Distributed Bayesian Filters for Multi-Vehicle Network by Using Latest-In-and-Full-Out Exchange Protocol of Observations

Chang Liu[1], Shengbo Eben Li[2] and J. Karl Hedrick[3]

***Abstract*—This paper presents a measurement dissemination-based distributed Bayesian filtering (DBF) approach for a network of unmanned ground vehicles (UGVs). The DBF utilizes the Latest-In-and-Full-Out (LIFO) local exchange protocol of sensor measurements for data communication within the network. Different from existing statistics dissemination-based approaches that transmit posterior distributions or likelihood functions, each UGV under LIFO only exchanges with neighboring UGVs a full communication buffer consisting of latest available measurements, which significantly reduces the transmission burden between each pair of UGVs to scale linearly with the size of the network. Under the condition of fixed and undirected topology, LIFO can guarantee non-intermittent dissemination of all observations over the network within finite time. Two types of LIFO-based DBF algorithms are then derived to estimate individual probability density function (PDF) for a static target and for a moving target, respectively. For the static target, each UGV locally fuses the newly received observations while for the moving target, a set of historical observations 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 demonstrated by comparing with consensus-based distributed filters and a centralized filter in simulations of target localization.**

(**TODO:** consider using ”measurement” instead of ”observations” as the main term.)

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], pedestrian tracking [2] and micro-environmental monitoring [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 additional 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] and is 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 multi-UGV network, which can be classified into two types: fusion center (FC)-based and neighborhood (NB)-based. In FC-based approaches, each UGV uses a filter to estimate local statistics of environment status based on its own measurement. The local statistics is then 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]. In NB-based approaches, a set of UGVs execute distributed filters to estimate individual posterior distribution. Consensus of individual estimates is achieved by solely communicating statistics and/or observations within local neighbors of these UGVs. The NB-based methods have become popular in recent years since such approaches do not require complex routing protocols or global knowledge of the network and therefore are robust to changes in network topology and to link failures.

So far, most studies on NB-based distributed filtering have mainly focused on the so-called *statistics dissemination* strategy that each UGV actually exchanges statistics, including posterior distributions and likelihood functions, with neighboring UGVs [12]. This strategy can be further categorized into two types: leader-based and consensus-based. In the former, statistics is sequentially passed and updated along a path formed by active UGVs, called leaders. Only leaders perform filtering based on its own measurement and 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 [13]. Sensors are grouped into multiple uncorrelated cliques, in each of which a leader is assigned to perform particle filtering and the particle information is then exchanged among leaders. In consensus-based distributed filters, every UGV diffuses statistics among neighbors, via which global agreement of the statistics is achieved by using consensus protocols [10], [14], [15]. For

*The first two authors, C. Liu and S. Li, have equally contributed to this research.

[1]Chang Liu is with the Vehicle Dynamics & Control Lab, Department of Mechanical Engineering, University of California, Berkeley, Berkeley, CA 94709, USA. Email: changliu@berkeley.edu

[2]Shengbo Eben Li is with the Department of Automotive Engineering, Tsinghua University, Beijing, 100084, China. He has worked at Department of Mechanical Engineering, University of California, Berkeley as a visiting scholar. Email: lisb04@gmail.com

[3]J. Karl Hedrick is with the Vehicle Dynamics & Control Lab, Department of Mechanical Engineering, University of California, Berkeley, Berkeley, CA 94709, USA. Email: khedrick@me.berkeley.edu

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 [16]. Saptarshi et al. (2014) presented a Bayesian consensus filter that uses logarithmic opinion pool for fusing posterior distributions of the tracked target [7]. Other examples can be found in [8], [17].

Despite the popularity of statistics dissemination strategy, exchanging statistics can consume high communication resources. One promising remedy is to disseminate measurement instead of statistics among neighbors, which, however, has not been fully exploited. One pioneering work was done by Coates et al. (2004), who used adaptive encoding of observations to minimize communication overhead [18]. Ribeiro et al. (2006) exchanged quantized observations along with error-variance limits considering more pragmatic signal models [19]. A recent work was conducted by Djuric et al. (2011), who proposed to broadcast raw measurements to other agents, and therefore each UGV has a complete set of observations 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 that every pair of distinct UGVs is directly connected by a unique edge, which is not always feasible in reality.

This paper extends existing works by introducing a Latest-In-and-Full-Out (LIFO) protocol into distributed Bayesian filters (DBF) for networked UGVs. Each UGV is only allowed to broadcast observations to its neighbors by using single-hopping and then implements individual Bayesian filter locally after receiving transmitted observations. The main benefit of using LIFO is on the reduction of communication burden, with the transmission data volume scaling linearly with the UGV number, while a statistics dissemination-based strategy can suffer from the order of environmental size. The proposed LIFO-based DBF has following properties: (1) For a fixed and undirected network, LIFO guarantees the global dissemination of observations over the network in a non-intermittent manner. (2) The corresponding DBF ensures the consistency of estimated target position, i.e., the estimated position converges in probability to the true target position when the number of observations 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 results are presented in Section V and Section VI concludes the paper.

II. PROBLEM FORMULATION

Consider a network of N UGVs in a bounded two-dimensional space S. The aim of UGVs is to efficiently localize a target in S. Each UGV is equipped with a noisy sensor for environmental perception. Due to the limit of communication range, each UGV can only share information locally by exchanging measurements with its neighbors. The Bayesian filter is run locally on each UGV based on its own and received measurements via single-hopping to estimate true position of the target.

A. Target and Sensor Model

The target motion takes a deterministic discrete-time model that can be described by

(**TODO:** this may make people wonder if LIFO-DBF can be applied to system with noise?)

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

where $x^g_k \in \mathbb{R}^2$ denotes the target position at time k; u^g_k represents the control input of the target.

Each UGV constantly measures the target position and the sensor measurement of i^{th} UGV can be described by a stochastic model:

$$z^i_k = g_i(x^g_k, w^i_k; y^i_k), \tag{2}$$

where w^i_k is the white measurement noise and $y^i_k = [x^i_k; \theta^i_k]$ represents the sensor state, consisting of the sensor position x^i_k and direction θ^i_k. g_i depends on the type of the sensor. Let $\mathcal{F}(y^i_k)$ denote the sensor field of view (FOV), g_i for several typical sensor types can be defined as follows:

Range-only sensors: when the target is within the sensor's FOV, the measurement only depends on the relative distance between the sensor and the target.

$$g_i(x^g_k, w^i_k; x^i_k) = \begin{cases} \|x^g_k - x^i_k\|_2 + w^i_k & \text{if } x^g_k \in \mathcal{F}(y^i_k), \\ \emptyset & \text{if } x^g_k \notin \mathcal{F}(y^i_k). \end{cases} \tag{3}$$

Bearing-only sensors: when the target is within the sensor's FOV, the measurement only depends on the relative bearing between the sensor and the target.

$$g_i(x^g_k, w^i_k; x^i_k) = \begin{cases} \measuredangle(x^g_k - x^i_k) + w^i_k & \text{if } x^g_k \in \mathcal{F}(y^i_k), \\ \emptyset & \text{if } x^g_k \notin \mathcal{F}(y^i_k). \end{cases} \tag{4}$$

Range-bearing sensors: when the target is within the sensor's FOV, the measurement is the relative distance and bearing between the sensor and the target.

$$g_i(x^g_k, w^i_k; x^i_k) = \begin{cases} x^g_k - x^i_k + w^i_k & \text{if } x^g_k \in \mathcal{F}(y^i_k), \\ \emptyset & \text{if } x^g_k \notin \mathcal{F}(y^i_k). \end{cases} \tag{5}$$

A probabilistic sensor model that describes the conditional probability of a certain measurement given sensor and target state is a key component for Bayesian filtering. We use a likelihood function to represent the probability of the target being detected by a sensor:

$$p^i_{1,k} = P(z^i_k \neq \emptyset | x^g_k; x^i_k) \in [0,1]\,,\; x^g_k \in S, \tag{6}$$

where x^i_k is the i^{th} sensor's position. Correspondingly, the likelihood function for no target being detected is:

$$p^i_{0,k} = P(z^i_k = \emptyset | x^g_k; x^i_k) = 1 - p^i_{1,k}. \tag{7}$$

The combination of Eq. (6) and Eq. (7) forms the probabilistic model for a sensor. If w^i_k is a zero-mean Gaussian white noise, then the probabilistic sensor model can be described as

$$p^i_{1,k} \sim \begin{cases} \mathcal{N}(\bar{z}^i_k, \Sigma^i_k) & \text{if } x^g_k \in \mathcal{F}(y^i_k) \\ 0 & \text{if } x^g_k \in \mathcal{F}(y^i_k), \end{cases} \tag{8}$$

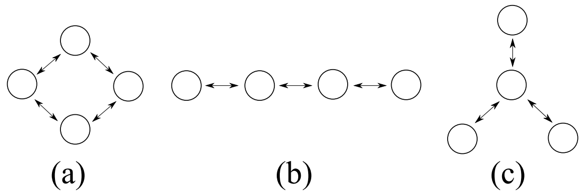


Fig. 1: Three types of topologies: (a) ring topology; (b) line topology; (c) star topology

where $\bar{z}_k^i$ is the nominal value of the measurement and equals $\|x_k^g - x_k^i\|$, $\measuredangle(x_k^g - x_k^i)$ and $x_k^g - x_k^i$ for range-only, bearing-only and range-bearing sensors, correspondingly.

Consequently,

$$p_{0,k}^i \sim \begin{cases} 0 & \text{if } x_k^g \in \mathcal{F}(y_k^i) \\ 1 & \text{if } x_k^g \in \mathcal{F}(y_k^i). \end{cases} \tag{9}$$

For the purpose of simplicity, we will not explicitly write x_k^i in the sensor model for the rest of the paper.

Remark 1: Given the knowledge of current target and UGV positions, current observation by each UGV can be considered conditionally independent from its own past observations and those by other UGVs [21].

Remark 2: The proposed LIFO protocol to be described in Section III and the consistency property proved in Section IV are applicable for general sensors, not limited to the ones described in this section. In addition, they also do not rely on the Gaussian noise assumption.

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 $V = \{1, \ldots, N\}$ represents the index set of UGVs and $E = V \times V$ denotes the edge set. The adjacency matrix $M = \left[M_{(ij)}\right]$ describes the communication topology of G:

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

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

The *direct neighborhood* of i^{th} UGV is defined as $\mathcal{N}_i = \left\{j | M_{(ij)} = 1, \forall j \in \{1, \ldots, N\}\right\}$. All the UGVs in $\mathcal{N}_i$ can directly exchange information with i^{th} UGV in one step. We also define another set $\mathcal{Q}_i$, called *available neighborhood*, that contains indices of UGVs whose observations can be received by the i^{th} UGV within single or multiple steps. Note that $\mathcal{N}_i \subseteq \mathcal{Q}_i$. Fig. 1 illustrates three types of typical topologies: ring [22], line [23], and star [24]. All of them are undirected and connected topologies.

C. Distributed Bayesian Filter for Multiple UGVs

The generic distributed Bayesian filter (DBF) is introduced in this section. Each UGV has its individual estimation of the probability density function (PDF) of target position, called *individual PDF*. The individual PDF of i^{th} UGV at time k is defined as $P_{pdf}^i(x_k^g|\mathbf{z}_{1:k}^i)$, where $\mathbf{z}_{1:k}^i$ denotes the set of observations by i^{th} UGV and by UGVs in $\mathcal{Q}_i$ that are transmitted to i^{th} UGV by time k. The individual PDF is initialized as $P_{pdf}^i(x_0^g|\mathbf{z}_0^i) = P(x_0^g)$, given all available prior information including past experience and environmental knowledge. Under the framework of DBF, the individual PDF is recursively estimated by two steps, i.e., prediction step and updating step, based on observations of i^{th} UGV and UGVs in $\mathcal{Q}_i$.

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

$$P_{pdf}^i(x_k^g|\mathbf{z}_{1:k-1}^i) = \int P(x_k^g|x_{k-1}^g) P_{pdf}^i(x_{k-1}^g|\mathbf{z}_{1:k-1}^i) dx_{k-1}^g \tag{10}$$

where $P(x_k^g|x_{k-1}^g)$ is a Markov motion model of the target defined by Eq. (1), which maps the state transition probability of the target from a prior state x_{k-1}^g to posterior state x_k^g. Note that the target is static in many search applications, such as the indoor search for stationary objects [25]. For the deterministic target motion model, as defined in Eq. (1), its Markov motion model is simplified to be

$$P(x_k^g|x_{k-1}^g) = \begin{cases} 1 & \text{if } x_k^g = x_{k-1}^g \\ 0 & \text{if } x_k^g \neq x_{k-1}^g \end{cases}. \tag{11}$$

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

$$P_{pdf}^i(x_k^g|\mathbf{z}_{1:k}^i) = K_i P_{pdf}^i(x_k^g|\mathbf{z}_{1:k-1}^i) P(\mathbf{z}_k^i|x_k^g) \tag{12}$$

where K_i is a normalization factor, given by:

$$K_i = 1 / \int P_{pdf}^i(x_k^g|\mathbf{z}_{1:k-1}^i) P(\mathbf{z}_k^i|x_k^g) dx_k^g$$

and $P_{pdf}^i(x_k^g|\mathbf{z}_{1:k}^i)$ is called posterior individual PDF; $P(z_k^i|x_k^g)$ is the likelihood function of observation z_k^i, described in **??** and **??**.

III. Distributed Bayesian Filter via Latest-In-and-Full-Out Protocol

This study proposes a Latest-In-and-Full-Out (LIFO) protocol for observation exchange and derives two corresponding distributed Bayesian filtering (DBF) algorithms, 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 observations 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 and consensus of distributed estimation while requiring much less communication burden than statistics dissemination-based methods.

Algorithm 1 LIFO Protocol

(1) Initialization: The CB of i^{th} 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 i^{th} 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 i^{th} UGV updates $z^j_{k^i_j}$ $(j=i)$ by its own observation at current step:

$$z^j_{k^i_j}=z^i_k,\ k^i_j=k,\ \text{if } j=i.$$

(2.3) Comparison Step:

The i^{th} 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 $j\neq i$,

$$l_{\text{latest}}=\underset{l\in\mathcal{N}_i,\,i}{\operatorname{argmax}}\left\{(k-1)^i_j,(k-1)^l_j\right\}$$
$$z^j_{k^i_j}=z^j_{(k-1)^{l_{\text{latest}}}_j},\ k^i_j=(k-1)^{l_{\text{latest}}}_j$$

(2.4) Sending Step:

The i^{th} UGV broadcasts its updated CB to all of its neighbors defined in $\mathcal{N}_i$.

(3) $k\leftarrow k+1$ until stop

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

Under LIFO, each UGV contains a communication buffer (CB) to store its latest knowledge of observations 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 observation 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. 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 for the purpose of simplicity. Note that under LIFO, $\mathcal{Q}_i=\{1,\ldots,N\}\setminus\{i\}$, which will be proved in Corollary 1. At time k, $z^j_{k^i_j}$ is received and stored in i^{th} UGV CB, in which k^i_j is the latest observation time of j^{th} UGV available to i^{th} UGV. Due to the communication delay, $k^i_j<k,\forall j\neq i$ and $k^i_i=k$ always holds. The **LIFO protocol** is stated in Algorithm 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 observations of i^{th} 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.

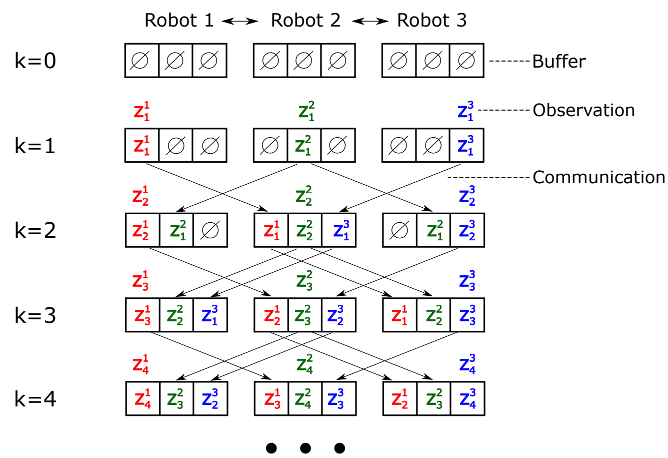


Fig. 2: Example of LIFO with three UGVs using line communication topology

Proof 1: Since the network is connected, there exists a minimum integer $\tau_{i,j}$ such that $M^{\tau_{i,j}}_{(i,j)}>0$ and $\tau_{i,j}$ is the length of a shortest path between i^{th} and j^{th} UGV. Under the LIFO, the latest observation of i^{th} UGV will always be received and then propagated in the communication buffer of the UGVs on a shortest path. Therefore, the latest observation that j^{th} UGV receives from i^{th} UGV is delayed by $\tau_{i,j}$ iterations of communication.

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

Corollary 2: For the same topology assumption in Proposition 1, once all elements in $\mathbf{z}^{CB,i}_k$ are filled, the updating of each element is non-intermittent.

Compared to statistics dissemination, LIFO is generally more communication-efficient for distributed filtering. To be specific, consider an $M\times M$ grid environment with 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 UGV positions where observations were made, the size of which is $O(N)$, scaling linearly with UGV number. On the contrary, the length of 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 environmental size. Since M is generally much larger than N in applications such as target localization, 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 UGV know the sensor models of other UGVs. Each UGV stores last-step individual PDF, i.e., $P^i_{pdf}(x^g|\mathbf{z}^i_{1:k-1})$. According to Corollary 2, $\mathbf{z}^i_k=\mathbf{z}^{CB,i}_k$ and $\mathbf{z}^i_{1:k}=\mathbf{z}^{CB,i}_{1:k}=\left[z^1_{1:k^i_1},\ldots,z^N_{1:k^i_N}\right]$. The assumption of static target can simplify the Bayesian filter as the prediction

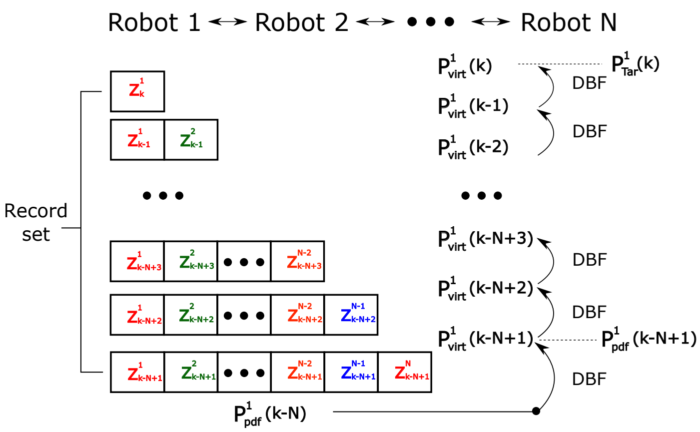


Fig. 3: Example of LIFO-DBF for 1st UGV at time k. Networked UGVs take a line topology.The stored individual PDF is $P^1_{pdf}(k-N)$. The UGV first calculates $P^1_{virt}(k-N+1)$ using DBF and stores it as $P^1_{pdf}(k-N+1)$. Repeating DBF until obtaining $P^1_{pdf}(k)$, which is then used as the target PDF estimation of 1st UGV at time k. In this example, $\Omega^1_\xi = \{1, 2, \dots, N+1-\xi\}$, $\xi = 1, \dots, N$.

step becomes unnecessary. Therefore, the i^{th} individual PDF is only updated by

$$\begin{aligned} P^i_{pdf}(x^g|\mathbf{z}^i_{1:k}) &= K_i P^i_{pdf}(x^g_k|\mathbf{z}^i_{1:k-1}) P(\mathbf{z}^i_k|x^g_k) \\ &= K_i P^i_{pdf}(x^g|\mathbf{z}^i_{1:k-1}) \prod_{j=1}^{N} P(z^j_{k^i_j}|x^g) \end{aligned} \tag{13}$$

where

$$K_i = 1/\int P^i_{pdf}(x^g|\mathbf{z}^i_{1:k-1}) \prod_{j=1}^{N} P(z^j_{k^i_j}|x^g) dx^g$$

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 historical observations, called the *record set*, from time $(k-N+1)$ to k. The i^{th} individual PDF is then alternatively predicted and updated by using the aforementioned Bayesian filter (Eq. (10) and Eq. (12)) from $(k-N)$ to k. Fig. 3 illustrates the LIFO-DBF procedure for the 1st UGV as an example. With a line topology, the record set of 1st UGV is shown as a triangle.

Let Ω^i_ξ $(\xi = 1, \dots, N)$ denote the index set of UGVs whose observation at time $(k-N+\xi)$ is stored in i^{th} UGV's record set. The **LIFO-DBF algorithm** for moving target is then 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 individual PDFs, only current-step CB is stored in UGV memory and all historical CBs can be discarded, which means that the size of occupied memory is $O(N)$. On the contrary, for the moving target, each UGV needs to store a triangular matrix of historical observation with size of $O(N^2)$ and an individual PDF with size $O(M^2)$, which means that the

Algorithm 2 LIFO-DBF Algorithm

For i^{th} UGV at k^{th} step:
After updating CB by Algorithm 1,
(1) The stored individual PDF for time $(k-N)$ is:

$$P^i_{pdf}(x^g_{k-N}|z^1_{1:k-N}, \dots, z^N_{1:k-N})$$

(2) Initialize a virtual PDF by assigning the individual PDF to it:

$$P^i_{virt}(x^g_{k-N}) = P^i_{pdf}(x^g_{k-N}|z^1_{1:k-N}, \dots, z^N_{1:k-N})$$

(3) From $\xi = 1$ to N, iteratively repeat two steps of Bayesian filtering:
(3.1) Prediction

$$\begin{aligned} &P^{pre}_{virt}(x^g_{k-N+\xi}) \\ &= \int P(x^g_{k-N+\xi}|x^g_{k-N+\xi-1}) P^i_{virt}(x^g_{k-N+\xi-1}) dx^g_{k-N+\xi-1} \end{aligned}$$

(3.2) Updating

$$P^i_{virt}(x^g_{k-N+\xi}) = K_\xi P^{pre}_{virt}(x^g_{k-N+\xi}) \prod_{j\in\Omega^i_\xi} P(z^j_{k-N+\xi}|x^g_{k-N+\xi})$$

$$K_\xi = 1/\int P^{pre}_{virt}(x^g_{k-N+\xi}) \prod_{j\in\Omega^i_\xi} P(z^j_{k-N+\xi}|x^g_{k-N+\xi}) dx^g_{k-N+\xi}$$

(3.3) When $\xi = 1$, store the virtual PDF as the individual PDF for time $(k-N+1)$

$$P^i_{pdf}(x^g_{k-N+1}|z^1_{1:k-N+1}, \dots, z^N_{1:k-N+1}) = P^i_{virt}(x^g_{k-N+1}).$$

(4) Individual PDF of i^{th} UGV at time k is $P^i_{pdf}(x^g_k|\mathbf{z}^i_{1:k}) = P^i_{virt}(x^g_k)$.

size of occupied memory in each UGV is $O(M^2+N^2)$. This is generally larger that statistics dissemination, the storage size of which is $O(M^2)$. Therefore, LIFO-DBF sacrifices storage size for reducing communication burden. This is desirable for real applications as local memory of vehicles are 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 [26]. 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 (true target position).

We first prove the consistency for static UGVs and static target. The consistency for moving UGVs and moving target are subsequently proved. For simplicity and clarity, we assume S is a finite set (e.g. a finely discretized field) and x^{g^*} is the true position of target.

A. Static UGVs and Static Target

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

Theorem 1: Assume both UGVs and the target are static, and the sensors are unbiased, the MAP estimator of target position converges in probability to the true position of the target using LIFO-DBF, i.e.,

$$\lim_{k\to\infty} P(x_{MAP} = x^{g^*}|\mathbf{z}^i_{1:k}) = 1,\ i = 1,\ldots,N,$$

where

$$x_{MAP} = \arg\max_{x^g} P^i_{pdf}(x^g|\mathbf{z}^i_{1:k}).$$

Proof 2: First look at the asymptotic behavior of $P^i_{pdf}(x^g|\mathbf{z}^i_{1:k})$. The batch form of DBF at k^{th} step is:

$$P^i_{pdf}(x^g|\mathbf{z}^i_{1:k}) = P^i_{pdf}(x^g|z^1_{1:k^i_1},\ldots,z^N_{1:k^i_N}) \tag{14a}$$

$$= \frac{P^i_{pdf}(x^g)\prod_{j=1}^{N}\prod_{l=1}^{k^i_j} P(z^j_l|x^g)}{\sum_{x^g\in S} P^i_{pdf}(x^g)\prod_{j=1}^{N}\prod_{l=1}^{k^i_j} P(z^j_l|x^g)}. \tag{14b}$$

The decomposition into multiplication in Eq. (14b) results from the conditional independence of observations given x^g.

Comparing $P^i_{pdf}(x^g|\mathbf{z}^i_{1:k})$ with $P^i_{pdf}(x^{g^*}|\mathbf{z}^i_{1:k})$ yields

$$\frac{P^i_{pdf}(x^g|\mathbf{z}^i_{1:k})}{P^i_{pdf}(x^{g^*}|\mathbf{z}^i_{1:k})} = \frac{P^i_{pdf}(x^g)\prod_{j=1}^{N}\prod_{l=1}^{k^i_j} P(z^j_l|x^g)}{P^i_{pdf}(x^{g^*})\prod_{j=1}^{N}\prod_{l=1}^{k^i_j} P(z^j_l|x^{g^*})}. \tag{15}$$

Take the logarithm of Eq. (15) and average it over k steps:

$$\frac{1}{k}\ln\frac{P^i_{pdf}(x^g|\mathbf{z}^i_{1:k})}{P^i_{pdf}(x^{g^*}|\mathbf{z}^i_{1:k})} = \frac{1}{k}\ln\frac{P^i_{pdf}(x^g)}{P^i_{pdf}(x^{g^*})} + \sum_{j=1}^{N}\frac{1}{k}\sum_{l=1}^{k^i_j}\ln\frac{P(z^j_l|x^g)}{P(z^j_l|x^{g^*})}. \tag{16}$$

Since $P^i_{pdf}(x^g)$ and $P^i_{pdf}(x^{g^*})$ are bounded, then

$$\lim_{k\to\infty}\frac{1}{k}\ln\frac{P^i_{pdf}(x^g)}{P^i_{pdf}(x^{g^*})} = 0. \tag{17}$$

Utilizing the facts: (1) z^j_l are conditionally independent samples from $P(z^j_l|x^g)$ and (2) $k-N < k^i_j \le k$, the law of large numbers yields

$$\frac{1}{k}\sum_{l=1}^{k^i_j}\ln\frac{P(z^j_l|x^g)}{P(z^j_l|x^{g^*})} \xrightarrow{P} \mathbb{E}_{z^j_l}\left[\ln\frac{P(z^j_l|x^g)}{P(z^j_l|x^{g^*})}\right] \tag{18a}$$

$$= \int_{z^j_l} P(z^j_l|x^{g^*})\ln\frac{P(z^j_l|x^g)}{P(z^j_l|x^{g^*})}dz^j_l \tag{18b}$$

$$= -D_{KL}\left(P(z^j_l|x^{g^*})\|P(z^j_l|x^g)\right), \tag{18c}$$

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

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

This leads to the following conclusion:

$$\lim_{k\to\infty}\frac{1}{k}\sum_{l=1}^{k^i_j}\ln\frac{P(z^j_l|x^g)}{P(z^j_l|x^{g^*})}\begin{cases}<0, & x^g \neq x^{g^*},\\ =0, & x^g = x^{g^*}.\end{cases} \tag{20}$$

Then by considering the limiting case of Eq. (16), we can get:

$$\lim_{k\to\infty}\frac{1}{k}\ln\frac{P^i_{pdf}(x^g|\mathbf{z}^i_{1:k})}{P^i_{pdf}(x^{g^*}|\mathbf{z}^i_{1:k})}\begin{cases}<0, & x^g \neq x^{g^*},\\ =0, & x^g = x^{g^*}.\end{cases} \tag{21}$$

Eq. (21) implies that

$$\frac{P^i_{pdf}(x^g|\mathbf{z}^i_{1:k})}{P^i_{pdf}(x^{g^*}|\mathbf{z}^i_{1:k})} \xrightarrow{P} \begin{cases}0, & x^g \neq x^{g^*},\\ 1, & x^g = x^{g^*}.\end{cases} \tag{22}$$

Therefore,

$$\lim_{k\to\infty} P^i_{pdf}(x^g = x^{g^*}|\mathbf{z}^i_{1:k}) = 1.$$

It is then straightforward to show

$$\lim_{k\to\infty} P(x_{MAP} = x^{g^*}|\mathbf{z}^i_{1:k}) = 1.$$

B. Static UGVs and Moving Target

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

Theorem 2: Assume UGVs are static, sensors are unbiased and the target is moving deterministically, the MAP estimator of the target position converges in probability to the true position of the target using LIFO-DBF, i.e.,

$$\lim_{k\to\infty} P(x_{MAP} = x^{g^*}_k|\mathbf{z}^i_{1:k}) = 1,\ i = 1,\ldots,N.$$

Proof 3: Since the target motion model is deterministic, as defined in Eq. (11), the integration in the prediction step (Eq. (10)) can be removed, as no uncertainty exists in $P(x^g_k|x^g_{k-1})$.

Eq. (14) now becomes

$$P^i_{pdf}(x^g_k|\mathbf{z}^i_{1:k}) = P^i_{pdf}(x^g_k|z^1_{1:k^i_1},\ldots,z^N_{1:k^i_N})$$

$$= \frac{P^i_{pdf}(x^g_0)\prod_{j=1}^{N}\prod_{l=1}^{k^i_j} P(z^j_l|x^g_l)P(x^g_l|x^g_{l-1})}{\sum_{x^g_k\in S} P^i_{pdf}(x^g_0)\prod_{j=1}^{N}\prod_{l=1}^{k^i_j} P(z^j_l|x^g_l)P(x^g_l|x^g_{l-1})}.$$

The updating terms $P(x^g_l|x^g_{l-1})$ cancel when comparing $P^i_{pdf}(x^g|\mathbf{z}^i_{1:k})$ with $P^i_{pdf}(x^{g^*}|\mathbf{z}^i_{1:k})$, yielding the same result as Eq. (15). Therefore, the rest of the proof is similar to that of Theorem 1.

C. Moving UGVs

This subsection considers the case of using moving UGVs to localize a target, either static or moving. The difficulty of consistency proof for this case lies in the fact that each UGV makes observations at multiple positions. Here, the main idea of proof is to classify UGV observation positions into two disjoint sets: *infinite-observation spots* that contain positions where a UGV makes infinitely many observations as time tends to infinity, and *finite-observation spots* that contain positions where the UGV makes finitely many observations (i.e. the positions where the UGV never visits 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 observations are made as k tends to infinity.

Proof 4: Let $n_j^{i,k}$ denote the times that i^{th} UGV visits j^{th} position up to time k. Then, $\sum_j n_j^{i,k} = k$. It is straightforward to see that $\exists n_j^{i,k}$, such that $n_j^{i,k} \to \infty$, as $k \to \infty$.

Theorem 3: Assume UGVs move within a collection of finite positions and sensors are unbiased, the MAP estimator of target position converges in probability to the true position of the target using LIFO-DBF, i.e.,

$$\lim_{k\to\infty} P(x_{MAP} = x^{g^*} | \mathbf{z}^i_{1:k}) = 1,\ i = 1, \dots, N.$$

Proof 5: Without loss of generality, we assume the target is static. The proof when target is moving deterministically is similar. According to Theorem 1, the batch form of DBF at k^{th} step is

$$\frac{P^i_{pdf}(x^g|\mathbf{z}^i_{1:k})}{P^i_{pdf}(x^{g^*}|\mathbf{z}^i_{1:k})} = \frac{P^i_{pdf}(x^g)\prod_{j=1}^{N}\prod_{l=1}^{k^i_j} P(z^j_l|x^g; x^R_l)}{P^i_{pdf}(x^{g^*})\prod_{j=1}^{N}\prod_{l=1}^{k^i_j} P(z^j_l|x^{g^*}; x^R_l)}. \tag{24}$$

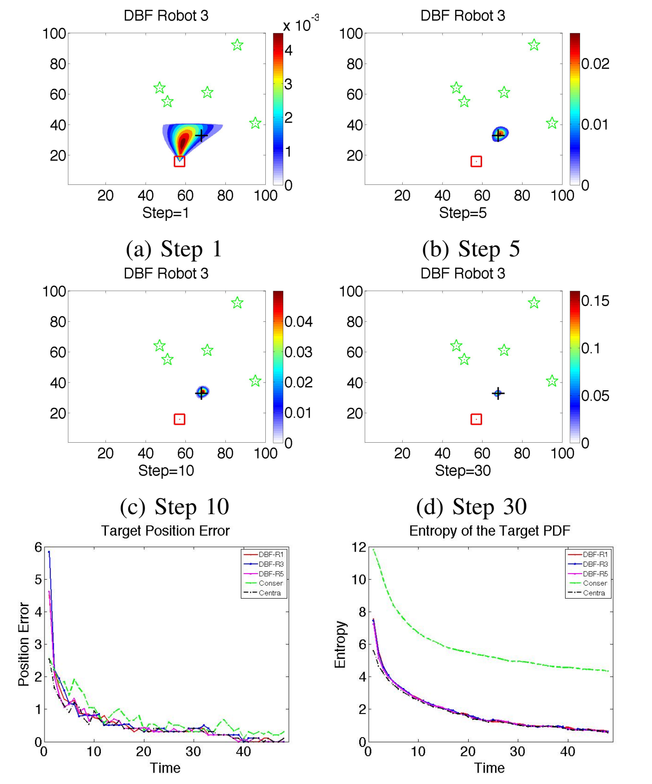
The only difference from Eq. (15) is that $P(z^j_l|x^g; x^R_l)$ in Eq. (24) varies as the UGV moves. For each UGV, there exists at least one position where infinite observations are made as $k \to \infty$, according to Lemma 1. All positions can be classified into finite-observation spots and infinite-observation spots. For the former, by referring to Eq. (18), it is easy to know that their contribution to Eq. (24) diminishes when $k \to \infty$. Therefore, Eq. (24) can be reduced to only considering infinite-observation spots and the rest of proof is similar to that of Theorem 1.

Remark 3: The assumption of unbiased sensors are important for the consistency. In fact, with unknown non-zero bias, the distribution of z^j_l differs from $P(z^j_l|x^{g^*})$, which invalidates the derivation in Eq. (18) and the consistency proof. This assumption also makes intuitive sense. In the extreme case, if each sensor has a very large 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 obtained when they fuse measurement from other sensors.

V. Simulation and Experiment

(**TODO:** This section needs a major revision. also notice to simplify the description of each plot.)

Simulation has been conducted to demonstrate 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 nearest neighbors. DBF is implemented using grid-based method, by which the space is finely discretized []. Other approaches, such as particle filters [], can also be used. Two types of scenarios are used: in first scenario, both UGVs and the target are static; the second scenario subsequently deals with six moving UGVs for localizing a moving target.



(a) Step 1 (b) Step 5 (c) Step 10 (d) Step 30 (e) Position Error (f) Entropy of Individual PDF

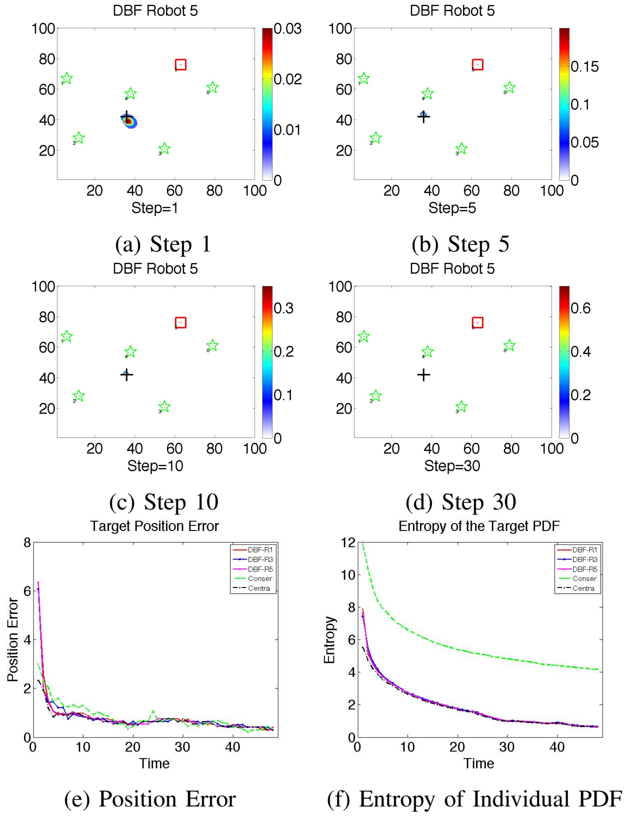
Fig. 4: (a)-(d) The 1st UGV's individual PDF at different times. The square denotes the current UGV and stars represent other UGVs. The cross stands for the target. (e) Average position estimation errors of 1st, 3rd and 5th UGV's LIFO-DBF, CbDF and CF. (f) Average entropy of individual PDFs.

In both scenarios, LIFO-DBF is compared with two commonly adopted approaches in multi-agent filtering: the consensus-based distributed filtering (CbDF) method and the centralized filtering (CF) method. The CbDF requires UGVs to continually exchange their individual PDFs with direct neighbors, computing the average of all received and its own target PDFs. Multiple rounds of communication and averaging are conducted during the "Sending Step" and "Receiving Step" in Algorithm 1 at each round to ensure the convergence of each UGV's individual PDFs. The CF assumes a central unit that can constantly receive and fuse all UGVs' latest observations 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 and true target position and the average entropy of individual PDFs of all 10 trials are compared among these three approaches.

A. Static UGVs, Static Target

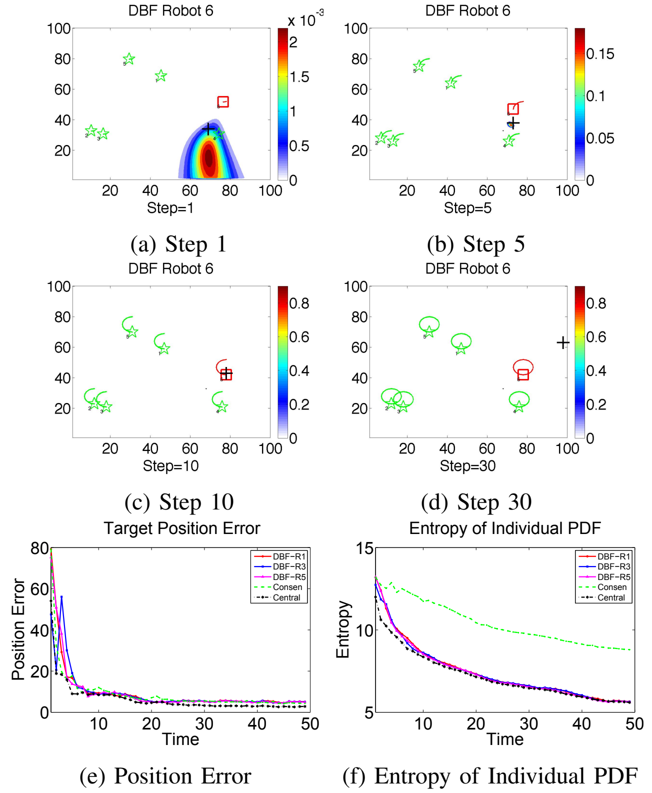
We assume all UGVs are equipped with bearing-only sensors.

1) Homogeneous Sensors: Six static UGVs, as represented by green stars and red square in Fig. 4a, are placed in the field.



(a) Step 1 (b) Step 5 (c) Step 10 (d) Step 30 (e) Position Error (f) Entropy of Individual PDF

Fig. 5: (a)-(d) The 1st UGV's individual PDF at different times. The square denotes the current UGV and stars represent other UGVs. The cross stands for the target. (e) Average position estimation errors of 1st, 3rd and 5th UGV's LIFO-DBF, CbDF and CF. (f) Average entropy of individual PDFs.



(a) Step 1 (b) Step 5 (c) Step 10 (d) Step 30 (e) Position Error (f) Entropy of Individual PDF

Fig. 6: (a)-(d) The 6th UGV's individual PDFs at different times. (e) Average position estimation errors. (f) Average entropy of individual PDFs.

The target is also assumed to be static, with A being an identity matrix and B_k being a zero matrix in eq. (1). The individual PDF of the first UGV at different time steps are shown in Figs. 4a to 4d. It can be noticed that, as more measurements are fused, the individual PDF asymptotically concentrates to the true location of the target (Fig. 4d), which accords with the consistency of LIFO-DBF.

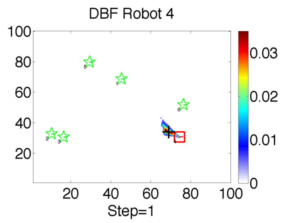
LIFO-DBF is compared with CbDF and CF, results of which are presented in Figs. 4e and 4f. Unsurprisingly, the CF achieves the best performance in terms of both small position estimation error and fast reduction of entropy. This happens because the central unit has access to the latest observations of all UGVs, thus making 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.

B. Moving UGVs, Moving Target

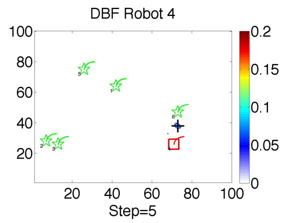
Six UGVs are equipped with heterogeneous sensors, with two having range-only sensors, two having bearing-only sensors and two having range-bearing sensors.

In this scenario, each UGV follows a pre-defined circular trajectory. The target motion is modeled as a single-integrator with process noise. Figs. 6a to 6d illustrates the test trial, in which the LIFO-DBF described in Section III-C is utilized for target localization. It can be noticed that the individual PDF concentrates to the true target location when the target constantly moves.

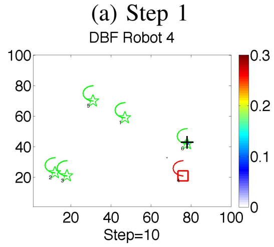
Figs. 6e and 6f compares LIFO-DBF with CbDF and CF. Similar to results in Section V-A1, CF achieves best performance with smallest position estimation error and fastest entropy reduction; LIFO-DBF shows similar asymptotic performance as the CF; and CbDF has the slowest entropy reduction among all three filtering approaches. However, CbDF achieves comparable position estimation error as the CF, which are marginally better than LIFO-DBF. This is not an unexpected result: due to the consensus procedure, CbDF is able to fuse the latest individual PDFs of each UGV via the averaging process. This is especially useful when the environment is dynamically changing. In contrast, LIFO-DBF for each UGV only uses delayed information from neighboring UGVs, thus gaining less knowledge about the target's current position. However, it is worth noting that, 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.



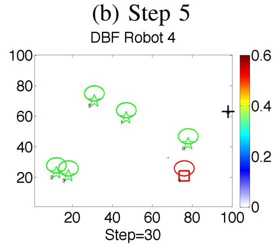
(a) Step 1



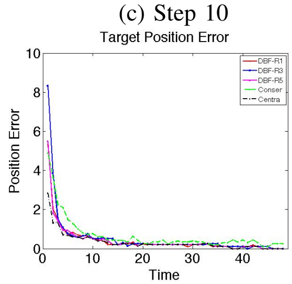
(b) Step 5



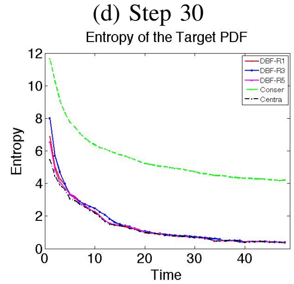
(c) Step 10



(d) Step 30



(e) Position Error



(f) Entropy of Individual PDF

Fig. 7: (a)-(d) The 1st UGV's individual PDF at different times. The square denotes the current UGV and stars represent other UGVs. The cross stands for the target. (e) Average position estimation errors of 1st, 3rd and 5th UGV's LIFO-DBF, CbDF and CF. (f) Average entropy of individual PDFs.

C. Experiment

VI. 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 exchanging full communication buffers among neighboring UGVs, LIFO significantly reduces the transmission burden between each pair of UGVs to scale linearly with the network size. It should be noted that LIFO is a general measurement exchange protocol and thus applicable to various sorts of sensors. Two types of LIFO-based DBF algorithms are proposed to estimate individual PDFs for a static target and a moving target, respectively. The consistency of LIFO-based 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.

Future work includes considering other types of sensors and imperfect communication between UGVs. Other types of sensors may have biased measurement , which complicates the design and analysis of LIFO-DBF. Imperfect communication, including package loss and transmission delay, requires extensions of current LIFO-DBF approach.

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] H. Wang, H. Lenz, A. Szabo, J. Bamberger, and U. D. Hanebeck, "Wlan-based pedestrian tracking using particle filters and low-cost mems sensors," in *Positioning, Navigation and Communication, 2007. WPNC'07. 4th Workshop on*. IEEE, 2007, pp. 1–7.

[3] X. Cao, J. Chen, Y. Zhang, and Y. Sun, "Development of an integrated wireless sensor network micro-environmental monitoring system," *ISA transactions*, vol. 47, no. 3, pp. 247–255, 2008.

[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] 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.

[13] 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.

[14] 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.

[15] 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.

[16] 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.

[17] 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.

[18] 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.

[19] A. Ribeiro and G. B. Giannakis, "Bandwidth-constrained distributed estimation for wireless sensor networks-part ii: unknown probability density function," *Signal Processing, IEEE Transactions on*, vol. 54, no. 7, pp. 2784–2796, 2006.

[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] 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.

[22] 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.

[23] 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.

[24] 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.

[25] M. Kulich, L. Preucil, and J. J. Miranda Bront, "Single robot search for a stationary object in an unknown environment," in *Robotics and Automation (ICRA), 2014 IEEE International Conference on*, pp. 5830–5835.

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