A CA-GCI Based Consensus Fusion Approach for Distributed Sensors with Different Fields of View

Feng Ma1, Huanzhang Lu1*, Luping Zhang1 and Xinglin Shen1

1National Key Laboratory of Science and Technology on Automatic Target Recognition, National University of Defense Technology, Changsha 410073, China
*Corresponding author’s e-mail: little_ma96@163.com

Abstract. This paper is concerned with the distributed fusion problem of sensors with different fields of view (FoVs). The recently derived CA-GCI method provides solution to the fusion problem of sensors with different FoVs. However, it does not mention about the way that how to share information among more than 3 sensors. Consensus method is suitable for information sharing in distributed network, but it always based on the assumption that the sensors all have the same FoV. Thus the existing method cannot be applied directly. Based on the combination of the consensus method and the newly derived CA-GCI method, this paper proposes a consensus fusion approach for the sensors with different FoVs in a distributed network. Furthermore, this paper derives the minimum iteration steps required to ensure the network consensus is $|\mathcal{N}| - 1$, where $|\mathcal{N}|$ represents the number of nodes in the networks. Simulation results demonstrate the effectiveness of the proposed approach.

1. Introduction

Multitarget tracking technology is of great practical relevance in both military and civil fields. Typical MTT techniques include joint probabilistic data association (JPDA) filter [1], multiple hypothesis tracker (MHT) [2], [3] and random finite set (RFS) based methods [4], [5]. Fusing the observations from multiple sensors can provide superior tracking capability and a more complete view of the environment. Thus multisensor multitarget tracking (MMT) technologies were proposed, and which can be classified into centralized [6]-[9] and distributed mode according to network architecture. Compared with centralized mode, distributed mode can provide lower communication cost and higher fault tolerance, which aroused more interests in recent years.

The consensus fusion method [9], [10] is an iterative information fusion method suitable for distributed sensor networks. According to the consensus method, at each iteration step, each node updates its multitarget density with the weighted average of the multitarget density of the nodes in its neighbourhood. After a certain number of iterations, each node can approximate the average of all nodes in the network. This iterative update approach provides a mechanism for information transmission in distributed networks. Based on consensus method, many solutions have been developed to solve distributed MMT (DMMT) problems. In [11], [12] the consensus method based distributed Kalman filter is developed. In [13], the consensus CPHD filter is introduced. In [14], [15], the combination of generalized covariance intersection (GCI) and labeled RFS filter is investigated and the consensus marginalized δ-generalized labeled multi-Bernoulli (Mδ-GLMB) and consensus labeled multi-Bernoulli (LMB) tracking filter are developed. However, it is assumed in these studies that all targets are in the FoV of all sensors, or that the sensors in the network share the same FoV, which is unreasonable in practical situations and limits their application. Recently, a modified GCI fusion strategy named
clustering algorithm GCI (CA-GCI) is proposed in [16], which is suitable for solving the fusion problem of sensors with different FoVs. But it does not mention how to share information among more than 3 sensors. Obviously, the consensus method is an appropriate choice, but there are still some problems need to be solved, such as: how to exploit the FoV information during consensus iteration; how many iteration steps are required to achieve an acceptable fusion results, etc. In this paper, the related problems are investigated and a consensus fusion approach for distributed sensors with different FoVs is proposed. Furthermore, the minimum number of iteration steps required to achieve an acceptable fusion result is given.

The rest of this paper is organized as follows. The CA-GCU approach and the consensus method are briefly introduced in Section II. Section III describes the proposed consensus fusion approach for distributed sensors with different FoVs. The required number of minimum iteration steps is analyzed in Section IV. Simulation experiments are reported in Section V and a final conclusion is given in Section VI.

2. Background

2.1. The CA-GCI Fusion Approach

Suppose that the network contains two nodes with known and fixed FoV knowledge. In each node, a local filter under the Bayesian framework is running. In this paper, the PHD filter is chosen as the local filter. At each time instant k, multitarget posterior densities \( \pi_k^{(1)}(X) \) and \( \pi_k^{(2)}(X) \) are prepared in node 1 and 2 respectively, where \( X \) represents the multi-target states. The GCI fusion rule can be formulated as [1]:

\[
\pi_k^{(1,2)}(X) = \frac{\int \pi_k^{(1)}(X) \omega_1}{\int \pi_k^{(1)}(X) \omega_1} \frac{\int \pi_k^{(2)}(X) \omega_2}{\int \pi_k^{(2)}(X) \omega_2} \delta X
\]

where \( \omega_1, \omega_2 \) represent the fusion weight assigned to node 1 and 2, satisfying \( \omega_1 + \omega_2 = 1 \). Since the discussion in this paper is all based on time k, for the sake of simplicity, the k is omitted below.

Suppose that the multitarget posterior density \( \pi(X) \) can be decomposed into multiple single-target posterior densities \( \tilde{\pi}_m^{(i)}(x) \), then

\[
\pi^{(1)}(X) = \sum_{m=1}^{N_1} \tilde{\pi}_m^{(1)}(x)
\]

\[
\pi^{(2)}(X) = \sum_{m=1}^{N_2} \tilde{\pi}_m^{(2)}(x)
\]

where \( N_i \) represents the number of targets contained in the FoV of the i-th sensor.

When the sensors share the same FoV, then \( N_1 = N_2 \) and \( \pi^{(1)}, \pi^{(2)} \) can be fused correctly according to the GCI rule. But when the sensors have different FoVs, the GCI fusion result is inappropriate that the targets outside the intersection of the FoVs (IoF) are lost. This is because the essence of GCI method can be interpreted as an intersection operation among the posteriors estimated by different sensors, thus only the information corresponding to the targets located in the IoF is retained. More specific, assuming that there are M targets within the IoF of the 2 sensors, there will be \( N_1 - M \) and \( N_2 - M \) single-target densities in \( \pi^{(1)} \) and \( \pi^{(2)} \), respectively, that are not involved in the fusion process, so they are lost in the GCI fusion result. To address this problem, [16] proposed CA-GCI fusion for sensors with different FoVs, the multi-target density outside the IoF are constructed as

\[
\tilde{\pi}^{(1)}(X) = \sum_{m=M+1}^{N_1} \tilde{\pi}_m^{(1)}(x)
\]
\[ \pi^{(2)}(X) = \sum_{m=M+1}^{N_2} \pi_m^{(2)}(x) \]  

where \( \pi^{(1)} \) corresponding to the density in region \( \text{FoV}_1/\text{FoV}_2 \), \( \pi^{(2)} \) corresponding to the density in region \( \text{FoV}_2/\text{FoV}_1 \). Let \( \pi^{(12)} \) represent the GCI fusion result corresponding to the region \( \text{FoV}_1 \cap \text{FoV}_2 \), then the complete fusion result is

\[ \pi^{(12)} = \pi^{(1)} + \pi^{(2)} \]  

An illustration of CA-GCI is shown in Fig. 1.

![An illustration of CA-GCI approach](image)

2.2. The Consensus Fusion Approach

Suppose that a distributed sensor network can be represented as a graph \( G = (\mathcal{N}, \mathcal{A}) \), where \( \mathcal{N} \) represents the set of nodes and \( \mathcal{A} \) represents the set of arcs. If \((i, j) \in \mathcal{A}\), means that node \( i \) and node \( j \) can communicate with each other, otherwise, two nodes cannot communicate. For each node \( i \in \mathcal{N} \), define the set of in-neighbors \( \mathcal{N}^{(i)} \triangleq \{ j \in \mathcal{N} : (i, j) \in \mathcal{A} \} \), represent the set of all the nodes that node \( i \) can communicate with (node \( i \) itself is also included).

In [13], [14], the consensus approach based on the same FoV assumption is introduced in detail. Let operation \( \text{GCI}(\cdot) \) denote the fusion process based on GCI approach, then (1) can be expressed as

\[ \pi^{(12)} = \text{GCI}(\omega_1, \pi^{(1)}_1, \omega_2, \pi^{(2)}_2) \]  

When there are more than 2 nodes in the in-neighbor set (i.e. \( |\mathcal{N}^{(i)}| \geq 3 \)), the strategy of sequential fusion is always adopted. And the fusion result of node \( i \) can be denoted as

\[ \hat{\pi}^{(i)} = \text{GCI}_{j \in \mathcal{N}^{(i)}}(\omega_{ij}, \hat{\pi}^{(j)}_i) \]  

Equation (8) is the initial density of the consensus iteration process. And the iteration process can be denoted by

\[ \hat{\pi}^{(i)}_n = \text{GCI}_{j \in \mathcal{N}^{(i)}}(\omega_{ij}, \hat{\pi}^{(j)}_{n-1}) \]  

where \( n \) represents the iteration step, \( \omega_{ij} \) represents the weight node \( j \) contribute to node \( i \). Consensus matrix \( \Omega \) can be formed by using \( \omega_{ij} \) be its \((i, j)\)-th element. If the consensus matrix \( \Omega \) is primitive and double stochastic, then the consensus iteration of each node approaches the collective unweighted KLA of the posterior densities over the entire network as the number of consensus steps tends to infinity. This paper follows the introduction of [13], [14] and selects the Metropolis weights, which can guarantee the primitive and double stochastic of the consensus matrix. The Metropolis weights is defined as follows.
\[
\omega_{i,j} = \begin{cases} 
1 + \max\{|\mathcal{N}(i)|, |\mathcal{N}(j)|\}, & i \in \mathcal{N}, j \in \mathcal{N}(i) \setminus \{i\} \\
1 - \sum_{j \in \mathcal{N}(i) \setminus \{i\}} \omega_{i,j}, & i \in \mathcal{N}, j = i
\end{cases}
\] (10)

3. Consensus Fusion for Sensors with Different FoV

Equation (8) (9) formulates the consensus fusion approach based on the GCI rule and the implicit assumption is that the sensors share the same FoV. As sensors have different FoVs, the GCI fusion rule will lose the targets information outside the IoF, which has been discussed in Section II. The CA-GCI is suitable for different FoV scenarios, and the targets information outside the IoF can be retained after fusion. For node \(i\), when \(|\mathcal{N}(i)| \geq 3\), the sequential fusion strategy is adopted. Different from the same FoV assumption, when the FoVs of sensors are different, the process of sequential fusion needs to consider the effect of that. And the effect is related to the process of CA-GCI fusion. Since the CA-GCI approach needs to use the FoV information to partition the single-target component of the multi-target density inside the IoF or not, the FoVs corresponding to the densities to be fused should be provided. Therefore, when sequential fusion carried on, the corresponding FoVs of the fusing densities should be involved as the parameter in the iteration process. Let the CA-GCI fusion approach introduced in Section II.A denoted as \(\text{CAGCI}(\cdot)\), and the process of sequential fusion for sensors with different FoVs can be expressed as:

\[
\text{for } i = 1: |\mathcal{N}(i)| - 1 \\
\left(\pi^{(i+1)}, \text{FoV}^{(i+1)}\right) = \text{CAGCI}\left(\omega_{\pi}, \pi^{(i)}, \text{FoV}^{(i)}, \omega_{\pi+1}, \pi^{(i+1)}, \text{FoV}^{(i+1)}\right) \\
\text{end}
\] (11)

where

\[
\text{FoV}^{(i+1)} = \text{FoV}^{(i)} \cup \text{FoV}^{(i+1)}
\] (12)

In (11), (12), two points need to be noted: (a) \(\text{FoV}^{(i)}\) does not represent the FoV of sensor \(i\), but the range of the FoV corresponding to the fused density \(\pi^{(i)}\); (b) In (12), \(\text{FoV}^{(i+1)}\) (output) represent the union of \(\text{FoV}^{(i)}\) and \(\text{FoV}^{(i+1)}\) (input). Equation (11), (12) can be understood as that after CA-GCI fusion, not only the density of the two nodes are fused, but their FoVs are also fused.

Similarly, this problem also exists in the process of consensus fusion. For (9), simply replacing GCI fusion with CA-GCI fusion could not obtain the correct iterative result. In the 0-th step of iteration, the provided density of node \(i\) is the fusion result of \(\mathcal{N}(i)\) and its corresponding FoV is the union of FoVs of the sensors in \(\mathcal{N}(i)\). As the iteration proceeds, node \(i\) will continuously fuse information from other nodes, and its corresponding FoV range will continue expanding. So (9) need to be modified accordingly.

Let the fusion result of \(\mathcal{N}(i)\) denoted by

\[
(\pi^{(i)}, \text{FoV}^{(i)}) = \text{CAGCI}\left(\omega_{\pi}, \pi^{(i)}, \text{FoV}^{(i)}\right)
\] (13)

And it should be viewed as the 0-th step of consensus iteration, i.e.

\[
(\pi_0^{(i)}, \text{FoV}_0^{(i)}) = \text{CAGCI}\left(\omega_{\pi}, \pi^{(i)}, \text{FoV}^{(i)}\right)
\] (14)

Then the iteration process can be denoted by

\[
(\pi_n^{(i)}, \text{FoV}_n^{(i)}) = \text{CAGCI}\left(\omega_{\pi}, \pi_n^{(i)}, \text{FoV}_{n-1}^{(i)}\right)
\] (15)

where

\[
\text{FoV}_n^{(i)} = \bigcup_{j \in \mathcal{N}(i)} \text{FoV}_{n-1}^{(j)}
\] (16)
Equation (15), (16) shows that in the process of consensus iteration, the FoV\(^{(i)}\) corresponding to the estimated PDF \(\pi^{(i)}_n\) is becoming larger and larger. After a certain number of iterations, the information contained in all FoVs of sensors in the network is fused. This process will be illustrated in the experiments in Section V. Therefore, the consensus fusion approach proposed in this paper can be summarized as follows, where \(N_{\text{iter}}\) represent the total iteration steps.

Algorithm 1. Consensus fusion with different FoV

\begin{align*}
\text{Init:} & \quad \text{Initialize } \pi^{(0)}_0 \text{ and } \text{FoV}^{(0)}_0 \text{ using (14)} \\
1: & \quad \text{for } n = 1:N_{\text{iter}} \\
2: & \quad \quad \text{Update } \pi^{(i)}_n \text{ and } \text{FoV}^{(i)}_n \text{ using (15) (16)} \\
3: & \quad \text{end} \\
\text{return: } & \quad \pi^{(N_{\text{iter}})}_n 
\end{align*}

4. The Minimal Iteration Steps

As consensus fusion method points out that when the number of iteration steps approaches infinity, each node of the distributed network approximates the average of the posterior densities over the entire network. However, in practical applications, only a limited step of iterations can be performed. Moreover, under the condition that the fusion result is acceptable, the fewer iteration steps the better. In the MMT problem, the evaluation of tracking performance often includes two aspects. On the one hand, it provides the state of all targets in the scene, that is, the correct targets number is estimated. On the other hand, it provides high target positioning accuracy. The "acceptable fusion result" discussed in this section refers to the first aspect, which is the more basic and important aspect of the tracking system, that is, each node can estimate the correct targets number of the whole scenario. Based on this goal, we analyze the minimum number of iteration steps required for the consensus fusion.

In distributed network \(\mathcal{G} = (\mathcal{N}, \mathcal{A})\), define distance \(d_{ij}\) denote the number of arcs between node \(i\) and \(j\). Analysis of (15) shows that in each iteration step, node \(i\) will fuse all the information of nodes contained in \(\mathcal{N}^{(i)}\), and the distance between these nodes and node \(i\) is 1 arc. In other words, the distance of information propagation is 1 arc per iteration step, i.e. the speed of information propagation is \(v_b = 1\text{arc/iter}\). Suppose the two farthest nodes in the network are \(I\) and \(J\), and there are some targets that could be observed by node \(J\) but could not be observed by node \(I\). To make \(I\) receive all the information of the targets in the scenario, the posteriors of node \(J\) need to be propagated to node \(I\), and the distance of propagation is \(d_{ij}\). Based on the above discussion, the iteration steps required for the propagation is \(d_{ij}/v_b\). With network \(\mathcal{G} = (\mathcal{N}, \mathcal{A})\), under the condition that the network topology is unknown, the distance between the two farthest nodes is \(\max_{(i,j) \in \mathcal{A}} d_{ij} = |\mathcal{N}|-1\), where \(|\mathcal{N}|\) represent the number of nodes in \(\mathcal{G}\). Then the number of iteration steps required to ensure that each node receives the information from the entire network is

\[\frac{\max_{(i,j) \in \mathcal{A}} d_{ij}}{v_b} = |\mathcal{N}|-1\]  \hspace{1cm} (17)

Therefore, the following proposition is given:

**Proposition** For distributed networks, under the condition that the network topology is unknown, the minimum number of iteration steps required to ensure each node can receive the information from the nodes of the entire network is \(|\mathcal{N}|-1\).

5. Simulation Experiment

Assume that the sensor network is distributed as shown in Fig. 2, where 5 sensors are contained. Suppose the dashed circle represents the communication range of each sensor. Thus \(\mathcal{N}^{(1)} = \{1, 2\}, \mathcal{N}^{(2)} = \{1, 2, 3\}, \mathcal{N}^{(3)} = \{2, 3, 4\}, \mathcal{N}^{(4)} = \{3, 4, 5\}, \mathcal{N}^{(5)} = \{4, 5\}\). For better verification of approach performance, this section includes 2 scenarios, as shown in Fig. 3 and Fig. 5. In order to verify the

---

5
iteration steps required for convergence of the consensus fusion approach, the targets in scenario 1 are set to be static. In order to verify the performance of the consensus fusion approach in complex tracking scenarios, the targets in scenario 2 is set to be moving. The FoV range of these sensors is 90°, the direction is the positive direction of the Y axis, and the sensor position is as shown in Table 1. In scenario 1, the true targets positions are represented by “+”. In scenario 2, circles represent the start points of the targets, and triangles represent the end points. The targets initial state in scenario 1 is shown in Table 2.

### Table 1. Sensor Position

| Sensor ID | Sensor position |
|-----------|-----------------|
| 1         | [400,0]         |
| 2         | [600,0]         |
| 3         | [800,0]         |
| 4         | [1000,0]        |
| 5         | [1200,0]        |

### Table 2. Target initial state

| Target ID | Initial state   | Target ID | Initial state   |
|-----------|-----------------|-----------|-----------------|
| 1         | [60,0,420,0]^T  | 6         | [930,0,740,0]^T |
| 2         | [390,0,310,0]^T | 7         | [1400,0,930,0]^T |
| 3         | [170,0,730,0]^T | 8         | [1100,0,440,0]^T |
| 4         | [610,0,490,0]^T | 9         | [1480,0,600,0]^T |
| 5         | [610,0,810,0]^T | 10        | [1360,0,280,0]^T |

It can be seen from Fig. 3 that the FoV of each sensor is different, and each sensor has some targets information that cannot be observed by other sensors. Therefore, to estimate the complete number of targets, each node must obtain the information of all sensors through the distributed networks. We first theoretically analyze the fusion process according to the network topology in Fig. 3 and the iteration steps in (14)-(16). In additional, we define the concept of targets number converges.

**Definition 1.** When the estimated targets number reaches the true value, it is called the estimated targets number of the node converges.

**Iteration Step 1.** Node 1 fuse the information of all nodes in \( \mathcal{N}(1) \) to obtain \( \bar{\pi}_1^{(1)} \), which includes 7 targets, corresponding to FoV_{1}^{(1)} = \text{ FoV}(1) \cup \text{ FoV}(2) \). Node 2 fuse the information of all nodes in \( \mathcal{N}(2) \) to obtain \( \bar{\pi}_1^{(2)} \), which includes 8 targets, corresponding to FoV_{1}^{(2)} = \text{ FoV}(1) \cup \text{ FoV}(2) \cup \text{ FoV}(3) \). Node 3 fuse the information of all nodes in \( \mathcal{N}(3) \) to obtain \( \bar{\pi}_1^{(3)} \), which includes 8 targets, corresponding to FoV_{1}^{(3)} = \text{ FoV}(2) \cup \text{ FoV}(3) \cup \text{ FoV}(4) \). In the same way, node 4 and 5 get \( \bar{\pi}_1^{(4)} \) including 8 targets and \( \bar{\pi}_1^{(5)} \) including 7 targets, respectively.

**Iteration step 2.** Node 1 fuse the 2 densities \( \bar{\pi}_1^{(1)} \) and \( \bar{\pi}_1^{(2)} \) in \( \mathcal{N}(1) \) to obtain \( \bar{\pi}_1^{(3)} \), which includes 8 targets, corresponding to FoV_{2}^{(1)} = \text{ FoV}(1) \cup \text{ FoV}(2) \cup \text{ FoV}(3) \cup \text{ FoV}(4) \). Node 2 fuse the 3 densities \( \bar{\pi}_1^{(1)} \), \( \bar{\pi}_1^{(2)} \) and \( \bar{\pi}_1^{(3)} \) in \( \mathcal{N}(2) \) to obtain \( \bar{\pi}_1^{(4)} \), which includes 9 targets, corresponding to FoV_{2}^{(2)} = \text{ FoV}(1) \cup \text{ FoV}(2) \cup \text{ FoV}(3) \cup \text{ FoV}(4) \). Node 3 fuse the 3

---

**Figure 2. Sensor network**

**Figure 3. Scenario 1**
densities \( \pi_1^{(2)}, \pi_1^{(3)} \) and \( \pi_1^{(4)} \) in \( \mathcal{N}^{(3)} \) to obtain \( \pi_1^{(3)} \), which includes 10 targets, corresponding to \( \text{FoV}_1^{(3)} = \text{FoV}_1^{(2)} \cup \text{FoV}_1^{(3)} \cup \text{FoV}_1^{(4)} = \text{FoV}^{(1)} \cup \text{FoV}^{(2)} \cup \text{FoV}^{(3)} \cup \text{FoV}^{(4)} \cup \text{FoV}^{(5)} \). The estimated targets number of node 3 converges.

Iteration step 3. Node 1 fuse the 2 densities \( \pi_1^{(2)} \) and \( \pi_1^{(3)} \) in \( \mathcal{N}^{(1)} \) to obtain \( \pi_1^{(1)} \), which includes 9 targets, corresponding to \( \text{FoV}_1^{(1)} = \text{FoV}_1^{(2)} \cup \text{FoV}_1^{(3)} \cup \text{FoV}_1^{(4)} = \text{FoV}^{(1)} \cup \text{FoV}^{(2)} \cup \text{FoV}^{(3)} \cup \text{FoV}^{(4)} \). Node 2 fuse the 3 densities \( \pi_2^{(1)}, \pi_2^{(2)} \) and \( \pi_3^{(2)} \) in \( \mathcal{N}^{(2)} \) to obtain \( \pi_3^{(2)} \), which includes 10 targets, corresponding to \( \text{FoV}_3^{(2)} = \text{FoV}_2^{(1)} \cup \text{FoV}_2^{(2)} \cup \text{FoV}_2^{(3)} = \text{FoV}^{(1)} \cup \text{FoV}^{(2)} \cup \text{FoV}^{(3)} \cup \text{FoV}^{(4)} \cup \text{FoV}^{(5)} \). The estimated targets number of node 2 converges.

Iteration step 4. Node 1 fuse the 2 densities \( \pi_1^{(1)} \) and \( \pi_3^{(2)} \) in \( \mathcal{N}^{(1)} \) to obtain \( \pi_4^{(1)} \), which includes 10 targets, corresponding to \( \text{FoV}_4^{(1)} = \text{FoV}_3^{(1)} \cup \text{FoV}_3^{(2)} = \text{FoV}^{(1)} \cup \text{FoV}^{(2)} \cup \text{FoV}^{(3)} \cup \text{FoV}^{(4)} \cup \text{FoV}^{(5)} \). The estimated targets number of node 1 converges.

Figure 4. The estimated targets number of each node versus iteration steps

Fig. 4 records the estimated targets number of each node in the experiment with the increase of iteration steps. The results take the mean value of 100 Monte Carlo experiments. In Fig. 4, the abscissa represents the step of iterations, the ordinate represents the estimated targets number, and different types of lines represent different nodes.

It can be seen from the experimental results that node 1 and node 5 take 4 steps to achieve convergence, node 2 and node 4 take 3 steps to achieve convergence, and node 3 take 2 steps to achieve convergence. This result is completely consistent with the above theoretical analysis. At the same time, we note that more iteration steps are needed to achieve convergence for the nodes located at the edge of the network than those located at the center of the network. This reflects the way distributed networks transmit information. It does not rely on the central node, but depends on the information exchange with their neighbor nodes whose information is updated through continuous iteration. When the iteration steps reach the number of Proposition 1, each node in the network receives the information of all nodes in the network.

To further verify the performance of the consensus fusion approach, a more complex scenario 2 is simulated, as shown in Fig. 5. On the basis of scenario 1, the targets are moving, and the complex cases such as the target crossing the FoV boundary is also included. The initial state of the targets is as shown in Table 3.
OSPA distance is chosen to assess the tracking result, and the cut-off parameter is 30. The representative node 1, 2 and 3 are selected for analysis, corresponding to Fig. 6 (a), (b) and (c), respectively. We separately count the OSPA distances of the iteration steps from 1-7 respectively, represented by different lines. On the whole trend, node 1 takes 4 steps to achieve convergence, node 2 takes 3 steps to achieve convergence, and node 3 takes 2 steps to achieve convergence, which is the same as the experimental results of scenario 1. The experimental results show that the main factor affecting the OSPA distance is the OSPA Card (targets number), and the continuous increase of the number of iteration steps after the node converges has little effect on the OSPA distance. By analyzing the OSPA distance at different times instance, it can be found that the number of iteration steps required for convergence at different times of the same node is not same, which is due to the change of the targets number in the FoV of sensors caused by the target moving. Taking node 3 as an example, at the initial moment, the nodes in \( \mathcal{N}(3) \) does not cover all the targets information of node 4 and 5, and it needs 2 steps to converge. When the target moves at 26 s, all the targets enter the FoVs covered by the nodes in \( \mathcal{N}(3) \), so only 1 iteration step is needed for convergence.

### Table 3. Target initial state

| Target ID | Initial state | Target ID | Initial state |
|-----------|---------------|-----------|---------------|
| 1         | [602, 8420, 2.8] \(^T\) | 6         | [930, -3.4, 740, -2] \(^T\) |
| 2         | [390, 2310, 3.4] \(^T\) | 7         | [1400, -4.930, 0] \(^T\) |
| 3         | [170, 34730, 2] \(^T\) | 8         | [1100, -2.8, 440, -2.8] \(^T\) |
| 4         | [610, 4490, 0] \(^T\) | 9         | [1480, -4.600, 0] \(^T\) |
| 5         | [610, 4810, 0] \(^T\) | 10        | [1360, 0280, 4] \(^T\) |
To further illustrate the performance of the consensus fusion approach, we statistically compare the estimated targets number of each local node with the consensus fusion result of node 1. The results take the mean of 100 Monte Carlo experiments. As shown in Fig. 7, the above 5 panels represent the targets number estimated by node 1-5 local filtering result. FR1 represents the targets number estimated by node 1 after the consensus fusion approach, in which the number of iteration step is 4. It can be seen from the experimental results that although each local node can only observe the targets in its own FoV, through the consensus fusion approach proposed in this paper, node 1 fused the targets information of all nodes in the network. Similarly, each node could estimate the complete number of targets, and the goal of distributed information fusion is achieved.

(c) Node 3
6. Conclusion
In this paper, a CA-GCI based consensus fusion approach for distributed sensors with different FoVs is proposed, which is more accord with the actual situation that different sensors have different FoVs and is of great practical significance. Theoretical analysis and experimental results verified the effectiveness of the proposed approach. Furthermore, as a conclusion the minimum iteration steps required to make distributed sensor network consensus is $|\mathcal{N}| - 1$, where $|\mathcal{N}|$ represents the number of nodes in the networks. The discussion in this paper is based on the active sensor network; the local filter employed the PHD filter. Next step, efforts can be focused on implementing the proposed method in the passive sensor network, and local filters can also be replaced by MB filters or labelled random set filters.

Acknowledgments
This work was supported by the National Natural Science Foundation of China (61901489).

References
[1] Y. Bar-Shalom and T. E. Fortmann, Tracking and Data Association. San Diego: Academic Press, 1988.
[2] S. Blackman and R. Popoli, Design and Analysis of Modern Tracking Systems. New York: Artech House, 1999.
[3] S. Blackman, “Multiple hypothesis tracking for multiple target tracking,” IEEE Aerospace & Electronic Systems Magazine, vol. 19(1), pp. 5–18, Jan 2004.
[4] R. Mahler, Statistical Multisource-Multitarget Information Fusion. Artech House, 2007.
[5] R. Mahler, Advances in Statistical Multisource-Multitarget Information Fusion, Artech House, 2014.
[6] R. Mahler, “The multisensor PHD filter: I. General solution via multitarget calculus,” in Proc. SPIE 7336, Signal Process., Sensor Fusion, Target Recogn. XVIII, 2009, pp. 1–12.
[7] R. Mahler, “Approximate multisensor CPHD and PHD filters,” in Proc. 13th IEEE Int. Conf. Inf. Fusion (FUSION), 2010, pp. 1–8.
[8] S. Nagappa and D. E. Clark, “On the ordering of the sensors in the iterated-corrector probability hypothesis density (PHD) filter,” in Proc. SPIE 8050, Signal Process., Sensor Fusion, Target Recogn. XX, Orlando, FL, USA, 2011, vol. 8050, pp. 0M-1–0M-6.
[9] Xiao L, Boyd S. “Fast linear iterations for distributed averaging”// 42nd IEEE International Conference on Decision and Control (IEEE Cat. No.03CH37475). IEEE, 2004.
[10] L. Xiao, S. Boyd and S. Lall, “A scheme for robust distributed sensor fusion based on average consensus,” Proc. 4th Int. Symp. on Information Processing in Sensor Networks, pp. 63-70, Los Angeles, CA, 2005
[11] R. Olfati-Saber, “Distributed Kalman filtering for sensor networks,” Proc. 46th IEEE Conf. on Decision and Control, pp. 5492-5498, 2007.
[12] R. Carli, A. Chiuso, L. Schenato, and S. Zampieri, “Distributed Kalman filtering based on consensus strategies,” IEEE Journal on Selected Areas in Communications, vol. 26, pp. 622-633, 2008.
[13] G. Battistelli, L. Chisci, C. Fantacci, A. Farina, and A. Graziano, “Consensus CPHD filter for distributed multitarget tracking,” IEEE J. Sel. Topics Signal Process., vol. 7, no. 3, pp. 508–520, Jun. 2013.
[14] C. Fantacci, B.-N. Vo, B.-T. Vo, G. Battistelli, and L. Chisci, “Consensus labeled random finite set filtering for distributed multi-object tracking,” arXiv:1501.01579v2, Jun. 2016
[15] Fantacci C, Vo B N, Vo B T, et al. “Robust Fusion for Multisensor Multiobject Tracking”. IEEE Signal Processing Letters, 2018:1-1.
[16] Li G, Battistelli G, Yi W, et al. “Distributed multi-sensor multi-view fusion based on generalized covariance intersection”. Signal processing, 2020, 166(Jan.):107246.1-107246.13.