Distributed Bayesian Estimation by Local Exchange of Observations

# Introduction

Several points:

Distributed estimation used for target-search

Statistics vs measurement-based

Mention consensus in statistics

Distributed estimation focuses on utilizing a group of networked agents to collectively infer the state of an environment, which have been adopted for various applications, such as object detection, target tracking and environmental monitoring etc [1]– [5]. The communication topology plays a vital role in distributed estimation algorithms. In, measurements of all agents are transmitted to a fusion center, where the environment state is estimated using all measurements in a batch manner. In, a distributed estimation is conducted, where no fusion center exists. Agents are fully connected in that they can communicate their measurement to all other agents directly in the network in a single transmission. Each agent estimates the environment state using all the observations. These communication topologies are beneficial in that all the observations are employed for state estimation in a single step. However, such full connection of the communication network is usually as ideal assumption.

More realistically, each agent can communicate only with neighboring agents and locally estimates environment state. In xxx.

Based on the type of the communicated data between agents, distributed estimation algorithms can be divided into two categories: statistics dissemination-based algorithms, and measurement dissemination-based DPFs. In statistics dissemination-based algorithms, processed data, such as posteriors or likelihood functions, are exchanged between the agents [34]–[55], [57], [60]–[66]. In, KF … . In, PF … . In, Bayesian filters … . Olfati-Saber et al. (2005) developed a consensus algorithm for Bayesian distributed hypothesis testing in a static sensor network. In our work, we use a consensus-based algorithm inspired by Olfati-Saber et al. (2005) to compute the joint measurement probabilities needed for the mutual information-based controller and the Bayesian filter calculations. (write about consensus algorithm)

An important class of algorithms for distributed estimation is based on the KF and its variations [23]–[29]. In the linear/Gaussian case, the “information form” of the KF is particularly amenable to a distributed implementation [23]. In [24] and

[25], distributed information filters based on consensus algorithms are presented. A diffusion-based distributed KF is proposed in [26], and a distributed KF using gossip algorithms is developed in [27]. For nonlinear/non-Gaussian models, distributed

versions of the extended KF or unscented KF are proposed in [23], [28], and [29]. A second large class of distributed algorithms for sequential Bayesian estimation is given by DPFs [30]–[66], which are addressed in the remainder of this article. (used for examples of KF)

In measurement dissemination-based DPFs, raw or quantized measurements are exchanged among agents [56]–[59]. Explain these references.

For certain applications, transmission limit poses a problem, for example, … . The measurement poses much less transmission burden than processed data (posterior or likelihood functions) in target-search scenarios. However, most of the works in distributed target search transmits processed data while few has considered the transmission of measurements. In this work, we propose a local exchange of observation (LEO) strategy for distributed Bayesian estimation. Each agent can make observations of the environment and communication the observations (both current and previous received observations) to neighboring agents. Local Bayesian estimation is conducted by each agent. The consistency of the algorithm is proved. The complexity analysis shows the great reduction in the transmitted data for target-search scenario using the proposed method than statistics dissemination-based methods.

# Problem definition

Each robot can only communicate with its neighboring agents. The set of neighbors of the ith robot is denoted as and the number of neighbors in is . The exchanged information is limited to the observation of each robot. Each robot has its individual estimation of the target PDF. Considering the limit of the communication range and bandwidth, no PDF is allowed to be transmitted. The individual PDF of robot i is initialized by the prior function at time k=0, given all available prior information including past experience and domain knowledge. Once determining the prior distribution, the ith individual PDF at time k, , can be estimated recursively by distributed Bayesian filter based on measurements from the neighborhood of robot i. The upper-script T represents the target, whose position is unknown for robots.



## Model of Binary Sensor

The sensor model is subject to a Gaussian distribution.



## Bayesian Filtering from Multi-senor

### Prediction

Suppose the system is at time step k-1 and the latest update for ith individual PDF is

. The prior PDF is predicted forward to time step k by using the Chapman-Kolmogorov equation:



where is a probabilistic Markov motion model of target, independent of robot states. This model describes the state transition probability of the target from the prior stateto the destination state . For a static target,



and the above equation can be reduced to .



### Updating

At time step k, the neighbors of the ith robot, denoted as , the observation of robot i is and its corresponding observation probability for given target state , is denoted as . This is referred to as the observation likelihood for a fixed . It is assumed that all observations are conditionally independent given the current state. Then the target PDF is updated by using the Bayes rule:



where is a normalization factor, given by:



.



If there is no prediction, it can be reduced to be



# Distributed Bayesian Filter via Observation Exchange

This section assumes to detect a static target.

## Algorithm for Local Exchange of observations (LEO)

We propose an observation local exchange (OLE) strategy for the network of robots. The observation of i-th robot at k-th step is denoted as . Besides its own observation, robot contains a buffer to store its latest knowledge of the observations of all agents:



where denotes that for and at k-th step, the latest information on j-th robot received by -th robot is the -th step observation of j-th robot (note that <=k).



The following broadcasting algorithm is used:

**(1) Initialization:** The storage buffer of i-th robot is initialized when k=0:



**(2) At k-th step and for i-th robot**

**(2.1) Receiving Step:** The i-th robot only receives the sending buffers from its neighboring nodes, i.e, . The received buffers are totally groups, each of which is actually the (k-1)-step buffer of a node in . For narrative simplicity, the *l*-th () received buffers is noted as



**(2.2) Observation Step:** The i-th robot updates , in its buffer according to its current-step observation



**(2.3) Comparison Step:**

**Besides** , **the other** values in i-th robot buffer, i.e, , is updated by using the latest information among all received buffers from .



For any ,



**End**

(2.4) Sending Step: Finally, send the updated buffer of i-th robot to all its neighbors in .



(3) Repeat step (2) until stop.

It can be proved that in a network of N robots, each robot will obtain the history observations of all other robots within a finite number of communication rounds, as stated in the following propostion:

*Proposition 1:* For an undirected and connected network of N robots, any element in buffer becomes nonempty within at most N-1 step, i.e., the information delay from node j to node i <=N-1; moreover, once becoming nonempty, the updating of each element in buffer  is non-intermittent, i.e.,  **is constant.**



*Proof:*

*Remark (1): Example*

Remark (2): why only l\_atest

Remark (3): Need complexity analysis.

## Algorithm for DBF

We are interested in distributed computation of the target PDF based on the measurement history. Once updating the history measurement, each robot locally runs the Bayesian filter for updating the target PDF. We first present the distributed Bayesian filter for a static target. Next a distributed Bayesian filter for a moving target will be proposed.

Since the target is static, the prediction step is unnecessary and we remove the subscript of . The update step becomes:



Note that only the observations with indices in are used for updating the target PDF at time k. Because the target is static, observations at different time equally contribute to the estimation process. The effectiveness of this strategy in estimating the target position is proved in the following proposition:



## Consistency proof of distributed Bayesian Filter

### Proof for static sensors

*Theorem 1* (consistency of DBF) Using multiple binary sensors to detect the single static target, the posterior probability given by the Bayesian estimator will concentrate on the true location of the target after infinitely many observations, i.e.



where  denotes the true location of the target. The proof of the proposition is presented in the Appendix.

*Proof*: The batch form of DBF at *K*-th step

|  |  |
| --- | --- |
|  | () |

where is initial guess of *i*-th local PDF. It is known from the proposition 1:



|  |  |
| --- | --- |
|  | () |

Since both target and sensors are static, (k=1:kj, j=1:N) are conditionally independent samples from sensor models (j=1:N) for given . Any binary observation subjects to Bernoulli distribution, yielding



|  |  |
| --- | --- |
|  | () |

where



Take the logarithm of (1) and average it over the K steps:

|  |  |
| --- | --- |
|  | () |

where



.



Utilizing the fact that (1) (k=1:kj, j=1:N) are conditionally independent, and (2) , recalling the law of large numbers yields



where . Then, the first term of Eq. (4) has the following limit



|  |  |
| --- | --- |
|  | () |

Note that the r.h.s of (5) achieves maximum iff .



Considering the equality



The third term of Eq. (4) is simplified to



Further, considering the equality



Considering Eq. (5), we have in the condition of when . Then,



Therefore, the limit of Eq. (4) becomes

|  |  |
| --- | --- |
|  | () |

It is known from Eq. (6):

(1) When , and ;



(2) When , and . (End of proof)



### Proof for moving sensors

Lemma: Consider N sensor in a finite set of sensor position,

*Theorem* *2.* Consider a finite set of target position. Using one binary sensor (sensors can move) to detect the single static target, the posterior probability given by the Bayesian estimator will concentrate on the true location of the target, i.e.



where denotes the true location of the target.



*Proof:*

The batch form of DBF at *K*-th step

|  |  |
| --- | --- |
|  | () |

where is initial guess of *i*-th local PDF. By converting



|  |  |
| --- | --- |
|  | () |

The only difference is that Eq. (2) does not hold, but for each sensor, at least there is one position has infinite observation. We can classify into finite-observation spot and infinite-observation spot. For the former, it is easy to know that



The corresponding item in has zero-limit. Therefore, the proof of Eq. (7) can be reduced to infinite-observation spot, which is similar to Theorem 1. (End of Proof)



Remark: It is interesting to find that even in Theorem 1, the consistency of DBF does not require that all sensors have infinite observation. The theorem 2 implies that if only one sensor has infinite observation, the consistency of DBF is achievable.

# Extend DBF for Moving Target

This section derives the DBF for a moving target. For the purpose of simplicity, we consider the update of the target PDF of the 1st robot with new measurement .



Following the Bayesian estimation framework:



Different from the DBF for the static target that utilizes the target PDF from previous time for updating, DBF for the moving target requires the ‘time-aligned’ target PDF and all available measurement after time k-2. Define the set , called the *local measurement history,* as the set that contains the previous measurement (not belong to ) necessary for updating the target PDF. In this three-robot example, . The robot needs to update and over time and implement the formula. Algorithm 1 gives the general formula of DBF for a moving target. Without loss of generality, assume and let .



For the ith robot

* Initialize



* At the time k,
  + Update ‘time-aligned’ target PDF from



* + Update the target PDF



For the network with N robots, the space complexity is.

# Simulation

## Setup

## Results and Discussion

# Appendix