# Distributed Bayesian Filter using Measurement Dissemination for Multiple UGVs with Dynamically Changing Interaction Topologies

**Chang Liu**
Department of Mechanical Engineering
University of California, Berkeley
Berkeley, CA 94720
Email: changliu@berkeley.edu

**Shengbo Eben Li***
Department of Automotive Engineering
Tsinghua University
Beijing, China 100084
Email: lisb04@gmail.com

**J. Karl Hedrick**
Department of Mechanical Engineering
University of California, Berkeley
Berkeley, CA 94720
Email: khedrick@me.berkeley.edu

This paper presents a novel distributed Bayesian filtering (DFB) method using measurement dissemination for multiple unmanned ground vehicles (UGVs) with dynamically changing interaction topologies. Different from statistics dissemination-based algorithms that transmit posterior distributions or likelihood functions, this method relies on a full-in-full-out (FIFO) transmission protocol, which significantly reduces the transmission burden between each pair of UGVs. Each UGV only sends a communication buffer and a track list to its neighbors, in which the former contains a history of measurements from all UGVs, and the latter trims the used measurements in the buffer to reduce communication overhead. It is proved that each UGV can disseminate its measurements over the whole network within the finite time and is able to achieve the consistency of environmental state estimation. The effectiveness of this method is validated by comparing with consensus-based distributed filters and centralized filter in a multi-target tracking problem.

## 1 INTRODUCTION

Estimation using a group of networked UGVs has been utilized to collectively measure environment status [1], such as intruder detection [2], and signal source seeking [3], and pollution field estimation [4], due to its merits on low cost, high efficiency, and good reliability. The widely adopted estimation approaches include the Kalman Filter, extended Kalman filter, and particle filter [5], and the most generic scheme might be the Bayesian filter because of its applicability for nonlinear systems with arbitrary noise distributions [6,7]. In fact, a Bayesian filter can be reduced to other methods in certain conditions. For example, under the assumption of linearity and Gaussian noise, a Bayesian filter can be reduced to the Kalman filter [8], and for general nonlinear systems, a Bayesian filter can be numerically implemented as a particle filter due to the advantage in computation [8]. Because of this generality, this study focuses on its networked variant, which can track targets using local communication between neighboring UGVs.

The interaction topology plays a central role on the design of networked Bayesian filter, of which two types are widely investigated in literature: centralized filters and distributed filters. In the former, local statistics estimated by each agent is transmitted to a single fusion center, where a global posterior distribution is calculated at each filtering cycle [9, 10]. In the latter, each agent individually executes distributed estimation and the agreement of local estimates is achieved by certain consensus strategies [11–13]. In general, the distributed filters are more suitable in practice since they do not require a fusion center with powerful computation capability and are more robust to changes in network topology and link failures. So far, the distributed filters have two mainstream schemes in terms of the transmitted data among agents, i.e., *statistics dissemination* (SD) and *measurement dissemination* (MD). In the SD scheme, each agent exchanges statistics such as posterior distributions and likelihood functions within neighboring nodes [14]. In the MD scheme, instead of exchanging statistics, each agent sends measurements to neighboring nodes.

The statistics dissemination scheme has been widely investigated during the last decade, especially in the field of signal processing, network control, and robotics. Madhavan et al. (2004) presented a distributed extended Kalman filter

*Address all correspondence to this author.

for nonlinear systems [15]. This filter was used to generate local terrain maps by using pose estimates to combine elevation gradient and vision-based depth with environmental features. Olfati-Saber (2005) proposed a distributed linear Kalman filter (DKF) for estimating states of linear systems with Gaussian process and measurement noise [16]. Each DKF used additional low-pass and band-pass consensus filters to compute the average of weighted measurements and inverse-covariance matrices. Gu (2007) proposed a distributed particle filter for Markovian target tracking over an undirected sensor network [17]. 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. 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 [18]. Saptarshi et al. (2014) presented a Bayesian consensus filter that uses logarithmic opinion pool for fusing posterior distributions of the tracked target [6]. Other examples can be found in [7] and [19].

Despite the popularity of statistics dissemination, exchanging statistics can consume high communication resources if the environment to be detected is relatively large in space and complicated in structure. One remedy is to approximate statistics with parametric models, e.g., Gaussian Mixture Model [20], which can reduce communication burden to a certain extent. However, such manipulation increases the computation burden of each agent and sacrifices filtering accuracy due to approximation. The measurement dissemination scheme is an alternative solution to address the issue of exchanging large-scale statistics. An early work on measurement dissemination was done by Coates et al. (2004), who used adaptive encoding of observations to minimize communication overhead [21]. Ribeiro et al. (2006) exchanged quantized observations along with error-variance limits considering more pragmatic signal models [22]. A recent work was conducted by Djuric et al. (2011), who proposed to broadcast raw measurements to other agents, and therefore each agent has a complete set of observations of other agents for executing particle filtering [23]. A shortcoming of aforementioned works is that their communication topologies are assumed to be a fixed and complete graph that every pair of distinct agents is constantly connected by a unique edge. In many real applications, the interaction topology can be time-varying due to unreliable links, external disturbances and/or range limits [24]. In such cases, dynamically changing topologies can cause random packet loss, variable transmission delay, and out-of-sequence measurement (OOSM) issues [25], thus decreasing the performance of distributed estimation. Leung et al. (2010) has explored a decentralized Bayesian filter for dynamic robot networks [26] in order to achieve centralized-equivalent filtering performance. However, it requires the communication of both measurements and statistics, which can still incur large communication overhead.

The main contribution of the paper is to design a distributed Bayesian filtering (DBF) method only using measurement dissemination for a group of networked UGVs with dynamically changing interaction topologies. In our previous work [27], we have proposed a Latest-In-and-Full-Out (LIFO) protocol for measurement exchange and developed a corresponding DBF algorithm. However, it only applies to static targets with simple binary sensor model because each UGV only transmits the latest measurements in the network, leading to the out-of-sequence measurement problem caused by moving targets. In this work, we introduce the a new protocol, called the Full-In-and-Full-Out (FIFO), which additionally includes a track list of communication history. The FIFO allows each UGV to broadcast a history of measurements to its neighbors by using single-hopping, thus enabling the tracking of moving targets with more general sensor models under time-varying topologies. An individual Bayesian filter is implemented locally by each UGV after exchanging observations using FIFO. Under the condition that the topologies is frequently jointly strongly connected, FIFO can disseminate measurements over the network within a finite time. The main benefit of using FIFO in the distributed Bayesian filter is on the reduction of communication burden while avoiding the OOSM issue and ensuring that no information loss occurs.

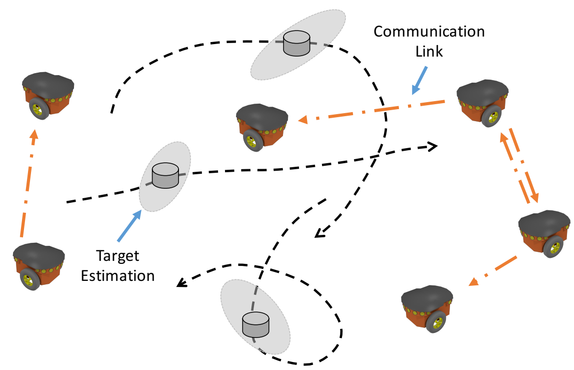



Fig. 1: Target tracking scenario. The interaction topology is dynamically changing and UGVs can only communicate with neighboring UGVs.

The rest of this paper is organized as follows: Section 2 formulates the target tracking problem using multiple UGVs; Section 3 proposes the FIFO protocol for measurement communication in dynamically changing interaction topologies; Section 4 introduces FIFO-based DBF algorithm and track list; simulation results are presented in Section 6 and Section 7 concludes the paper.

## 2 Problem Formulation

Consider a network of $N$ UGVs in a bounded two-dimensional space $S$, as shown in Fig. 1. The interaction topology can be dynamically changing due to limited communication range, team reconfiguration, or intermittently link failure. Each UGV is equipped with a sensor for target detection. Due to the limit of communication range, each

UGV can only exchange sensor measurements with its local neighbors. Every UGV locally runs a Bayesian filter to estimate the target position in $S$ utilizing its own measurements and the received measurements from other UGVs.

### 2.1 Target and Sensor Model

The target motion uses a stochastic discrete-time model:

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

where the superscript $g$ represents the target and $x^g_k \in S$ is the target position at time $k$; $v_k$ is the white process noise.

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 the 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 design of the Bayesian filter relies on the conditional probability of obtaining a certain measurement $z^i_k$ given the current target and sensor states, which is denoted by $P(z^i_k|x^g_k;x^i_k)$ [5]. The conditional probability $P(z^i_k|x^g_k;x^i_k)$ depends on both $h_i$ and/or $w^i_k$ in Eq. (2). For example, if $w^i_k$ is a zero-mean Gaussian white noise with covariance $\Gamma^i_k$, $P(z^i_k|x^g_k;x^i_k)$ can be described as $P(z^i_k|x^g_k;x^i_k) = N(h_i(x^g_k,x^i_k),\Gamma^i_k)$. For non-Gaussian noise, such as Poisson noise or Cauchy noise [28], $P(z^i_k|x^g_k;x^i_k)$ can also be similarly defined (for the purpose of simplicity, $P(z^i_k|x^g_k;x^i_k)$ is shorted as $P(z^i_k|x^g_k)$ for the rest of the paper). It should be noted that this work is not confined to any specific distribution of the noise. The measurement function $h_i$ for several typical sensors are listed as follows [29]:

**Range-only sensors:** $h_i$ is a function of the relative Euclidean distance between the sensor and the target:

$$h_i(x^g_k, x^i_k) = \|x^g_k - x^i_k\|_2,$$

where $\|\cdot\|_2$ is the Euclidean distance in $S$.

**Bearing-only sensors:** $h_i$ is a function of the relative bearing between the sensor and the target:

$$h_i(x^g_k, x^i_k) = \measuredangle(x^g_k - x^i_k),$$

where $\measuredangle$ denotes the angle from the sensor to the target.

**Range-bearing sensors:** $h_i$ includes both the relative distance and bearing:

$$h_i(x^g_k, x^i_k) = x^g_k - x^i_k.$$

**Remark 1.** *Given the knowledge of current target and UGV positions, the current measurement by each UGV can be considered conditionally independent from its own past measurements and those by other UGVs [30].*

### 2.2 Graphical Model of Interaction Topology

We consider a simple[1] graph $G = (V,E)$ to represent the interaction topology of N networked UGVs, where the vertex set $V = \{1,\dots,N\}$ represents the index set of UGVs and $E = V \times V$ denotes the edge set. For the purpose of narrative simplicity, we use directed graphs to describe our approach in this work. The undirected graphs can be treated as bidirectional directed graphs.

The *adjacency matrix* $A = [a_{ij}]$ of the graph $G$ describes the interaction topology:

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

where $a_{ij}$ is the entry on the $i^{\text{th}}$ row and $j^{\text{th}}$ column of the adjacency matrix. The notation $a_{ij} = 1$ indicates that the $i^{\text{th}}$ UGV can directly communicate to the $j^{\text{th}}$ UGV and $a_{ij} = 0$ indicates no direct communication from $i$ to $j$. A directed graph is *strongly connected* if there is a directed path connecting any two arbitrary vertices in $V$.

Define the *union* of a collection of simple directed graphs as the graph with the vertices in $V$ and the edge set given by the union of each member's edge sets. Such collection is *jointly strongly connected* if the union of its members forms a strongly connected graph. We use $G_k$ to represent the interaction topology of time $k$ and define the *inbound neighbors* and *outbound neighbors* of the $i^{\text{th}}$ UGV under $G_k$ as the set $\mathcal{N}^{\text{in}}_i(G_k) = \left\{j|a^k_{ji} = 1, j \in V\right\}$and $\mathcal{N}^{\text{out}}_i(G_k) = \left\{j|a^k_{ij} = 1, j \in V\right\}$. All UGVs in $\mathcal{N}_i(G_k)$ can directly receive information from the $i^{\text{th}}$ UGV via the single hopping.

## 3 Full-In-and-Full-Out (FIFO) Protocol

This study proposes a Full-In-and-Full-Out (FIFO) protocol for measurement exchange in dynamically changing interaction topologies for the purpose of applying distributed Bayesian filters to target tracking. Let $Y^i_{\mathcal{K}} = \left\{\left[x^i_k, z^i_k\right] | k \in \mathcal{K}\right\}$be the set of state-measurement pairs of the $i^{\text{th}}$ UGV, where $\mathcal{K}$ is an index set of time steps. Each UGV contains a communication buffer (CB) and a track list (TL). The CB stores state-measurement pairs of all UGVs:

$$\mathcal{B}^i_k = \left[Y^1_{\mathcal{K}^{i1}_k}, \dots, Y^N_{\mathcal{K}^{iN}_k}\right],$$

where $\mathcal{B}^i_k$ is the CB of $i^{\text{th}}$ UGV at time $k$ and $\mathcal{K}^{ij}_k (j \in V)$ is the time index set. $Y^j_{\mathcal{K}^{ij}_k}$ represents the set of $j^{\text{th}}$ UGV's state-measurement pairs of time steps in $\mathcal{K}^{ij}_k$ that are stored in $i^{\text{th}}$ UGV's CB at time $k$. The TL stores the information of each UGV's reception of other UGVs' measurements and is used for trimming old state-measurement pairs in the CB to reduce the communication burden. The details of TL will be introduced in Section 4.2. Each UGV sends its CB and TL to its outbound neighbors at every time step.

The **FIFO protocol** is stated in Algorithm 1. Fig. 2 illustrates the FIFO cycles of a network of 3 UGVs with switching topologies. The following facts can be observed from Fig. 2: (1) the topologies are jointly strongly connected in the time interval $[0,6)$; (2) each UGV can receive the state-measurement pairs of other UGVs within finite steps. Extending these facts to a network of $N$ UGVs, we have the following properties of FIFO:

[1] A (directed/undirected) graph $G = (V,E)$ is *simple* if it has no self-loops (i.e., $(i,j) \in E$ only if $i \neq j$) or multiple edges with the same source and target nodes (i.e., $E$ only contains distinct elements).

**Algorithm 1** FIFO Protocol

**(1)** Initialization.

***CB***: The CB of $i^{\text{th}}$ UGV is initialized as an empty set at $k=0$:

$$\mathcal{B}_0^i = \left[Y^1_{\mathcal{K}_0^{i1}}, \ldots, Y^N_{\mathcal{K}_0^{iN}}\right], \text{ where } Y^j_{\mathcal{K}_0^{ij}} = \{[\varnothing, \varnothing]\}.$$

***TL***: The TL of $i^{\text{th}}$ UGV is initialized at $k=0$:

$$\mathbf{q}_1^{ij} = [0, \ldots, 0, 1], \text{ i.e. } q_1^{jl} = 0, \forall j, l \in \{1 \ldots, N\}.$$

**(2)** At time $k\,(k \geq 1)$ for $i^{\text{th}}$ UGV:

(2.1) Receiving Step.

***CB***: The $i^{\text{th}}$ UGV receives all CBs of its inbound neighbors $\mathcal{N}_i^{\text{in}}(G_{k-1})$, corresponding to the $(k-1)^{\text{th}}$ step CBs. The received CB from the $l^{\text{th}}$ UGV is $\mathcal{B}_{k-1}^l\,(l \in \mathcal{N}_i^{\text{in}}(G_{k-1}))$.

***TL***: The $i^{\text{th}}$ UGV receives all TLs of its inbound neighbors $\mathcal{N}_i^{\text{in}}(G_{k-1})$. The received TL from the $l^{\text{th}}$ UGV is $Q_{k-1}^l\,(l \in \mathcal{N}_i^{\text{in}}(G_{k-1}))$.

(2.2) Observation Step.

***CB***: The $i^{\text{th}}$ UGV updates $Y^i_{\mathcal{K}_k^{i,i}}$ by its own state-measurement pair:

$$Y^i_{\mathcal{K}_k^{ii}} = Y^i_{\mathcal{K}_{k-1}^{ii}} \cup \left\{\left[x_k^i, z_k^i\right]\right\}.$$

(2.3) Updating Step.

***CB***: The $i^{\text{th}}$ UGV updates other entries of its own CB, $Y^j_{\mathcal{K}_k^{ij}}\,(j \neq i)$, by merging with all received CBs:

$$Y^j_{\mathcal{K}_k^{ij}} = Y^j_{\mathcal{K}_{k-1}^{ij}} \cup Y^j_{\mathcal{K}_{k-1}^{lj}}, \forall j \neq i, \forall l \in \mathcal{N}_i^{\text{in}}(G_{k-1}).$$

***TL***: The $i^{\text{th}}$ UGV updates its own TL, $Q_k^i$, using the received TLs (see Algorithm 3). Trim the CB based on the updated track lists (see Algorithm 4).

(2.4) Sending Step.

***CB***: The $i^{\text{th}}$ UGV sends its updated CB, $\mathcal{B}_k^i$, to all of its outbound neighbors defined in $\mathcal{N}_i^{\text{out}}(G_k)$.

***TL***: The $i^{\text{th}}$ UGV sends its updated track list, $Q_k^i$, to its outbound neighbors $\mathcal{N}_i^{\text{out}}(G_k)$.

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

**Theorem 1.** *Consider a network of $N$ UGVs with dynamically changing interaction topologies $\mathbb{G} = \{G_1, G_2, G_3 \ldots,\}$. If $\mathbb{G}$ is frequently jointly strongly connected, i.e.,*

1. *there exists an infinite sequence of time intervals $[k_m, k_{m+1})$, $m = 1, 2, \ldots,$ starting at $k_1 = 0$ and are contiguous, nonempty and uniformly bounded;*
2. *the union of graphs across each such interval is jointly strongly connected,*

*then each pair of UGVs can exchange measurements under FIFO. In addition, it takes no more than $NT_u$ steps for a UGV to communicate to another one, where $T_u = \sup_{m=1,2,\ldots}(k_{m+1}-k_m)T$ is the upper bound of interval lengths.*

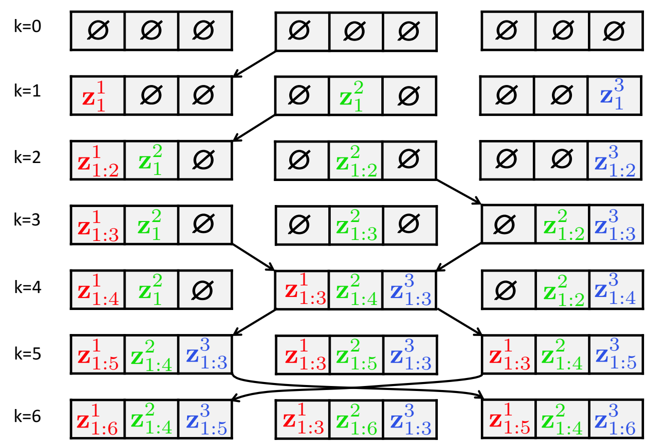



Fig. 2: Example of FIFO with three UGVs under dynamically changing interaction topologies. The arrows represent a directed communication link between two UGVs. $\varnothing$ denotes the empty set. For the purpose of clarity, we only show measurements, not the states, in the CB in this example.

*Proof.* Without loss of generality, we consider the transmission of $\mathcal{B}_{t_1}^i$ from the $i^{\text{th}}$ UGV to an arbitrary $j^{\text{th}}$ UGV $(j \in V \setminus \{i\})$, where $t_1 \in [k_1, k_2)$. Since each UGV will receive inbound neighbors' CBs and send the merged CB to its outbound neighbors at the next time step, the $i^{\text{th}}$ UGV can transmit $\mathcal{B}_{t_1}^i$ to $j^{\text{th}}$ UGV if and only if a path $[l_1, \ldots, l_n]$ exists, with $l_1 = i$, $l_n = j$, $l_2, \ldots, l_{n-1} \in V \setminus \{i, j\}$, and the edges $(l_s, l_{s+1})$ appears no later than $(l_{s+1}, l_{s+2})$, $s = 1, \ldots, n-2$.

As the union of graphs across the time interval $[k_2, k_3)$ is jointly connected, $i^{\text{th}}$ UGV can directly send $\mathcal{B}_{t_1}^i$ to at least one another UGV at a time instance, i.e., $\exists l_2 \in V \setminus \{i\}$, $\exists t_2 \in [k_2, k_3)$ s.t. $l_2 \in \mathcal{N}_i^{\text{out}}(G_{t_2})$. If $l_2 = j$, then $\mathcal{B}_{t_1}^i$ has been sent to $j$. If $l_2 \neq j$, $\mathcal{B}_{t_1}^i$ has been merged into $\mathcal{B}_{t_2+1}^{l_2}$ and will be sent out in the next time step.

Using the similar reasoning for time intervals $[k_m, k_{m+1})$, , $m = 3, 4, \ldots,$ it can be shown that all UGVs can receive the state-measurement pairs in $\mathcal{B}_{t_1}^i$ no later by $k_{N+1}$. Therefore, the transmission time from an arbitrary UGV to any other UGVs is no greater than $NT_u$. □

**Corollary 1.** *For a frequently jointly strongly connected network, each UGV receives the CBs of all other UGVs under FIFO within finite time.*

*Proof.* According to Theorem 1, each UGV is guaranteed to receive $\mathcal{B}_t^j$ $(\forall t \geq 0, j \in V)$ when $k \geq t + NT_u$. □

## 4 Distributed Bayesian Filter via FIFO Protocol

We first introduce the generic distributed Bayesian filter (DBF). Let $X_k \in S$ be the random variable representing the position of the target at time $k$. Define $\mathcal{Z}_k^i$ as the set of measurements of time $k$ that are in the $i^{\text{th}}$ UGV's CB, i.e.,

$\mathcal{Z}_k^i = \left\{ z_k^j | \left[ x_k^j, z_k^j \right] \in \mathcal{B}_k^i, \forall j \in V \right\}$ and let $\mathcal{Z}_{1:k}^i = \bigcup_{t=1}^{k} \mathcal{Z}_t^i$. We also define $z_{1:k}^i = \left[ z_1^i, \ldots, z_k^i \right]$ as the set of the $i^{\text{th}}$ UGV's measurements of times 1 through $k$. The probability density function (PDF) of $X_k$, called *individual PDF*, of the $i^{\text{th}}$ UGV is represented by $P_{pdf}^i(X_k|\mathcal{Z}_{1:k}^i)$. It is the estimation of the target position given all the measurements that the $i^{\text{th}}$ UGV has received. The initial individual PDF, $P_{pdf}^i(X_0)$, is constructed given prior information including past experience and environment knowledge. It is necessary to initialize $P_{pdf}^i(X_0)$ such that the probability density of the true target position is nonzero, i.e., $P_{pdf}^i(X_0 = x_0^g) > 0$.

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

**Prediction.** At time $k$, the prior individual PDF $P_{pdf}^i(X_{k-1}|\mathcal{Z}_{1:k-1}^i)$ is first predicted forward by using the Chapman-Kolmogorov equation:

$$P_{pdf}^i(X_k|\mathcal{Z}_{1:k-1}^i) = \int\limits_{X_{k-1} \in S} P(X_k|X_{k-1}) P_{pdf}^i(X_{k-1}|\mathcal{Z}_{1:k-1}^i) dX_{k-1}, \tag{3}$$

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

**Updating.** The $i^{\text{th}}$ individual PDF is then updated by the Bayes' rule using the set of newly received measurements at time $k$, i.e., $\mathcal{Z}_k^i$:

$$\begin{aligned} P_{pdf}^i(X_k|\mathcal{Z}_{1:k}^i) &= K_i P_{pdf}^i(X_k|\mathcal{Z}_{1:k-1}^i) P(\mathcal{Z}_k^i|X_k) \\ &= K_i P_{pdf}^i(X_k|\mathcal{Z}_{1:k-1}^i) \prod_{z_k^j \in \mathcal{Z}_k^i} P(z_k^j|X_k) \end{aligned} \tag{4}$$

where $P(z_k^j|X_k)$ is the sensor model and $K_i$ is a normalization factor, given by:

$$K_i = \left[ \int\limits_{X_k \in S} P_{pdf}^i(X_k|\mathcal{Z}_{1:k-1}^i) P(\mathcal{Z}_k^i|X_k) dX_k \right]^{-1}.$$

The factorization of $P(\mathcal{Z}_k^i|X_k)$ comes from the conditional independence of measurements from each UGV given the target position and the corresponding UGV's position.

### 4.1 The FIFO-DBF Algorithm

The generic DBF is not directly applicable to time-varying interaction topologies. This is because changing topologies can cause intermittent and out-of-sequence arrival of measurements from different UGVs, giving rise to the OOSM problem. One possible solution is to ignore all measurements that are out of the temporal order. This is undesirable since this will cause significant information loss. Another possible remedy is to fuse all measurements by running the filtering algorithm from the beginning at each time step. This solution naturally causes excessive computational burden. To avoid both OOSM problem and unnecessary computational complexity, we add a new PDF, namely the *stored PDF*, $P_{sto}^i(X_t)$, that is updated from the $i^{\text{th}}$ UGV's initial PDF by fusing the state-measurement pairs of *all* UGVs up to a certain time $t \leq k$. The choice of $t$ is described in Section 4.2.

---

**Algorithm 2** FIFO-DBF Algorithm

For $i^{\text{th}}$ UGV at $k^{\text{th}}$ step ($\forall i \in V$):

**(1)** Initialize a *temporary PDF* by assigning the stored individual PDF to it:

$$P_{tmp}^i(X_t) = P_{sto}^i(X_t),$$

where

$$P_{sto}^i(X_t) = P_{pdf}^i(X_t|z_{1:t}^1, \ldots, z_{1:t}^N).$$

**(2)** For $\xi = t+1$ to $k$, iteratively repeat two steps of Bayesian filtering:

(2.1) Prediction

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

(2.2) Updating

$$P_{tmp}^i(X_\xi) = K_\xi P_{tmp}^{pre}(X_\xi) P(\mathcal{Z}_\xi^i|X_\xi),$$

$$K_\xi = \left[ \int_S P_{tmp}^{pre}(X_\xi) P(\mathcal{Z}_\xi^i|X_\xi) dX_\xi \right]^{-1}.$$

(2.3) If $z_\xi^j \neq \emptyset$ for $\forall j \in V$, update the stored PDF:

$$P_{sto}^i(X_\xi) = P_{tmp}^i(X_\xi).$$

**(3)** The individual PDF of $i^{\text{th}}$ UGV at time $k$ is $P_{pdf}^i(X_k|\mathcal{Z}_{1:k}^i) = P_{tmp}^i(X_k)$.

---

The individual PDF, $P_{pdf}^i(X_k|\mathcal{Z}_{1:k}^i)$, is then computed by fusing the measurements from time $t+1$ to $k$ in the CB into $P_{sto}^i(X_t)$, running the Bayesian filter (Eq. (3) and (4)). Note that initially, $P_{sto}^i(X_0) = P_{pdf}^i(X_0)$.

The **FIFO-DBF algorithm** is stated in Algorithm 2. Each UGV runs FIFO-DBF after its CB is updated in the Updating Step in Algorithm 1. At the beginning, we assign the stored PDF to a temporary PDF, which will then be updated by sequentially fusing measurements in the CB to obtain the individual PDF. It should be noted that, when the UGV's CB contains all UGVs' state-measurement pairs from $t$ to $\xi$, the temporary PDF of $\xi$ is assigned as the stored PDF. Fig. 3 illustrates the FIFO-DBF procedure for the 1st UGV as an example. It can be noticed that, the purpose of using the stored PDF is to avoid running the Bayesian filtering from the initial PDF at every time step. Since the stored PDF has incorporated all UGVs' measurements up to some time step $t$, the information loss is prevented. We point out that the time $t$ of each UGV's stored PDF can be different from others. The stored PDF is saved locally by each UGV and not transmitted to others.

### 4.2 Track Lists for Trimming CBs

The size of CBs can keep increasing as measurements cumulate over time. The use of the stored PDF has made it feasible to trim excessive measurements from the CBs. To avoid information loss, a state-measurement pair can only be trimmed from a UGV's CB when *all* UGVs have received

---

[2] '$\vee$' is the notation of the logical 'OR' operator.

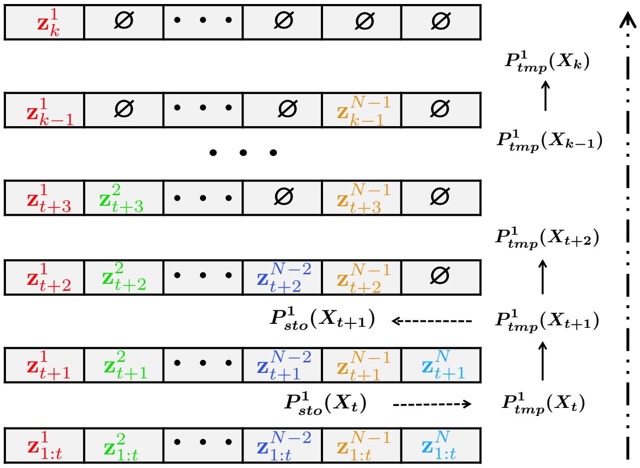


Fig. 3: Example of FIFO-DBF for the 1st UGV at time $k$. Only the measurement (not the state) is shown in the figure. The UGV first calculates $P^1_{tmp}(X_{t+1})$. Since the UGV has received all UGVs' measurements of $t+1$, the $P^1_{tmp}(X_{t+1})$ is assigned as the new stored PDF. The dashed arrow on the right shows the order to fuse measurements in the CB.

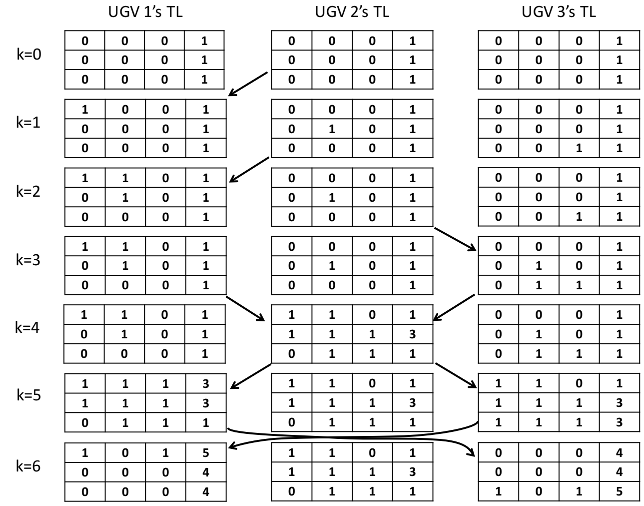



Fig. 4: Example of updating TLs. For the 1st UGV's TL, the $j^{\text{th}}$ $(j \in V)$ entry on the $i^{\text{th}}$ $(i \in V)$ row represents this UGV's knowledge about whether the $i^{\text{th}}$ UGV has received the $j^{\text{th}}$ UGV's state-measurement pair of time $k^{1i}$, where $k^{1i}$ is the last entry of the $i^{\text{th}}$ row. TLs are updated using Algorithms 3 and 4.

**Algorithm 3** Updating TLs

Consider updating the $i^{\text{th}}$ UGV's TL, $Q^i_k$, using the received $r^{\text{th}}$ UGV's TL, $Q^r_{k-1}(r \in \mathcal{N}^{\text{in}}_i(G_{k-1}))$.

**(1)** Update the $i^{\text{th}}$ row of $Q^i_k$, using $\mathcal{B}^i_k$:

choose $k^{ii}$ as the minimum integer that satisfies the following conditions: (a) $\exists l \in V$ s.t. $\left[x^l_{k^{ii}}, z^l_{k^{ii}}\right] \notin \mathcal{B}^i_k$ and (b) $k^{ii} \geq t_m - 1$, where $t_m$ is the minimum time of state-measurement pairs in $\mathcal{B}^i_k$.

**(2)** Update other rows of $Q^i_k$: $\forall j \in V \setminus \{i\}$

if $k^{ij} > k^{rj}$, keep current $\mathbf{q}^{ij}_{k^{ij}}$;
if $k^{ij} = k^{rj}$, $\mathbf{q}^{ij}_{k^{ij}} = \mathbf{q}^{ij}_{k^{ij}} \vee^2 \mathbf{q}^{rj}_{k^{rj}}$;
if $k^{ij} < k^{rj}$, $\mathbf{q}^{ij}_{k^{ij}} = \mathbf{q}^{rj}_{k^{rj}}$ and $k^{ij} = k^{rj}$.

it. To keep track of each UGV's reception of other UGVs' measurements, every UGV maintains a *track list* (TL), $Q^i_k = \left[\mathbf{q}^{i1}_{k^{i1}}, \dots, \mathbf{q}^{iN}_{k^{iN}}\right]^T$ $(\forall i \in V)$, where $\mathbf{q}^{ij}_{k^{ij}} = \left[q^{j1}_{k^{ij}}, \dots, q^{jN}_{k^{ij}}, k^{ij}\right]^T$ $(j \in V)$ is a $(N+1) \times 1$ binary vector. Therefore, a TL can be represented by a binary matrix of size $N \times (N+1)$, with the last column corresponding to the measurement time. $Q^i_k$ represents the $i^{\text{th}}$ UGV's knowledge of its own and other UGVs' measurements of the times specified by $k^{ij}$ $(j \in V)$: the entry $q^{jl}_{k^{ij}}$ equals 1 if the $i^{\text{th}}$ robot knows that the $j^{\text{th}}$ UGV has received the state-measurement pair of the $l^{\text{th}}$ UGV of time $k^{ij}$, $\left[x^l_{k^{ij}}, z^l_{k^{ij}}\right]$, and equals 0 if the $i^{\text{th}}$ robot cannot determine whether $\left[x^l_{k^{ij}}, z^l_{k^{ij}}\right]$ has been received by the $j^{\text{th}}$ robot, i.e.,

$$q^{jl}_{k^{ij}} = \begin{cases} 1 & \text{if} \quad \exists t \in \left[k^{ij}, k\right] \text{ s.t. } \left[x^l_{k^{ij}}, z^l_{k^{ij}}\right] \in \mathcal{B}^j_t, \\ 0 & \text{if} \quad \nexists t \in \left[k^{ij}, k\right] \text{ s.t. } \left[x^l_{k^{ij}}, z^l_{k^{ij}}\right] \in \mathcal{B}^j_t. \end{cases} \tag{5}$$

Therefore it can happen that $\left[x^l_{k^{ij}}, z^l_{k^{ij}}\right]$ has been received by the $j^{\text{th}}$ UGV but the $i^{\text{th}}$ UGV does not know this and thus $q^{jl}_{k^{ij}} = 0$.

The exchange and updating of TLs are described in Algorithm 1, with the updating details presented in Algorithm 3. For the $i^{\text{th}}$ UGV, it updates the $i^{\text{th}}$ row of its TL matrix using the entries of its CB, and updates other rows of the TL using the received TLs from inbound neighbors. The updating rule guarantees that, if the last term of the $j^{\text{th}}$ row is $k^{ij}$, the $i^{\text{th}}$ UGV is ensured that every UGV has received state-measurement pairs of times earlier than $k^{ij}$ from all UGVs. The Algorithm 4 describes the approach to trim CBs using TLs. Fig. 4 shows the updating of each UGV's TLs using the Algorithm 3. In the example, the 1st and 3rd UGV's CB will be trimmed at $k = 6$ and the trimmed state-measurement pairs corresponds to times $1, 2$, and $3$.

The use of TLs can avoid the excessive size of CBs and guarantee that trimming the CBs will not lose any information; the trimmed measurements have been encoded into the stored PDF. The following theorem formalizes this property.

**Theorem 2.** *Each UGV's estimation result using the trimmed CB is the same as that using the non-trimmed CB.*

*Proof.* Consider the $i^{\text{th}}$ UGV. Let $k^i_m = \min_j k^{ij}$. Trimming $\mathcal{B}^i_k$ happens when all entries in $Q^i_k$ corresponding to time $k^i_m$ equal 1. This indicates that each UGV has received the state-measurement pairs of time $k^i_m$ from all UGVs. A UGV has either stored the pairs in its CB or already fused them to obtain the stored PDF. In both cases, such pairs are no longer needed to be transmitted. Therefore, it causes no loss to trim theses measurements. $\square$

The following theorem describes when CBs get trimmed. Consider trimming all the state-measurement pairs of time $t$ in the $i^{\text{th}}$ UGV's CB. Let $k_t^{lj} (> t)$ be the first time that the $l^{\text{th}}$ UGV communicates to the $j^{\text{th}}$ UGV in the time interval $(t, \infty)$. Define $\tilde{k}_t^j = \max_l k_t^{lj}$, which is the time that the $j^{\text{th}}$ UGV receives all other UGVs' measurements of $t$. Similarly, let $k_t^{ji} (> \tilde{k}_t^j)$ be the first time that the $j^{\text{th}}$ UGV communicates to the $i^{\text{th}}$ UGV in the time interval $(\tilde{k}_t^j, \infty)$ and define $\tilde{k}_t^i = \max_j k_t^{ji}$. The following theorem gives the time when the $i^{\text{th}}$ UGV $(\forall i \in V)$ trims all state-measurement pairs of time $t$ in its own CB.

**Theorem 3.** *The $i^{th}$ UGV trims $\left\{ \left[x_t^l, z_t^l\right] (\forall l \in V) \right\}$ from its CB at the time $\tilde{k}_t^i$.*

*Proof.* The $i^{\text{th}}$ UGV can trim $\left\{ \left[x_t^l, z_t^l\right] (\forall l \in V) \right\}$ only when it is sure that all other UGVs have also received these state-measurement pairs. This happens at $\tilde{k}_t^i$ and is thus the time when the trim occurs.

□

**Corollary 2.** *Under the frequently jointly strongly connectedness condition, the size of any UGV's CB is no greater than $2N(N-1)T_u$.*

*Proof.* We consider an arbitrary $i^{\text{th}}$ $(i \in V)$ UGV. According to Theorem 1, a UGV can communicate to any other UGV within $NT_u$ steps. Therefore, $\tilde{k}_t^i \leq 2NT_u$, since it first requires each UGV communicate to all other UGVs and then each UGV communicate to the $i^{\text{th}}$ UGV. This implies that, the state-measurement pairs of a certain time of all UGVs will be trimmed from each UGV's CB within $2NT_u$ steps.

The maximum size the of CB occurs when the state-measurement pairs of a certain time from all but one UGV are saved in the $i^{\text{th}}$ UGV's CB. Therefore, the size of any UGV's CB is no greater than $2N(N-1)T_u$.

□

---

**Algorithm 4** Trimming CBs using TLs

---

For the $i^{\text{th}}$ UGV: find the smallest time in $Q_k^i$: $k_m^i = \min\left\{k^{i1}, \ldots, k^{iN}\right\}$.

1. Remove state-measurement pairs in $\mathcal{B}_k^i$ that corresponds to measurement times earlier than $k_m^i$, i.e., $\mathcal{B}_k^i = \mathcal{B}_k^i \setminus \left\{ \left[x_t^l, z_t^l\right] \right\}, \forall t < k_m^i, \forall l \in V$.
2. If entries associated with time $k_m^i$ in $Q_k^i$ are 1's, then
   (a) set these entries to be 0.
   (b) update the $i^{\text{th}}$ row of $Q_k^i$ using the current CB, i.e., $q_{k^{ij}}^{il} = 1$ if $\left[x_{k^{ij}}^l, z_{k^{ij}}^l\right] \in \mathcal{B}_k^i, \forall l \in V$.
   (c) remove all corresponding state-measurement pairs in $\mathcal{B}_k^i$, i.e., $\mathcal{B}_k^i = \mathcal{B}_k^i \setminus \left\{ \left[x_{k_m^i}^l, z_{k_m^i}^l\right] \right\}, \forall l \in V$.
   (d) $k_m^i \leftarrow k_m^i + 1$.

---

### 4.3 Complexity Analysis of FIFO-DBF

Compared to statistics dissemination, FIFO is usually more communication-efficient for distributed filtering. To be specific, consider a grid representation of the environment with the size $D \times D$. The transmitted data between each pair of UGVs are the CB and TL of each UGV. The size of the CB is upper bounded by $O(N^2 T_u)$, according to Corollary 2. On the contrary, the communicated data of a statistics dissemination approach that transmits unparameterized posterior distributions or likelihood functions is $O(D^2)$. In applications such as the target localization, $D$ is generally much larger than $N$ and the consensus filter usually requires multiple rounds to arrive at consensual results. Therefore, when $T_u$ is not comparable to $D^2$, the FIFO protocol requires much less communication burden.

It is worth noting that each UGV needs to store an individual PDF and a sotred PDF, each of which has size $O(D^2)$. In addition, each UGV needs to keep the CB and TL. This is generally larger than that of statistics dissemination-based methods, which only stores the individual PDF. Therefore, the FIFO-DBF sacrifices the local memory for reducing the communication burden. This is actually desirable for real applications as local memory of vehicles is usually abundant compared to the limited bandwidth for communication.

**Remark 2.** *Under certain interaction topologies, CBs can grow to undesirable sizes and causes excessive communication burden if the trim cannot happen frequently. In this case, we can use a time window to constrain the measurements that are saved in CBs. This will cause information loss to the measurements. However, with a decently long time window, FIFO-DBF can still effectively estimate the target position.*

## 5 Proof of Consistency

This section proves the consistency of the *maximum a posteriori* (MAP) estimator of LIFO-DBF under unbiased sensors (sensors without offset). A state estimator is *consistent* if it converges in probability to the true value of the state [31]. Consistency is an important metric for stochastic filtering approaches [8], which 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 and then for moving UGVs. Here we assume that $S$ is a finite set (e.g. a finely discretized field) and the target is relatively slow compared to the filtering dynamics. In addition, the target position can be uniquely determined by the multi-UGV network with proper placement (i.e., excluding the special case of ghost targets [32]).

### 5.1 Static UGVs

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

**Theorem 4.** *Assume the UGVs are static and the sensors are unbiased. If the network of N UGVs is frequently jointly strongly connected, then the MAP estimator of target position converges in probability to the true position of the target* *using FIFO-DBF, i.e.,*

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

*where*

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

*Proof.* Define the time set of $i^{\text{th}}$ UGV, $\mathcal{K}_k^{i,j}\,(j \in V)$, that contains the time steps of measurements by the $j^{\text{th}}$ UGV that are contained in $B_k^i$. The batch form of DBF at $k^{\text{th}}$ step is

$$P_{pdf}^i(X|\mathcal{Z}_{1:k}^i) = \frac{P_{pdf}^i(X)\prod_{j=1}^{N}\prod_{t\in\mathcal{K}_k^{i,j}} P(z_t^j|X)}{\sum_{X\in S} P_{pdf}^i(X)\prod_{j=1}^{N}\prod_{t\in\mathcal{K}_k^{i,j}} P(z_t^j|X)}.$$

Comparing $P_{pdf}^i(X_k = x|\mathcal{Z}_{1:k}^i)$ with $P_{pdf}^i(X_k = x^g|\mathcal{Z}_{1:k}^i)$[3] yields

$$\frac{P_{pdf}^i(x|\mathcal{Z}_{1:k}^i)}{P_{pdf}^i(x^g|\mathcal{Z}_{1:k}^i)} = \frac{P_{pdf}^i(x)\prod_{j=1}^{N}\prod_{t\in\mathcal{K}_k^{i,j}} P(z_t^j|x)}{P_{pdf}^i(x^g)\prod_{j=1}^{N}\prod_{t\in\mathcal{K}_k^{i,j}} P(z_t^j|x^g)}. \tag{6}$$

Take the logarithm of Eq. (6) and average it over $k$ steps:

$$\frac{1}{k}\ln\frac{P_{pdf}^i(x|\mathcal{Z}_{1:k}^i)}{P_{pdf}^i(x^g|\mathcal{Z}_{1:k}^i)} = \frac{1}{k}\ln\frac{P_{pdf}^i(x)}{P_{pdf}^i(x^g)} + \sum_{j=1}^{N}\frac{1}{k}\sum_{t\in\mathcal{K}_k^{i,j}}\ln\frac{P(z_t^j|x)}{P(z_t^j|x^g)}. \tag{7}$$

Since $P_{pdf}^i(x)$ and $P_{pdf}^i(x^g)$ are bounded and nonzero by the choice of the initial PDF, $\lim_{k\to\infty}\frac{1}{k}\ln\frac{P_{pdf}^i(x)}{P_{pdf}^i(x^g)} = 0$. The law of large numbers yields

$$\frac{1}{k}\sum_{t\in\mathcal{K}_k^{i,j}}\ln\frac{P(z_t^j|x)}{P(z_t^j|x^g)} \xrightarrow{P} \mathbb{E}_{z_t^j}\left[\frac{P(z_t^j|x)}{P(z_t^j|x^g)}\right] \tag{8a}$$

$$= \int_{z_t^j} P(z_t^j|x^g)\frac{P(z_t^j|x)}{P(z_t^j|x^g)}dz_t^j \tag{8b}$$

$$= -D_{KL}\left(P(z_t^j|x)\|P(z_t^j|x^g)\right), \tag{8c}$$

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 $P_2$. KL divergence has the property that $\forall P_1, P_2,\ D_{KL}(P_1\|P_2) \le 0$, and the equality holds if and only if $P_1 = P_2$. Therefore

$$\lim_{k\to\infty}\frac{1}{k}\sum_{t\in\mathcal{K}_k^{i,j}}\ln\frac{P(z_t^j|x)}{P(z_t^j|x^g)} < 0, \quad x \neq x^g$$

$$\lim_{k\to\infty}\frac{1}{k}\sum_{t\in\mathcal{K}_k^{i,j}}\ln\frac{P(z_t^j|x)}{P(z_t^j|x^g)} = 0, \quad x = x^g.$$

Considering the limiting case of Eq. (7), we get

$$\lim_{k\to\infty}\frac{1}{k}\ln\frac{P_{pdf}^i(x|\mathcal{Z}_{1:k}^i)}{P_{pdf}^i(x^g|\mathcal{Z}_{1:k}^i)} < 0, \quad x \neq x^g \tag{9}$$

$$\lim_{k\to\infty}\frac{1}{k}\ln\frac{P_{pdf}^i(x|\mathcal{Z}_{1:k}^i)}{P_{pdf}^i(x^g|\mathcal{Z}_{1:k}^i)} = 0, \quad x = x^g. \tag{10}$$

Eq. (9) and (10) imply that

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

Therefore,

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

□

### 5.2 Moving UGVs

The consistency proof for the moving UGVs case is different from the static UGVs case in that each moving UGV makes measurements at multiple different positions. We classify UGV measurement positions into two disjoint sets: *infinite-measurement spots* that contain positions where a UGV keeps revisiting 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). It is easy to know that each UGV has at least one position where it revisits infinitely many times as $k$ tends to infinity.

**Theorem 5.** *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 FIFO-DBF, i.e.,*

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

*Proof.* Similar to Eq. (6), comparing $P_{pdf}^i(x|\mathcal{Z}_k^i)$ and $P_{pdf}^i(x^g|\mathcal{Z}_k^i)$ yields

$$\frac{P_{pdf}^i(x|\mathcal{Z}_k^i)}{P_{pdf}^i(x^g|\mathcal{Z}_k^i)} = \frac{P_{pdf}^i(x)\prod_{j=1}^{N}\prod_{t\in\mathcal{K}_k^{i,j}} P(z_t^j|x;x_t^j)}{P_{pdf}^i(x^g)\prod_{j=1}^{N}\prod_{t\in\mathcal{K}_k^{i,j}} P(z_t^j|x^g;x_t^j)}. \tag{11}$$

The only difference from Eq. (6) is that $P(z_l^j|x;x_l^j)$ in Eq. (11) varies as the UGV moves. For the finite-measurement spots, by referring to Eq. (8), it is easy to know that their contribution to Eq. (7) diminishes when $k \to \infty$. Therefore, proof using Eq. (11) can be reduced to only considering the infinite-measurement spots and the rest of the proof is similar to that of Theorem 4.

□

## 6 Simulation

We conduct a simulation that uses a team of six UGVs to localize three moving targets. Every UGV maintains three

[3] For the purpose of simplicity, we use $P_{pdf}^i(x|\mathcal{Z}_{1:k}^i)$ to represent $P_{pdf}^i(X_k = x|\mathcal{Z}_{1:k}^i)$ in this proof.

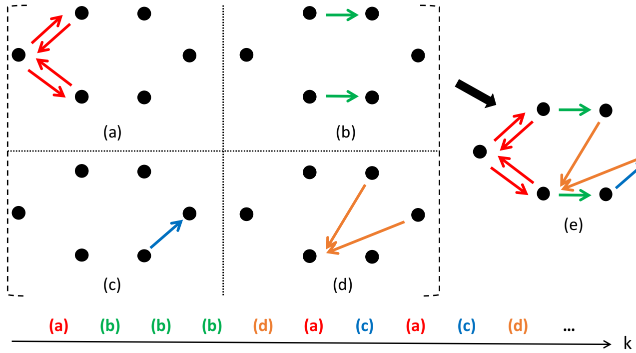



Fig. 5: The dynamically changing interaction topologies used in the simulation: (a)-(d) four types of topologies; (e) the union of these topologies is jointly strongly connected. The bottom axis shows a randomly generated sequence of topologies that satisfy the frequently jointly strongly connectedness condition.

individual PDFs, each corresponding to a target. At each time step, a UGV's sensor can measurement the positions of three targets. We assume that the UGVs know the association between the measurement and the corresponding target. The targets have different motion models, including the linear motion (target 1), sinusoidal motion (target 2), and circular motion (target 3). Three of the UGVs have range-only sensors and the other three UGVs have bearing-only sensors. The interaction topology of the UGVs is time-varying and consists of four types, as shown in Fig. 5(a)-(d). A randomly generated sequence of topologies is used (Fig. 5f). It can be noticed that, the interaction topology is frequently jointly strongly connectedwhen all four types appears repeatedly (Fig. 5e). Ten layouts of the initial positions of UGVs and targets are randomly generated. We compare FIFO-DBF with a consensus-based filter (CbDF) and a centralized filter (CF).

Fig. 6 show the simulation results of a specific layout. The sum of the $1^{\text{st}}$ UGV's individual PDFs are shown in the figures. Figs. 6a to 6f show that the FIFO-DBF can successfully localize and track moving target's positions and effectively reduce the estimation uncertainty, which is similar to the performance of the CF (Fig. 7b). On the contrary, CbDF is less effective to reduce the estimation uncertainty (Fig. 7a).

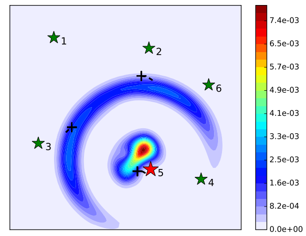
We quantitatively compare the three filters in terms of the estimation error and entropy of the uncertainty. The estimation error is defined as the difference between the true target position and the MAP estimate of the individual PDF:

$$\Delta_k = \|X_k^{\text{MAP}} - x_k^g\|_2.$$

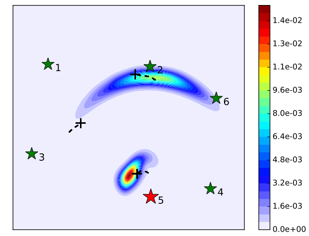
The entropy of the uncertainty is

$$H_k = \sum_{X_k \in S} -P_{pdf}(X_k)\log(P_{pdf}(X_k)).$$

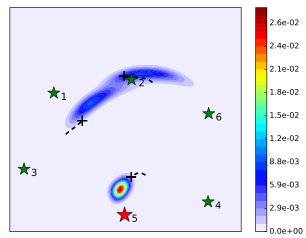
The average of the estimation error and entropy of each target across ten layouts are shown in Fig. 8. It can be noticed that, the CF achieves the most accurate position estimation and fastest entropy reduction. This is an expected result since the CF utilizes all sensor measurements. The FIFO-DBF



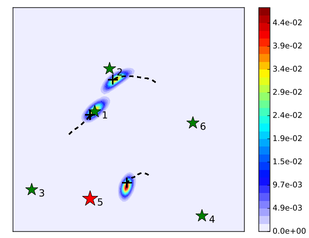
(a) FIFO-DBF at step 3



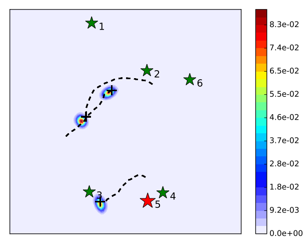
(b) FIFO-DBF at step 5



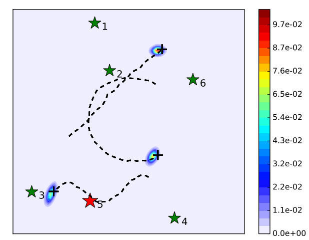
(c) FIFO-DBF at step 7



(d) FIFO-DBF at step 10

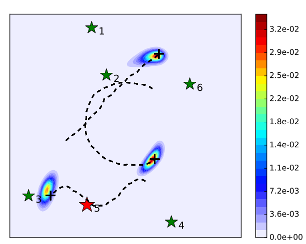


(e) FIFO-DBF at step 20

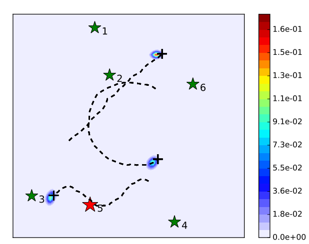


(f) FIFO-DBF at step 40

Fig. 6: Evolution of the target estimation using FIFO-DBF. The colorful background represents the sum of the individual PDF of three targets.



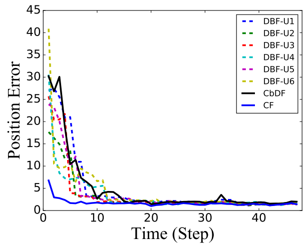
(a) CbDF at Step 40



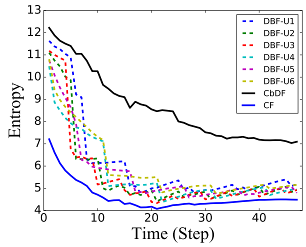
(b) CF at Step 40

Fig. 7: Target estimation using (a) CbDF and (b) CF. The estimation uncertainty remains large for CbDF.

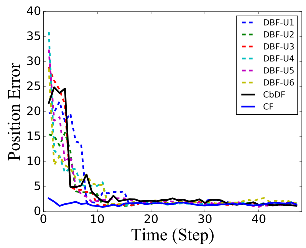
achieves similar results as the CF asymptotically. This is a very interesting results, since FIFO-DBF only communicates with neighboring UGVs and have a subset of other UGVs' measurements. The CbDF achieves similar position estimation performance as the CF and FIFO-DBF. However, it fails to effectively reduce the estimation entropy. This is because that, the linear combination of PDFs used in the CbDF does not follow the nonlinear nature of Bayesian filtering, thus information is loss during the combination. The FIFO-DBF, on the other hand, rigorously follows the procedure of Bayesian filtering, and therefore achieves better performance.



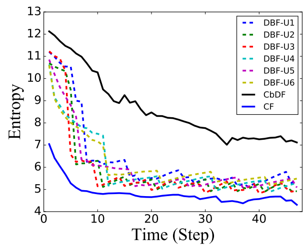

(a) Position Error of Target 1



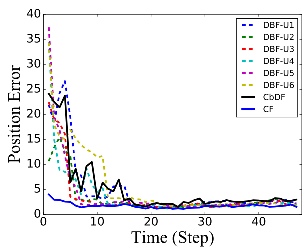

(b) Entropy of Target 1



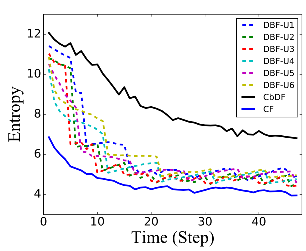

(c) Position Error of Target 2




(d) Entropy of Target 2




(e) Position Error of Target 3




(f) Entropy of Target 3

Fig. 8: First scenario: (a) two types of topologies; (b) individual PDF of the 3rd UGV after initial observation; (c)-(e) PDFs at the end of simulation using different filters; (f) average position estimation errors; (g) average entropy of PDF. In last two figures, metrics are based on the PDFs of the 1st, 3rd and 5th UGV using FIFO-DBF, the common PDF using CbDF and using CF.

## 7 Conclusion

This paper presents a general measurement dissemination-based distributed Bayesian filter (DBF) method for a network of multiple unmanned ground vehicles (UGVs) under dynamically changing interaction topologies. The information exchange among UGVs relies on the Full-In-and-Full-Out (FIFO) protocol, under which UGVs exchange the communication buffers and track lists with neighbors. Under the condition that the union of the interaction topologies is frequently jointly strongly connected, FIFO can disseminate measurements over the network within finite time. By using the track list, the CBs can be trimmed without causing information loss. The FIFO-DBF algorithm is developed to estimate individual probability density function for target localization. The FIFO-DBF can significantly reduce the transmission burden between each pair of UGVs compared to the statistics dissemination methods. Simulations comparing FIFO-DBF with consensus-based distributed filters (CbDF) and the centralized filter (CF) show that FIFO-DBF achieves similar performance as the CF and superior performance over the CbDF.

## References


[1] J. K. Hedrick, B. Basso, J. Love, and B. M. Lavis, "Tools and techniques for mobile sensor network control," *Journal of Dynamic Systems, Measurement, and Control*, vol. 133, no. 2, p. 024001, 2011.

[2] 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.

[3] N. A. Atanasov, J. Le Ny, and G. J. Pappas, "Distributed algorithms for stochastic source seeking with mobile robot networks," *Journal of Dynamic Systems, Measurement, and Control*, vol. 137, no. 3, p. 031004, 2015.

[4] A. Madhag and J. Choi, "A distributed navigation strategy for mobile sensor networks with the probabilistic wireless links," *Journal of Dynamic Systems, Measurement, and Control*, vol. 139, no. 3, p. 031004, 2017.

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

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

[7] 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.

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

[9] 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*, pp. 1–4.

[10] M. Vemula, M. F. Bugallo, and P. M. Djurić, "Target tracking in a two-tiered hierarchical sensor network," in *ICASSP 2006 Proceedings.*, vol. 4, pp. IV–IV.

[11] A. Jadbabaie, J. Lin, and A. S. Morse, "Coordination of groups of mobile autonomous agents using nearest neighbor rules," *IEEE Transactions on automatic control*, vol. 48, no. 6, pp. 988–1001, 2003.

[12] 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.

[13] 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.

[14] 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.

[15] 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.
[16] 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.
[17] D. Gu, "Distributed particle filter for target tracking," in *Robotics and Automation, 2007 IEEE International Conference on*, pp. 3856–3861.
[18] 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.
[19] J. Beaudeau, M. F. Bugallo, and P. M. Djuric, "Target tracking with asynchronous measurements by a network of distributed mobile agents," in *in ICASSP 2012 Proceedings*, pp. 3857–3860.
[20] 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.
[21] 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.
[22] 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.
[23] P. M. Djurić, J. Beaudeau, and M. F. Bugallo, "Non-centralized target tracking with mobile agents," in *ICASSP 2011 Proceedings*, pp. 5928–5931.
[24] F. Xiao and L. Wang, "Asynchronous consensus in continuous-time multi-agent systems with switching topology and time-varying delays," *Automatic Control, IEEE Transactions on*, vol. 53, no. 8, pp. 1804–1816, 2008.
[25] Y. Xia, J. Shang, J. Chen, and G.-P. Liu, "Networked data fusion with packet losses and variable delays," *IEEE Transactions on Systems, Man, and Cybernetics, Part B (Cybernetics)*, vol. 39, no. 5, pp. 1107–1120, 2009.
[26] 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.
[27] C. Liu, S. E. Li, and J. K. Hedrick, "Distributed target localization using a group of ugvs under dynamically changing interaction topologies," in *American Control Conference (ACC), 2016*. IEEE, 2016, pp. 2585–2590.
[28] 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.
[29] 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.
[30] 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.
[31] T. Amemiya, *Advanced econometrics*. Harvard university press, 1985.
[32] M. Malanowski and K. Kulpa, "Two methods for target localization in multistatic passive radar," *IEEE transactions on Aerospace and Electronic Systems*, vol. 48, no. 1, pp. 572–580, 2012.