Measurement Dissemination-based Distributed Bayesian Filter using the Latest-In-and-Full-Out Exchange Protocol for Networked Unmanned Vehicles

Anonymous

Abstract—This paper presents a measurement dissemination-based distributed Bayesian filter (DBF) for a network of unmanned ground vehicles (UGVs). The DBF utilizes the Latest-In-and-Full-Out (LIFO) exchange protocol to disseminate the sensor measurements within local neighbors. Different from existing statistics dissemination strategies that transmit posterior distributions or likelihood functions, each UGV under LIFO only exchanges latest available measurements, which significantly reduces the transmission burden between each pair of UGVs to scale linearly with the network size. Under the condition of fixed and undirected topology, LIFO can guarantee non-intermittent dissemination of all measurements over the network within a finite time. Two types of LIFO-based DBF algorithms are then derived to estimate individual probability density function (PDF) for the static target and moving target, respectively. For the former, each UGV locally fuses the newly received measurements, while for the latter, a set of measurement history is stored and sequentially fused. The consistency of LIFO-based DBF is proved that the estimated target position converges in probability to the true target position. The effectiveness of this method is experimentally demonstrated by the target localization via multiple mobile robots with sonar sensors in an indoor environment.

Index Terms—Bayesian filter, Distributed estimation, Networked vehicles, Nonlinear filter, Unmanned vehicles.

I. INTRODUCTION

Distributed filtering that focuses on using a group of networked UGVs to collectively infer environment status has been used for various applications, such as intruder detection [1], moving target tracking [2] and object localization [3]. Several techniques have been developed for distributed filtering. For example, Olfati-Saber (2005) proposed a distributed Kalman filter (DKF) for estimating states of linear systems with Gaussian process and measurement noise [4]. Each DKF used low-pass and band-pass consensus filters to compute the average of weighted measurements and inverse-covariance matrices. Madhavan et al. (2004) presented a distributed extended Kalman filter for nonlinear systems [5]. This filter was used to generate local terrain maps by using pose estimates to combine elevation gradient and vision-based depth with environmental features. Gu (2007) proposed a distributed particle filter for Markovian target tracking over an undirected sensor network [6]. Gaussian mixture models (GMM) were adopted to approximate the posterior distribution from weighted particles, and the parameters of GMM were exchanged via average consensus filter. As a generic filtering scheme for general system dynamics and arbitrary noise distributions, distributed Bayesian filters (DBF) have received increasing interest during

past years [7], [8], which are also the focus of this study. It is worth noting that Bayesian filters can be reduced to Kalman filters and particle filters under appropriate conditions [9].

The design of distributed filtering algorithms depends on the communication topology of a multi-UGV network, which can be classified into two types: fusion center (FC)-based and neighborhood (NB)-based. In the FC-based approaches, each UGV uses a filter to estimate local statistics of environment status based on its own measurement. The local statistics are transmitted (possibly via multi-hopping) to a single FC, where a global posterior distribution (or statistical moments in DKF [10]) is calculated at each filtering cycle after receiving all local information [11], [12]. In the NB-based approaches, a set of UGVs executes distributed filters to estimate individual posterior distribution. The consensus of individual estimates is achieved by solely communicating statistics and/or measurements within local neighbors. The NB-based methods have become popular in recent years, since such approaches do not require complex routing protocols and global network knowledge and are therefore rather robust to topological changes and link failures.

So far, most studies on NB-based distributed filtering have focused on the so-called *statistics dissemination* strategy. This strategy directly exchanges environment statistics, including posterior distributions and likelihood functions, within neighboring UGVs [13]. It can be further categorized into two types: leader-based and consensus-based. In the former, statistics are sequentially passed and updated along a path formed by active UGVs, called leaders. Only the leaders can perform filtering based on its own and the received measurements from local neighbors. For example, Sheng et al. (2005) proposed a multiple leader-based distributed particle filter with Gaussian Mixer for target tracking [14]. Sensors are grouped into multiple uncorrelated cliques, in each of which a leader is assigned to perform particle filtering and exchanges particle information with other leaders. In the consensus-based distributed filters, every UGV diffuses environment statistics among neighbors, via which the global agreement of the statistics is achieved by using multi-agent consensus laws [10], [15], [16]. For example, Hlinka et al. (2012) proposed a distributed method for computing an approximation of the joint (all-sensors) likelihood function by means of weighted-linear-average consensus algorithm when local likelihood functions belong to the exponential family of distributions [17]. Saptarshi et al. (2014) presented a Bayesian consensus filter that uses a logarithmic opinion pool for fusing posterior distributions of the tracked target [7]. Other examples can be found in [8] and [18].

Despite the popularity of environment statistics dissemina-

tion strategies, exchanging statistics can consume excessive communication resources. One promising remedy is to disseminate sensor measurement instead of environment statistics among neighbors. One early work was done by Coates et al. (2004), which used adaptive encoding of sensor measurements to minimize communication overhead [19]. A recent study was conducted by Djuric et al. (2011), which proposed to broadcast raw measurements to other agents, and therefore each UGV had a complete set of measurements of other UGVs for executing particle filtering [20]. A shortcoming of aforementioned works is that their communication topologies are assumed to be a complete graph: a UGV can directly exchange information with any other UGV via a single hopping even they are far from each other. This is a strong assumption on the communication network and is not always feasible in reality [7], [16]. For incomplete graphs, Leung et al. (2010) explored a decentralized filter for dynamic robot networks [21]. The algorithm was shown to achieve centralized-equivalent filtering performance in simulations. However, it required the communication of both measurements and statistics, which incurred large communication overhead.

In this work, we propose a novel measurement dissemination-based distributed Bayesian filtering approach for target localization using networked UGVs. The main contributions of this paper include: (a) Different from existing works that assume full connectivity of the communication topology, each UGV only needs to broadcast the sensor measurements to its neighbors via single hopping and then implements individual Bayesian filter locally using its own measurements and the ones transmitted from neighbors. (b) We introduce the Latest-In-and-Full-Out (LIFO) protocol to reduce the communication burden, with the transmission data scaling linearly with the UGV number. (c) The proposed LIFO-based DBF has the following properties: For a fixed and undirected network, LIFO guarantees the global dissemination of measurements over the network in a non-intermittent manner. The corresponding DBF ensures the consistency of estimated target position, i.e., the estimate converges in probability to the true value when the number of measurements tends to infinity.

The rest of this paper is organized as follows: The problem of distributed Bayesian filtering is formulated in Section II. The LIFO-based DBF algorithm is described in Section III, followed by the proof of consistency in Section IV. Simulation and experiment results are presented in Section V and Section VI, respectively. The Section VII concludes the paper.

II. PROBLEM FORMULATION

Consider a network of N UGVs to localize a target in a bounded two-dimensional space $S \in \mathbb{R}^2$. Each UGV is equipped with a sensor for environment perception. Due to the limit of communication range, each UGV can only share information with its local neighbors. The Bayesian filter is run on each UGV based on its own measurements and the measurements of other UGVs to estimate the target position.

A. Target and Sensor Model

The target motion takes a deterministic discrete-time model:

$$x^g_{k+1} = f(x^g_k, u^g_k), \tag{1}$$

where the superscript g represents the target; $x^g_k \in S$ is the target position at time k and u^g_k is the target control input.

The sensor measurement is described by a stochastic model:

$$z^i_k = h_i(x^g_k, x^i_k) + w^i_k, \tag{2}$$

where the superscript $i \in \{1, \dots, N\}$ represents the index of each UGV; $x^i_k \in S$ is the sensor position and w^i_k is the white measurement noise. The measurement function h_i depends on the type of the sensor.

The information of the conditional probability of a certain measurement z^i_k is critical to designing the updating procedure of a Bayesian filter [22]. The conditional probability of z^i_k given the sensor state x^i_k and target state x^g_k is denoted as $P(z^i_k|x^g_k; x^i_k)$, which depends on the distribution of the measurement noise. For example, if w^i_k is a zero-mean Gaussian white noise with covariance Γ^i_k, then, according to Eq. (2), $P(z^i_k|x^g_k; x^i_k)$ can be described as

$$P(z^i_k|x^g_k; x^i_k) = \mathcal{N}(h_i(x^g_k, x^i_k), \Gamma^i_k)^1.$$

For non-Gaussian noise distributions, such as the Poisson noise or Cauchy noise [23], $P(z^i_k|x^g_k)$ can also be defined. It should be noted that, the approaches presented in this work do not rely on specific distributions of noise.

The h_i of several typical sensors are defined as follows [24]:

Range-only sensors: $h_i(x^g_k, x^i_k) = \|x^g_k - x^i_k\|_2$, where $\|\cdot\|_2$ is the Euclidean distance in S.

Bearing-only sensors: $h_i(x^g_k, x^i_k) = \measuredangle(x^g_k - x^i_k)$, where $\measuredangle$ denotes the angle from the sensor to the target.

Range-bearing sensors: $h_i(x^g_k, x^i_k) = x^g_k - x^i_k$.

Remark 1. Given the knowledge of current target and UGV positions, current measurement by each UGV can be considered conditionally independent from its own past measurements and those by other UGVs [25].

B. Graphical Model of Communication Topology

The UGV network is assumed to be connected, i.e., there exists a path, either direct or indirect, between every pair of UGVs. Under this assumption, consider an undirected and fixed graph $G = (V, E)$, where $V = \{1, \dots, N\}$ represents the index set of UGVs and $E = V \times V$ denotes the edge set. The adjacency matrix $A = [A_{(ij)}]$ describes the communication topology of G:

$$A_{(ij)} = \begin{cases} 1 & \text{if } (i,j) \in E \\ 0 & \text{if } (i,j) \notin E \end{cases},$$

where $A_{(ij)}$ denotes the entity of the adjacency matrix. The notation $A_{(ij)} = 1$ indicates that a communication link exists between i^{th} and j^{th} UGV and $A_{(ij)} = 0$ indicates no direct communication between them.

The *direct neighborhood* of ith UGV is defined as $\mathcal{N}_i = \{j | A_{(ij)} = 1, \forall j \in \{1, \dots, N\} \setminus \{i\}\}$. All the UGVs in $\mathcal{N}_i$ can directly exchange information with i^{th} UGV via single hopping. We also define another set $\mathcal{Q}_i$, called *available*

---

[1] For the purpose of simplicity, we will not explicitly write the parameter x^i_k in $P(z^i_k|x^g_k; x^i_k)$ for the rest of the paper.
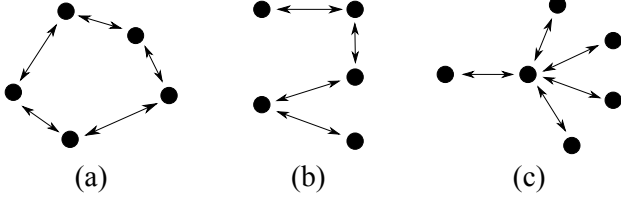
Fig. 1: Three types of topologies: (a) ring topology; (b) line topology; (c) star topology

neighborhood, that contains indices of UGVs whose measurements can be received by the ith UGV via single or multiple hoppings. Therefore $\mathcal{N}_i \subseteq \mathcal{Q}_i$. Fig. 1 illustrates three types of typical undirected and connected topologies: ring [26], line [27], and star [28].

C. Distributed Bayesian Filter for Multiple UGVs

The generic distributed Bayesian filter (DBF) is introduced in this section. Let $X_k \in S$ be the random variable that represents the position of the target at time k. The probability density function (PDF) of X_k, called *individual PDF*, of ith UGV is then represented by $P^i_{pdf}(X_k|\mathbf{z}^i_{1:k})$, where $\mathbf{z}^i_{1:k}$ denotes the set of measurements by i^{th} UGV and by UGVs in $\mathcal{Q}_i$, that have been received by i^{th} UGV until time k. The initial individual PDF, $P^i_{pdf}(X_0)$, is constructed given all available prior information including past experience and environment knowledge. It is necessary to initialize the individual PDF such that the probability density of true target position is nonzero, i.e., $P^i_{pdf}(X_0 = x^g_0) \neq 0$.

Under the framework of DBF, the individual PDF is recursively estimated by two steps: the prediction step and the updating step.

1) Prediction: At time k, the prior individual PDF $P^i_{pdf}(X_{k-1}|\mathbf{z}^i_{1:k-1})$ is first predicted forward by using the Chapman-Kolmogorov equation:

$$P^i_{pdf}(X_k|\mathbf{z}^i_{1:k-1}) = \int\limits_{X_{k-1}\in S} P(X_k|X_{k-1})P^i_{pdf}(X_{k-1}|\mathbf{z}^i_{1:k-1})dX_{k-1},$$

(3)

where $P(X_k|X_{k-1})$ represents the state transition probability of the target, based on the Markovian motion model (Eq. (1)). For the deterministic motion model, the state transition probability is simplified to be

$$P(X_k = c_k|X_{k-1} = c_{k-1}) = \begin{cases} 1 & \text{if } c_k = f(c_{k-1}, u^g_{k-1}) \\ 0 & \text{otherwise} \end{cases}. \quad (4)$$

2) Updating: The ith individual PDF is then updated by Bayes' theorem using the set of newly received measurements at time k, i.e., $\mathbf{z}^i_k$:

$$P^i_{pdf}(X_k|\mathbf{z}^i_{1:k}) = K_i P^i_{pdf}(X_k|\mathbf{z}^i_{1:k-1})P(\mathbf{z}^i_k|X_k), \quad (5)$$

where $P(\mathbf{z}^i_k|X_k)$ comes from the sensor model and K_i is a normalization factor, given by:

$$K_i = \left[\int\limits_{X_k\in S} P^i_{pdf}(X_k|\mathbf{z}^i_{1:k-1})P(\mathbf{z}^i_k|X_k)dX_k \right]^{-1}.$$

III. DISTRIBUTED BAYESIAN FILTER VIA LATEST-IN-AND-FULL-OUT PROTOCOL

This study proposes a Latest-In-and-Full-Out (LIFO) protocol for measurement exchange and derives corresponding distributed Bayesian filtering (DBF) algorithm, shorted as LIFO-DBF. The data communication in LIFO is synchronized with the execution of DBF. In each step, LIFO only allows single-hopping communication within the direct neighborhood, but is able to broadcast measurements of each UGV to any other agent after a finite number of steps. The individual PDF is forward predicted and updated in DBF after each LIFO cycle. The theoretical analysis show that LIFO-DBF can ensure the consistency of distributed estimation while requiring much less communication burden than statistics dissemination-based methods.

A. Latest-In-and-Full-Out (LIFO) Protocol

Under LIFO, each UGV contains a communication buffer (CB) to store its latest knowledge of measurements of all UGVs:

$$\mathbf{z}^{CB,i}_k = \left[z^1_{k^i_1}, \ldots, z^N_{k^i_N}, x^1_{k^i_1}, \ldots, x^N_{k^i_N} \right],$$

where $z^j_{k^i_j}$ represents the measurement made by j^{th} UGV at time k^i_j and $x^j_{k^i_j}$ denotes the sensor position when the associated measurement $z^j_{k^i_j}$ is made[2]. At time k, $z^j_{k^i_j}$ is received and stored in i^{th} UGV's CB, in which k^i_j is the latest time of jth UGV's measurement that is available to i^{th} UGV. Due to the communication delay, $k^i_j < k, \forall j \neq i$ and $k^i_i = k$ always hold. The **LIFO protocol** is stated in Algorithm 1. Note that under LIFO, $\mathcal{Q}_i = \{1, \ldots, N\} \setminus \{i\}$, which will be proved in Corollary 1. Fig. 2 illustrates the LIFO cycles with 3 UGVs using a line topology. For general graphs, we have the following proposition:

Proposition 1. *For a fixed and undirected network of N UGVs, the latest measurements of ith and j^{th} UGV are exchanged via the shortest path(s) under LIFO. The delay of exchange $\tau_{i,j}$ is equivalent to the length of the shortest path(s) between them.*

Proof. Since the network is connected, there exists a minimum integer $\tau_{i,j}$ such that $A^{\tau_{i,j}}_{(i,j)} > 0$ and $\tau_{i,j}$ is the length of a shortest path between i^{th} and j^{th} UGV [29]. Under the LIFO, the latest measurement of ith UGV will always be propagated via the CBs of UGVs along a shortest path between i^{th} and j^{th} UGV. Therefore, the latest measurement that j^{th} UGV receives from i^{th} UGV is delayed by $\tau_{i,j}$ rounds of communication. $\square$

Corollary 1. *For the same topology assumption in Proposition 1, all elements in $\mathbf{z}^{CB,i}_k$ under LIFO become nonempty when $k \geq N$. This implies $\mathcal{Q}_i = \{1, \ldots, N\} \setminus \{i\}$.*

[2]For the purpose of simplicity, we will not explicitly write $\left[x^1_{k^i_1}, \ldots, x^N_{k^i_N} \right]$ in $\mathbf{z}^{CB,i}_k$ for the rest of the paper.

Algorithm 1 LIFO Protocol

(1) Initialization: The CB of ith UGV is initialized when $k = 0$:
$$z^j_{k^i_j} = \varnothing, \; k^i_j = 0, \; j = 1, \ldots, N$$

(2) At k^{th} step for i^{th} UGV :

(2.1) Receiving Step:

The ith UGV receives all CBs of its direct neighborhood $\mathcal{N}_i$, each of which corresponds to the (k-1)-step CB of a UGV in $\mathcal{N}_i$. The received CB from l^{th} ($l \in \mathcal{N}_i$) UGV is denoted as
$$\mathbf{z}^{CB,l}_{k-1} = \left[z^1_{(k-1)^l_1}, \ldots, z^N_{(k-1)^l_N} \right], \; l \in \mathcal{N}_i$$

(2.2) Observation Step:

The ith UGV updates $z^j_{k^i_j}$ ($j = i$) by its own measurement at current step:
$$z^j_{k^i_j} = z^i_k, \; k^i_j = k, \; \text{if } j = i.$$

(2.3) Comparison Step:

The ith UGV updates other elements of its own CB, i.e., $z^j_{k^i_j}$ ($j \neq i$), by selecting the latest information among all received CBs from $\mathcal{N}_i$. For all $j \neq i$,
$$l_{\text{latest}} = \operatorname*{argmax}_{\{l \in \mathcal{N}_i, \, i\}} \left\{ (k-1)^i_j, (k-1)^l_j \right\}$$
$$z^j_{k^i_j} = z^j_{(k-1)^{l_{\text{latest}}}_j}, \; k^i_j = (k-1)^{l_{\text{latest}}}_j$$

(2.4) Sending Step:

The ith UGV broadcasts its updated CB to all of its neighbors defined in $\mathcal{N}_i$.
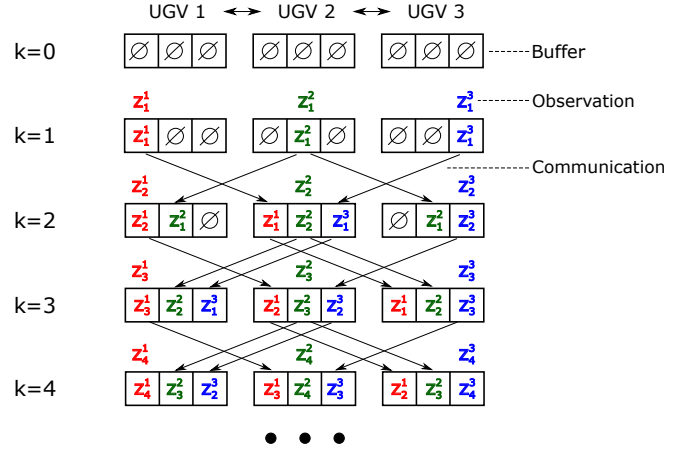
(3) $k \leftarrow k + 1$ until stop

Fig. 2: Example of LIFO with three UGVs using line communication topology. In this case, $\mathcal{N}_1 = \{2\}$, $\mathcal{Q}_1 = \{2, 3\}$; $\mathcal{N}_2 = \{1, 3\}$, $\mathcal{Q}_2 = \{1, 3\}$; $\mathcal{N}_3 = \{2\}$, $\mathcal{Q}_3 = \{1, 2\}$.

Corollary 2. *For the same topology assumption in Proposition 1, once elements in $\mathbf{z}^{CB,i}_k$ are nonempty, the updating of each element is non-intermittent. Moreover, $k - N < k^i_j \leq k$.*

Compared to statistics dissemination, LIFO is generally more communication-efficient for distributed filtering. To be specific, assume the environment S is represented by an $M \times M$ grid. For a network of N UGVs, the transmitted data of LIFO between each pair of UGVs are only the CB of each UGV and the corresponding sensor states, the size of which is $O(N)$, scaling linearly with UGV number. On the contrary, the transmitted data for a statistics dissemination approach that transmits unparameterized posterior distributions or likelihood functions is $O(M^2)$, which is in the order of the environment size. Since M is generally much larger than N in applications such as target localization and environment exploration, LIFO requires much less communication resources.

B. Algorithm of LIFO-DBF for Static Target

This section derives the LIFO-DBF algorithm for localizing a static target. It is assumed that all UGVs know the sensor model of other UGVs. According to Corollary 2, $\mathbf{z}^i_k$ and $\mathbf{z}^i_{1:k}$ in Eq. (5) equals $\mathbf{z}^{CB,i}_k$ and $\mathbf{z}^{CB,i}_{1:k}$, respectively. The assumption of static target can simplify the Bayesian filter as the prediction step becomes trivial. At time k, the ith UGV

starts from the last-step individual PDF that is locally stored, i.e., $P^i_{pdf}(X|\mathbf{z}^i_{1:k-1})$[3]. The ith individual PDF is then updated by fusing all measurements in $\mathbf{z}^i_k$:

$$P^i_{pdf}(X|\mathbf{z}^i_{1:k}) = K_i P^i_{pdf}(X|\mathbf{z}^i_{1:k-1}) P(\mathbf{z}^i_k|X)$$
$$= K_i P^i_{pdf}(X|\mathbf{z}^i_{1:k-1}) \prod_{j=1}^{N} P(z^j_{k^i_j}|X),$$

where
$$K_i = \left[\int_{X \in S} P^i_{pdf}(X|\mathbf{z}^i_{1:k-1}) \prod_{j=1}^{N} P(z^j_{k^i_j}|X) dX \right]^{-1}.$$

C. Algorithm of LIFO-DBF for Moving Target

This section derives the LIFO-DBF for localizing a moving target. Instead of storing last-step PDF, at time k each UGV maintains an individual PDF of time $(k - N)$ and a collection of measurement history, called the *record set*, from time $(k - N + 1)$ to k. The ith individual PDF is then alternatively predicted and updated by using the aforementioned Bayesian filter (Eq. (3) and (5)) from $(k - N)$ to k to obtain $P^i_{pdf}(X_k|\mathbf{z}^i_{1:k})$.

Fig. 3 illustrates the LIFO-DBF procedure for the 1st UGV as an example [4]. Ω^i_ξ ($\xi = 1, \ldots, N$) denote the index set of UGVs whose measurement at time $(k - N + \xi)$ is stored in i^{th} UGV's record set, i.e. $\Omega^i_\xi = \left\{ j \in \mathcal{Q}_i \bigcup \{i\} \,|\, z^j_{k-N+\xi} \text{ is in the record set} \right\}$. The **LIFO-DBF algorithm** for moving target is stated in Algorithm 2.

It is worth noting that, for the static target, each UGV only needs current-step CB to update individual PDFs. Therefore, besides storing its own individual PDF of size $O(M^2)$, only current-step CB of size $O(N)$ is stored in an UGV's memory and all previous CBs can be discarded, which means that the

---

[3]Since the target is static, we drop the subscript k in X_k.

[4]Due to the space limit, in this figure we use $P^i_{pdf}(k)$, $P^i_{pdf}(k - N)$ and $P^i_{pdf}(k - N + 1)$ to represent $P^i_{pdf}(X|\mathbf{z}^i_{1:k})$, $P^i_{pdf}(X|\mathbf{z}^i_{1:k-N})$ and $P^i_{pdf}(X|\mathbf{z}^i_{1:k-N+1})$, respectively.

UGV 1 ↔ UGV 2 ↔ • • • ↔ UGV N
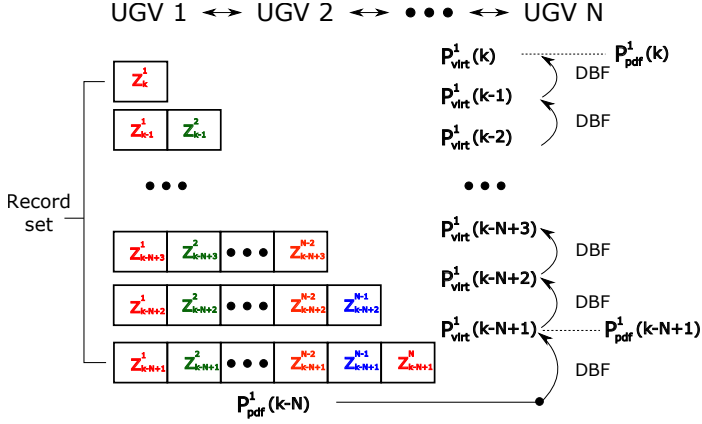
Fig. 3: Example of LIFO-DBF for 1st UGV at time k. Networked UGVs take a line topology. The stored individual PDF is represented by $P_{pdf}^1(k-N)$. The UGV first calculates $P_{virt}^1(k-N+1)$, defined in Algorithm 2, and then stores it as $P_{pdf}^1(k-N+1)$. Repeating DBF until obtaining $P_{pdf}^1(k)$. In this example, $\Omega_\xi^1 = \{1, 2, \ldots, N+1-\xi\}$, $\xi = 1, \ldots, N$.

size of needed memory is $O(N + M^2)$. On the contrary, for the moving target, each UGV needs to store a set of measurement history of size $O(N^2)$ and an individual PDF of size $O(M^2)$. Therefore the size of the needed memory for each UGV is $O(M^2 + N^2)$. This is generally larger than that of statistics dissemination-based methods, the memory of which is $O(M^2)$. Besides, additional computation power is needed for LIFO-DBF compared to statistics dissemination-based methods. Therefore, LIFO-DBF sacrifices storage space and computation resource for reducing communication burden. This is actually desirable for real applications as local memory of vehicles is usually abundant compared to the limited bandwidth for communication.

IV. PROOF OF CONSISTENCY

This section proves the consistency of the maximum a posteriori (MAP) estimator of LIFO-DBF under unbiased sensors (sensors without offset). An estimator of a state is said to be consistent if it converges in probability to the true value of the state [30]. Consistency is an important metric for stochastic filtering approaches [9] and it differs from the concept of consensus; consensus implies that the estimation results of all sensors converge to a same value, while consistency not only implies achieving consensus asymptotically, but also requires the estimated value converge to the true value.

We first prove the consistency for static UGVs. The consistency for moving UGVs is subsequently proved. For simplicity and clarity, we assume S is a finite set (e.g. a finely discretized field) and the target motion is relatively slow compared to the filtering dynamics.

A. Static UGVs

The consistency of LIFO-DBF for static UGVs is stated as follows:

Theorem 1. *Assume the UGVs are static and the sensors are unbiased, then the MAP estimator of target position converges*

Algorithm 2 LIFO-DBF Algorithm for Moving Target

For i^{th} UGV at k^{th} step:
After updating CB by Algorithm 1,
(1) Initialize a *virtual PDF* by assigning the stored individual PDF to it:

$$P_{virt}^i(X_{k-N}) = P_{stored}^i,$$

where the stored individual PDF is for time $(k-N)$:

$$P_{stored}^i = P_{pdf}^i(X_{k-N}|z_{1:k-N}^1, \ldots, z_{1:k-N}^N).$$

(2) For $\xi = 1$ to N, iteratively repeat two steps of Bayesian filtering:
(2.1) Prediction

$$P_{virt}^{pre}(X_{k-N+\xi})$$
$$= \int_S P(X_{k-N+\xi}|X_{k-N+\xi-1})P_{virt}^i(X_{k-N+\xi-1})dX_{k-N+\xi-1}.$$

(2.2) Updating

$$P_{virt}^i(X_{k-N+\xi}) = K_\xi P_{virt}^{pre}(X_{k-N+\xi}) \prod_{j\in\Omega_\xi^i} P(z_{k-N+\xi}^j|X_{k-N+\xi}).$$

$$K_\xi = \left[\int_S P_{virt}^{pre}(X_{k-N+\xi}) \prod_{j\in\Omega_\xi^i} P(z_{k-N+\xi}^j|X_{k-N+\xi})dX_{k-N+\xi} \right]^{-1}.$$

(2.3) When $\xi = 1$, if $z_{k-N+1}^j \neq \emptyset$ for $\forall j \in \{1, \ldots, N\}$, then the virtual PDF is equivalent to the individual PDF for time $(k-N+1)$. Store it to replace the old PDF:

$$P_{stored}^i = P_{virt}^i(X_{k-N+1}),$$

where

$$P_{virt}^i(X_{k-N+1}) = P_{pdf}^i(X_{k-N+1}|z_{1:k-N+1}^1, \ldots, z_{1:k-N+1}^N).$$

(3) Individual PDF of ith UGV at time k is $P_{pdf}^i(X_k|\mathbf{z}_{1:k}^i) = P_{virt}^i(X_k)$.

---

in probability to the true position of the target using LIFO-DBF, i.e.,

$$\lim_{k\to\infty} P(X_k^{i,MAP} = x^g) = 1, \ i = 1, \ldots, N,$$

where

$$X_k^{i,MAP} = \arg\max_X P_{pdf}^i(X|\mathbf{z}_{1:k}^i).$$

Proof. First we look at the asymptotic behavior of $P_{pdf}^i(X|\mathbf{z}_{1:k}^i)$. The batch form of DBF at k^{th} step is:

$$P_{pdf}^i(X|\mathbf{z}_{1:k}^i) = P_{pdf}^i(X|z_{1:k_1^i}^1, \ldots, z_{1:k_N^i}^N) \quad (6a)$$

$$= \frac{P_{pdf}^i(X) \prod_{j=1}^N \prod_{l=1}^{k_j^i} P(z_l^j|X)}{\sum_{X\in S} P_{pdf}^i(X) \prod_{j=1}^N \prod_{l=1}^{k_j^i} P(z_l^j|X)}. \quad (6b)$$

The decomposition into multiplication in Eq. (6b) results from the conditional independence of measurements given X.

Comparing the conditional probability of an arbitrary position $x \in S$ with that of target position x^g:

$$\frac{P_{pdf}^i(X = x|\mathbf{z}_{1:k}^i)}{P_{pdf}^i(X = x^g|\mathbf{z}_{1:k}^i)} = \frac{P_{pdf}^i(x) \prod\limits_{j=1}^{N} \prod\limits_{l=1}^{k_j^i} P(z_l^j|x)}{P_{pdf}^i(x^g) \prod\limits_{j=1}^{N} \prod\limits_{l=1}^{k_j^i} P(z_l^j|x^g)}.^5 \quad (7)$$

Take the logarithm of Eq. (7) and average it over k steps:

$$\frac{1}{k} \ln \frac{P_{pdf}^i(X = x|\mathbf{z}_{1:k}^i)}{P_{pdf}^i(X = x^g|\mathbf{z}_{1:k}^i)} = \frac{1}{k} \ln \frac{P_{pdf}^i(x)}{P_{pdf}^i(x^g)} + \sum_{j=1}^{N} \frac{1}{k} \sum_{l=1}^{k_j^i} \ln \frac{P(z_l^j|x)}{P(z_l^j|x^g)}.$$
$$(8)$$

Since $P_{pdf}^i(x)$ and $P_{pdf}^i(x^g)$ are bounded and $P_{pdf}^i(X)$ is initialized such that $P_{pdf}^i(x^g) \neq 0$, then

$$\lim_{k \to \infty} \frac{1}{k} \ln \frac{P_{pdf}^i(x)}{P_{pdf}^i(x^g)} = 0.$$

Utilizing the facts: (1) z_l^j are conditionally independent samples from $P(z_l^j|x^g)$ and (2) $k - N < k_j^i \leq k$, the law of large numbers yields

$$\frac{1}{k} \sum_{l=1}^{k_j^i} \ln \frac{P(z_l^j|x)}{P(z_l^j|x^g)} \xrightarrow{P} \mathbb{E}_{z_l^j} \left[\ln \frac{P(z_l^j|x)}{P(z_l^j|x^g)} \right] \quad (9a)$$

$$= \int_{z_l^j} P(z_l^j|x^g) \ln \frac{P(z_l^j|x)}{P(z_l^j|x^g)} dz_l^j \quad (9b)$$

$$= -D_{KL}\left(P(z_l^j|x^g) \| P(z_l^j|x) \right), \quad (9c)$$

where "$\xrightarrow{P}$" represents "convergence in probability" and $D_{KL}(P_1\|P_2)$ denotes the Kullback-Leibler (KL) divergence between two probability distributions P_1 and P_2. The KL divergence has the property [31] that $\forall P_1, P_2,$

$$D_{KL}(P_1\|P_2) \leq 0, \text{ equality holds iff } P_1 = P_2.$$

This leads to the following results:

$$\lim_{k \to \infty} \frac{1}{k} \sum_{l=1}^{k_j^i} \ln \frac{P(z_l^j|x)}{P(z_l^j|x^g)} \begin{cases} < 0, & \text{if } x \neq x^g \\ = 0, & \text{if } x = x^g \end{cases}.$$

Then by considering the limit of Eq. (8) when $k \to \infty$, we get:

$$\lim_{k \to \infty} \frac{1}{k} \ln \frac{P_{pdf}^i(x|\mathbf{z}_{1:k}^i)}{P_{pdf}^i(x^g|\mathbf{z}_{1:k}^i)} \begin{cases} < 0, & \text{if } x \neq x^g \\ = 0, & \text{if } x = x^g \end{cases}. \quad (10)$$

Eq. (10) implies that

$$\frac{P_{pdf}^i(x|\mathbf{z}_{1:k}^i)}{P_{pdf}^i(x^g|\mathbf{z}_{1:k}^i)} \xrightarrow{P} \begin{cases} 0, & \text{if } x \neq x^g \\ 1, & \text{if } x = x^g \end{cases}.$$

Therefore,

$$\lim_{k \to \infty} P_{pdf}^i(X = x^g|\mathbf{z}_{1:k}^i) = 1.$$

Thus

$$\lim_{k \to \infty} P(X_k^{i,MAP} = x^g) = 1.$$

$\square$

---

^{5}For the purpose of simplicity, we abbreviate $P_{pdf}^i(X = x)$ and $P(z_l^j|X = x)$ as $P_{pdf}^i(x)$ and $P(z_l^j|x)$ in this proof.

B. Moving UGVs

This subsection considers the case of using moving UGVs to localize a target. The difficulty of consistency proof for this case lies in the fact that each UGV makes measurements at multiple positions. Here, the main idea of the proof is to classify UGV measurement positions into two disjoint sets: *infinite-measurement spots* that contain positions where a UGV visits infinitely many times as time tends to infinity, and *finite-measurement spots* that contain positions where the UGV visits finitely many times (i.e., the UGV does not visit again after a finite time period). Before stating the main theorem, the following lemma is introduced.

Lemma 1. *For a set of UGVs moving within a collection of finite positions, each UGV has at least one position where infinite measurements are made as k tends to infinity.*

Proof. Let $n_s^{i,k}$ denote the times that i^{th} UGV visits s^{th} position up to time k. Then, $\sum_{s \in S} n_s^{i,k} = k$. It is straightforward to see that $\exists n_s^{i,k}$, such that $n_s^{i,k} \to \infty$, as $k \to \infty$. $\square$

Theorem 2. *Assume UGVs move within a collection of finite positions and sensors are unbiased, then the MAP estimator of target position converges in probability to the true position of the target using LIFO-DBF, i.e.,*

$$\lim_{k \to \infty} P(X_k^{i,MAP} = x_k^g) = 1, \ i = 1, \ldots, N.$$

Proof. According to Theorem 1, the batch form of DBF at k^{th} step is

$$\frac{P_{pdf}^i(X = x|\mathbf{z}_{1:k}^i)}{P_{pdf}^i(X = x^g|\mathbf{z}_{1:k}^i)} = \frac{P_{pdf}^i(x) \prod\limits_{j=1}^{N} \prod\limits_{l=1}^{k_j^i} P(z_l^j|x; x_l^j)}{P_{pdf}^i(x^g) \prod\limits_{j=1}^{N} \prod\limits_{l=1}^{k_j^i} P(z_l^j|x^g; x_l^j)}. \quad (11)$$

The only difference from Eq. (7) is that $P(z_l^j|x; x_l^j)$ in Eq. (11) varies as the UGV moves. For each UGV, there exists at least one position where infinite measurements are made as $k \to \infty$, according to Lemma 1. For the finite-measurement spots, by referring to Eq. (9), it is easy to know that their contribution to Eq. (8) diminishes when $k \to \infty$. Therefore, proof using Eq. (11) can be reduced to only considering infinite-measurement spots and the rest of proof is similar to that of Theorem 1. $\square$

***Remark* 2.** The assumption of unbiased sensors are important for the consistency of the estimator. In fact, with unknown non-zero bias, the distribution of z_l^j differs from $P(z_l^j|x^g)$, which invalidates the derivation in Eq. (9) and the consistency proof. This assumption also makes intuitive sense. In the extreme case, if each sensor has a very large unknown measurement offset, then the estimated target position of each sensor (without communicating with other sensors) will be very different from each other's. Therefore, no common target position can be correctly obtained when they fuse measurements.
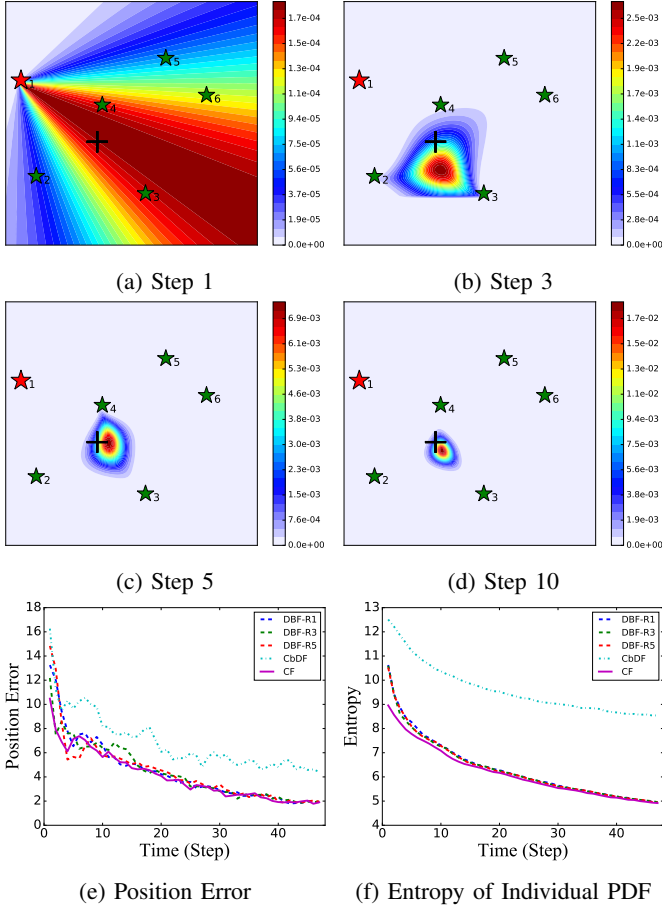
(a) Step 1   (b) Step 3

(c) Step 5   (d) Step 10

(e) Position Error   (f) Entropy of Individual PDF

Fig. 4: (a)-(d) The 1st UGV's individual PDF at different times. The cross stands for the target. Note that the value of the color bar varies among different figures. (e) Average position estimation error of the 1st, 3rd and 5th UGV's LIFO-DBF, the CbDF and the CF. (f) Average entropy of individual PDFs.

(a) Step 1   (b) Step 3

(c) Step 5   (d) Step 10

(e) Position Error   (f) Entropy of Individual PDF

Fig. 5: (a)-(d) The 5th UGV's individual PDF at different times. (e) Average position estimation error. (f) Average entropy of individual PDFs.

V. SIMULATION FOR TARGET LOCALIZATION

Simulation has been conducted to evaluate the effectiveness of LIFO-DBF for target localization using a team of six UGVs. The networked UGVs take a ring communication topology that each UGV can communicate with two fixed neighbors. DBF is implemented using the histogram filter method, by which the continuous space is finely discretized into finitely many regions and the individual PDF is approximated by the cumulative probability of each region [22]. Two types of scenarios are used: in the first scenario, both UGVs and the target are static; the second scenario subsequently deals with moving UGVs for localizing a moving target. In each scenario, we also test two different kinds of UGV team: the first UGV team, called the homogeneous team, is equipped with bearing-only sensors; in the second team, called the heterogeneous team, three UGVs are equipped with bearing-only sensor and the other three equipped with range-only sensors. The bearing-only sensors are assumed to have large enough sensing range to cover the simulation field and the measurement noise is a zero-mean Gaussian white noise with standard deviation being 0.5. The range-only sensors are assumed to have 360° field
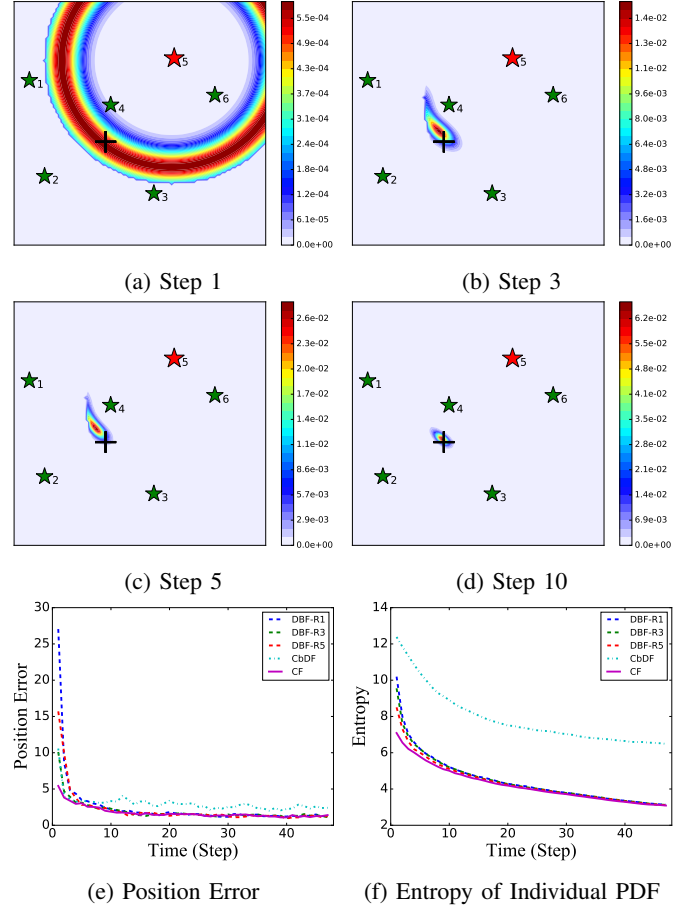
of view and the measurement noise is a zero-mean Gaussian white noise with standard deviation being 5.

In both scenarios, LIFO-DBF is compared with two commonly adopted approaches in multi-agent filtering: the consensus-based distributed filtering (CbDF) method [32] and the centralized filtering (CF) method [33]. The CbDF requires UGVs to continually exchange their individual PDFs with direct neighbors, computing the average of its own and the received PDFs. Multiple rounds of communication and averaging are conducted during the "Sending Step" and "Receiving Step" in Algorithm 1 at each step to ensure the convergence of UGV's individual PDFs. The CF assumes a central unit that can constantly receive and fuse all UGVs' latest measurements into a single PDF. Ten test trials with randomly generated initial UGV and target positions are run and each trial is terminated after 50 time steps. The average error between the estimated (using the MAP estimator) and true target position and the average entropy of individual PDFs of all 10 trials using these three approaches are compared.

A. Static UGVs, Static Target

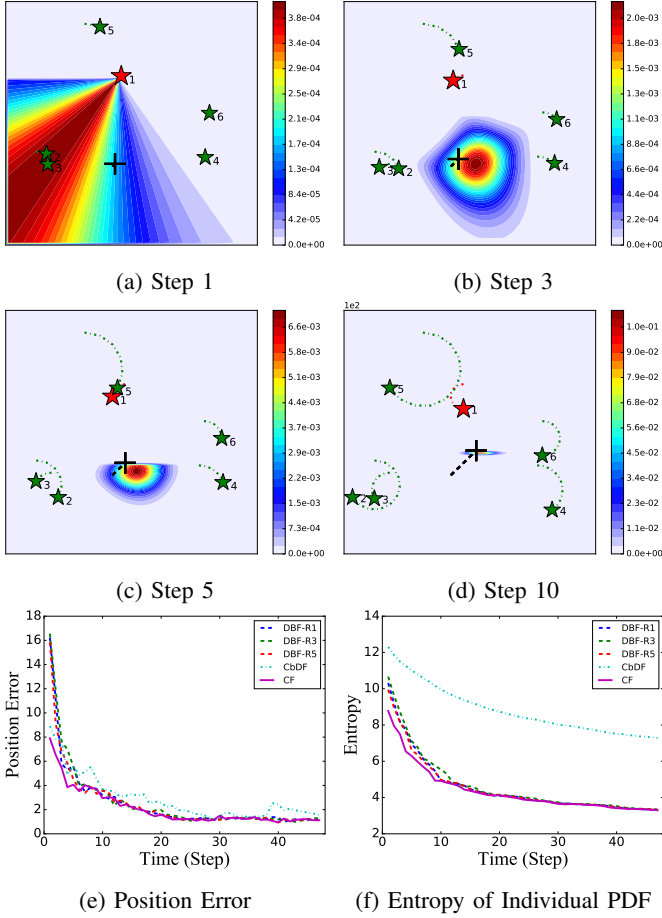The individual PDF of each UGV is initialized as a uniform distribution over a bounded space. At each time step, each

(a) Step 1

(b) Step 3

(c) Step 5

(d) Step 10

(e) Position Error

(f) Entropy of Individual PDF

Fig. 6: (a)-(d) The 1st UGV's individual PDFs at different times. The green dashed lines represent robots' trajectories. (e) Average position estimation error. (f) Average entropy of individual PDFs.

(a) Step 1

(b) Step 3

(c) Step 5

(d) Step 10

(e) Position Error

(f) Entropy of Individual PDF

Fig. 7: (a)-(d) The 4th UGV's individual PDF at different times. (e) Average position estimation error. (f) Average entropy of individual PDFs.

UGV executes the LIFO-DBF for static target (Section III-B) to update their estimation of target position. The evolution of the 1st robot's individual PDF is shown in Figures 4a to 4d. Since the robot is equipped with a noisy bearing-only sensor, the measurement at step 1 results in a wide distribution with the peak centered along the noise-corrupted measured direction from the sensor to the target, as Figure 4a illustrates. As more measurements are fused, the individual PDF asymptotically concentrates to the true location of the target, which accords with the consistency of LIFO-DBF.

The Figures 4e and 4f shows the comparison of LIFO-DBF with CbDF and CF. Unsurprisingly, the CF achieves the best performance in terms of both small position estimation error and fast reduction of entropy. This happens because the central unit has access to the latest measurements of all UGVs, thus being able to make the most use of all available information. It is worth noting that, LIFO-DBF achieves similar asymptotic performance as the CF, both in position estimation error and entropy reduction; this is achieved even though each UGV only communicates with its two neighboring UGVs, which requires less communication burden than the CF. The CbDF has the worst performance among these three filtering approaches. It
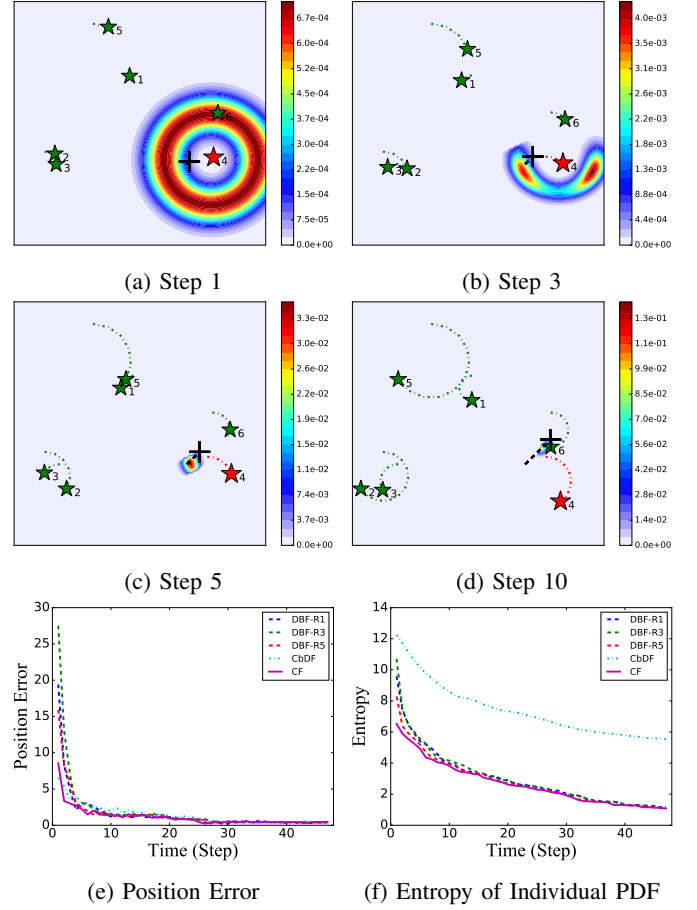
results in slow reduction of entropy and the position error remains large. This happens because Bayes filtering (Eq. (3) and (5)) is a nonlinear process. Using the linear average consensus law to fuse individual PDFs thus deviates from the actual Bayesian filtering and therefore cannot fully exploit the information of new measurements to reduce the uncertainty of estimation.

The Figure 5 shows the simulation results of the heterogeneous team. The individual PDF of the UGV with a noisy range-only sensor is presented in Figures 5a to 5d. At step 1, the target is detected and the individual PDF is updated such that the peak of the distribution centers at the positions whose distance to the sensor equals the noise-corrupted measured value. Similar to the case of using the homogeneous team, the individual PDF asymptotically concentrates to the true target position. Due to the use of various sensors, the estimation accuracy increases faster than the homogeneous team case, as shown in Figures 5e and 5f. It can also be noticed that, LIFO-DBF achieves similar performance as CF, while the performance of CbDF is the worst.

B. Moving UGVs, Moving Target

In this scenario, each UGV follows a pre-defined circular trajectory. The target motion is modeled as a single-integrator
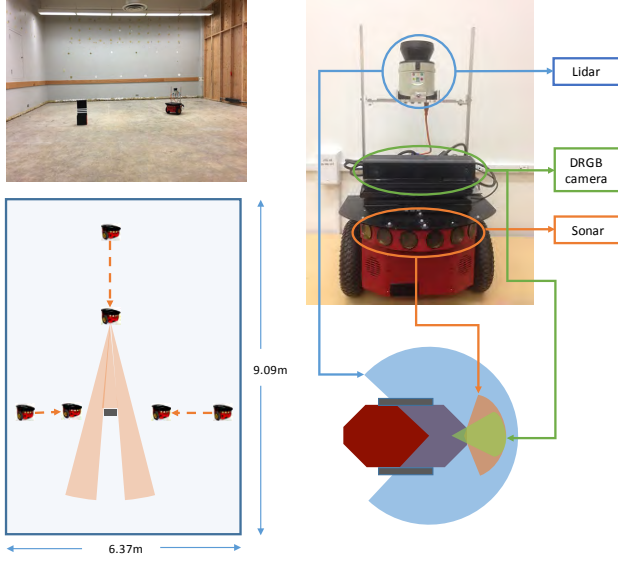
Fig. 8: Experiment Setup. Left: Three robots are used for localizing the target (black box). The room size is $9.09m \times 6.37m$. The orange sectors show the sonar sensing domain. Red dashed lines show the trajectory of each robot. Right: The mobile robot platform is equipped with different sensors. The sonar sensor is used for target localization.

with unit velocity on both directions. Each UGV executes the LIFO-DBF for moving target (Section III-C) for estimation. Figures 6a to 6d and Figures 7a to 7d illustrate the evolution of individual PDF for homogeneous and heterogeneous teams, respectively. They all present similar asymptotic behavior of individual PDFs as in aforementioned simulations.

Figs. 6e and 6f and Figs. 7e and 7f compare LIFO-DBF with CbDF and CF. It is worth noting that, for the heterogeneous team, CbDF achieves comparable position estimation error performance as the CF and LIFO-DBF. However, CbDF requires multiple rounds of exchanging individual PDFs, which incurs much higher communication burden than LIFO-DBF at each time step. Considering the small difference in position estimation error and significantly faster entropy reduction, LIFO-DBF is still preferable over CbDF for moving target scenario.

VI. Indoor Experiment using Mobile Robot

We have conducted a target localization experiment to validate the LIFO-DBF approach in an indoor environment. The experiment equipments is illustrated in Figure 8. Each robot is equipped with an onboard sonar sensor to measure the target (a small cardboard box) position. The sonar sensor has the sensing field of view of $25°$ and the range of $4.9m$. Its measurement noise is modeled as a zero-mean Gaussian random variable with standard deviation $0.51m$, which is rather inaccurate to use single robot to localize the target. The robots locally run the LIFO-DBF to estimate the target position. Each robot's probability map is constructed on an evenly spaced grid with $0.1m$ interval on each axis. In the experiment, robots move to different locations to measure the target position. We manually measure the robot positions where the sensor
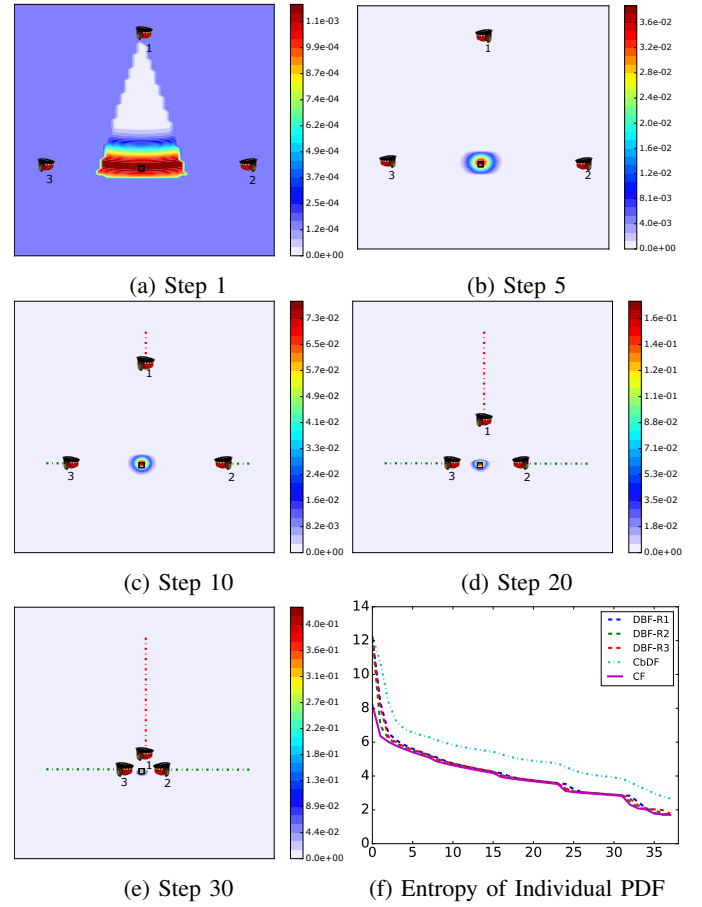

Fig. 9: The 1st robot's individual PDF at different time steps.

measurements are obtained to reduce the effects of localization error. The effect of the target size has been compensated during the post-processing of the measurement data.

Figures 9a to 9e show the evolution of the 1st robot's individual PDF. At step 1, the probability mass concentrates to a stripe due to the fact that the sonar sensor used in this experiment has limited FOV and noise in range measurement. As new measurements from different robots are continually fused, the probability mass gradually concentrates to the true position of the target. Similar to previous simulations, we compare our method with CbDF and CF. All three approaches achieve accurate position estimation. However, they differ in the uncertainty reduction, as shown in Figure 9f. The CF has the fastest entropy reduction and the LIFO-DBF achieves comparable performance, while the CbDF shows the slowest entropy reduction. These experiment results validate the consistency of LIFO-DBF and demonstrate its effectiveness for distributed filtering.

VII. CONCLUSION

This paper presents a measurement dissemination-based distributed Bayesian filtering (DBF) approach for a multi-UGV network, utilizing the Latest-In-and-Full-Out (LIFO) protocol for measurement exchange. By communicating the communication buffers among neighboring UGVs, LIFO significantly reduces the transmission burden between each pair of UGVs

to scale linearly with the network size. Two types of LIFO-DBF algorithms are proposed to estimate individual PDFs for static target and moving target, respectively. The consistency of LIFO-DBF is proved by utilizing the law of large numbers, which ensures that the estimated target position converges in probability to the true target position. The LIFO-DBF is compared with the consensus filter and the centralized filter in various simulations. Results have shown that LIFO-DBF achieves very similar performance as the centralized filter and is advantageous over the consensus filter. An experiment is also conducted and has demonstrated the effectiveness of LIFO-DBF for distributed filtering. The LIFO-DBF is promising for a wide range of applications using multiple robots, such as the environment monitoring, precision farming, and vehicle localization and mapping.

This work has opened up several directions for future work. First, we plan to extend the current LIFO-DBF to deal with some common issues in the measurement for networked unmanned vehicles, including the asynchronous clock, communication delay, dynamically changing network, and packet loss. Second, LIFO-DBF provides a general framework for distributed nonlinear filtering. In our future work, we will develop computationally efficient implementation of LIFO-DBF by using particle filters and Unscented Kalman filters. Lastly, we will combine the distributed filtering with path planning approaches [34] so that multiple robots can actively localize and track targets, which can be applied to search and rescue and the navigation of autonomous vehicles.

REFERENCES

[1] J.-F. Chamberland and V. V. Veeravalli, "Wireless sensors in distributed detection applications," *Signal Processing Magazine, IEEE*, vol. 24, no. 3, pp. 16–25, 2007.

[2] Z. Wang and D. Gu, "Cooperative target tracking control of multiple robots," *IEEE Transactions on Industrial Electronics*, vol. 59, no. 8, pp. 3232–3240, 2012.

[3] H. Song, V. Shin, and M. Jeon, "Mobile node localization using fusion prediction-based interacting multiple model in cricket sensor network," *IEEE Transactions on Industrial Electronics*, vol. 59, no. 11, pp. 4349–4359, 2012.

[4] R. Olfati-Saber, "Distributed kalman filter with embedded consensus filters," in *Decision and Control, 2005 and 2005 European Control Conference. CDC-ECC'05. 44th IEEE Conference on*, pp. 8179–8184.

[5] R. Madhavan, K. Fregene, and L. E. Parker, "Distributed cooperative outdoor multirobot localization and mapping," *Autonomous Robots*, vol. 17, no. 1, pp. 23–39, 2004.

[6] D. Gu, "Distributed particle filter for target tracking," in *Robotics and Automation, 2007 IEEE International Conference on*, pp. 3856–3861.

[7] S. Bandyopadhyay and S.-J. Chung, "Distributed estimation using bayesian consensus filtering," in *American Control Conference (ACC), 2014*, pp. 634–641.

[8] B. J. Julian, M. Angermann, M. Schwager, and D. Rus, "Distributed robotic sensor networks: An information-theoretic approach," *The International Journal of Robotics Research*, vol. 31, no. 10, pp. 1134–1154, 2012.

[9] Z. Chen, "Bayesian filtering: From kalman filters to particle filters, and beyond," *Statistics*, vol. 182, no. 1, pp. 1–69, 2003.

[10] R. Olfati-Saber, A. Fax, and R. M. Murray, "Consensus and cooperation in networked multi-agent systems," *Proceedings of the IEEE*, vol. 95, no. 1, pp. 215–233, 2007.

[11] L. Zuo, K. Mehrotra, P. K. Varshney, and C. K. Mohan, "Bandwidth-efficient target tracking in distributed sensor networks using particle filters," in *Information Fusion, 2006 9th International Conference on*.

[12] X. He, Z. Wang, X. Wang, and D. Zhou, "Networked strong tracking filtering with multiple packet dropouts: algorithms and applications," *IEEE Transactions on Industrial Electronics*, vol. 61, no. 3, pp. 1454–1463, 2014.

[13] O. Hlinka, F. Hlawatsch, and P. M. Djuric, "Distributed particle filtering in agent networks: A survey, classification, and comparison," *Signal Processing Magazine, IEEE*, vol. 30, no. 1, pp. 61–81, 2013.

[14] X. Sheng, Y.-H. Hu, and P. Ramanathan, "Distributed particle filter with gmm approximation for multiple targets localization and tracking in wireless sensor network," in *Proceedings of the 4th international symposium on Information processing in sensor networks*, p. 24.

[15] W. Ren, R. W. Beard, *et al.*, "Consensus seeking in multiagent systems under dynamically changing interaction topologies," *IEEE Transactions on automatic control*, vol. 50, no. 5, pp. 655–661, 2005.

[16] A. Jadbabaie, J. Lin, *et al.*, "Coordination of groups of mobile autonomous agents using nearest neighbor rules," *Automatic Control, IEEE Transactions on*, vol. 48, no. 6, pp. 988–1001, 2003.

[17] O. Hlinka, O. Slučiak, F. Hlawatsch, P. M. Djurić, and M. Rupp, "Likelihood consensus and its application to distributed particle filtering," *Signal Processing, IEEE Transactions on*, vol. 60, no. 8, pp. 4334–4349, 2012.

[18] J. Beaudeau, M. F. Bugallo, and P. M. Djuric, "Target tracking with asynchronous measurements by a network of distributed mobile agents," in *Acoustics, Speech and Signal Processing (ICASSP), 2012 IEEE International Conference on*, pp. 3857–3860.

[19] M. Coates, "Distributed particle filters for sensor networks," in *Proceedings of the 3rd international symposium on Information processing in sensor networks*. ACM, 2004, pp. 99–107.

[20] P. M. Djurić, J. Beaudeau, and M. F. Bugallo, "Non-centralized target tracking with mobile agents," in *Acoustics, Speech and Signal Processing (ICASSP), 2011 IEEE International Conference on*, pp. 5928–5931.

[21] K. Y. Leung, T. D. Barfoot, and H. H. Liu, "Decentralized localization of sparsely-communicating robot networks: A centralized-equivalent approach," *IEEE Transactions on Robotics*, vol. 26, no. 1, pp. 62–77, 2010.

[22] S. Thrun, W. Burgard, and D. Fox, *Probabilistic robotics*. MIT press, 2005.

[23] G. Kitagawa, "Monte carlo filter and smoother for non-gaussian nonlinear state space models," *Journal of computational and graphical statistics*, vol. 5, no. 1, pp. 1–25, 1996.

[24] A. N. Bishop, B. Fidan, B. D. Anderson, K. Doğançay, and P. N. Pathirana, "Optimality analysis of sensor-target localization geometries," *Automatica*, vol. 46, no. 3, pp. 479–492, 2010.

[25] F. Bourgault, T. Furukawa, and H. F. Durrant-Whyte, "Optimal search for a lost target in a bayesian world," in *Field and service robotics*. Springer, 2003, pp. 209–222.

[26] J. R. Lawton, R. W. Beard, and B. J. Young, "A decentralized approach to formation maneuvers," *Robotics and Automation, IEEE Transactions on*, vol. 19, no. 6, pp. 933–941, 2003.

[27] H. Liu, X. Chu, Y.-W. Leung, and R. Du, "Simple movement control algorithm for bi-connectivity in robotic sensor networks," *Selected Areas in Communications, IEEE Journal on*, vol. 28, no. 7, pp. 994–1005, 2010.

[28] G. Thatte and U. Mitra, "Sensor selection and power allocation for distributed estimation in sensor networks: Beyond the star topology," *Signal Processing, IEEE Transactions on*, vol. 56, no. 7, pp. 2649–2661, 2008.

[29] N. Deo, *Graph theory with applications to engineering and computer science*. Courier Dover Publications, 2016.

[30] T. Amemiya, *Advanced econometrics*. Harvard university press, 1985.

[31] D. J. MacKay, *Information theory, inference and learning algorithms*. Cambridge university press, 2003.

[32] R. Olfati-Saber, E. Franco, E. Frazzoli, and J. S. Shamma, "Belief consensus and distributed hypothesis testing in sensor networks," in *Networked Embedded Sensing and Control*. Springer, 2006, pp. 169–182.

[33] V. V. Veeravalli and P. K. Varshney, "Distributed inference in wireless sensor networks," *Philosophical Transactions of the Royal Society of London A: Mathematical, Physical and Engineering Sciences*, vol. 370, no. 1958, pp. 100–117, 2012.

[34] C. Liu, S. E. Li, and J. K. Hedrick, "Model predictive control-based probabilistic search method for autonomous ground robot in a dynamic environment," in *ASME 2015 Dynamic Systems and Control Conference*.