Multi-Rate Asynchronous Distributed Filtering Under Randomized Gossip Strategy

Teng Shao, Zhangsheng Duan
Center for Information Engineering Science Research
Xi'an Jiaotong University
Xi'an, Shaanxi 710049, China
Email: tshaoking@stu.xjtu.edu.cn, zsduan@mail.xjtu.edu.cn

Uwe D. Hanebeck
Intelligent Sensor-Actuator-Systems Laboratory (ISAS)
Institute for Anthropomatics and Robotics
Karlsruhe Institute of Technology (KIT), Germany
Email: uwe.hanebeck@kit.edu

***Abstract*—In this paper, we consider the extension of synchronous distributed filtering under randomized gossip strategy to multi-rate asynchronous sensor network. To deal with asynchronous measurements due to multi-rate sampling, two processing schemes are proposed, i.e., a batch one and a sequential one. In batch processing, the common fusion period is chosen as the least common multiple of the sampling periods of all sensors. Each measurement of a sensor is propagated to the nearest common fusion time and saved locally. At the common fusion time, multiple propagated measurements of a sensor are compressed to a single measurement and then exchanged with its neighbors. Whereas in sequential processing, the common fusion period is chosen as the greatest common divisor of the sampling periods of all sensors to sequentially process each measurement once it is received. Depending on whether a real measurement is available locally, three exchanging strategies are developed. It is found that unlike in the traditional distributed fusion with a common fusion center, the batch and sequential processing schemes are not equivalent. Pros and cons of these two schemes are analyzed. Numerical experimental results further verify the effectiveness of the proposed schemes.**

***Index Terms*—Multi-rate sampling, asynchronous fusion, distributed filtering, sensor network, batch processing, sequential processing, randomized gossip**

I. INTRODUCTION

In the past decades, distributed filtering over sensor network has received much attention [1], [2] due to its wide application in indoor localization, vehicle tracking, orbital determination, robot navigation, etc. Depending on whether a fusion center is available or not, distributed filtering can be categorized into two types, i.e., distributed filtering with a fusion center and without a fusion center. For distributed filtering with a fusion center, the communication part is simple, whereas the fusion functionality will malfunction if the fusion center fails. For distributed filtering without a fusion center, however, even if some fusion nodes fail, the fusion functionality can still work. The difficulty of distributed filtering without a fusion center lies in how and what to communicate between sensor nodes.

In the past few years, distributed filtering without a fusion center has been extensively studied under the consensus strategy and the diffusion strategy. In both strategies, each node only communicates with its neighbors. However, the diffusion strategy focuses on estimation accuracy, whereas the consensus strategy focuses on estimation consensus, i.e., all sensor estimates are expected to achieve an agreement. Under diffusion strategy, [3] proposed a distributed Kalman filter and smoother. A diffusion Kalman filter based on covariance intersection method was developed in [4]. Different from the fusion of local Kalman filters by a convex combination regardless of the error covariance information in [3], [4] incorporates the estimation error covariance information as an important role to improve the estimation performance. In [5], a distributed Kalman filter was developed with adaptive weights by optimizing a locally defined cost function. The diffusion strategy was proven to be more efficient than the consensus strategy for static estimation in [6]. A diffusion strategy based distributed fusion algorithm was proposed in [7] without exchanging raw measurements, which is more suitable for the case with intermittent observations. Under consensus strategy, [8] and [9] proposed a class of distributed filtering by imposing a consensus term on each local state estimate to achieve consensus. [10] developed the globally optimal Kalman consensus filter. In [11], a hybrid consensus based linear and nonlinear filtering was proposed which is a combination of two existing consensus methods, i.e., consensus on measurement and consensus on information. The randomized gossip strategy, which is the fastest consensus strategy, was studied in [12] and [13]. At each communication round, only a pair of nodes are allowed to communicate with each other according to the asynchronous time model. [14] proposed a synchronous distributed filtering under randomized gossip strategy.

For simplicity, many well-known existing estimation fusion algorithms assume that all sensors observe synchronously, i.e., all sensors have the same sampling rates. However, different sensors may have different sampling rates in practice. To deal with asynchronous measurements due to multi-rate sampling, asynchronous distributed filtering with a fusion center have been extensively studied. These asynchronous distributed filtering approaches can be categorized into two classes, i.e., batch processing and sequential processing. In the batch processing class, asynchronous fusion based on equivalent pseudo measurements was considered in [15] and optimal

This work was supported in part by the National Natural Science Foundation of China (NSFC) through Grants 61673317, 61773313, by the Fundamental Research Funds for the Central Universities of China.

asynchronous estimation fusion methods was proposed in [16]. The optimal asynchronous multirate with unreliable measurements estimation method was studied in [17]. The distributed estimation fusion problem for a class of multi-sensor asynchronous sampling systems with correlated noises was discussed in [18], where the state is uniformly updated and the sensors randomly sample measurements. In sequential processing class, a sequential asynchronous estimation fusion algorithm based on the predicted estimates was proposed in [19]. A sequential estimation fusion algorithm was presented in [20] for asynchronous measurement with communication uncertainties. However, asynchronous distributed filtering without a fusion center was rarely studied in the existing work. In this paper, we consider the distributed filtering under randomized gossip strategy for multi-rate asynchronous sensor network.

The rest of this paper is organized as follows. Section II formulates the problem. We briefly summary the randomized gossip strategy in Section III. Two asynchronous distributed filtering schemes are proposed in Section IV. Section V verifies performance of the proposed schemes through numerical examples and the conclusions are drawn in Section VI.

II. Problem Formulation

Consider the following typical discrete-time linear dynamic system observed by N sensors with different sampling rates

$$x_k = F_{k-1}x_{k-1} + w_{k-1} \ , \tag{1}$$

$$z^i_{k_i} = H^i_{k_i}x^i_{k_i} + v^i_{k_i}, \quad i = 1, 2, \ldots, N \ , \tag{2}$$

where $x_k \in \Re^n$ is the system state at discrete time k and its sampling period is assumed to be T; the superscript i is the sensor node index and N is the number of sensor nodes in a given network; $F_{k-1} \in \Re^{n\times n}$ is the state transition matrix and $w_k \in \Re^n$ is the driven process noise, assumed to be zero-mean white with covariance Q_k; $z^i_{k_i} \in \Re^{m_i}$ is the k-th measurement of sensor i with sampling period T_i; it is assumed that $T_i = n_iT$ and n_i is a positive integer; $x^i_{k_i} \in \Re^n$ and $H^i_{k_i} \in \Re^{m_i\times n}$ are the corresponding system state and measurement matrix; $v^i_{k_i} \in \Re^{m_i}$ is the driven measurement noise, assumed to be zero-mean white with covariance $R^i_{k_i}$; the initial state x_0, with mean $\bar{x}_0$ and covariance P_0; the two noises and x_0 are assumed to be mutually independent.

The observing process of four asynchronous sensors with multi-rate sampling is illustrated in Fig. 1.

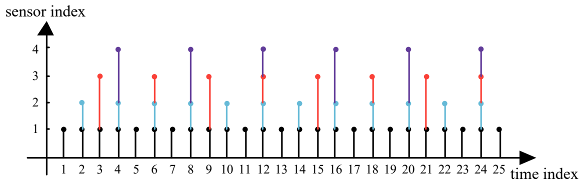


Fig. 1. An illustration of asynchronous measurements of four sensors.

Suppose that the sampling periods of all sensors are $T_1 = T$, $T_2 = 2T$, $T_3 = 3T$, and $T_4 = 4T$, respectively. Then we can see from Fig. 1 that at time 8, sensors 1, 2, and 4 have measurements, whereas sensor 3 has no measurement. And at time 9, sensors 1 and 3 have measurements, whereas sensors 2 and 4 have no measurements.

The system formulation of (1) and (2) can be equivalently rewritten as

$$x_k = F_{k-1}x_{k-1} + w_{k-1} \tag{3}$$

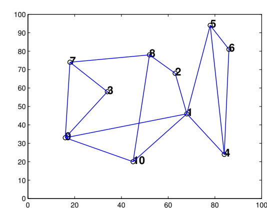
$$z^i_k = \begin{cases} H^i_k x_k + v^i_k, & \text{if } \operatorname{mod}(k, n_i) = 0 \\ \text{null}, & \text{otherwise} \end{cases} \tag{4}$$

where z^i_k, H^i_k, v^i_k represent the measurement, measurement matrix, measurement noise of sensor i at continuous-time kT and

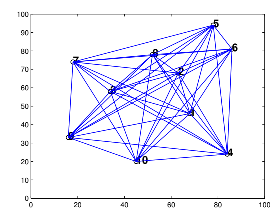
$$z^i_k = z^i_{(\frac{k}{n_i})_i}, H^i_k = H^i_{(\frac{k}{n_i})_i}, x_k = x^i_{(\frac{k}{n_i})_i}, v^i_k = v^i_{(\frac{k}{n_i})_i}$$

if $\operatorname{mod}(k, n_i) = 0$.

The topology of the sensor network of the above N sensor nodes is represented by an undirected graph $\mathbb{G} = (\mathbb{V}, \mathbb{E})$, e.g., Fig. 2, where $\mathbb{V} = \{1, 2, \cdots, N\}$ is the set of nodes and $\mathbb{E} \subset \mathbb{V} \times \mathbb{V}$ is the set of edges. The existence of an edge $(i, j) \in \mathbb{E}$ means that nodes i and j can communicate with each other. The set of neighbors of sensor node i is denoted by $\mathbb{N}_i = \{j \in \mathbb{V} | (i, j) \in \mathbb{E}\}$.



(a) Random mesh topology.



(b) Full mesh topology.

Fig. 2. The topology of a sensor network with 10 nodes.

In this paper, we consider coordinated/cooperative estimation of the above asynchronous sensor network, i.e., achieving consensus in the whole network. Randomized gossip is the fastest strategy to achieve consensus and has been successfully applied to distributed filtering in synchronous sensor networks [13]. We will briefly summarize this in the next section. However, different sensors may have different sampling rates in practice. Therefore the main goal of this paper is to extend randomized gossip strategy also to distributed filtering in multi-rate asynchronous sensor networks.

III. Randomized Gossip

Consider a synchronous sensor network with N sensor nodes. Let $d(0) = [d^1, \cdots, d^N]'$, where $d^i \in \Re$, $i = 1, 2, \cdots, N$, denote the measurements of all sensors. The purpose of average consensus is to reach the consensus $\bar{d} = \frac{1}{N}\sum_{i=1}^N d^i$ at each sensor node after certain number of communication rounds. Randomized gossip provides the fastest strategy to achieve this. Next we will give a brief summary to randomized gossip strategy first.

A. Asynchronous Randomized Gossip

Two types of time models for randomized gossip were given in [13], i.e., the synchronous and the asynchronous[1] time model. In the synchronous time model, multiple node pairs communicate with each other at the same time independently. Whereas in the asynchronous time model, at each communication round, only one pair of nodes are allowed to communicate with each other. Both of these time models can be used to achieve average consensus. However, in terms of simplicity for implementation, the asynchronous time model is preferred. That is also why only asynchronous time model will be considered in this paper.

The averaging consensus problem under the asynchronous gossip constraint was analyzed in [12] and [13]. An asynchronous gossip constraint on the communication protocol means that, at each communication round, each node can only communicate with one of its neighbors, i.e., the node communicates with no more than one of its neighbor in every time slot. Denote by l the communication round index. Suppose that at the l-th communication round, a pair of nodes i and j are selected to communicate with each other. The data exchange procedure between them can then be described as

$$d(l) = W(l)d(l-1), \quad l = 1, 2, \ldots \tag{5}$$

where

$$W_{ij} = \begin{cases} I - (e_i - e_j)(e_i - e_j)'/2, & \text{if } (i,j) \in \mathbb{E} \\ 0, & \text{if } (i,j) \notin \mathbb{E} \end{cases} \tag{6}$$

$e_i = [0, 0, \cdots, 0, 1, 0, \cdots, 0]'$ is an $N \times 1$ unit vector with the i-th component equal to 1.

B. Communication Pair Selection

At the l-th communication round of asynchronous randomized gossip, a node i, $i \in \mathbb{V}$, is chosen among all N equally possible nodes first. Then one of its neighboring nodes is chosen with the probability for each of its neighbor to be selected is P_{ij}, where $P_{ij} \geq 0$, $\sum_j P_{ij} = 1$ and $P_{ij} = 0$ if $(i,j) \notin \mathbb{E}$. After that, the two selected nodes exchange their data and use the average of their data as their new data, respectively. In this communication pair selection procedure, the most critical problem is how to determine the probability P_{ij}.

Define $\overline{W}$ as

$$\overline{W} = \sum_{(i,j)\in E} P_{ij} W_{ij} \ . \tag{7}$$

This $\overline{W}$ is a doubly stochastic matrix and satisfies

$$\mathit{1}'\overline{W} = \mathit{1}', \quad \overline{W}\mathit{1} = \mathit{1}, \quad \rho(\overline{W} - \frac{\mathit{11}'}{N}) < 1 \ ,$$

where $\rho(\cdot)$ is the spectral radius of a matrix.

[1]Note that "asynchronous" in gossip is different from "asynchronous" in sensor networks, although theirs names are the same. In gossip, "asynchronous" means that at each communication round, only one pair of nodes are allowed to communicate with each other. However, in sensor networks, "asynchronous" means that the sampling rates of sensors are different.

In [12], it was found that the converging speed of consensus depends on the second largest eigenvalue of $\overline{W}$: the smaller the the second largest eigenvalue of $\overline{W}$, the faster the converging speed of consensus. In view of this, the optimal P_{ij} can then be determined by the following semi-definite programming problem

$$\min \quad s \tag{8}$$

$$\text{subject to} \quad \begin{cases} \overline{W} - \frac{\mathit{11}'}{N} \preceq sI \\ \overline{W} = \sum\limits_{i,j=1}^{N} \frac{P_{ij} W_{ij}}{N} \\ P_{ij} \geq 0, \sum\limits_{j} P_{ij} = 1 \\ P_{ij} = 0 \text{ if } (i,j) \notin \mathbb{E} \end{cases} \tag{9}$$

where the matrix inequality $A \succeq B$ means that the difference $A - B$ is a positive semi-definite matrix and $\mathit{1}$ denotes the vector of all ones.

Remark 1. Note that the same optimal P_{ij} also works for vector measurements, i.e., when $d^i \in \Re^m, m > 1, \quad i = 1, 2, \cdots, N$. The only difference is that at the l-th communication round, (5) should be applied to the local data component by component.

IV. Asynchronous Distributed Filtering

In [14], distributed filtering under the randomized gossip strategy for synchronous sampling was studied. In practical networks, however, different sensors may have different sampling rates. For such cases, synchronous distributed filtering under randomized gossip strategy can not be applied directly due to missing measurements of some sensors at some times. In this section, to deal with asynchronous measurements due to multi-rate sampling, two processing schemes, i.e., a batch one and a sequential one, will be discussed.

A. Batch Multi-Rate Asynchronous Distributed Filtering

In the batch scheme, the common fusion period T_f^b is chosen as the least common multiple of the sampling periods of all sensors, i.e., $T_f^b = \text{lcm}(T_1, \cdots, T_N)$, where the function $\text{lcm}(\cdot)$ finds the least common multiple of a set of numbers.

The basic idea of batch processing scheme is that each measurement of a sensor is propagated to the nearest common fusion time once it is received. At the common fusion time, multiple propagated measurements of a sensor are transformed to a single measurement and then exchanged with its neighbors. For illustrative purposes, a batch processing scheme with two sensors is given in this subsection. Suppose that the sampling periods of sensors i and j are $T_i = T$ and $T_j = 2T$, respectively, see, e.g., Fig. 3. Then the common fusion period will be $T_f^b = \text{lcm}(T, 2T) = 2T$. Next, we will illustrate how the batch processing scheme can be applied to these two sensors for the time interval $((k-2)T, kT]$.

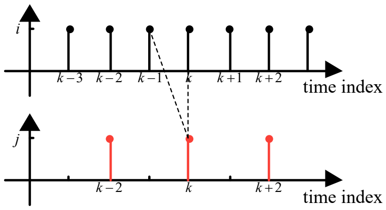


Fig. 3. An illustration of batch scheme

1) Propagating to the Nearest Common Fusion Time: For sensor i, its measurement at time $k-1$ needs to be propagated to the nearest common fusion time k, whereas sensor j does not need to propagate anything. The propagation for sensor i can be done as

$$z_{k-1}^{i} = H_{k-1}^{i} F_{k-1}^{-1} x_k - H_{k-1}^{i} F_{k-1}^{-1} w_{k-1} + v_{k-1}^{i} \ . \tag{10}$$

2) Transformation: At the common fusion time k, the measurements of sensor i can be written in an augmented form as

$$z_k^{i,a} = H_k^{i,a} x_k + v_k^{i,a}, \tag{11}$$

where

$$z_k^{i,a} = \begin{bmatrix} z_{k-1}^{i} \\ z_k^{i} \end{bmatrix}, \ H_k^{i,a} = \begin{bmatrix} H_{k-1}^{i} F_{k-1}^{-1} \\ H_k^{i} \end{bmatrix}, \ v_k^{i,a} = \begin{bmatrix} v_k^{i,1} \\ v_k^{i} \end{bmatrix}$$

$$v_k^{i,1} = v_{k-1}^{i} - H_{k-1}^{i} F_{k-1}^{-1} w_{k-1} \ ,$$

and the augmented measurement noise is still zero-mean white with covariance

$$\begin{aligned} R_k^{i,a} &= \mathrm{cov}(v_k^{i,a}) \\ &= \begin{bmatrix} R_{k-1} + H_{k-1}^{i} F_{k-1}^{-1} Q_{k-1} (H_{k-1}^{i} F_{k-1}^{-1})' & 0 \\ 0 & R_k \end{bmatrix} . \end{aligned}$$

However, for sensor j, there is only a single measurement at k, i.e.,

$$z_k^{j} = H_k^{j} x_k + v_k^{j} \ . \tag{12}$$

From (11) and (12), in general we can not exchange the measurement $z_k^{i,a}$ at sensor i and the measurement z_k^{j} at sensor j directly. This is because their dimensions may be different. Also the physical meaning of each component of them may be different. However, the randomized gossip strategy requires that the dimensions of all local data should be the same and the physical meaning of each component of them should also be the same. For this purpose, we apply the following linear transformation to the local data at each sensor

$$y_k^{i} = (H_k^{i,a})' (R_k^{i,a})^{-1} z_k^{i,a} \ , \tag{13}$$

$$y_k^{j} = (H_k^{j})' (R_k^{j})^{-1} z_k^{j} \ . \tag{14}$$

The state transition from $k-2$ to k can be written as

$$x_k = F_{k-1} F_{k-2} x_{k-2} + F_{k-1} w_{k-2} + w_{k-1} \ .$$

Then it can easily seen that the two-step transition process noise and the augmented measurement noise $v_k^{i,a}$ are correlated because

$$\mathrm{cov}(F_{k-1} w_{k-2} + w_{k-1}, v_k^{i,a}) = -Q_{k-1} (H_{k-1}^{i} F_{k-1}^{-1})'$$

This is against the linear Gaussian assumption where they are assumed uncorrelated with each other in the standard Kalman filter. As a result of this, performance of distributed estimation using the transformed measurement y_k^i without considering the correlation may be degraded to certain extent.

3) Exchanging: After applying the above linear transformations to all local data, we apply the asynchronous randomized gossip strategy to exchange data within the whole network. For example, at the l-th communication round, a pair of nodes i and j exchange their data as

$$y_{k,l}^{i} = (y_{k,l-1}^{i} + y_{k,l-1}^{j})/2 \ , \tag{15}$$

$$y_{k,l}^{j} = (y_{k,l-1}^{i} + y_{k,l-1}^{j})/2 \ . \tag{16}$$

This batch multi-rate asynchronous distributed filtering under randomized gossip strategy can be summarized as in Algorithm 1 (A1).

Algorithm 1: (A1) Batch multi-rate asynchronous distributed filtering under randomized gossip strategy

Input: sensor nodes N, common fusion period T_f^b, number of communication rounds L;

```
Initialization: x̂^i_{0|0} = x̄_0 and P^i_{0|0} = P_0,  i = 1,2,···,N;
for k = 1; k < ∞; k = ++ do
    Propagate measurements to the nearest common
    fusion time as done in (10);
    while mod(kT, T_f^b) == 0 do
        j = kT/T_f^b,  m = jT_f^b,  n = (j − 1)T_f^b;
        for i = 1; i ≤ N; i + + do
            x̂^i_{m|n} = F_{m,n} x̂^i_{n|n},  F_{m,n} = (F_n)^{T_f^b};
            P^i_{m|n} = F_{m,n} P^i_{n|n} F'_{m,n} + Q_n
        end
        Transform multiple propagated measurements to
        a single measurement as done in (11)-(14);
        for l = 1; l ≤ L; l + + do
            Choose a node r among all N equally
            possible nodes;
            Choose one of its neighboring nodes with the
            probability for each of its neighbor to be
            selected is P_rs;
            Exchange data between the two selected
            nodes as done in (15) and (16);
        end
        for i = 1; i ≤ N; i + + do
            (P^i_{m|m})^{-1} x̂^i_{m|m} = (P^i_{m|n})^{-1} x̂^i_{m|n} + N y^i_m;
            (P^i_{m|m})^{-1} =
            (P^i_{m|n})^{-1} + Σ_{i=1}^N (H^i_m)'(R^i_m)^{-1} H^i_m
        end
    end
end
```

Output: state estimate $\hat{x}_{m|m}^{i}$ and its error covariance $P_{m|m}^{i}$;

B. Sequential Multi-Rate Asynchronous Distributed Filtering

In the sequential scheme, the common fusion period T_f^s is chosen as the greatest common divisor of the sampling periods of all sensors, i.e., $T_f^s = \gcd(T_1, \cdots, T_N)$, where the function $\gcd(\cdot)$ finds the greatest common divisor of a set of numbers.

The core idea of sequential processing scheme is to sequentially process all measurements according to their sampled temporal order. For illustrative purpose, sequential processing scheme with two sensors is given in this subsection. Suppose that the sampling periods of sensors i and j are $T_i = 2T$ and $T_j = 3T$, respectively, see, e.g., Fig. 4. Then the common fusion period is $T_f^s = \gcd(2T, 3T) = T$. At the fusion time, for its l-th communication round, suppose that a pair of nodes i and j are selected to exchange their data. Depending on whether a real measurement is available locally, three data exchange cases should be considered.

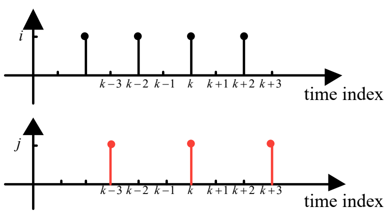


Fig. 4. An illustration of sequential scheme

Case 1: If both sensors have measurements, see, e.g., fusion time kT, in Fig. 4, then their measurements are transformed and exchanged as done in (14)-(16).

Case 2: If sensor i (or j) has measurement and sensor j (or i) has no measurement, see, e.g, fusion time $(k-3)T$, $(k-2)T$, $(k+2)T$ or $(k+3)T$, in Fig. 4, then the real data and predicted data are transformed and exchanged, where the predicted data is given by

$$\hat{z}_k^j = H_k^j F_{k-1} \hat{x}_{k-1|k-1}^j \quad (\text{or} \quad \hat{z}_k^i = H_k^i F_{k-1} \hat{x}_{k-1|k-1}^i)$$

Case 3: If both sensors have no measurements, see, e.g., fusion time $(k-1)T$ or $(k+1)T$, in Fig. 4, then the predicted data are transformed and exchanged, where the predicted data is given by

$$\hat{z}_k^i = H_k^i F_{k-1} \hat{x}_{k-1|k-1}^i$$

$$\hat{z}_k^j = H_k^j F_{k-1} \hat{x}_{k-1|k-1}^j$$

For *Case 2* and *Case 3*, the missing measurement is replaced by predicted data and the predicted data can be rewritten as

$$\hat{z}_k^i = H_k^i x_k + v_k^i - \tilde{z}_k^i = H_k^i x_k + \tilde{v}_k^i$$

$$\hat{z}_k^j = H_k^j x_k + v_k^j - \tilde{z}_k^j = H_k^j x_k + \tilde{v}_k^j$$

where the pseudo measurement noises $\tilde{v}_k^i$ and $\tilde{v}_k^j$ are zero-mean white with covariance

$$\tilde{R}_k^i = \text{cov}(\tilde{v}_k^i) = H_k^i P_{k|k-1}^i (H_k^i)'$$

$$\tilde{R}_k^j = \text{cov}(\tilde{v}_k^j) = H_k^j P_{k|k-1}^j (H_k^j)'$$

Then the transformation of the predicted data will be

$$y_k^i = (H_k^i)'(\tilde{R}_k^i)^{-1} \hat{z}_k^i \ , \tag{17}$$

$$y_k^j = (H_k^j)'(\tilde{R}_k^j)^{-1} \hat{z}_k^j \ . \tag{18}$$

Correspondingly, for *Case 2*, the real and predicted data are transformed and exchanged as done in (14)-(17); for *Case 3*, the predicted data are transformed and exchanged as done in (15)-(18).

This sequential multi-rate asynchronous distributed filtering under randomized gossip strategy can be summarized as in Algorithm 2 (A2).

Algorithm 2: (A2) Sequential multi-rate asynchronous distributed filtering under randomized gossip strategy

Input: sensor nodes N, common fusion period T_f^s, number of communication rounds L;

Initialization: $\hat{x}_{0|0}^i = \bar{x}_0$ and $P_{0|0}^i = P_0, \ \ i = 1, 2, \cdots, N$;
for $k = 1; k < \infty; k++$ **do**
  while $mod(kT, T_f^s) == 0$ **do**
    $j = kT/T_f^s, \ \ m = jT_f^s, \ \ n = (j-1)T_f^s$;
    for $i = 1; i \leq N; i++$ **do**
      $\hat{x}_{m|n}^i = F_{m,n} \hat{x}_{n|n}^i, \ \ F_{m,n} = (F_n)^{T_f^s}$;
      $P_{m|n}^i = F_{m,n} P_{n|n}^i F_{m,n}' + Q_n$
    end
    for $l = 1; l \leq L; l++$ **do**
      Choose a node r among all N equally possible nodes;
      Choose one of its neighboring nodes with the probability for each of its neighbor to be selected is P_{rs};
      Determine which case the data of sensor r and s belonging to and then transform and exchange their data;
    end
    for $i = 1; i \leq N; i++$ **do**
      $(P_{m|m}^i)^{-1} \hat{x}_{m|m}^i = (P_{m|n}^i)^{-1} \hat{x}_{m|n}^i + N y_m^i$;
      $(P_{m|m}^i)^{-1} = (P_{m|n}^i)^{-1} + \sum_{i=1}^N (H_m^i)'(R_m^i)^{-1} H_m^i$
    end
  end
end

Output: state estimate $\hat{x}_{m|m}^i$ and its error covariance $P_{m|m}^i$;

Remark 2. Due to the finer selection of the common fusion period, the sequential multi-rate asynchronous distributed filtering can help circumvent measurement synchronization in batch processing and has a better continuity for the fused track. However, the communication cost is higher. Whereas in batch multi-rate asynchronous distributed filtering, less communication cost is required. However, measurement synchronization is necessary and the continuity of the fused track will be poorer.

V. Illustrative Examples

In this section, we verify the effectiveness of the proposed algorithms. For this purpose, four examples are provided and all results are averaged over 500 Monte Carlo runs. The target motion in a two-dimensional plane is considered and the target state is denoted as $x_k = [x_k, \dot{x}_k, y_k, \dot{y}_k]'$, where x_k and y_k are target displacement from the origin along the horizontal and vertical directions and $\dot{x}_k$ and $\dot{y}_k$ are the corresponding velocities.

The involved parameters in (3) and (4) for target motion and sensor measurements are given as

$$F_{k-1} = \begin{bmatrix} 1 & T & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & T \\ 0 & 0 & 0 & 1 \end{bmatrix}, \quad Q_{k-1} = \mathrm{diag}([0.04, 0.09]) ,$$

$$\bar{x}_0 = \mathrm{diag}([4, 24, 9, 15]), \quad P_0 = \mathrm{diag}([25, 1, 64, 2]) ,$$

$$H_k^i = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 \end{bmatrix}, \quad R_k^i = \mathrm{diag}([4, 9]) ,$$

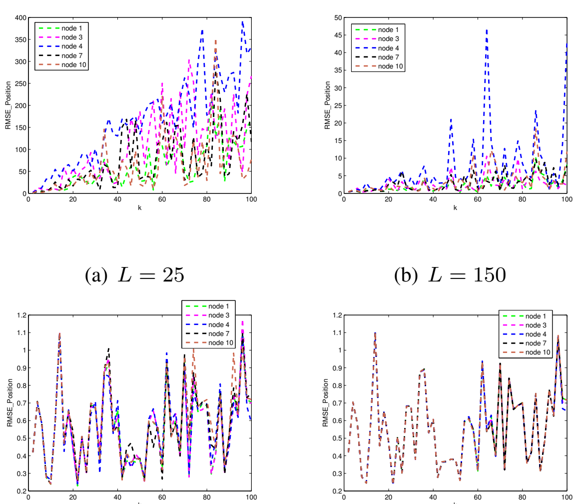
where $T = 1$ s.

A. Example 1: $N = 10$, A1, $T_i^{odd} = T, T_i^{even} = 2T$

In this example, we verify performance of the proposed batch multi-rate asynchronous distributed filtering. The two mesh topologies in Fig. 2 are considered and the sampling periods of all sensors are that

$$T_i = \begin{cases} T, & \text{if } \mathrm{mod}(i, 2) = 0 \\ 2T, & \text{if } \mathrm{mod}(i, 2) = 1 \end{cases}, \quad i = 1, \ldots, 10$$

The position root mean squared errors (RMSEs) are provided in Figs. 5 and 6.

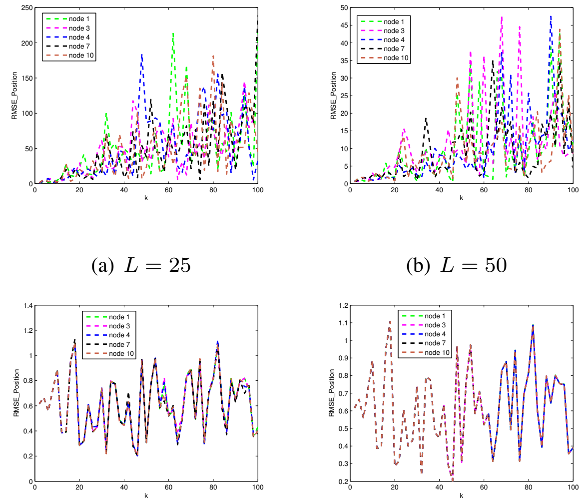


(a) $L = 25$ (b) $L = 150$

(c) $L = 300$ (d) $L = 400$

Fig. 5. Position RMSE using A1 for random mesh topology.

From Figs. 5 and 6, we can see that the consensus of the estimation is affected by the number of communication rounds.



(a) $L = 25$ (b) $L = 50$

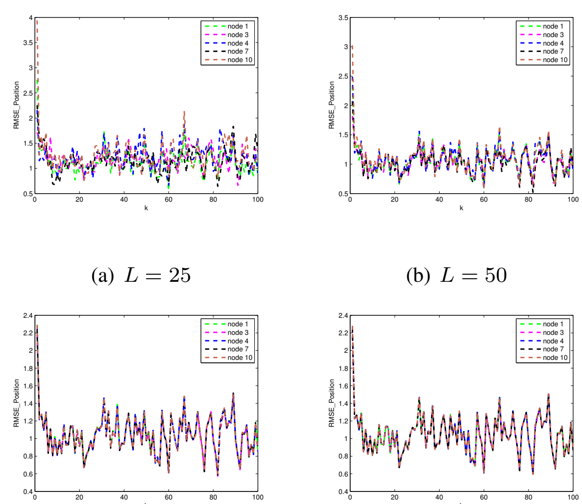
(c) $L = 150$ (d) $L = 200$

Fig. 6. Position RMSE using A1 for full mesh topology.

The more rounds of communication, the more consensus. It can be seen that full mesh topology needs less communication rounds than random mesh topology to achieve estimation consensus.

B. Example 2: $N = 10$, A2, $T_i^{odd} = T, T_i^{even} = 2T$

In this example, we verify performance of the proposed sequential multi-rate asynchronous distributed filtering. The position RMSEs are provided in Figs. 7 and 8.

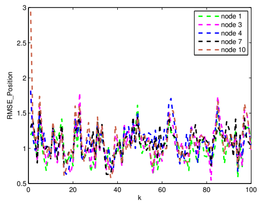


(a) $L = 25$ (b) $L = 50$

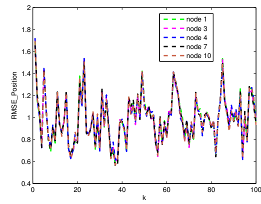
(c) $L = 100$ (d) $L = 200$

Fig. 7. Position RMSE using A2 for random mesh topology.

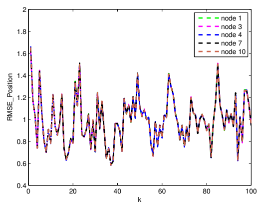
From Figs. 7 and 8, it can be seen that the random mesh topology needs more communication rounds than the full mesh topology to achieve estimation consensus.



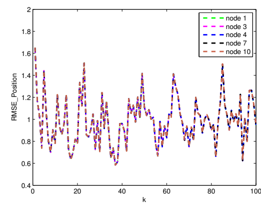
(a) $L = 25$



(b) $L = 50$

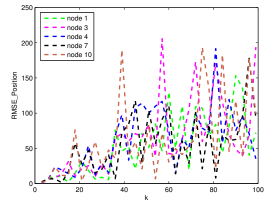


(c) $L = 100$

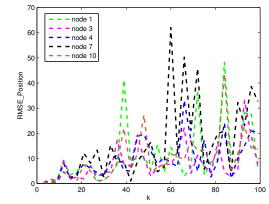


(d) $L = 150$

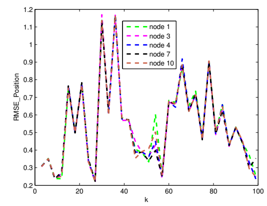
Fig. 8. Position RMSE using A2 for full mesh topology.



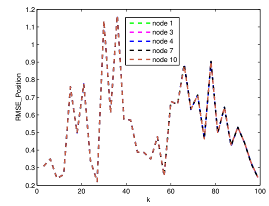
(a) $L = 25$



(b) $L = 50$



(c) $L = 150$

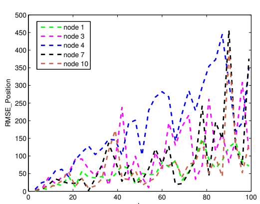


(d) $L = 200$

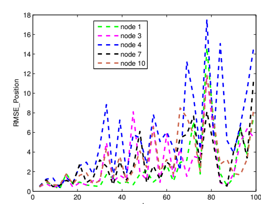
Fig. 10. Position RMSE using A1 for full mesh topology.

C. Example 3: $N = 10$, A1, $T_i^{odd} = T, T_i^{even} = 3T$

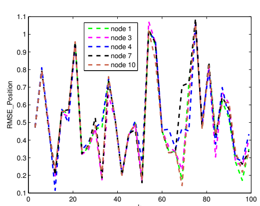
In this example, we further verify performance of the proposed batch multi-rate asynchronous distributed filtering. The difference form Example 1 is that the sampling periods of sensors with even indices are changed from $2T$ to $3T$. The position RMSEs are shown in Figs. 9 and 10.



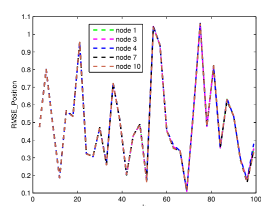
(a) $L = 25$



(b) $L = 150$



(c) $L = 300$



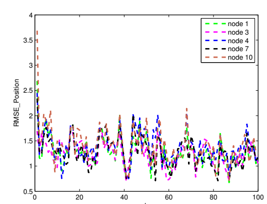
(d) $L = 400$

Fig. 9. Position RMSE using A1 for random mesh topology.

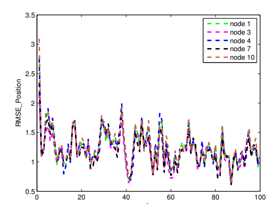
From Figs. 9 and 10, we can see that the consensus of the estimation is affected by the number of communication rounds. The more rounds of communication, the more consensus.

D. Example 4: $N = 10$, A2, $T_i^{odd} = T, T_i^{even} = 3T$

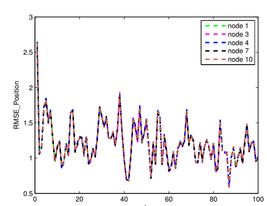
In this example, we further verify performance of the proposed sequential multi-rate asynchronous distributed filtering. The difference form Example 2 is that the sampling periods of sensors with even indices are changed from $2T$ to $3T$. The position RMSEs are shown in Figs. 11 and 12.



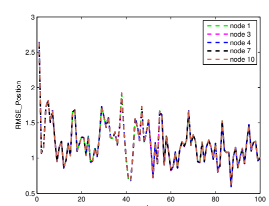
(a) $L = 25$



(b) $L = 50$



(c) $L = 100$



(d) $L = 200$

Fig. 11. Position RMSE using A2 for random mesh topology.

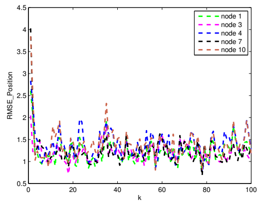
From Fig. 11 and 12, it can be seen that both random and full mesh topologies can achieve estimation consensus.

E. Communication Rounds Comparison

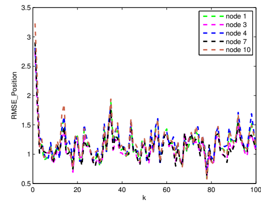
To measure the disagreement of the estimates, we use the following metric

$$\Psi_k = \left(\frac{1}{N} \sum_{i=1}^{N} \delta_{k,i}^2\right)^{1/2}, \tag{19}$$

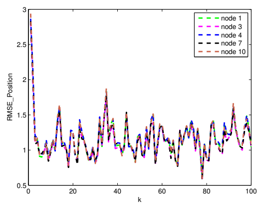
where $\delta_{k,i} = \hat{x}_{k|k}^i - \bar{x}_{k|k}$, $\bar{x}_{k|k} = \frac{1}{N} \sum_{i=1}^{N} \hat{x}_{k|k}^i$.



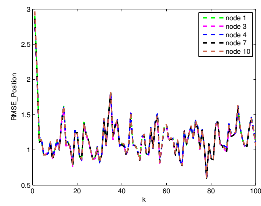
(a) $L = 25$



(b) $L = 50$



(c) $L = 100$



(d) $L = 150$

Fig. 12. Position RMSE using A2 for full mesh topology.

TABLE I lists the minimum communication rounds L needed such that Ψ_k satisfies $\Psi_k \leq 0.001$ $(k = 1, 2, \ldots, 100)$.

TABLE I
THE MINIMUM COMMUNICATION ROUNDS

		random mesh topology				full mesh topology			
	N	5	10	15	20	5	10	15	20
A1	$(T, 2T)$	86	308	475	727	67	145	221	352
A1	$(T, 3T)$	103	336	588	892	79	187	255	374
A2	$(T, 2T)$	47	164	257	376	48	89	127	194
A2	$(T, 3T)$	68	181	275	408	52	112	153	213

From TABLE I, we can see that, first, the larger the number of sensor nodes, the more communication cost are needed to achieve consensus. Second, the higher the degree of network connectivity, the less communication rounds are needed. Third, the sequential scheme has higher communication costs than the batch scheme to achieve the same consensus criterion.

VI. CONCLUSIONS

Multi-rate asynchronous distributed filtering under randomized gossip strategies has been studied in this paper. To deal with asynchronous measurements due to multi-rate sampling, two processing schemes are proposed, i.e., a batch one and a sequential one. Applying these two schemes is not like in traditional distributed fusion with a common fusion center. For the batch processing scheme, the key difference is that the propagated measurements should be transformed due to the possible dimensionality difference and physical meaning inconsistency. For the sequential processing scheme, depending on whether a real measurement is available locally, three data exchange cases are considered and a transformation method for missing measurement is provided. The random and full mesh topologies are used to verify the effectiveness of these two proposed schemes. Numerical experimental results show that the number of sensor nodes, sampling periods and network topology plays key roles in the determination of the minimum number of communication rounds to achieve consensus.

REFERENCES

[1] D. L. Hall, *Mathematical techniques in multisensor data fusion*, Norwood, MA: Artech House, 1992.

[2] Y. Bar-Shalom, and X. R. Li, *Multitarget-mutisensor tracking: Principles and techniques*, YBS Publishing, Storrs, CT, 1995.

[3] F. S. Cattivelli, and A. H. Sayed, “Diffusion strategies for distributed Kalman filtering and smoothing,” *IEEE Transactions Automation Control*, vol. 55, no. 9, pp. 2069-2084, 2010.

[4] J. W. Hu, L. H. Xie, and C. S. Zhang, “Diffusion Kalman filtering based on covariance intersection,” *IEEE Transactions on Signal Processing*, vol. 60, no. 2, pp. 891-902, 2012.

[5] F. S. Cattivelli, and A. H. Sayed, “Diffusion distributed Kalman filtering with adaptive weights,” in *Proceedings of the Conference Record of the Forty-Third Asilomar Conference on Signals, Systems and Computers*, pp. 908-912, 2009.

[6] S. Y. Tu, and A. H. Sayed, “Diffusion strategies outperform consensus strategies for distributed estimation over adaptive networks,” *IEEE Transactions on Signal Processing*, vol. 60, no. 12, pp. 6127-6234, 2012.

[7] G. Q. Wang, N. Li, and Y. G. Zhang, “Diffusion distributed Kalman filter over sensor networks without exchanging raw measurements,” *Signal Processing*, vol. 132, pp. 1-7, 2017.

[8] R. Olfati-Saber, “Distributed Kalman filtering for sensor networks,” in *Proceedings of the 46th IEEE Conference Decision and Control*, pp. 5492-5498, 2007.

[9] R.Olfati-Saber, “Kalman consensus filter: optimality, stability, and performance,” in *Proceedings of the Joint 48th IEEE Conference on Decision and Control and 28th Chinese Control Conference*, pp. 7036-7042, 2009.

[10] R. Deshmukh, C. Kwon, and I. Hwang, “Optimal discrete-time Kalman consensus filter,” in *Proceedings of the American Control Conference*, pp. 5801-5806, 2017.

[11] G. Battistelli, L. Chisci, G. Mugnai, A. Farina, and A. Graziano, “Consensus-based linear and nonlinear filtering,” *IEEE Transactions on Automatic Control*, vol. 60, no. 5, pp. 1410-1415, 2015.

[12] S. Boyd, A. Ghosh, B. Prabhakar, and S. Devavrat, “Analysis and optimization of randomized gossip algorithms,” in *Proceedings of the 43rd IEEE Conference on Decision and Control*, pp. 5310-5315, 2004.

[13] S. Boyd, A. Ghosh, B. Prabhakar, and S. Devavrat, “Randomized gossip algorithms,” *IEEE Transactions on Information Theory*, vol. 52, no. 6, pp. 2508-2530, 2006.

[14] L. Y. Jiang, C. Wan, Y. X. Gao, and Z. S. Duan, “Distributed filtering based on randomized gossip strategy,” in *Proceedings of the IEEE International Conference on Multisensor Fusion and Integration for Intelligent Systems*, pp. 306-311, 2017.

[15] A. T. Alouani, and T. R. Rice, “On asynchronous data fusion,” in *Proceedings of the 26th Southeastern Symposium on System Theory*, pp. 143-146, 1994.

[16] Y. Y. Hu, Z. S. Duan, and D. H. Zhou, “Estimation fusion with general asynchronous multi-rate sensors,” *IEEE Transactions on Aerospace and Electronic Systems*, vol. 46, no. 4, pp. 2090-2102, 2010.

[17] L. P. Yan, X. R. Li, Y. Q. Xia, and M. F. Fu, “Modeling and estimation of asynchronous multirate multisensor system with unreliable measurements,” *IEEE Transactions on Aerospace and Electronic Systems*, vol. 51, no. 3, pp. 2012-2026, 2015.

[18] H. L. Lin, and S. L. Sun, “Distributed fusion estimation for multi-sensor asynchronous sampling systems with correlated noises,” *International Journal of Systems Science*, vol. 48, no. 5, pp. 952-960, 2017.

[19] G. M. Zhu, F. Zhou, L. Xie, R. X. Jiang, and Y. W. Chen, “Sequential asynchronous filters for target tracking in wireless sensor networks,” *IEEE Sensors Journal*, vol. 14, no. 9, pp. 3174-3182, 2014.

[20] X. S. Yang, W. A. Zhang, Z. Q. Chen, and L. Yu, “Hybrid sequential fusion estimation for asynchronous sensor network-based target tracking,” *IEEE Transactions on Control Systems Technology*, vol. 25, no. 2, pp. 669-676, 2017.