Distributed Bayesian Filters for Multi-Robot Network by Using Local-Exchange-of-Observation Strategy

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

Abstract— **This paper proposes a local-exchange-of-observation (LEO) strategy for distributed Bayesian filters (DBF) in a multi-robot network with the application of target search and tracking. Different from statistics dissemination approaches that transmit posterior distributions or likelihood functions, LEO only exchanges observations within communication-available neighbors, which significantly reduces the transmission burden of each pair from the order of environmental size to that of robot number. Under the condition of fixed and undirected topology, LEO can guarantee the global dissemination of all observations over the network within finite time, with each robot non-intermittently receiving observations of all others. Two LEO-DBFs are proposed for SAT of static and moving target, respectively. For the static target, each robot locally fuses the latest available knowledge of all robots' observations by only considering the updating step of the Bayesian filter. For the moving target, a triangle matrix of historical observations is maintained by each robot. Upon obtaining the latest available observations of all robots, an iterative Bayesian filtering procedure is applied that alternates between prediction and updating steps. The consistency of LEO-DBF is proved, ensuring the agreement between robots' estimated target position and the actual position. Simulations are generated to demonstrate the effectiveness of LEO-DBF for SAT of both static and moving targets.**

I. INTRODUCTION

Distributed filtering in multi-robot network that focuses on using a group of networked robots to collectively infer the state of an environment has been used for various applications, such as object detection [1], target tracking [2] and environmental monitoring [3]. Several techniques have been developed for distributed filtering. For example, in [4], a distributed Kalman filter (DKF) was proposed for estimating states of linear systems with Gaussian process and measurement noise. Each DKF used low-pass and band-pass consensus filters for the average-censensus of weighted measurements and inverse-covariance matrices. Gu [5] presented a distributed particle filter for target tracking over sensor networks. Gaussian mixture model (GMM) was adopted to approximate the posterior distribution from weighted particles and the parameters of GMM was exchanged via average consensus filter. This study focuses on developing a distributed Bayesian filter (DBF) that is applicable for state estimation of general nonlinear systems and the proposed DBF is applied to search and tracking (SAT) of both static and moving targets.

The communication topology of multi-robot network plays a vital role in distributed filtering algorithms. Fusion center (FC)-based DBF has been a common structure for distributed filtering, in which local information collected by robots is transmitted (possibly via multi-hopping) to the fusion center for forming global estimation [6], [7]. FC-based DBF is efficient for estimation in that it can collectively utilize all robots' information and thus useful for applications that only require information at a single central unit, such as in environmental monitoring.

Neighborhood(NB)-based DBF is another commonly adopted structure for distributed filtering. Instead of communicating with a fusion center, each robot only exchanges information with neighboring robots and forms local estimation of the environment state. NB-based DBF is advantageous over FC-based DBF in that no central unit is required, thus suitable for applications in which maintaining communication link between robots and center is challenging, such as in disaster situations. Besides, state estimation is locally conducted on each robot, which requires less computation power compared to that in the fusion center.

Thus far, works on NB-based DBF have mainly focused on the so-called *statistics dissemination-based* strategy that each robot actually exchanges posterior distributions or likelihood functions to neighboring robots for distributed estimation. For example, Sheng et al.[8] proposed a leader agent-based distributed particle filter (DPF) with Gaussian Mixer to track multiple moving targets. DPFs were run on a set of uncorrelated sensor cliques and the particles were approximated as GMMs, the parameters of which were then exchanged among cliques for global estimation of targets. A popular strategy for statistics dissemination is to use consensus-based approaches that all robots perform distributed filtering simultaneously, exchange statistics with neighbors and executes consensus algorithms, as proposed in [9], [10], [11], for fusion of statistics. For example, Saptarshi et al. [12] presented a Bayesian consensus filter (BCF) that uses logarithmic opinion pool for fusing posterior distributions of the tracked target among neighboring robots. A consensus-based distributed particle filter (DPF) is proposed by Julian et al. [13] for estimating environment state by exchanging and fusing posterior functions of the state among neighbors using a discrete-time linear consensus algorithm.

Despite the popularity of statistics dissemination-based

*This work is supported by the Embedded Humans: Provably Correct Decision Making for Networks of Humans and Unmanned Systems project, a MURI project funded by the Office of Naval Research.

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

[2]Shengbo Eben Li is with the Dept. of Automotive Engineering, Tsinghua University, Beijing, 100084, China lisb04@gmail.com

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

approaches, exchanging posterior distributions or likelihood functions can consume high communication resources, which may be infeasible for applications in vast area or complex environment, such as marine search, seismological rescue, etc. This study focuses on the strategy of exchanging observations in the neighborhood of each robot, called the *measurement dissemination-based* strategy, for the purpose of achieving a consensus of the probability density function (PDF) of the tracked target. Some pioneering studies have been done on measurement dissemination strategies, in which raw or quantized observations are exchanged among robots. For example, Coates et al.[14] used adaptive encoding of observations to minimize communication overhead for tracking a manoeuvring object. Djuric et al. [15] proposed a decentralized particle filter for tracking targets. At each time instant, a subset of robots that are in proximity of the tracked targets share their observations for target position estimation. Another example can be found in [16], in which both observations and statistics were exchanged among sensors for distributed surveillance of the environment. In aforementioned works, either communication topology is assumed fully connected that each robot can broadcast observations to all other robots in single transmission step or the communication involves large amount of exchanged data.

This paper proposes a local-exchange-of-observation (LEO) strategy for distributed Bayesian filters (DBF) for undirected and connected communication topology. Each robot only broadcasts observations to its neighbors and implements Bayesian filter locally after receiving observations transmitted from neighboring robots. The main benefit of LEO is the reduction of communication burden, with the transmission data volume scaling linearly with the robot number, while statistics dissemination-based strategies suffer from the data volume on the order of environmental size. The proposed LEO-DBF has following properties: (1) For an undirected and connected network with fixed topology, LEO guarantees the global dissemination of all robots' observations among the network via multi-hopping, with each robot non-intermittently receiving (delayed) observations of all other robots via local communication. (2) LEO-DBF ensures consistency of the estimation of states, which refers to the agreement between robots' estimates of target position and the true position of the target. Moreover, consistency implies the consensus of robots' target PDFs. In this study, formal proofs of the consistency of LEO-DBF and consensus of robots' target PDFs are provided.

The rest of this paper is organized as follows: The distributed SAT of target is formulated in Section II. The LEO-DBF algorithm is described in Section III, followed by the proof of consistency and consensus in Section IV. Simulation results of LEO-DBF and conclusions are presented in Section V and Section VI, respectively.

II. PROBLEM FORMULATION

Consider a network of N robots in a bounded two-dimensional space S. Each robot is equipped with a binary sensor for environmental perception. Due to the limit of communication range, each robot can only exchange observations with its neighbors. Bayesian filter is run locally on each robot for target search and tracking based on its own and received observations.

A. Probabilistic Model of Binary Sensor

The binary sensor only gives two types of observation: 1 if the target is detected, and 0 if no target is detected. The observation of i^{th} sensor at k^{th} time step is denoted as z_k^i. The likelihood function that the target is detected is:

$$P(z_k^i = 1 | x_k^T; y_k^R) \in [0, 1]\,,\ x_k^T \in X^T \subseteq S \qquad (1)$$

where x_k^T denotes the target positions; y_k^R is the robot state (position, sensor orientation etc.); X^T represents the set of all possible target positions.

Correspondingly, the likelihood function that no target is detected is:

$$P(z_k^i = 0 | x_k^T; x_k^R) = 1 - P(z_k^i = 1 | x_k^T; x_k^R) \qquad (2)$$

The combination of Eqn. (1) and Eqn. (2) is actually binary sensor model, parameterized by x_k^T. The goal of Bayesian filter is to estimate the actual target position $x_k^{T^*}$, with which all observations are generated. For the purpose of simplicity, we will not explicitly write x_k^R in binary sensor model for the rest of the paper.

The commonly used likelihood functions for binary sensor include Gaussian function [17], [18] and step function [19]. In addition, LEO also works for other types of sensors, including imaging sensors, such as laser scanners [20] and cameras [21].

Remark 1: The current observation of each robot is conditionally independent from both its own past observations and those of other robots, given the knowledge of current target position.

Remark 2: This study is applicable for both homogeneous and heterogeneous binary sensors. A homogeneous model can simplify the analysis of completeness, while the heterogeneous model is more close to real sensing characteristics.

B. Graphical model of communication topology

Consider an undirected and connected graph $G = (V, E)$, where $V = \{1, \ldots, N\}$ represents the index set of robots and $E = V \times V$ denotes the edge set. The adjacency matrix $M = [m_{i,j}]$ 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} robot and $m_{ij} = 0$ indicates no communication between them.

The *direct neighborhood* of i^{th} robot is defined as $\mathcal{N}_i = \{j | m_{ij} = 1, \forall j \in \{1, \ldots, N\}\}$. All the robots in $\mathcal{N}_i$ can directly exchange information with i^{th} robot. In addition to

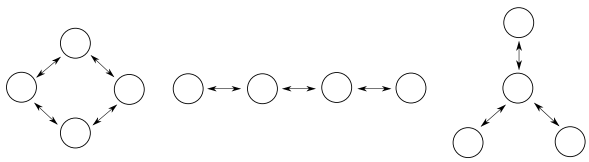


Fig. 1: Three types of topologies: (a)circular topology; (b)serial topology; (c)star topology

direct neighborhood, another set, called *available neighborhood*, is defined as $\mathcal{Q}_i$, which contains indices of robots whose observations are received by the i^{th} robot given a specific observation exchange algorithm. Note that in general $\mathcal{N}_i \subseteq \mathcal{Q}_i$, but if only single-hopping is allowed, $\mathcal{Q}_i = \mathcal{N}_i$. Fig. 1 illustrates three types of typical topologies: circular [22], serial [23], and star [8]. All of them are undirected and connected topologies.

C. Distributed Bayesian Filter for Multiple Robots

The generic distributed Bayesian filter (DBF) is introduced in this section, which was also stated in [24] and [25]. Each robot has its individual estimation of PDF of target position, called *individual PDF*. The i^{th} individual PDF at time k is defined as $P^i_{pdf}(x^T_k|z^i_{1:k}, z^{\mathcal{Q}_i}_{1:k})$. Before starting DBF, the individual PDF is initialized by the prior function $P^i_{pdf}(x^T_0|z^i_0, z^{\mathcal{Q}_i}_0) = P(x^T_0)$, given all available prior information including past experience and environmental knowledge. Then the individual PDF is recursively estimated by two steps, i.e., prediction step and updating step, based on the observations of i^{th} robot and robots in $\mathcal{Q}_i$.

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

$$\begin{aligned} &P^i_{pdf}(x^T_k|z^i_{1:k-1}, z^{\mathcal{Q}_i}_{1:k-1}) \\ &= \int P(x^T_k|x^T_{k-1}) P^i_{pdf}(x^T_{k-1}|z^i_{1:k-1}, z^{\mathcal{Q}_i}_{1:k-1}) dx^T_{k-1} \end{aligned} \tag{3}$$

where $P(x^T_k|x^T_{k-1})$ is a Markov motion model of the target, independent of robot states. This model describes the state transition probability of the target from a prior state x^T_{k-1} to posterior state x^T_k. Note that the target can be static in many search applications, such as the indoor search for stationary objects[26]. For a static target, its Markov motion model is simplified to be

$$P(x^T_k|x^T_{k-1}) = \begin{cases} 1 & \text{if } x^T_k = x^T_{k-1} \\ 0 & \text{if } x^T_k \neq x^T_{k-1} \end{cases}.$$

2) Updating: At time k, the i^{th} individual PDF is then updated by Bayes' formula using the latest available observations at time k:

$$\begin{aligned} &P^i_{pdf}(x^T_k|z^i_{1:k}, z^{\mathcal{Q}_i}_{1:k}) \\ &= K_i P^i_{pdf}(x^T_k|z^i_{1:k-1}, z^{\mathcal{Q}_i}_{1:k-1}) P(z^i_k|x^T_k) \prod_{j\in\mathcal{Q}_i} P(z^j_k|x^T_k) \end{aligned} \tag{4}$$

where K_i is a normalization factor, given by:

$$K_i = 1/\int P^i_{pdf}(x^T_k|z^i_{1:k-1}, z^{\mathcal{Q}_i}_{1:k-1}) P(z^i_k|x^T_k) \prod_{j\in\mathcal{Q}_i} P(z^j_k|x^T_k) dx^T_k$$

where $P^i_{pdf}(x^T_k|z^i_{1:k}, z^{\mathcal{Q}_i}_{1:k})$ is called posterior individual PDF; $P(z^i_k|x^T_k)$ is the likelihood function of observation z^i_k, described in Eqn. (1) and Eqn. (2).

III. Distributed Bayesian Filter via Local Exchange of Observations

This study proposes a distributed Bayesian filtering (DBF) algorithm based on LEO strategy, shorted as LEO-DBF. The LEO only uses the local communication within the neighborhood of each robot, but allows to broadcast observations of each robot to any other nodes by multi-hopping along the shortest path in the undirected and connected network. The theoretical analysis show that LEO-DBF can ensure the consistency and consensus of individual PDF while requiring much less communication burden than any statistics dissemination-based DBFs.

A. Algorithm for Local Exchange of Observations (LEO)

Under the LEO, each robot contains a communication buffer (CB) to store its latest knowledge of the observations of all robots:

$$\mathbf{z}^i_k = \left[z^1_{k^i_1}, \dots, z^N_{k^i_N}\right]$$

where $z^j_{k^i_j}$ represents the observation made by j^{th} robot at time k^i_j. Note that under LEO, $\mathcal{Q}_i = \{1, \dots, N\}$, which will be proved in Corollary 1. At time k, $z^j_{k^i_j}$ is received and stored in i^{th} robot CB, in which k^i_j is the latest available observation time of j^{th} robot known by i^{th} robot. Due to the communication delay of multi-hopping, $k^i_j < k, \forall j \neq i$ always holds in practice.

The **LEO algorithm** is stated as follows:

(1) Initialization: The buffer of i^{th} robot is initialized when $k = 0$:

$$z^j_{k^i_j} = \emptyset,\ k^i_j = 0,\ j = 1:N$$

(2) Repeat the following steps for i-th robot until stop:

(2.1) Receiving Step:

The i^{th} robot receives all CBs of its neighboring robots. The received CBs are totally $|\mathcal{N}_i|$ groups, each of which corresponding to the (k-1)-step CB of a robot in $\mathcal{N}_i$. The received CB from l^{th} ($l \in \mathcal{N}_i$) robot is denoted as

$$\mathbf{z}^l_{k-1} = \left[z^1_{(k-1)^l_1}, \dots, z^N_{(k-1)^l_N}\right],\ l \in \mathcal{N}_i$$

(2.2) Observation Step:

The i^{th} robot 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:

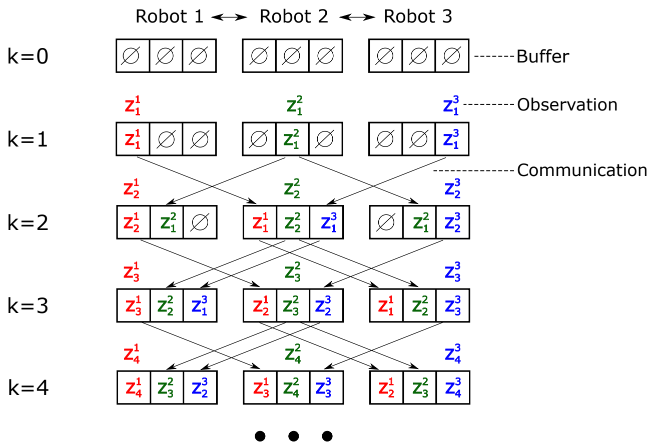


Fig. 2: Example of LEO with three robots using serial communication topology

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

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

(2.4) Sending Step:

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

Fig. 2 illustrates the LEO algorithm with 3 robots using a serial topology. Two facts can be observed in Fig. 2: (1) all robot CBs are filled within 3 steps, which means under LEO each robot has a maximum delay of 2 steps for receiving observations from other robots; (2) after filled, the updating of CBs are non-intermittent, which means each robot continuously receives newer observations of other robots. Extending the two facts to a network of N robots, we have the following proposition:

Proposition 1: For an undirected and connected network of N robots with fixed topology, LEO uses the shortest path(s) between i^{th} and j^{th} robot to exchange observation. Moreover, the delay $\tau_{i,j}$ between i^{th} and j^{th} robot is equivalent to the length of shortest path(s) connecting them.

Proof: Without loss of generality, assume that there is a unique shortest path between i and j, denoted by $T^{j,i}_{n^*} = (v_1, \dots, v_{n^*})$, with $v_1 = j, v_{n^*} = i, v_{m+1} \in \mathcal{N}_{v_m}$. Then, the distance between i and j is $d(j,i) = n^* - 1$. The following mathematical induction will prove Proposition 1.

Step (1): For $d(j,i) = 1$, this means $j \in \mathcal{N}_i$ and j can directly send z^j_k to i. Then z^j_k is stored in i^{th} CB at time $k+1$, i.e. $\tau_{i,j} = 1$. Proposition 1 holds for $d(j,i) = 1, \; i,j \in \{1, \dots, N\}$.

Step (2): Suppose that Proposition 1 holds for $d(j,i) = s, \; s \geq 2$. Then for $d(j,i) = s+1$, i.e., $n^* = s+2$, by the Bellman's principle of optimality, the path $T^{j,l}_{n^*-1} = (v_1, \dots, v_{n^*-1})$ is a shortest path between j and l, where $v_{n^*-1} = l$ and $i \in \mathcal{N}_l$. The assumption that Proposition 1 holds for $d(j,i) = s$ implies that z^j_k is received and stored in l^{th} robot's CB at time $k+s$. Since $i \in \mathcal{N}_i$, i^{th} robot receives z^j_k at $k+s+1$. For any other path $T^{j,i}_n = (v_1, \dots, v_n)$ with $n > n^*$, z^j_k cannot be received by i earlier than $k+s+1$. Therefore $\tau_{i,j} = s+1$. This proves the Proposition 1 for $d(j,i) = s+1$. ∎

Corollary 1: For the same topology assumption in Proposition 1, all elements in $\mathbf{z}^i_k$ under LEO become filled when $k \geq N$.

Proof: In a network of N robots, the maximal length of shortest paths is no greater than $N-1$. Based on Proposition 1, $\tau_{i,j} \leq N-1$ and thus all elements of $\mathbf{z}^i_k$ become filled when $k \geq N$. ∎

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

Proof: For a network with fixed topology, the shortest path between any nodes is fixed. Therefore, based on Proposition 1, $\tau_{i,j}$ is constant and the updating of each element in $\mathbf{z}^i_k$ is non-intermittent. ∎

Remark 3: Compared to statistics dissemination, LEO is a more communication-efficient approach for distributed filtering. To be specific, consider an $M \times M$ grid environment with a network of N robots, the transmitted data of LEO between each pair of robots are only the CB of each robot, the length of which is $O(N)$. On the contrary, the length of transmitted data for a statistics-dissemination approach is $O(M^2)$, which is the size of the environment. Since M is generally much larger than N in target search and tracking, LEO requires much less communication resources than the statistics-dissemination approaches.

B. Algorithm of LEO-DBF for Static Target

This section gives the LEO-DBF for a static target. Each robot stores last-step individual PDF, i.e., $(k-1)^{\text{th}}$ step. The assumption of static target can simplify the Bayesian filter as the prediction step becomes unnecessary. Therefore, the i^{th} individual PDF is only updated by

$$P^i_{pdf}(x^T | \mathbf{z}^i_{1:k}) = K_i P^i_{pdf}(x^T | \mathbf{z}^i_{1:k-1}) \prod_{j=1}^{N} P(z^j_{k^i_j} | x^T) \quad (5)$$

where

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

C. Algorithm of LEO-DBF for Moving target

This section gives the LEO-DBF for a moving target. Instead of storing last-step PDF, each robot maintains an individual PDF for the time $(k-N)$ and a triangular matrix of historical observations from time $(k-N+1)$ to k. The i^{th} individual PDF is then alternatively predicted and updated by using aforementioned Bayesian filter (Eqn. (3) and Eqn. (4)) from time $(k-N)$ to k. Fig. 3 illustrates the LEO-DBF procedure for the 1^{st} robot.

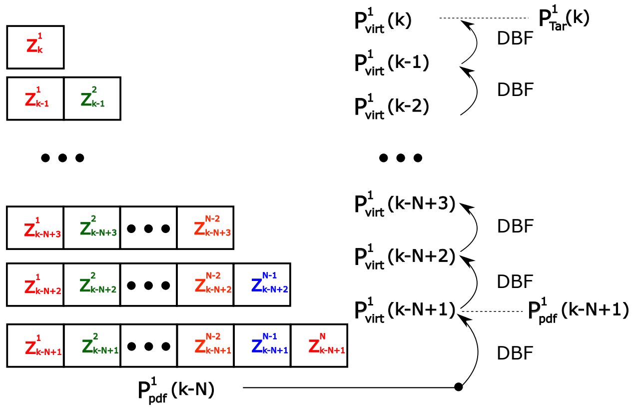


Fig. 3: Example of LEO-DBF for 1st robot at time k. The current individual PDF is $P^1_{pdf}(x^T_{k-N}|z^1_{1:k-N},\dots,z^N_{1:k-N})$, denoted as $P^1_{pdf}(k-N)$ in the figure. The robot first calculates $P^1_{virt}(k-N+1)$ using DBF and stores it as $P^1_{pdf}(k-N+1)$. Repeating DBF until obtaining $P^1_{pdf}(k)$, which is then used as the target PDF estimation of i-th robot at time k.

For the i^{th} robot at k^{th} step, the **LEO-DBF algorithm** for moving target is as follows:

(1) The stored individual PDF for time $(k-N)$ is:

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

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

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

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

(3.1) Prediction

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

(3.2) Updating

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

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

(4) Store the first-step virtual PDF as the individual PDF for time $(k-N+1)$

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

■

Note that Ω^i_ξ denotes the index set of robots whose observation at time $(k-N+\xi)$ is stored in i^{th} robot's CB. The estimated target PDF of i^{th} robot at time k is $P^i_{virt}(x^T_k)$.

Remark 4: For the static target, each robot only needs current step CB to update individual PDFs. Therefore, except storing individual PDFs, all historical CBs can be discarded and only current-step CB is stored in robot memory, the length of which is $O(N)$. On the contrary, for the moving target, each robot needs to store a triangular matrix of history observation (except current step CB) with size of $O(N^2)$ and an individual PDF with size $O(M^2)$, which means that the size of occupied memory in each robot is $O(M^2+N^2)$.

IV. Proof of Consistency and Consensus

This section presents consistency and consensus proof of LEO-DBF. Only proofs for static target SAT using static robots and moving robots are presented. The proof of LEO-DBF for moving target SAT is similar to that of static target by utilizing the dynamic model of the target, but with more complicated algebraic manipulation.

Assume that X^T is finite. Define an *equi-parameter* set $X^T_{eq} \subseteq X^T$ such that

$$P(z_k|x^{T,1}) = P(z_k|x^{T,2}),\ \forall x^{T,1},x^{T,2}\in X^T_{eq},\ \forall z_k\in\{0,1\},$$

i.e., the probability for generating the same observation z_k is equivalent among sensor models with different parameters in X^T_{eq}. Since X^T is finite, X^T_{eq} is also finite. Let $X^T_{eq,1},\dots,X^T_{eq,u}$ denote all equi-parameter sets that partition X^T such that following properties hold:

1) $\bigcup_{i=1}^u X^T_{eq,i} = X^T$
2) $X^T_{eq,i}\cap X^T_{eq,j} = \emptyset,\ i,j\in\{1,\dots,u\},i\neq j.$

Without loss of generality, assume $x^{T^*}\in X^T_{eq,1}$, where x^{T^*} denotes the actual position of the target.

Remark 5: the equi-parameter set depends on the property of the sensor. For example, for a laser scanner with high-fidelity sensing capability, each equi-parameter set contains only a small number of target positions.

A. *Proof for static robots*

The consistency of LEO-DBF for static robots is stated as follows:

Theorem 1: For static robots, each individual PDF converges to $X^T_{eq,1}$ using LEO-DBF when the number of observations tends to infinity, i.e.

$$P^i_{pdf}(x^T\in X^T_{eq,i}|\mathbf{z}^i_{1:k}) \longrightarrow \begin{cases}1 & \text{if } i=1\\ 0 & \text{if } i\neq 1\end{cases},\ k\to\infty$$

where $\mathbf{z}^i_{1:k} = (z^1_{1:k_1},\dots,z^N_{1:k_N})$, $\{k_1,\dots,k_N\}$ are the timestamps of the i^{th} robot's latest knowledge of all robots' observations.

Proof: Considering the conditional independence of observations z^j_k for given $x^T\in X^T_{eq,i}$, the batch form of DBF at k^{th} step is:

$$\begin{aligned}P^i_{pdf}(x^T|\mathbf{z}^i_{1:k}) &= P^i_{pdf}(x^T|z^1_{1:k_1},\dots,z^N_{1:k_N})\\ &= \frac{P^i_{pdf}(x^T)\prod\limits_{j=1}^N\prod\limits_{l=1}^{k_j}P(z^j_l|x^T)}{\int_{X^T}\prod\limits_{j=1}^N\prod\limits_{l=1}^{k_j}P(z^j_l|x^T)dx^T},\end{aligned}$$

where P^i_{pdf} is i^{th} robot's initial individual PDF. It is known from Corollary 1 and Corollary 2 that $k-N<k_j\leq k$.

Compare $P^i_{pdf}(x^T|\mathbf{z}^i_{1:k})$ with $P^i_{pdf}(x^{T*}|\mathbf{z}^i_{1:k})$:

$$\frac{P^i_{pdf}(x^T|\mathbf{z}^i_{1:k})}{P^i_{pdf}(x^{T*}|\mathbf{z}^i_{1:k})} = \frac{P^i_{pdf}(x^T)\prod_{j=1}^{N}\prod_{l=1}^{k_j} P(z^j_l|x^T)}{P^i_{pdf}(x^{T*})\prod_{j=1}^{N}\prod_{l=1}^{k_j} P(z^j_l|x^{T*})} \tag{6}$$

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

$$\frac{1}{k}\ln\frac{P^i_{pdf}(x^T|\mathbf{z}^i_{1:k})}{P^i_{pdf}(x^{T^*}|\mathbf{z}^i_{1:k})} = \frac{1}{k}\ln\frac{P^i_{pdf}(x^T)}{P^i_{pdf}(x^{T^*})}+\sum_{j=1}^{N}\frac{1}{k}\sum_{l=1}^{k_j}\ln\frac{P(z^j_l|x^T)}{P(z^j_l|x^{T^*})} \tag{7}$$

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

$$\lim_{k\to\infty}\frac{1}{k}\ln\frac{P^i_{pdf}(x^T)}{P^i_{pdf}(x^{T^*})} \longrightarrow 0 \tag{8}$$

The binary observations subject to Bernoulli distribution $B(1,p_j)$, yielding

$$P^i_{pdf}(z^j_k|x^T) = p_j^{z^j_k}(1-p_j)^{1-z^j_k}$$

where $p_j = P(z^j_k = 1|x^T)$. Utilizing the facts: (1) z^j_l are conditionally independent samples from $B(1,p^*_j)$ and (2) $k-N < k_j \le k$, the law of large numbers yields

$$\lim_{k\to\infty}\frac{1}{k}\sum_{l=1}^{k_j} z^j_l = p^*_j, \quad \lim_{k\to\infty}\frac{1}{k}(k_j - \sum_{l=1}^{k_j} z^j_l) = 1-p^*_j$$

where $p^*_j = P(z^j_k = 1|x^{T^*})$. Then,

$$\lim_{k\to\infty}\frac{1}{k}\sum_{l=1}^{k_j}\ln\frac{P(z^j_l|x^T)}{P(z^j_l|x^{T^*})} = p^*_j\ln\frac{p_j}{p^*_j} + (1-p^*_j)\ln\frac{1-p_j}{1-p^*_j} \tag{9}$$

Note that the right-hand side of Eqn. (9) achieves maximum if and only if $p_j = p^*_j$. Considering Eqn. (8) and Eqn. (9), the limit of Eqn. (7) can be obtained:

$$\lim_{k\to\infty}\frac{1}{k}\ln\frac{P^i_{pdf}(x^T|\mathbf{z}^i_{1:k})}{P^i_{pdf}(x^{T^*}|\mathbf{z}^i_{1:k})} = \sum_{j=1}^{N} p^*_j\ln\frac{p_j}{p^*_j} + (1-p^*_j)\ln\frac{1-p_j}{1-p^*_j} \tag{10}$$

It is known from Eqn. (10) that

1) When $p_j \neq p^*_j$, that is $x^T \notin X^T_{eq,1}$,

$$\lim_{k\to\infty}\frac{1}{k}\ln\frac{P^i_{pdf}(x^T|\mathbf{z}^i_{1:k})}{P^i_{pdf}(x^{T^*}|\mathbf{z}^i_{1:k})} < 0, \text{ thus}$$
$$\lim_{k\to\infty}\frac{P^i_{pdf}(x^T|\mathbf{z}^i_{1:k})}{P^i_{pdf}(x^{T^*}|\mathbf{z}^i_{1:k})} = 0$$

2) When $p_j = p^*_j$, that is $x^T \in X^T_{eq,1}$,

$$\lim_{k\to\infty}\frac{1}{k}\ln\frac{P^i_{pdf}(x^T|\mathbf{z}^i_{1:k})}{P^i_{pdf}(x^{T^*}|\mathbf{z}^i_{1:k})} = 0, \text{ thus}$$
$$\lim_{k\to\infty}\frac{P^i_{pdf}(x^T|\mathbf{z}^i_{1:k})}{P^i_{pdf}(x^{T^*}|\mathbf{z}^i_{1:k})} = 1$$

■

B. Proof for moving robots

The difficulty of consistency proof for moving robots lies in their time-variant states. The main idea is to classify robot's states into two disjoint sets: the *infinite-observation set* that contains states with which a robot makes infinite observations, and the *finite-observation set* that contains states with which the robot makes finite observations, as $k\to\infty$. Before stating main theorem, the following lemma is introduced:

Lemma 1: For a set of robots with a collection of finite possible states Y^R, there exists at least one state for each robot that it visits for infinite times as k tends to infinity.

Proof: Let $n^{i,k}_j$ denote the times that i^{th} robot visits j^{th} state in Y^R up to time k. Then, $\sum_j n^{i,k}_j = k$. It is straightforward to see that $\exists n^{i,k}_j$, such that $n^{i,k}_j \to \infty$, as $k\to\infty$. ■

The consistency of LEO-DBF for moving robots is stated as follows:

Theorem 2: Under the condition of moving robots, each individual PDF by LEO-DBF converges to $X^T_{eq,1}$ when the number of observations tends to infinity, i.e.,

$$P^i_{pdf}(x^T \in X^T_{eq,i}|\mathbf{z}^i_{1:k}) \longrightarrow \begin{cases} 1 & \text{if } i=1 \\ 0 & \text{if } i\neq 1 \end{cases}, \ k\to\infty$$

where $\mathbf{z}^i_{1:k} = (z^1_{1:k_1},\dots,z^N_{1:k_N})$, $\{k_1,\dots,k_N\}$ are the timestamps of i^{th} robot's latest knowledge of all robots' observations.

Proof: Let $M \subseteq \{1,\dots,k_j\}\times Y^R$ denote the set of (time, state) pairs to indicate j^{th} robot's state at the corresponding time of observation.

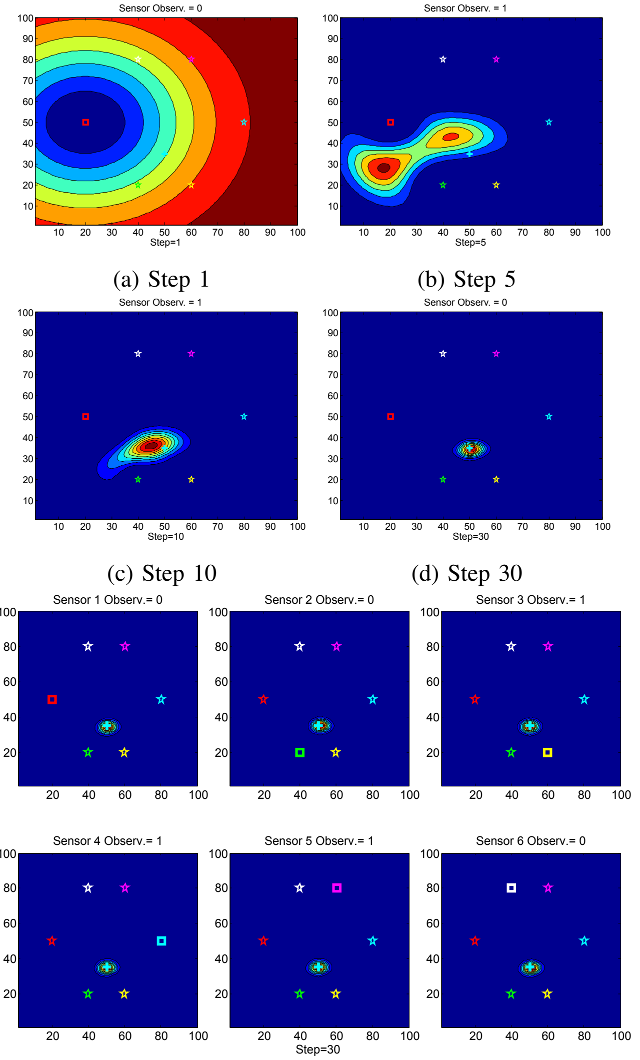
Similar to Theorem 1, the batch form of DBF at k^{th} step is:

$$\frac{P^i_{pdf}(x^T|\mathbf{z}^i_{1:k})}{P^i_{pdf}(x^{T*}|\mathbf{z}^i_{1:k})} = \frac{P^i_{pdf}(x^T)\prod_{j=1}^{N}\prod_{l\in M^i} P(z^j_l|x^T)}{P^i_{pdf}(x^{T*})\prod_{j=1}^{N}\prod_{l\in M^i} P(z^j_l|x^{T*})} \tag{11}$$

The only difference is that $P(z^j_l|X^T)$ needs to be grouped according to the robot state. For each robot, there exists at least one state from which the robot obtains infinite observations as $k\to\infty$, according to Lemma 1. All states in Y^R can be classified into finite-observation states and infinite-observation states. For the former, it is straightforward to know that their contribution to Eqn. (10) is zero. Therefore, the proof of Eqn. (11) can be reduced by only considering infinite-observation states, which is similar to Theorem 1. ■

Remark 6: Since $X^T_{eq,1}$ is unique, the consistency of LEO-DBF guarantees the consensus of individual PDFs.

Remark 7: The consistency property differentiates LEO-DBF from the statistics dissemination-based filtering approaches. In fact, the statistics dissemination-based methods can ensure the convergence of the state estimate among robots. However, there's no guarantee whether the agreed estimate is close to the actual target position.

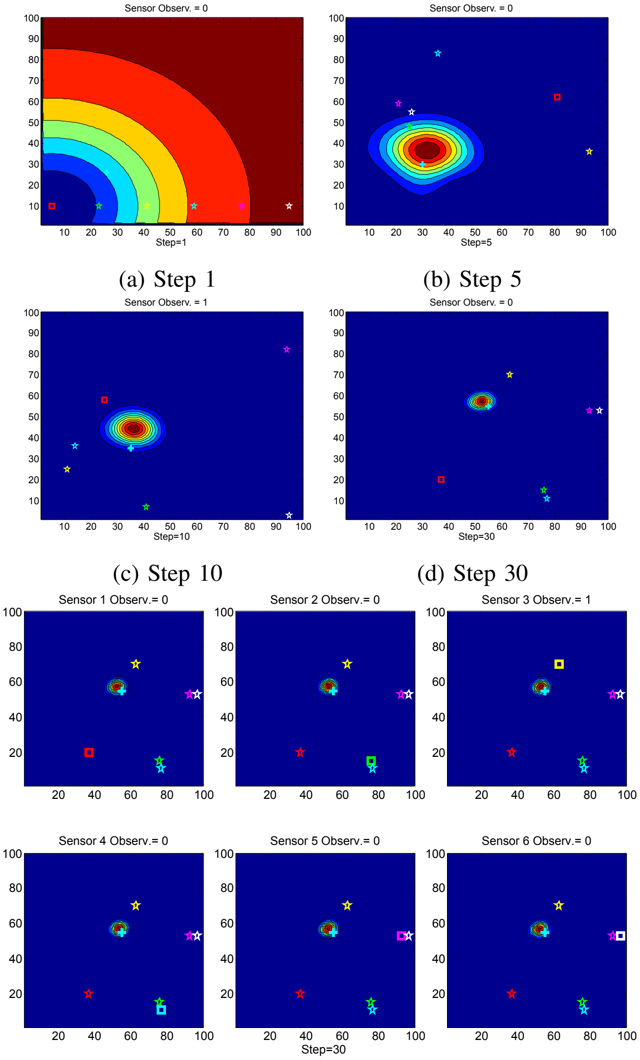


(a) Step 1 (b) Step 5

(c) Step 10 (d) Step 30

(e) Step 30

Fig. 4: (a)-(d): The 1st robot's individual PDFs at different time; (e) All robots' individual PDFs at time 30.



(a) Step 1 (b) Step 5

(c) Step 10 (d) Step 30

(e) Step 30

Fig. 5: (a)-(d): The 1st robot's individual PDFs at different time; (e) All robots' individual PDFs at time 30.

V. Simulation

This section simulates three searching scenarios in order to demonstrate the effectiveness of LEO-BDF. In all scenarios, six robots are utilized and each robot is equipped with a binary sensor. All sensors are modeled with identical Gaussian functions:

$$P(z=1|x^T;y^R) = \exp\left\{-\frac{1}{2}(x^T-x^R)^T\Sigma^{-1}(x^T-x^R)\right\} \tag{12a}$$

$$P(z=0|x^T;y^R) = 1 - P(z=1|x^T;x^R). \tag{12b}$$

where x^R denotes the robot position, which is included in robot state y^R. The first scenario consists of six static robots and single static target, which acts as a proof of concept of LEO-DBF for static target. The second scenario subsequently deals with six moving robots for searching the single static target. Robot positions are randomly generated at each time step. The third scenario is presented that contains six moving sensors and one moving target. A single-integrator dynamics is used for the target motion model.

A. Static Robots, Static Target

The positions of six static robots are shown as stars in Fig. 4e. Each robot constantly receives binary observations of the target. LEO-DBF (Eqn. (5)) is implemented on each robot for target position estimation.

Fig. 4 shows the estimation results of the static target. After the initial observation, each robot forms a circular individual PDF, centered at the corresponding robot position. Fig. 4a shows the individual PDF after initial observation for 1st robot. This happens because the Gaussian sensor model (Eqn. (12)) only depends on the distance between robot and target. As more observations are received, the posterior individual PDF concentrates on the true location of the target (Fig. 4d), which is an expected result from the consistency of LEO-DBF. Fig. 6 (a) shows the decrease of the entropy

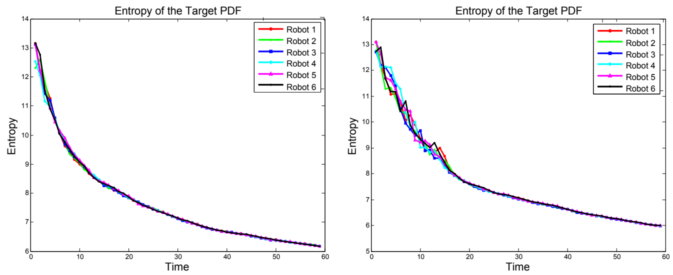


Fig. 6: Entropy of individual PDFs over time: (a) static robots and static target; (b) moving robots and moving target.

of each robot's individual PDF, indicating the reduction of the uncertainty in estimated target position.

B. Moving Robots, Moving Target

The six robots move within the field to estimate the target position. The motion planning of robots for effective target search has received much attention in the past years. Readers interested in this topic can refer to [21], [25]. In this work, the robot positions are randomly generated at each time in order to demonstrate the effectiveness of LEO-DBF approach. The target moves on the horizontal plane and the dynamics is given by a single-integrator model:

$$x^T(k+1) = x^T(k) + v$$

where v is the constant velocity of the target.

The LEO-DBF described in Section III-C is utilized for target SAT. Fig. 5 shows the estimation results of the moving target. It is interesting to notice that the posterior individual PDFs concentrate to the true target location at each time, even when the target constantly moves. Fig. 6 (b) shows the decrease of the entropy of the posterior distribution.

VI. CONCLUSION

In this study, we proposed the local-exchange-of-observation (LEO) strategy for distributed Bayesian filters (LEO-DBF) in a multi-robot network with the application of distributed search and tracking (SAT) of target. With fixed communication topology, LEO guarantees the global dissemination of all robots' observations over the network only via local exchange of observations among neighbors. Once elements in communication buffer (CB) gets filled, each robot can receive and update its CB non-intermittently under LEO. Two LEO-DBFs are proposed for SAT of a static and a moving target, respectively. For the static target, each robot locally fuses the latest knowledge of all robots' observations by only considering the updating step of the Bayesian filter. For the moving target, each robot maintains a triangle matrix of historical observations and an iterative Bayesian filtering procedure is applied that alternates between prediction and updating steps upon obtaining the latest available observations of all robots. The consistency of LEO-DBF is proved by showing the asymptotic concentration of posterior individual PDF to the equi-parameter set containing the actual state, ensuring the agreement between robots' state estimate using LEO-DBF and the actual environment state. Simulations demonstrate the effectiveness of LEO-DBF for SAT of both static and moving targets.

Future work includes several extensions to the proposed LEO-DBF. First, LEO under switching topology can lead to unpredictable delay and intermittent transmission. Therefore consistency of individual PDFs requires further analysis. In addition, combining LEO-DBF with robot motion planning is promising for more effective SAT of target.

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] 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*. IEEE, 2012, pp. 3857–3860.

[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*. IEEE, 2005, pp. 8179–8184.

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

[6] 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*. IEEE, 2006, pp. 1–4.

[7] 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.

[8] 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*. IEEE Press, 2005, p. 24.

[9] 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.

[10] 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.

[11] 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.

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

[13] 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.

[14] 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.

[15] 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*. IEEE, 2011, pp. 5928–5931.

[16] M. Rosencrantz, G. Gordon, and S. Thrun, "Decentralized sensor fusion with distributed particle filters," in *Proceedings of the Nineteenth conference on Uncertainty in Artificial Intelligence*. Morgan Kaufmann Publishers Inc., 2002, pp. 493–500.

[17] D. Bonnie, S. Candido, T. Bretl, and S. Hutchinson, "Modelling search with a binary sensor utilizing self-conjugacy of the exponential family," in *Robotics and Automation (ICRA), 2012 IEEE International Conference on*. IEEE, 2012, pp. 3975–3982.

[18] C. Liu, S.-Y. Liu, E. L. Carano, and J. K. Hedrick, "A framework for autonomous vehicles with goal inference and task allocation capabilities to support peer collaboration with human agents," in *ASME 2014 Dynamic Systems and Control Conference*. American Society of Mechanical Engineers, 2014, pp. V002T30A005–V002T30A005.

[19] P. M. Djurić, M. Vemula, and M. F. Bugallo, "Target tracking by particle filtering in binary sensor networks," *Signal Processing, IEEE Transactions on*, vol. 56, no. 6, pp. 2229–2238, 2008.

[20] D. Hahnel, W. Burgard, D. Fox, and S. Thrun, "An efficient fastslam algorithm for generating maps of large-scale cyclic environments from raw laser range measurements," in *Intelligent Robots and Systems, 2003.(IROS 2003). Proceedings. 2003 IEEE/RSJ International Conference on*, vol. 1. IEEE, 2003, pp. 206–211.

[21] J. Tisdale, Z. Kim, and J. K. Hedrick, "Autonomous uav path planning and estimation," *Robotics & Automation Magazine, IEEE*, vol. 16, no. 2, pp. 35–42, 2009.

[22] S. S. Ram, A. Nedic, and V. Veeravalli, "Stochastic incremental gradient descent for estimation in sensor networks," in *Signals, Systems and Computers, 2007. ACSSC 2007. Conference Record of the Forty-First Asilomar Conference on*. IEEE, 2007, pp. 582–586.

[23] I. Bahceci, G. Al-Regib, and Y. Altunbasak, "Serial distributed detection for wireless sensor networks," in *Information Theory, 2005. ISIT 2005. Proceedings. International Symposium on*. IEEE, 2005, pp. 830–834.

[24] O. Hlinka, F. Hlawatsch, and P. Djuric, "Distributed sequential estimation in asynchronous wireless sensor networks," 2015.

[25] T. Furukawa, F. Bourgault, B. Lavis, and H. F. Durrant-Whyte, "Recursive bayesian search-and-tracking using coordinated uavs for lost targets," in *Robotics and Automation, 2006. ICRA 2006. Proceedings 2006 IEEE International Conference on*. IEEE, 2006, pp. 2521–2526.

[26] 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*. IEEE, 2014, pp. 5830–5835.