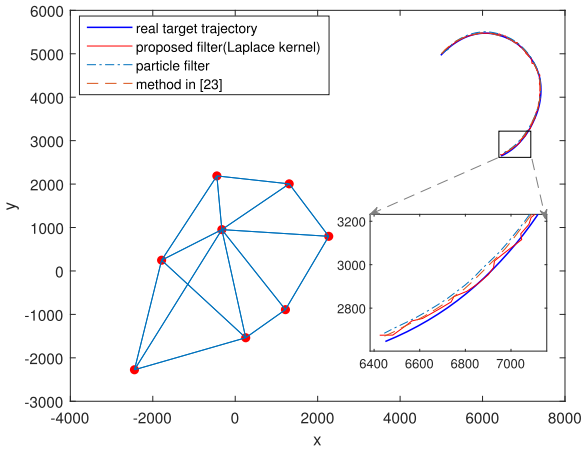


Fig. 3. Communication topology, real target trajectory, and estimated trajectories (Example B).

TABLE III
Comparison Results of Averaged RMSEs and AEEs of Position Estimates (m) and Velocity Estimates (m/s) and Turn Rate Estimates (rad) (Example B)

error \ method / Measure	Method in [23]	Particle filter	Proposed filter (Gaussian kernel)	Proposed filter (Laplace kernel)
Position RMSEs	27.85	34.92	19.43	17.69
Velocity RMSEs	4.00	4.36	3.16	2.82
Position AEEs	23.46	27.50	17.11	17.07
Velocity AEEs	3.20	3.50	2.72	2.40
Turn rate RMSEs	0.0325	0.0329	0.0273	0.0263
Turn rate AEEs	0.0257	0.0261	0.0248	0.0239

VII. CONCLUSION

We have developed the distributed nonlinear filter by approximating the posterior distribution with KME. By introducing the RKHS, the nonlinear measurement function has been linearly represented, and the recursive estimate for the KME of posterior distribution has been well established.

Authorized licensed use limited to: Yunnan University. Downloaded on April 14,2025 at 13:45:04 UTC from IEEE Xplore. Restrictions apply.

Furthermore, by extending the standard Kalman filter to the reproducing kernel Hilbert space, we have proposed the centralized nonlinear filter, which is much more efficient than the existing kernel-based filters for single-sensor setups. In addition, to demonstrate that the proposed distributed filter has centralized estimation accuracy, we have proved that it is equivalent to the centralized filter. Finally, two target-tracking examples have demonstrated the effectiveness of the proposed filters in different scenarios.

For distributed sensor networks, privacy issues may occur due to the information interaction between sensors, so privacy-preserving distributed nonlinear filtering will be studied in the future.

APPENDIX

A. Proof of Theorem 1

Let $\Lambda=[\lambda_1,\ldots,\lambda_{n_{y_i}}]$, where $\lambda_s\in\mathcal{H}$, with $s=1,\ldots,n_{y_i}$. Then, benefiting from the representer theorem [56], each λ_s can be expressed as

$$\lambda_s=\sum_{l=1}^{m} b_l\phi(\mathbf{x}_{i,l}^{+}),\ s=1,\ldots,n_{y_i}$$

where $b_1,\ldots,b_m$ are the coefficients. Then, Λ is written as

$$\Lambda=\begin{bmatrix}\sum_{l=1}^{m} b_{l,1}\phi(\mathbf{x}_{i,l}^{+}) & \ldots & \sum_{l=1}^{m} b_{l,n_{y_i}}\phi(\mathbf{x}_{i,l}^{+})\end{bmatrix}=\Phi B$$

with $B\in\mathbb{R}^{m\times n_{y_i}}$. Denoting $\sqrt{\mathbf{w}_{i,k-1}}=[\sqrt{w_{i,k-1,1}},\ldots,\sqrt{w_{i,k-2,m}}]^{\mathrm{T}}$, and $V=\operatorname{diag}(\sqrt{\mathbf{w}_{i,k-1}})$, we rewrite (11) in a more comprehensive fashion as

$$\min_{B}\|Y_{i,k}V-B^{\mathrm{T}}KV\|^2. \tag{36}$$

Then, the solution to (11) is given by $\widehat{\Lambda}_{i,k}=\Phi B$. The derivative of the objective function of (36) with respect to B is $-2KVVY_{i,k}^{\mathrm{T}}+2KVVKB$. Setting the derivative to zero, we have $KVVKB=KVVY_{i,k}^{\mathrm{T}}$. Then, the solution to (36) is

$$\begin{aligned}B&=(KVVK)^{-1}KVVY_{i,k}^{\mathrm{T}}\\&=\left(K\operatorname{diag}(\mathbf{w}_{i,k-1})K\right)^{-1}K\operatorname{diag}(\mathbf{w}_{i,k-1})Y_{i,k}^{\mathrm{T}}\\&=K^{-1}Y_{i,k}^{\mathrm{T}}\end{aligned}$$

and thus, $\widehat{\Lambda}_{i,k}=\Phi K^{-1}Y_{i,k}^{\mathrm{T}}$.

B. Proof of Theorem 2

Let $\widetilde{Y}_{i,k}=Y_{i,k}W_{i,k-1}^{\frac{1}{2}}$. By some calculations, we have

$$\begin{aligned}\mathbf{1}^{\mathrm{T}}\widetilde{\mathbf{w}}_{i,k}&=\mathbf{1}^{\mathrm{T}}(\mathbf{w}_{i,k}+W_{i,k-1}^{\frac{1}{2}}\Gamma_{i,k}^{-1}\boldsymbol{\xi}_{i,k})\\&=1+\mathbf{1}^{\mathrm{T}}W_{i,k-1}^{\frac{1}{2}}\Gamma_{i,k}^{-1}\boldsymbol{\xi}_{i,k}\\&=1+\mathbf{1}^{\mathrm{T}}W_{i,k-1}^{\frac{1}{2}}\left(\frac{1}{n}\sum_{i\in\mathcal{V}}\left(n\widetilde{Y}_{i,k}^{\mathrm{T}}R_{i,k}^{-1}\widetilde{Y}_{i,k}+I\right)\right)^{-1}\\&\quad\cdot\frac{1}{n}\sum_{i\in\mathcal{V}}n\widetilde{Y}_{i,k}^{\mathrm{T}}R_{i,k}^{-1}(\mathbf{y}_{i,k}-\widehat{\Lambda}_{i,k}^{*}\mu_{i,k|k-1})\\&=1+\mathbf{1}^{\mathrm{T}}W_{i,k-1}^{\frac{1}{2}}\left(\sum_{i\in\mathcal{V}}\widetilde{Y}_{i,k}^{\mathrm{T}}R_{i,k}^{-1}\widetilde{Y}_{i,k}+I\right)^{-1}\\&\quad\cdot\sum_{i\in\mathcal{V}}\widetilde{Y}_{i,k}^{\mathrm{T}}R_{i,k}^{-1}(\mathbf{y}_{i,k}-\widehat{\Lambda}_{i,k}^{*}\mu_{i,k|k-1}).\end{aligned} \tag{37}$$

Denote

$$\begin{aligned}\widetilde{Y}_k&=\left[\widetilde{Y}_{1,k}^{\mathrm{T}},\ldots,\widetilde{Y}_{n,k}^{\mathrm{T}}\right]^{\mathrm{T}}\\R_k&=\text{block-diag}(R_{1,k},\ldots,R_{n,k}).\end{aligned}$$

Then, using Woodbury matrix identity (see [57, p. 258]), the right-hand side of (37) is equivalent to

$$\begin{aligned}&1+\mathbf{1}^{\mathrm{T}}W_{i,k-1}^{\frac{1}{2}}\left(I-\widetilde{Y}_k^{\mathrm{T}}\left(R_k+\widetilde{Y}_k\widetilde{Y}_k^{\mathrm{T}}\right)^{-1}\widetilde{Y}_k\right)\\&\quad\cdot\sum_{i\in\mathcal{V}}\widetilde{Y}_{i,k}^{\mathrm{T}}R_{i,k}^{-1}(\mathbf{y}_{i,k}-\widehat{\Lambda}_{i,k}^{*}\mu_{i,k|k-1})\\&=1+\left(\mathbf{1}^{\mathrm{T}}W_{i,k-1}^{\frac{1}{2}}-\mathbf{1}^{\mathrm{T}}W_{i,k-1}^{\frac{1}{2}}\widetilde{Y}_k^{\mathrm{T}}\left(R_k+\widetilde{Y}_k\widetilde{Y}_k^{\mathrm{T}}\right)^{-1}\widetilde{Y}_k\right)\\&\quad\cdot W_{i,k-1}^{\frac{1}{2}}\sum_{i\in\mathcal{V}}Y_{i,k}^{\mathrm{T}}R_{i,k}^{-1}(\mathbf{y}_{i,k}-\widehat{\Lambda}_{i,k}^{*}\mu_{i,k|k-1})\\&=1+\mathbf{1}^{\mathrm{T}}W_{i,k-1}\sum_{i\in\mathcal{V}}Y_{i,k}^{\mathrm{T}}R_{i,k}^{-1}(\mathbf{y}_{i,k}-\widehat{\Lambda}_{i,k}^{*}\mu_{i,k|k-1})\\&=1.\end{aligned}$$

Here, we use the fact that $\mathbf{1}^{\mathrm{T}}W_{i,k-1}=\mathbf{w}_{i,k-1}^{\mathrm{T}}-\mathbf{w}_{i,k-1}^{\mathrm{T}}=0$.

C. Derivation of (30)

From (23), we have $\mu_{k|k-1}=\Phi\mathbf{w}_{k-1}$. From (24), we have

$$\begin{aligned}P_{k|k-1}^{\phi}&=(\Phi-\Phi\mathbf{w}_{k-1}\mathbf{1}^{\mathrm{T}})\operatorname{diag}(\mathbf{w}_{k-1})(\Phi-\Phi\mathbf{w}_{k-1}\mathbf{1}^{\mathrm{T}})^{\mathrm{T}}\\&=\Phi(I-\mathbf{w}_{k-1}\mathbf{1}^{\mathrm{T}})\operatorname{diag}(\mathbf{w}_{k-1})(I-\mathbf{1}\mathbf{w}_{k-1}^{\mathrm{T}})\Phi^{*}\\&=\Phi\left(\operatorname{diag}(\mathbf{w}_{k-1})+\mathbf{w}_{k-1}\mathbf{1}^{\mathrm{T}}\operatorname{diag}(\mathbf{w}_{k-1})\mathbf{1}\mathbf{w}_{k-1}^{\mathrm{T}}\right.\\&\quad\left.-2\mathbf{w}_{k-1}\mathbf{1}^{\mathrm{T}}\operatorname{diag}(\mathbf{w}_{k-1})\right)\Phi^{*}\\&=\Phi\left(\operatorname{diag}(\mathbf{w}_{k-1})+\mathbf{w}_{k-1}\mathbf{w}_{k-1}^{\mathrm{T}}\mathbf{1}\mathbf{w}_{k-1}^{\mathrm{T}}\right.\\&\quad\left.-2\mathbf{w}_{k-1}\mathbf{w}_{k-1}^{\mathrm{T}}\right)\Phi^{*}\\&=\Phi\left(\operatorname{diag}(\mathbf{w}_{k-1})-\mathbf{w}_{k-1}\mathbf{w}_{k-1}^{\mathrm{T}}\right)\Phi^{*}\\&=\Phi W_{k-1}\Phi^{*}.\end{aligned}$$

Substituting $\mu_{k|k-1}$, $P_{k|k-1}^{\phi}$, and $\widehat{\Lambda}_k$ given by (27) into (29), we have

$$\begin{aligned}G_k&=\Phi W_{k-1}\Phi^{*}\Phi K^{-1}Y_k^{\mathrm{T}}\\&\quad\cdot(Y_kK^{-1}\Phi^{*}\Phi W_{k-1}\Phi^{*}\Phi K^{-1}Y_k^{\mathrm{T}}+R_k)^{-1}\\&=\Phi W_{k-1}Y_k^{\mathrm{T}}(Y_kW_{k-1}Y_k^{\mathrm{T}}+R_k)^{-1}\end{aligned}$$

and then

$$\begin{aligned}\mu_{k|k}&=\Phi\mathbf{w}_{k-1}+\Phi W_{k-1}Y_k^{\mathrm{T}}(Y_kW_{k-1}Y_k^{\mathrm{T}}+R_k)^{-1}\\&\quad\cdot(\mathbf{y}_k-Y_kK^{-1}\Phi^{*}\Phi\mathbf{w}_{k-1})\\&=\Phi\mathbf{w}_{k-1}+\Phi W_{k-1}Y_k^{\mathrm{T}}(Y_kW_{k-1}Y_k^{\mathrm{T}}+R_k)^{-1}\\&\quad\cdot(\mathbf{y}_k-Y_k\mathbf{w}_{k-1}).\end{aligned}$$

Here, we use the fact that $\Phi^{*}\Phi=K$.

Authorized licensed use limited to: Yunnan University. Downloaded on April 14,2025 at 13:45:04 UTC from IEEE Xplore. Restrictions apply.

D. Proof of Lemma 1

Note that $Y_k = [Y_{1,k}^{\mathrm{T}}, \ldots, Y_{n,k}^{\mathrm{T}}]^{\mathrm{T}}$, where each $Y_{i,k}$ for $i \in \mathcal{V}$ corresponds to a sample of the noiseless measurement obtained at Node i, as given by (10). Then, we have

$$\begin{aligned}\widehat{\Lambda}_k &= \Phi K^{-1}\left[Y_{1,k}^{\mathrm{T}}, \ldots, Y_{n,k}^{\mathrm{T}}\right] \\ &= \left[\Phi K^{-1} Y_{1,k}^{\mathrm{T}}, \ldots, \Phi K^{-1} Y_{n,k}^{\mathrm{T}}\right] \\ &= \left[\widehat{\Lambda}_{1,k}, \ldots, \widehat{\Lambda}_{n,k}\right].\end{aligned}$$

E. Proof of Theorem 3

We adopt the mathematical induction. At time $k = 1$, suppose that the KMEs of posterior distributions are the same for the distributed and centralized algorithms, i.e., $\mu_{i,0|0} = \mu_{0|0}$. At time $k-1$, suppose that $\mu_{i,k-1|k-1} = \mu_{k-1|k-1}$. Next, we prove that at time k, $\mu_{i,k|k} = \mu_{k|k}$ holds. Since the sample of the process noise is deterministic, from $\mu_{i,k-1|k-1} = \mu_{k-1|k-1}$, we immediately have $\mu_{i,k|k-1} = \mu_{k|k-1}$, which indicates $W_{i,k-1} = W_{k-1}$. Then, at time k, the KME of posterior distribution for each node i is calculated as

$$\mu_{i,k|k} = \mu_{k|k-1} + \Phi W_{k-1}^{\frac{1}{2}} (\Gamma_{i,k}^*)^{-1} \boldsymbol{\xi}_{i,k}^*$$

where $\Gamma_{i,k}^*$ and $\boldsymbol{\xi}_{i,k}^*$ are given by (15) and (16), respectively. Let $\widetilde{Y}_k = [\widetilde{Y}_{1,k}^{\mathrm{T}}, \ldots, \widetilde{Y}_{n,k}^{\mathrm{T}}]^{\mathrm{T}}$ with $\widetilde{Y}_{i,k} = Y_{i,k} W_{k-1}^{\frac{1}{2}}$. Then,

$$\begin{aligned}&\Phi W_{k-1}^{\frac{1}{2}} (\Gamma_{i,k}^*)^{-1} \boldsymbol{\xi}_{i,k}^* \\ &= \Phi W_{k-1}^{\frac{1}{2}} \left(\sum_{i\in\mathcal{V}} W_{k-1}^{\frac{1}{2}} Y_{i,k}^{\mathrm{T}} R_{i,k}^{-1} Y_{i,k} W_{k-1}^{\frac{1}{2}} + I \right)^{-1} \\ &\quad \cdot \left(\sum_{i\in\mathcal{V}} \widetilde{Y}_{i,k}^{\mathrm{T}} R_{i,k}^{-1} (\mathbf{y}_{i,k} - \widehat{\Lambda}_{i,k}^* \mu_{i,k|k-1}) \right) \\ &= \Phi W_{k-1}^{\frac{1}{2}} \left(\left[\widetilde{Y}_{1,k}^{\mathrm{T}}, \ldots, \widetilde{Y}_{n,k}^{\mathrm{T}}\right] \cdot \mathrm{block-diag}\left(R_{1,k}^{-1}, \ldots, R_{n,k}^{-1}\right) \right. \\ &\quad \left. \cdot \left[\widetilde{Y}_{1,k}^{\mathrm{T}}, \ldots, \widetilde{Y}_{n,k}^{\mathrm{T}}\right]^{\mathrm{T}} + I \right)^{-1} \left[\widetilde{Y}_{1,k}^{\mathrm{T}}, \ldots, \widetilde{Y}_{n,k}^{\mathrm{T}}\right] \\ &\quad \cdot \mathrm{block-diag}\left(R_{1,k}^{-1}, \ldots, R_{n,k}^{-1}\right) \left(\left[\mathbf{y}_{1,k}^{\mathrm{T}}, \ldots, \mathbf{y}_{n,k}^{\mathrm{T}}\right]^{\mathrm{T}} \right. \\ &\quad \left. - \left[\left(\widehat{\Lambda}_{1,k}^* \mu_{1,k|k-1}\right)^{\mathrm{T}}, \ldots, \left(\widehat{\Lambda}_{n,k}^* \mu_{n,k|k-1}\right)^{\mathrm{T}} \right]^{\mathrm{T}} \right) \\ &= \Phi W_{k-1}^{\frac{1}{2}} \left(\widetilde{Y}_k^{\mathrm{T}} R_k^{-1} \widetilde{Y}_k + I \right)^{-1} \widetilde{Y}_k^{\mathrm{T}} R_k^{-1} \left(\mathbf{y}_k - \widehat{\Lambda}_k^* \mu_{k|k-1} \right) \\ &= \Phi W_{k-1}^{\frac{1}{2}} \left(W_{k-1}^{\frac{1}{2}} Y_k^{\mathrm{T}} R_k^{-1} Y_k W_{k-1}^{\frac{1}{2}} + I \right)^{-1} \\ &\quad \cdot W_{k-1}^{\frac{1}{2}} Y_k^{\mathrm{T}} R_k^{-1} \left(\mathbf{y}_k - \widehat{\Lambda}_k^* \mu_{k|k-1} \right) \\ &= G_k' \left(\mathbf{y}_k - \widehat{\Lambda}_k^* \mu_{k|k-1} \right)\end{aligned}$$

where $G_k' = \Phi W_{k-1}^{\frac{1}{2}} (W_{k-1}^{\frac{1}{2}} Y_k^{\mathrm{T}} R_k^{-1} Y_k W_{k-1}^{\frac{1}{2}} + I)^{-1} W_{k-1}^{\frac{1}{2}} Y_k^{\mathrm{T}} R_k^{-1}$. Letting $P = W_{k-1}^{\frac{1}{2}} Y_k^{\mathrm{T}} R_k^{-1} Y_k W_{k-1}^{\frac{1}{2}}$, we have

$$\begin{aligned}G_k' &= \Phi W_{k-1}^{\frac{1}{2}} \left(I - W_{k-1}^{\frac{1}{2}} Y_k^{\mathrm{T}} R_k^{-1} Y_k W_{k-1}^{\frac{1}{2}} \right. \\ &\quad \left. \cdot \left(I + W_{k-1}^{\frac{1}{2}} Y_k^{\mathrm{T}} R_k^{-1} Y_k W_{k-1}^{\frac{1}{2}} \right)^{-1} \right) W_{k-1}^{\frac{1}{2}} Y_k^{\mathrm{T}} R_k^{-1} \\ &= \Phi W_{k-1} Y_k^{\mathrm{T}} \left(R_k^{-1} - R_k^{-1} Y_k W_{k-1}^{\frac{1}{2}} \right. \\ &\quad \left. \cdot \left(I + W_{k-1}^{\frac{1}{2}} Y_k^{\mathrm{T}} R_k^{-1} Y_k W_{k-1}^{\frac{1}{2}} \right)^{-1} W_{k-1}^{\frac{1}{2}} Y_k^{\mathrm{T}} R_k^{-1} \right).\end{aligned}$$

Using the Woodbury matrix identity (see [57, p. 258]), we have

$$\begin{aligned}G_k' &= \Phi W_{k-1} Y_k^{\mathrm{T}} \left(Y_k W_{k-1} Y_k^{\mathrm{T}} + R_k \right)^{-1} \\ &= \Phi W_{k-1} \Phi^* \Phi K^{-1} Y_k^{\mathrm{T}} \\ &\quad \cdot \left(Y_k K^{-1} \Phi^* \Phi W_{k-1} \Phi^* \Phi K^{-1} Y_k^{\mathrm{T}} + R_k \right)^{-1} \\ &= P_{k|k-1}^{\phi} \widehat{\Lambda}_k (\widehat{\Lambda}_k^* P_{k|k-1}^{\phi} \widehat{\Lambda}_k + R_k)^{-1} = G_k.\end{aligned}$$

This means that

$$\mu_{i,k|k} = \mu_{k|k-1} + G_k(\mathbf{y}_k - \widehat{\Lambda}_k^* \mu_{k|k-1}) = \mu_{k|k}$$

which completes the proof.

REFERENCES

[1] B. Ristic, S. Arulampalam, and N. Gordon, *Beyond the Kalman Filter: Particle Filters for Tracking Applications*. Norwood, MA, USA: Artech House, 2004.

[2] S. V. Bordonaro, P. Willett, Y. Bar-Shalom, and T. Luginbuhl, "Converted measurement sigma point Kalman filter for bistatic sonar and radar tracking," *IEEE Trans. Aerosp. Electron. Syst.*, vol. 55, no. 1, pp. 147–159, Feb. 2019.

[3] N. Forti, L. M. Millefiori, P. Braca, and P. Willett, "Bayesian filtering for dynamic anomaly detection and tracking," *IEEE Trans. Aerosp. Electron. Syst.*, vol. 58, no. 3, pp. 1528–1544, Jun. 2022.

[4] J. Lan, "Generalized conversion based nonlinear filtering using deterministic sampling for target tracking," *IEEE Trans. Aerosp. Electron. Syst.*, vol. 59, no. 5, pp. 7295–7307, Oct. 2023.

[5] H. Cui, L. Mihaylova, X. Wang, and S. Gao, "Uncertainty-aware variational inference for target tracking," *IEEE Trans. Aerosp. Electron. Syst.*, vol. 59, no. 1, pp. 258–273, Feb. 2023.

[6] K. Li and G. Zhou, "Maneuvering target state estimation in range-squared coordinate," *IEEE Trans. Aerosp. Electron. Syst.*, vol. 60, no. 1, pp. 574–590, Feb. 2024.

[7] L. Guo, S. Hu, J. Zhou, and X. Rong Li, "Nonlinear filtering based on third-degree approximation via sequential homotopy," *IEEE Trans. Signal Process.*, vol. 70, pp. 5334–5348, Nov. 2022.

[8] L. Guo, S. Hu, J. Zhou, and X. Rong Li, "Recursive nonlinear filtering via Gaussian approximation with minimized Kullback–Leibler divergence," *IEEE Trans. Aerosp. Electron. Syst.*, vol. 60, no. 1, pp. 965–979, Feb. 2024.

[9] Y. Choe, J. W. Song, and C. G. Park, "Lightweight marginalized particle filtering with enhanced consistency for terrain referenced navigation," *IEEE Trans. Aerosp. Electron. Syst.*, vol. 58, no. 3, pp. 2493–2504, Jun. 2022.

[10] M. Tian, Z. Chen, H. Wang, and L. Liu, "An intelligent particle filter for infrared dim small target detection and tracking," *IEEE Trans. Aerosp. Electron. Syst.*, vol. 58, no. 6, pp. 5318–5333, Dec. 2022.

[11] R. P. S. Mahler, *Advances in Statistical Multisource-Multitarget Information Fusion*. Norwood, MA, USA: Artech House, 2014.

[12] S. Xu, K. Doğançay, and H. Hmam, "Distributed pseudolinear estimation and UAV path optimization for 3D AOA target tracking," *Signal Process.*, vol. 133, pp. 64–78, 2017.

[13] W. Li, Y. Jia, and J. Du, "Distributed Kalman filter for cooperative localization with integrated measurements," *IEEE Trans. Aerosp. Electron. Syst.*, vol. 56, no. 4, pp. 3302–3310, Aug. 2020.

[14] M. Zhu, R. Wang, X. Miao, and T. Sui, "Accuracy analysis for distributed dynamic state estimation in large-scale systems with a cyclic network graph," *Sci. China Inf. Sci.*, vol. 66, 2023, Art. no. 190206.

[15] J. Hu, Z. Hu, H. Dong, and H. Liu, "Distributed resilient fusion filtering for nonlinear systems with random sensor delays: Optimized algorithm design and boundedness analysis," *J. Syst. Sci. Complexity*, vol. 36, no. 4, pp. 1423–1442, 2023.

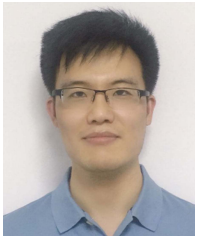
Authorized licensed use limited to: Yunnan University. Downloaded on April 14,2025 at 13:45:04 UTC from IEEE Xplore. Restrictions apply.

[16] Y. Tan, P. Weng, B. Chen, and L. Yu, "Nonlinear fusion estimation for false data injection attack signals in cyber-physical systems," *Sci. China Inf. Sci.*, vol. 66, no. 7, 2023, Art. no. 179203.
[17] R. Olfati-Saber and J. S. Shamma, "Consensus filters for sensor networks and distributed sensor fusion," in *Proc. 44th IEEE Conf. Decis. Control*, Seville, Spain, 2005, pp. 6698–6703.
[18] R. Olfati-Saber, "Distributed Kalman filter with embedded consensus filters," in *Proc. 44th IEEE Conf. Decis. Control*, Seville, Spain, 2005, pp. 8179–8184.
[19] R. Olfati-Saber, "Distributed Kalman filtering for sensor networks," in *Proc. 46th IEEE Conf. Decis. Control*, New Orleans, LA, USA, 2007, pp. 5492–5498.
[20] S. P. Talebi and S. Werner, "Distributed Kalman filtering and control through embedded average consensus information fusion," *IEEE Trans. Autom. Control*, vol. 64, no. 10, pp. 4396–4403, Oct. 2019.
[21] S. J. Julier and J. K. Uhlmann, "Unscented filtering and nonlinear estimation," *Proc. IEEE*, vol. 92, no. 3, pp. 401–422, Mar. 2004.
[22] W. Li and Y. Jia, "Distributed consensus filtering for discrete-time nonlinear systems with non-Gaussian noise," *Signal Process.*, vol. 92, no. 10, pp. 2464–2470, 2012.
[23] Q. Chen, W. Wang, C. Yin, X. Jin, and J. Zhou, "Distributed cubature information filtering based on weighted average consensus," *Neurocomputing*, vol. 243, pp. 115–124, 2017.
[24] B. Xiao, Q. J. Wu, and L. Yan, "Multisensor fusion estimation of nonlinear systems with intermittent observations and heavy-tailed noises," *Sci. China Inf. Sci.*, vol. 65, 2022, Art. no. 192203.
[25] D. Gu, J. Sun, Z. Hu, and H. Li, "Consensus based distributed particle filter in sensor networks," in *Proc. Int. Conf. Inf. Autom.*, Changsha, China, 2008, pp. 302–307.
[26] O. Hlinka, O. Slučiak, F. Hlawatsch, P. M. Djurić, and M. Rupp, "Distributed Gaussian particle filtering using likelihood consensus," in *Proc. IEEE Int. Conf. Acoust. Speech Signal Process.*, Prague, Czech Republic, 2011, pp. 3756–3759.
[27] O. Hlinka, F. Hlawatsch, and P. M. Djurić, "Consensus-based distributed particle filtering with distributed proposal adaptation," *IEEE Trans. Signal Process.*, vol. 62, no. 12, pp. 3029–3041, Jun. 2014.
[28] W. Xia, M. Sun, and Q. Wang, "Direct target tracking by distributed Gaussian particle filtering for heterogeneous networks," *IEEE Trans. Signal Process.*, vol. 68, pp. 1361–1373, Feb. 2020.
[29] W. Song, Z. Wang, J. Wang, F. E. Alsaadi, and J. Shan, "Distributed auxiliary particle filtering with diffusion strategy for target tracking: A dynamic event-triggered approach," *IEEE Trans. Signal Process.*, vol. 69, pp. 328–340, Dec. 2021.
[30] W. Song, Z. Wang, J. Wang, F. E. Alsaadi, and J. Shan, "Particle filtering for nonlinear/non-Gaussian systems with energy harvesting sensors subject to randomly occurring sensor saturations," *IEEE Trans. Signal Process.*, vol. 69, pp. 15–27, 2021.
[31] A. Smola, A. Gretton, L. Song, and B. Schölkopf, "A Hilbert space embedding for distributions," in *Algorithmic Learning Theory*, M. Hutter, R. A. Servedio, and E. Takimoto, Eds. Berlin, Germany: Springer, 2007, pp. 13–31.
[32] L. Song, J. Huang, A. Smola, and K. Fukumizu, "Hilbert space embeddings of conditional distributions with applications to dynamical systems," in *Proc. 26th Annu. Int. Conf. Mach. Learn.*, Montreal Quebec, Canada, 2009, pp. 961–968.
[33] B.K. Sriperumbudur, A. Gretton, K. Fukumizu, B. Schölkopf, and G. R. Lanckriet, "Hilbert space embeddings and metrics on probability measures," *J. Mach. Learn. Res.*, vol. 11, pp. 1517–1561, 2010.
[34] L. Song, K. Fukumizu, and A. Gretton, "Kernel embeddings of conditional distributions: A unified kernel framework for nonparametric inference in graphical models," *IEEE Signal Process. Mag.*, vol. 30, no. 4, pp. 98–111, Jul. 2013.
[35] M. Kanagawa, Y. Nishiyama, A. Gretton, and K. Fukumizu, "Monte Carlo filtering using kernel embedding of distributions," in *Proc. AAAI Conf. Artif. Intell.*, 2014, pp. 1897–1903.
[36] K. Muandet, K. Fukumizu, B. Sriperumbudur, and B. Schölkopf, "Kernel mean embedding of distributions: A review and beyond," *Found. Trends Mach. Learn.*, vol. 10, no. 1–2, pp. 1–141, 2017.
[37] K. Fukumizu, L. Song, and A. Gretton, "Kernel Bayes' rule: Bayesian inference with positive definite kernels," *J. Mach. Learn. Res.*, vol. 14, no. 1, pp. 3753–3783, 2013.
[38] G. Gebhardt, A. Kupcsik, and G. Neumann, "The kernel Kalman rule—efficient nonparametric inference with recursive least squares," in *Proc. AAAI Conf. Artif. Intell.*, 2017, pp. 3754–3760.
[39] M. Sun, M. E. Davies, I. K. Proudler, and J. R. Hopgood, "Adaptive kernel Kalman filter," *IEEE Trans. Signal Process.*, vol. 71, pp. 713–726, Mar. 2023.
[40] S. Hu, L. Guo, and J. Zhou, "An iterative nonlinear filter based on posterior distribution approximation via penalized Kullback–Leibler divergence minimization," *IEEE Signal Process. Lett.*, vol. 29, pp. 1137–1141, Apr. 2022.
[41] L. Guo, S. Hu, J. Zhou, and X. Rong Li, "Gaussian approximation filter based on divergence minimization for nonlinear dynamic systems," in *Proc. 25th Int. Conf. Inf. Fusion*, Linköping, Sweden, 2022, pp. 1–7.
[42] C. Lemieux, *Monte Carlo and Quasi-Monte Carlo Sampling*. New York, NY, USA: Springer, 2009.
[43] V. Elvira, L. Martino, and P. Closas, "Importance Gaussian quadrature," *IEEE Trans. Signal Process.*, vol. 69, pp. 474–488, Jan. 2021.
[44] M. F. Bugallo, V. Elvira, L. Martino, D. Luengo, J. Miguez, and P. M. Djuric, "Adaptive importance sampling: The past, the present, and the future," *IEEE Signal Process. Mag.*, vol. 34, no. 4, pp. 60–79, Jul. 2017.
[45] I. Steinwart and A. Christmann, *Support Vector Machines*. New York, NY, USA: Springer, 2008.
[46] V. A. Zorich, *Mathematical Analysis I*, 2nd ed. Berlin, Germany: Springer, 2015.
[47] S. Sundaram and C. N. Hadjicostis, "Finite-time distributed consensus in graphs with time-invariant topologies," in *Proc. Amer. Control Conf.*, New York, NY, USA, 2007, pp. 711–716.
[48] R. J. Vanderbei and T. J. Carpenter, "Symmetric indefinite systems for interior point methods," *Math. Program.*, vol. 58, no. 1-3, pp. 1–32, 1993.
[49] H. Yang, H. Li, Y. Xia, and L. Li, "Distributed Kalman filtering over sensor networks with transmission delays," *IEEE Trans. Cybern.*, vol. 51, no. 11, pp. 5511–5521, Nov. 2021.
[50] L. Shi, L. Xie, and R. M. Murray, "Kalman filtering over a packet-delaying network: A probabilistic approach," *Automatica*, vol. 45, no. 9, pp. 2134–2140, 2009.
[51] M. S. Arulampalam, S. Maskell, N. Gordon, and T. Clapp, "A tutorial on particle filters for online nonlinear/non-Gaussian Bayesian tracking," *IEEE Trans. Signal Process.*, vol. 50, no. 2, pp. 174–188, Feb. 2002.
[52] X. R. Li and Z. Zhao, "Evaluation of estimation algorithms part I: Incomprehensive measures of performance," *IEEE Trans. Aerosp. Electron. Syst.*, vol. 42, no. 4, pp. 1340–1358, Oct. 2006.
[53] J. Lan and X. R. Li, "Nonlinear estimation by LMMSE-based estimation with optimized uncorrelated augmentation," *IEEE Trans. Signal Process.*, vol. 63, no. 16, pp. 4270–4283, Aug. 2015.
[54] J. Lan and X. R. Li, "Multiple conversions of measurements for nonlinear estimation," *IEEE Trans. Signal Process.*, vol. 65, no. 18, pp. 4956–4970, Sep. 2017.
[55] B. Jia, M. Xin, and Y. Cheng, "High-degree cubature Kalman filter," *Automatica*, vol. 49, no. 2, pp. 510–518, 2013.
[56] B. Schölkopf, R. Herbrich, and A. J. Smola, "A generalized representer theorem," in *Computational Learning Theory*, D. Helmbold and B. Williamson, Eds, vol. 2111 Berlin, Germany: Springer, 2001, pp. 416–426.
[57] N. J. Higham, *Accuracy and Stability of Numerical Algorithms*, 2nd ed. Philadelphia, PA, USA: SIAM, 2002.

Authorized licensed use limited to: Yunnan University. Downloaded on April 14,2025 at 13:45:04 UTC from IEEE Xplore. Restrictions apply.

Liping Guo received the B.S. and Ph.D. degrees in statistics and applied statistics from Sichuan University, Chengdu, China, in 2019 and 2023, respectively.

She is currently a Postdoctoral Research Associate with the Institute of Systems Science, Academy of Mathematics and Systems Science, Chinese Academy of Sciences, Beijing, China. Her current research interests include state estimation, privacy preservation, target tracking, and optimization algorithms.



Jimin Wang (Member, IEEE) received the B.S. degree in mathematics from Shandong Normal University, Jinan, China, in 2012, and the Ph.D. degree in mathematics from the School of Mathematics, Shandong University, Jinan, in 2018.

From 2017 to 2018, he was a joint Ph.D. student with the School of Electrical Engineering and Computing, The University of Newcastle, Callaghan NSW, Australia. From 2018 to 2020, he was a Postdoctoral Researcher with the Institute of Systems Science (ISS), Chinese Academy of Sciences (CAS), Beijing, China. He is currently an Associate Professor with the School of Automation and Electrical Engineering, University of Science and Technology Beijing. His current research interests include privacy and security in cyber-physical systems, stochastic systems, and networked control systems.

Dr. Wang was the recipient of the Shandong University's excellent doctoral dissertation. He is a Member of the IEEE CSS Technical Committee on Security and Privacy, the IEEE CSS Technical Committee on Networks and Communication Systems, and the IFAC Technical Committee 1.5 on Networked Systems.

Yanlong Zhao (Senior Member, IEEE) received the B.S. degree in mathematics from Shandong University, Jinan, China, in 2002, and the Ph.D. degree in systems theory from the Academy of Mathematics and Systems Science (AMSS), Chinese Academy of Sciences (CAS), Beijing, China, in 2007.

Since 2007, he has been with the AMSS, CAS, where he is currently a full Professor. His research interests include identification and control of quantized systems, information theory, and modeling of financial systems.

Dr. Zhao was a Vice-President of Asian Control Association and a Vice-President of IEEE CSS Beijing Chapter, and is currently a Vice General Secretary of Chinese Association of Automation (CAA) and the Chair of Technical Committee on Control Theory, CAA. He has been a Deputy Editor-in-Chief of *Journal of Systems and Science and Complexity*, an Associate Editor of *Automatica*, *SIAM Journal on Control and Optimization*, and IEEE TRANSACTIONS ON SYSTEMS, MAN AND CYBERNETICS: SYSTEMS.

Ji-Feng Zhang (Fellow, IEEE) received the B.S. degree in mathematics from Shandong University, China, in 1985, and the Ph.D. degree in systems theory from the Institute of Systems Science (ISS), Chinese Academy of Sciences (CAS), China, in 1991.

Since 1985, he has been with the ISS, CAS. He is currently with the School of Automation and Electrical Engineering, Zhongyuan University of Technology, Zhengzhou, China. His current research interests include system modeling, adaptive control, stochastic systems, and multiagent systems.

Dr. Zhang is an IFAC Fellow, CAA Fellow, CSIAM Fellow, Member of the European Academy of Sciences and Arts, and Academician of the International Academy for Systems and Cybernetic Sciences. He was the recipient of the second prize of the State Natural Science Award of China in 2010 and 2015, respectively. He was a Vice-Chair of the IFAC Technical Board, Member of the Board of Governors, IEEE Control Systems Society; Convenor of Systems Science Discipline, Academic Degree Committee of the State Council of China; Vice-President of the Systems Engineering Society of China, the Chinese Mathematical Society, and the Chinese Association of Automation. He was the Editor-in-Chief, Deputy Editor-in-Chief, Senior Editor, or Associate Editor for more than 10 journals, including *Science China Information Sciences*, *National Science Review*, IEEE TRANSACTIONS ON AUTOMATIC CONTROL, and *SIAM Journal on Control and Optimization* etc.

Authorized licensed use limited to: Yunnan University. Downloaded on April 14,2025 at 13:45:04 UTC from IEEE Xplore. Restrictions apply.