sensors

MDPI

Article

Multisensor Multi-Target Tracking Based on GM-PHD Using Out-Of-Sequence Measurements

Meiqin Liu [1,2,*], Tianyi Huai [2], Ronghao Zheng [2] and Senlin Zhang [2]

[1] State Key Laboratory of Industrial Control Technology, Hangzhou 310027, China
[2] College of Electrical Engineering, Zhejiang University, Hangzhou 310027, China; 21710159@zju.edu.cn (T.H.); rzheng@zju.edu.cn (R.Z.); slzhang@zju.edu.cn (S.Z.)
* Correspondence: liumeiqin@zju.edu.cn; Tel.: +86-139-5800-7313

Received: 30 August 2019; Accepted: 1 October 2019; Published: 5 October 2019

Abstract: In this paper, we study the issue of out-of-sequence measurement (OOSM) in a multi-target scenario to improve tracking performance. The OOSM is very common in tracking systems, and it would result in performance degradation if we used it inappropriately. Thus, OOSM should be fully utilized as far as possible. To improve the performance of the tracking system and use OOSM sufficiently, firstly, the problem of OOSM is formulated. Then the classical B1 algorithm for OOSM problem of single target tracking is given. Next, the random finite set (RFS)-based Gaussian mixture probability hypothesis density (GM-PHD) is introduced. Consequently, we derived the equation for re-updating of posterior intensity with OOSM. Implementation of GM-PHD using OOSM is also given. Finally, several simulations are given, and results show that tracking performance of GM-PHD using OOSM is better than GM-PHD using in-sequence measurement (ISM), which can strongly demonstrate the effectiveness of our proposed algorithm.

Keywords: out-of-sequence; multi-target tracking; random finite set; gaussian mixture probability hypothesis density

1. Introduction

Multi-target tracking has been studied widely in recent years. It has attracted a lot of attention and is a major research hotspot in the field of target tracking. Multi-target tracking is not a simple extension of single-target tracking; it involves many situations that will not occur in single-target tracking. For example, in a multi-target tracking scenario, the number of targets is time-varying because new targets may appear and original targets may die. Therefore, a multi-target tracking algorithm needs to estimate the number of targets and the state of each target, at the same time. Because of the existence of clutter, it cannot guarantee that every measurement comes from a real target.

Multi-sensor measurements at each sampling time correspond to many different targets (or clutters). The problem of correlating measurements with targets is called data association, which is the key point and difficulty of a multi-target tracking algorithm and has been the focus of multi-target tracking research. Today, joint probability data association (JPDA) and multiple hypothesis tracking (MHT) are regarded as representative algorithms to solve data association problems [1–4]. There are also many improved algorithms based on JPDA and MHT. For example, [5] studied the JPDA filter with unknown detection probability and clutter rate, and [6] studied the problem of passive sonar tracking using MHT algorithm. It is noted that key developments of MHT over the past 40 years have been already reviewed in [7].

Probability hypothesis density (PHD) filter is a multi-target tracking algorithm based on random finite set (RFS) that has emerged in recent years. It was first proposed by Mahler in the literature [8]. This algorithm avoids the problem of data association. Thus, it can effectively solve the problem of a large amount of computation in a classical data association algorithm. A comparison of the RFS approach and traditional multi-target tracking methods has been given in [9]. Under the framework of RFS, the set of states of targets is regarded as a set-valued state, and the set of observations is regarded as a set-valued observation. Thus, RFS is the natural representation of the states of targets and observations of the targets. After the state and observation are modeled as RFS, a multi-target tracking problem can be formulated in Bayes filtering. Because multi-target Bayesian filtering involves multiple integrals, it will result in computational intractability. The PHD filter can alleviate the computational intractability because it operates on the single-target state space. However, there is no closed-form solution for PHD filter, so it is a common way to use the sequential Mento Carlo method to approximate it [10]. Another common implementation of PHD is Gaussian mixture probability hypothesis density (GM-PHD) [11], in which multiple Gaussian components are used to approximate PHD. Under the assumption of the linear Gaussian multi-target model, the analytical solution of GM-PHD can be obtained. This filtering algorithm is also adopted in this paper. In addition, these three multi-target tracking approaches mentioned above have been widely used in sonar [12,13], autonomous vehicles [14,15], fusion [16–18], visual target tracking [19,20], and sensor networks [21,22]. There are also some multi-target tracking algorithms and related work in the framework of a particle filter; for example, [23] introduced an efficient particle filter for multi-target tracking which solved the problem of the curse of dimensionality, [24] proposed two sampling techniques based on group importance sampling, and the problem of model selection in the framework of particle filter is studied in [25,26].

In a multi-target tracking system, target information measured by sensors is transmitted to the fusion center by the wireless network. However, due to network delay and other reasons, it is inevitable that some of measurements may arrive at the fusion center later. In this situation, we call it out-of-sequence measurement (OOSM). OOSM is a widely studied problem, and the main difficulty is how to use past measurements to update the current state. The research of one-step-lag OOSM in single target tracking is mature. The representative works based on Kalman filter framework are A1, B1, and C1 algorithms [27–29], where A1 is the optimal algorithm, and B1 and C1 are sub-optimal algorithms. In recent years, a particle filter has been widely used in the tracking system. Thus, OOSM based on a particle filter framework has also been extensively studied. Y-bar Shalom gives the optimal algorithm based on a particle filter [30]; however, this method requires excessive computational resources. To overcome this drawback, an efficient particle filter using OOSM is proposed in [31].

Although the OOSM problem and the multi-target tracking problem may appear in the tracking system simultaneously, there is little related work on the combination of them. Moreover, to the best of our knowledge, there is no related work about multi-target tracking based on RFS using OOSM that has been before. Obviously, in the actual tracking system, using OOSM in multi-target tracking is very common. In view of this, a novel multi-target tracking algorithm based on RFS using OOSM is proposed. When OOSM arrives, it is used to re-update the current state, instead of being discarded directly.

The structure of this paper is as follows. Section 2 gives the formulations of the B1 algorithm which is used to solve the problem of single-target tracking when OOSM arrives, and the PHD filter is also introduced in this section. Section 3 presents the main contribution of this paper, namely the formulation of re-update of GM-PHD filter using OOSM. And the implementation of GM-PHD filter using OOSM is also given in this section. To demonstrate the effectiveness of our proposed algorithm, several simulation results are given in Section 4. Finally, the analysis and conclusion are given in Section 5.

2. Problem Formulation

This section is arranged as follows. At first, the model of the system is given. Next, the problem of OOSM is formulated. Finally, the RFS based GM-PHD filter is summarized.

2.1. System Model

Consider a dynamic system as follows:

$$x_{k+1} = F_k x_k + w_k, \tag{1}$$

where F_k is state transition matrix to time t_{k+1} from time t_k, x_k is the state vector, and w_k is the process noise for this interval. The F_k has different forms when the target adopts different motions, and specific F_k related to our experiment is given in the Section 4. The state of target at time t_k is given by $x_k = [x, v_x, y, v_y]^T$, where (v_x, v_y) represent the velocities in the x,y coordinate, and (x, y) is the target location. And the process noise w_k is always assumed as zero-mean, white Gaussian noise:

$$w_k \sim N(0, Q_k), \tag{2}$$

where $Q_k = \mathrm{E}\{w_k w_k^T\}$. The measurement equation of system is given as follows:

$$z_k = H_k x_k + v_k, \tag{3}$$

where H_k is the measurement matrix of sensor; it also has different forms when sensors are different, and v_k is the measurement noise which is always assumed as zero-mean, white Gaussian noise:

$$v_k \sim N(0, R_k), \tag{4}$$

where $R_k = \mathrm{E}\{v_k v_k^T\}$.

2.2. OOSM B1 One-Step-Lag Algorithm

The OOSM's problem can be formulated as follows. At time t_k, we use the measurement Z_k to update the target state to get $\mathbf{x}_k$; subsequently, an earlier measurement's Z_d with time stamp t_d arrives, which satisfies the following in Equality (5), where Z_d is called an one-step-lag OOSM. The goal is to re-estimate the current state x_k with OOSM Z_d.

$$t_{k-1} < t_d < st_k. \tag{5}$$

Applying Equation (1), we can get the backward transition equation as follows:

$$x_k = F_{k,d} x_d + w_{k,d}, \tag{6}$$

where x_d is the state of target at time d, $F(k, d)$ is the state transition matrix to time t_k from time t_d, and $w_{k,d}$ is the process noise during this interval. And then, we can get the backward transition equation from (5):

$$x_d = F_{d,k}(x_k - w_{k,d}), \tag{7}$$

where $F_{d,k}$ is the backward transition matrix, and it is also the inverse matrix of $F_{k,d}$, which means $F_{d,k} = F_{k,d}^{-1}$. For a one-lag measurement problem, in which $t_{k-1} < t_d < t_k$, the schematic diagram is shown in Figure 1.

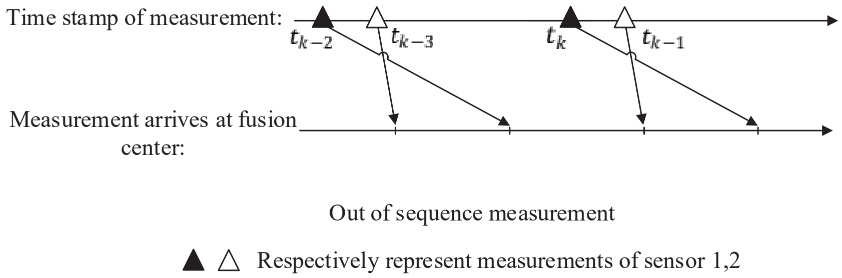

Time stamp of measurement:
t_{k-2}
t_{k-3}
t_k
t_{k-1}
Measurement arrives at fusion center:
Out of sequence measurement
▲ △ Respectively represent measurements of sensor 1,2

Figure 1. Out-of-sequence measurement (OOSM).

Based on the LMMSE criterion, using the basic equation of linear estimation, we can get the backward prediction of the state from t_k to t_d as follows:

$$\hat{x}_{d|k} = F_{k,d}(\hat{x}_{k|k} - \hat{w}_{k,d|k}), \tag{8}$$

where $\hat{w}_{k,d|k}$ is the estimation of process noise of backward prediction, and it is assumed to be zero in the sub-optimal B1 algorithm, which means under this situation $\hat{w}_{k,d|k} = 0$, and based on this point, we can get the following equation:

$$\hat{x}_{d|k} = F_{k,d}\hat{x}_{k|k}. \tag{9}$$

Then the covariance matrix of backward state prediction can be calculated as follows:

$$P_{d|k} = F_{k,d}[P_{k|k} + P^{ww}_{k,d|k} - P^{xw}_{k,d|k} - (P^{xw}_{k,d|k})^T]F^T_{k,d}. \tag{10}$$

The related covariance matrix in the Equation (10) can be calculated as follows:

$$P^{ww}_{k,d|k} = Q_{k,d}, \tag{11}$$

$$P^{xw}_{k,d|k} = Q_{k,d} - P_{k|k-1}H^T_k S^{-1}_k H_k Q_{k,d}, \tag{12}$$

$$P_{k|k} = P_{k|k-1} - P_{k|k-1}H^T_k S^{-1}_k H_k P_{k|k-1}. \tag{13}$$

Here, we just give the equations of the B1 algorithm, and the derivation of the equations in detail can be found in [27]. The covariance matrix of the measurement of backward state prediction is

$$S_d = H_d P_{d|k} H^T_d + R_d. \tag{14}$$

The covariance matrix between state $\mathbf{x}_k$ and measurement z_d is

$$P^{xz}_{k,d|k} = (P_{k|k} - P^{xw}_{k,d|k})F^T_{d,k}H^T_d. \tag{15}$$

Then we can calculate the filter gain as follows:

$$K_d = P^{xz}_{k,d|k}(S_d)^{-1}; \tag{16}$$

therefore, the re-update of current state $\hat{x}_{k|k}$ using the OOSM z_d is

$$\hat{x}_{k|d} = \hat{x}_{k|k} + K_d(z_d - H_d\hat{x}_{k|d}). \tag{17}$$

2.3. GM-PHD Filter

As mentioned in the previous section, the multi-target tracking algorithm based on random sets effectively avoids the problem of data association. Moreover, it does not need to know the number of targets in advance. In a word, it brings a new idea to solve the problem of multi-target tracking.

In multi-target tracking scenarios, the birth and death of the target should also be considered except from the target motion. Thus, it is necessary not only to estimate the state of multiple targets, but also to estimate the number of targets. This means that the number of elements in the multi-target state set and measurement set is variable. So, in a multi-target situation, the states of targets and measurements can be represented by the following:

$$X_k = \{x_k^1, ..., x_k^{n(k)}\} \in \mathcal{F}(E_s), \tag{18}$$

$$Z_k = \{z_k^1, ..., z_k^{m(k)}\} \in \mathcal{F}(E_o), \tag{19}$$

where $n(k)$ and $m(k)$ are the number of targets and the number of measurements at time t_k, respectively, and $\mathcal{F}(E_s)$ and $\mathcal{F}(E_o)$ are the multi-target state space and measurement space.

In the multi-target scenario, the birth and death of the target should also be considered separately from the target motion. At time t_{k-1}, each target either survives at time t_k with the probability $p_{s,k}$, or dies with probability $1 - p_{s,k}$. At time t_k, a new target may arise either by newborn or by spawning from a target at time t_{k-1}. Therefore, the multi-target state of next time t_k can be modeled by the following RFS:

$$X_k = S_k(X_{k-1}) \cup B_k(X_{k-1}) \cup \Gamma_k(X_{k-1}), \tag{20}$$

where $S_k(X_{k-1})$ is the RFS of multi-target state at time t_k, which evolves from time t_{k-1}, $B_k(X_{k-1})$ is the RFS of the newborn target, and $\Gamma_k(X_{k-1})$ is RFS of targets spawned from the previous state. Due to the existence of clutters in measurement, the RFS of measurement can be modeled as:

$$Z_k = \Theta_k(X_{k-1}) \cup K_k(X_{k-1}), \tag{21}$$

where $\Theta_k(X_{k-1})$ is the RFS of measurement of targets, and $K_k(X_{k-1})$ is the RFS of measurement of clutters.

The PHD filter is a typical RFS-based multi-target tracking algorithm. However, there is no closed form solution to PHD recursion, in general. But the PHD recursion has a closed-form solution in a certain class of situation, in which the multi-target model satisfies the linear Gaussian assumption. In this situation, the Gaussian mixture is used to approximate the intensity.

For clarity, the Gaussian mixture PHD (GM-PHD) filter is summarized in this paper, and readers can find more details in [11]. Under linear Gaussian assumption, the intensity at time t_{k-1} can be represented by the following equation:

$$v_{k-1}(x) = \sum_{i=1}^{J_{k-1}} \omega_{k-1}^{(i)} \mathcal{N}(x; m_{k-1}^{(i)}, P_{k-1}^{(i)}). \tag{22}$$

Then, the predicted intensity is given as follows:

$$\begin{aligned} v_{k|k-1}(x) =& v_{S,k|k-1}(x) + v_{\beta,k|k-1}(x) + v_{\gamma,k|k-1}(x) \\ =& \sum_{i=1}^{J_{k|k-1}} \omega_{k|k-1}^{(i)} \mathcal{N}(x; m_{k|k-1}^{(i)}, P_{k|k-1}^{(i)}), \end{aligned} \tag{23}$$

where $v_{S,k|k-1}(x)$, $v_{\beta,k|k-1}(x)$, $v_{\gamma,k|k-1}(x)$ represent intensity of survival target, intensity of new target spawned by the previous target, and intensity of newborn target, respectively. These three intensities are also Gaussian mixture, and their specific expressions can be found in [11]. Next, we can get the posterior intensity, which is given as follows:

$$v_k(x) = (1 - p_{D,k})v_{k|k-1}(x) + \sum_{z \in Z_k} v_{D,k}(x;z), \tag{24}$$

$$v_{D,k}(x;z) = \sum_{i=1}^{J_{k|k-1}} \omega_k^{(i)}(z)\mathcal{N}(x; m_{k|k}^{(i)}, P_{k|k}^{(i)}), \tag{25}$$

where $p_{D,k}$ is the probability of detect, $v_{D,k}(x;z)$ is the update item of predicted intensity, and the expression of $\omega_k^{(i)}(z)$, $m_{k|k}^{(i)}$, $P_{k|k}^{(i)}$ can be found in [11]. Finally, the pruning and merging is performed, and readers can also find the details in [11]

3. Multi-Target OOSM Tracking Algorithm

In the above two sections, the problems of OOSM and multi-target tracking based on RFS were discussed. In fact, these two issues have been widely studied. And the issue of multi-target tracking using OOSM has also attracted a lot of attention recently. However, to the best of our knowledge, there is no relative work about RFS-based multi-target tracking using OOSM. In the following, a novel multi-target tracking algorithm based on RFS is proposed, in which OOSM is used to re-update the posterior intensity. As we can see from Equations (9) and (12), predicted covariance $P_{k|k-1}$ is needed to calculate the retrodiction covariance $P_{d|k}$. However, in the GM-PHD algorithm, after pruning and merging, there is only posterior information left in the Gaussian component. In this situation, OOSM cannot be used to re-update the posterior intensity. Therefore, when measurement arrives, the judgment of whether the measurement is a delay should be executed first. If the timestamp of measurement shows that the measurement is in sequence, the update process of the GM-PHD filter is performed. On the contrary, if the measurement is a delay, our proposed algorithm is performed to re-update the multi-target state at time t_k. The flowchart of the proposed algorithm is given in Figure 2. Recalling the update equation of GM-PHD filter, substituting Equations (23) and (25) into update Equation (24), then the update equation can be rewritten as follows:

$$\begin{aligned} v_k(x) &= (1 - p_{D,k})v_{k|k-1}(x) + \sum_{z \in Z_k} \sum_{i=1}^{J_{k|k-1}} \omega_k^{(i)}(z)\mathcal{N}(x; m_{k|k}^{(i)}(z), P_{k|k}^{(i)}) \\ &= (1 - p_{D,k}) \sum_{i=1}^{J_{k|k-1}} \omega_{k|k-1}^{(i)}\mathcal{N}(x; m_{k|k-1}^{(i)}, P_{k|k-1}^{(i)}) + \sum_{i=1}^{|Z_k|J_{k|k-1}} \omega_k^{(i)}\mathcal{N}(x; m_{k|k}^{(i)}, P_{k|k}^{(i)}), \end{aligned} \tag{26}$$

where the $|Z_k|$ is the number of elements of set Z_k. In Equation (26), the first item in the right side is the prior intensity, which is associated with the estimated state $\hat{x}_{k|k-1}$. It also represents targets which are not detected, and we refer to it herein a 'prior target'. The second item represents detected targets, which we refer to herein as 'posterior target'.

The 'prior target' is the target which is not detected at time t_k, but when OOSM arrives, it may contain information about the 'prior target'. Therefore, when OOSM arrives, the estimated state $\hat{x}_{k|k-1}$ associated with 'prior target' should be updated. Analogously, the 'posterior target' is also needed to re-update using OOSM, and the weights of Gaussian components also need to be modified. The diagram of the classification processing algorithm using OOSM is shown in Figure 3.

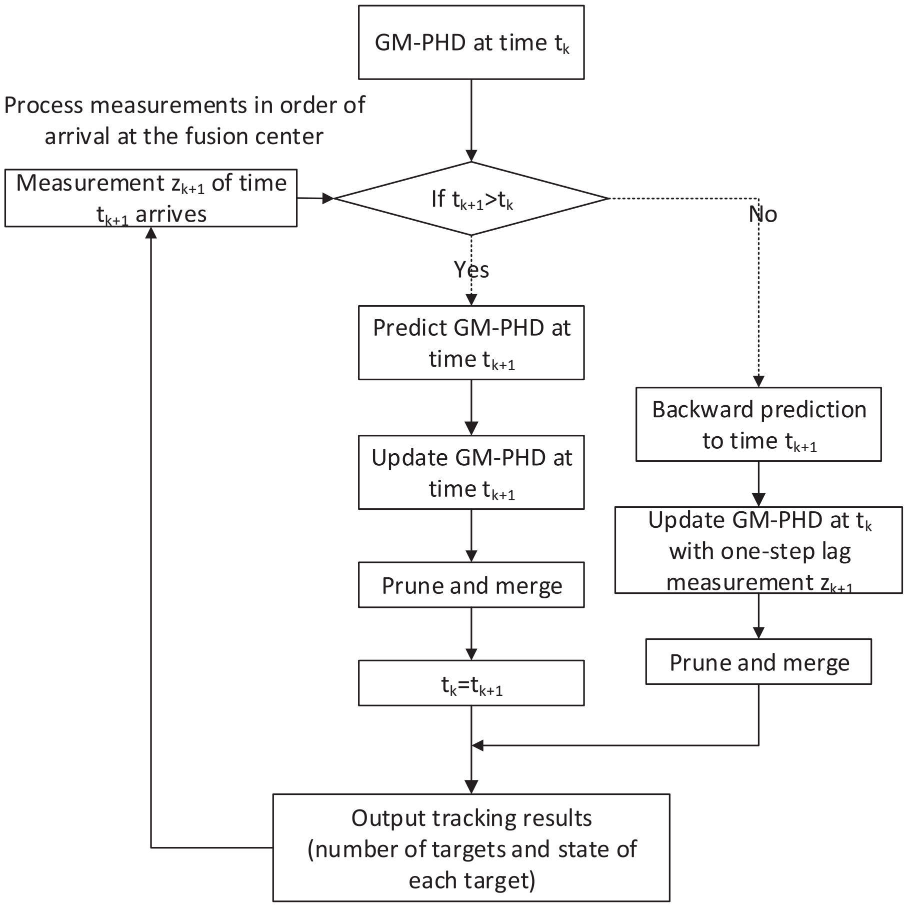

GM-PHD at time t_k
Process measurements in order of arrival at the fusion center
Measurement z_{k+1} of time t_{k+1} arrives
If t_{k+1}>t_k
No
Yes
Predict GM-PHD at time t_{k+1}
Update GM-PHD at time t_{k+1}
Prune and merge
t_k=t_{k+1}
Backward prediction to time t_{k+1}
Update GM-PHD at t_k with one-step lag measurement z_{k+1}
Prune and merge
Output tracking results (number of targets and state of each target)

Figure 2. Flowchart of proposed algorithm. GM-PHD = Gaussian mixture probability hypothesis density.

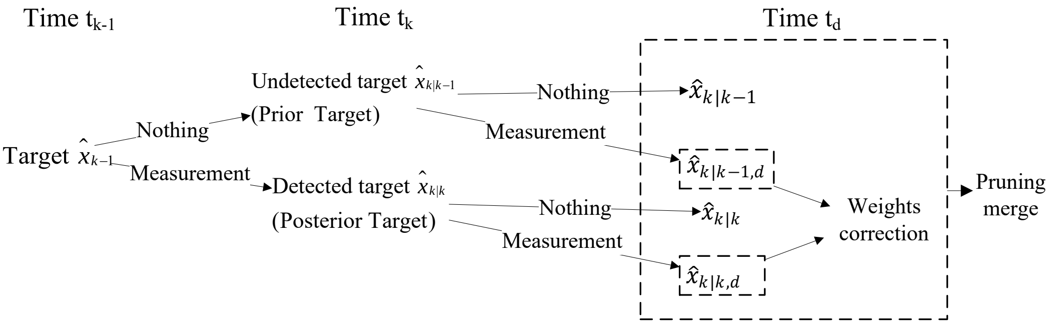


Figure 3. OOSM.

In the following, the derivation of the proposed algorithm is given.

3.1. Backward State Prediction

For simplicity, we assumed that there is no targets newborning and spawning from exist targets during time t_d to t_k, which means the number of targets at time t_k and time t_d are the same. The multi-target state at time t_d can be represented by the following:

$$X_d = \bigcup_{x \in X_{k|k-1} \cup X_{D,k}} S_{d|k}(x), \tag{27}$$

where $S_{d|k}(x)$ is the RFS of multi-target state at time t_d which evolve from time t_k, $X_{k|k-1}$ is the RFS of undetected targets' state, and $X_{D,k}$ is the RFS of detected targets' state. The intensity at time t_d can be divided into two items. One is retrodicted from the undetected intensity $(1-p_{D,k})v_{k|k-1}(x)$. The other is retrodicted from the detected intensity $\sum_{z\in Z_k} v_{D,k}(x;z)$. Applying retrodiction Equations (9) and (10) of the B1 algorithm, the intensity retrodicted from the undetected intensity is given as follows:

$$\begin{aligned} v_{d|k-1}(x) = v_{S,d|k-1} =& p_{S,d}(1-p_{D,k}) \sum_{i=1}^{J_{k|k-1}} \omega_{k|k-1}^{(i)} \mathcal{N}(x; m_{d|k-1}^{(i)}, P_{d|k-1}^{(i)}) \\ =& \sum_{i=1}^{J_{k|k-1}} \omega_{d|k-1}^{(i)} \mathcal{N}(x; m_{d|k-1}^{(i)}, P_{d|k-1}^{(i)}), \end{aligned} \tag{28}$$

where

$$\omega_{d|k-1}^{(i)} = (1-p_{D,k})p_{S,d}\omega_{k|k-1}^{(i)},\ m_{d|k-1}^{(i)} = F_{d,k}m_{k|k-1}^{(i)}, \tag{29}$$

$$P_{d|k-1}^{(i)} = F_{d,k}[P_{k|k-1}^{(i)} + P_{k,d|k-1}^{ww(i)} - P_{k,d|k-1}^{xw(i)} - (P_{k,d|k-1}^{xw(i)})^T]F_{d,k}^T \tag{30}$$

where $P_{k,d|k-1}^{ww(i)}$ is the ith Gaussian component's covariance of process noise from time t_d to time t_k, and $P_{k,d|k}^{xw(i)}$ is the ith Gaussian component's cross-covariance between state and process noise. They can be represented as follows:

$$P_{k,d|k-1}^{ww(i)} = \mathrm{cov}\{w_{k,d}|Z^{k-1}\} = Q_{k,d}, \tag{31}$$

$$\begin{aligned} P_{k,d|k-1}^{xw(i)} &= \mathrm{cov}\{w_{k,d}, x_k|Z^{k-1}\} \\ &= \mathrm{E}\{w_{k,d}[x_k - \hat{x}_{k|k-1}]^T|Z^{k-1}\} \\ &= \mathrm{E}\{w_{k,d}[F_{k,d}x_d + w_{k,d} - \hat{x}_{k|k-1}]^T|Z^{k-1}\} \\ &= \mathrm{E}\{w_{k,d}w_{k,d}^T|Z^{k-1}\} \\ &= Q_{k,d}. \end{aligned} \tag{32}$$

Substituting Equations (31) and (32) into (30), we can rewrite Equation (30) as follows:

$$\begin{aligned} P_{d|k-1}^{(i)} &= F_{d,k}[P_{k|k-1}^{(i)} + P_{k,d|k-1}^{ww(i)} - P_{k,d|k}^{xw(i)} - (P_{k,d|k}^{xw(i)})^T]F_{d,k}^T \\ &= F_{d,k}[P_{k|k-1}^{(i)} - Q_{k,d}]F_{d,k}^T. \end{aligned} \tag{33}$$

Analogously, we can easily get the intensity retrodicted from the detected intensity:

$$\begin{aligned} v_{d|k}(x) = v_{S,d|k} =& p_{S,d} \sum_{i=1}^{|Z_k|J_{k|k-1}} \omega_{k|k}^{(i)} \mathcal{N}(x; m_{d|k}^{(i)}, P_{d|k}^{(i)}) \\ =& \sum_{i=1}^{|Z_k|J_{k|k-1}} \omega_{d|k}^{(i)} \mathcal{N}(x; m_{d|k}^{(i)}, P_{d|k}^{(i)}), \end{aligned} \tag{34}$$

where

$$\omega_{d|k}^{(i)} = p_{S,d}\omega_{k|k}^{(i)},\ m_{d|k}^{(i)} = F_{d,k}m_{k|k}^{(i)}, \tag{35}$$

$$P_{d|k}^{(i)} = F_{d,k}[P_{k|k}^{(i)} + P_{k,d|k}^{ww,(i)} - P_{k,d|k}^{xw,(i)} - (P_{k,d|k}^{xw,(i)})^T]F_{d,k}^T, \tag{36}$$

where $P_{k|k}^{(i)}$, $P_{k,d|k}^{ww,(i)}$, and $P_{k,d|k}^{xw,(i)}$ can be derived from Equations (10)–(12). And the pseudo-code for backward state prediction is given in Algorithm 1.

Algorithm 1 Backward state prediction

```
Input $\{w_k^{(i)}, m_k^{(i)}, P_k^{(i)}\}_{i=1}^{J_{k|k}} = \{w_{k|k-1}^{(i)}, m_{k|k-1}^{(i)}, P_{k|k-1}^{(i)}\}_{i=1}^{J_{k|k-1}} \cup \{w_{k|k}^{(i)}, m_{k|k}^{(i)}, P_{k|k}^{(i)}\}_{i=1}^{|Z_k|J_{k|k-1}}$, and the out of sequence measurement $Z_d$
Step 1. Backward prediction
for $i = 1, 2, ..., J_{k|k-1}$ do
    $\omega_{d|k-1}^{(j)} = (1 - p_{D,d})p_{S,d}\omega_{k|k-1}^{(i)}, m_{d|k-1}^{(i)} = F_{d,k}m_{k|k-1}^{(i)}$
    $P_{d|k-1}^{(i)} = F_{d,k}[P_{k|k-1}^{(i)} + P_{k,d|k-1}^{ww,(i)} - P_{k,d|k-1}^{xw,(i)} - (P_{k,d|k-1}^{xw,(i)})^T]F_{d,k}^T$
end for
for $i = 1, ..., |Z_k|J_{k|k-1}$ do
    $\omega_{d|k}^{(j)} = p_{S,d}\omega_{k|k}^{(i)}, m_{d|k}^{(i)} = F_{d,k}m_{k|k}^{(i)}$
    $P_{d|k}^{(i)} = F_{d,k}[P_{k|k}^{(i)} + P_{k,d|k}^{ww,(i)} - P_{k,d|k}^{xw,(i)} - (P_{k,d|k}^{xw,(i)})^T]F_{d,k}^T$
end for
Output $\{w_{d|k-1}^{(i)}, m_{d|k-1}^{(i)}, P_{d|k-1}^{(i)}\}_{i=1}^{J_{k|k-1}} \cup \{w_{d|k}^{(i)}, m_{d|k}^{(i)}, P_{d|k}^{(i)}\}_{i=1}^{|Z_k|J_{k|k-1}}$
```

3.2. Re-Update with OOSM

It has been mentioned in Equation (26) that intensity $v_k(x)$ can be divided into two items. Therefore, when OOSM arrives, it is used for re-updating these two items, respectively, by using OOSM, and we can get the following re-update equation:

$$\begin{aligned} v_{k|d}(x) =& (1 - p_{D,d})\{[1 - p_{D,k}]v_{k|k-1}(x) + \sum_{z \in Z_k} v_{D,k}(x;z)\} + \sum_{z \in Z_d}\{v_{D,k|k-1,d}(x;z) + v_{D,k|k,d}(x;z)\} \\ =& (1 - p_{D,d})v_{k|k} + \sum_{z \in Z_d}\{v_{D,k|k-1,d}(x;z) + v_{D,k|k,d}(x;z)\} \end{aligned} \tag{37}$$

where the first item represents the undetected intensity at time t_d, and the second item represents the detected intensity at time t_d. And intensities $v_{D,k|k-1,d}(x;z)$, $v_{D,k|k,d}(x;z)$ are given by:

$$v_{D,k|k-1,d}(x;z) = \sum_{i=1}^{J_{k|k-1}} \omega_{k|k-1,d}^{(i)} \mathcal{N}(x; m_{k|k-1,d}^{(i)}, P_{k|k-1,d}^{(i)}), \tag{38}$$

$$m_{k|k-1,d}^{(i)} = m_{k|k-1}^{(i)} + K_{d,k-1}^{(i)}(z - H_d m_{d|k-1}^{(i)}), \tag{39}$$

$$P_{k|k-1,d}^{(i)} = P_{k|k-1}^{(i)} - P_{k,d|k-1}^{xz,(i)}(S_{d|k-1}^{(i)})^{-1}(P_{k,d|k-1}^{xz,(i)})^T, \tag{40}$$

$$v_{D,k|k,d}(x;z) = \sum_{i=1}^{|Z_k|J_{k|k-1}} \omega_{k|k,d}^{(i)} \mathcal{N}(x; m_{k|d}^{(i)}, P_{k|d}^{(i)}), \tag{41}$$

$$m^{(i)}_{k|d} = m^{(i)}_{k|k} + K^{(i)}_{d,k}(z - H_d m^{(i)}_{d|k}), \tag{42}$$

$$P^{(i)}_{k|k,d} = P^{(i)}_{k|k} - P^{xz,(i)}_{k,d|k}(S^{(i)}_{d|k})^{-1}(P^{xz,(i)}_{k,d|k})^T, \tag{43}$$

where $K^{(i)}_{d,k-1}$, $S^{(i)}_{d|k-1}$, $K^{(i)}_{d,k}$ and $S^{(i)}_{d|k}$ can be derived from the Equations (13) and (15). The pseudo-code is given in Algorithm 2. And weights $\omega^{(i)}_{k|k-1,d}$, $\omega^{(i)}_{k|k,d}$ are given in the next section.

Algorithm 2 Re-update using OOSM

```
Input the output of backward state prediction $\{w^{(i)}_{d|k-1}, m^{(i)}_{d|k-1}, P^{(i)}_{d|k-1}\}_{i=1}^{J_{k|k-1}} \cup \{w^{(i)}_{d|k}, m^{(i)}_{d|k}, P^{(i)}_{d|k}\}_{i=1}^{|Z_k|J_{k|k-1}}$
Step 1. PHD update components
for $i = 1, 2, ..., J_{k|k-1}$ do
    $P^{xz,(i)}_{k,d|k-1} = (P_{k|k-1} - Q_{k,d})F^T_{d,k}H^T_d$
    $S^{(i)}_{d|k-1} = H_d P^{(i)}_{d|k-1} H^T_d + R_d$
    $K^{(i)}_{k,d|k-1} = P^{xz,(i)}_{k,d|k-1}(S^{(i)}_{d|k-1})^{-1}$
end for
for $i = J_{k|k-1} + 1, ..., |Z_k|J_{k|k-1}$ do
    $P^{xw,(i)}_{k,d|k} = Q_{k,d} - P^{(i)}_{k|k-1}H^T_k(S^{(i)}_k)^{-1}H_k Q_{k,d}$
    $P^{xz,(i)}_{k,d|k} = (P^{(i)}_{k|k} - P^{xw,(i)}_{k,d|k})F^T_{d,k}H^T_d$
    $S^{(i)}_{d|k} = H_d P^{(i)}_{d|k} H^T_d + R_d$
    $K^{(i)}_{k,d|k} = P^{xz,(i)}_{k,d|k}(S^{(i)}_{d|k})^{-1}$
end for
Step 2. Re-update using OOSM
for $l = 0, 1, ..., |Z_d| - 1$ do
    for $j = 1, 2, ..., J_{k|k-1}$ do
        $\omega^{(lJ_{k|k-1}+j)}_{k|k-1,d} = p_{D,d}\omega^{(j)}_{d|k-1}N(z^{(l+1)}; H_d m^{(j)}_{d|k-1}, S^{(j)}_{d|k-1})$
        $m^{(lJ_{k|k-1}+j)}_{k|k-1,d} = m^{(j)}_{k|k-1} + K^{(j)}_{k,d|k-1}(z - H_d m^{(j)}_{d|k-1})$
        $P^{(lJ_{k|k-1}+j)}_{k|k-1,d} = P^{(j)}_{k|k-1} - P^{xz,(j)}_{k,d|k-1}(S^{(j)}_{d|k-1})^{-1}(P^{xz,(j)}_{k,d|k-1})^T$
    end for
    for $j = 1, 2, ..., |Z_k|J_{k|k-1}$ do
        $\omega^{(l|Z_k|J_{k|k-1}+j)}_{k|k,d} = p_{D,d}\omega^{(j)}_{d|k}N(z^{(l+1)}; H_d m^{(j)}_{d|k}, S^{(j)}_{d|k})$
        $m^{(l|Z_k|J_{k|k-1}+j)}_{k|d} = m^{(j)}_{k|k} + K^{(i)}_{k,d|k}(z - H_d m^{(i)}_{d|k})$
        $P^{(l|Z_k|J_{k|k-1}+j)}_{k|d} = P^{(j)}_{k|k} - P^{xz,(j)}_{k,d|k}(S^{(j)}_{d|k})^{-1}(P^{xz,(j)}_{k,d|k})^T$
    end for
end for
Output $\{\omega^{(i)}_{k|k-1,d}, m^{(i)}_{k|k-1,d}, P^{(i)}_{k|k-1,d}\}_{i=1}^{|Z_d|J_{k|k-1}} \cup \{w^{(i)}_{k|k,d}, m^{(i)}_{k|k,d}, P^{(i)}_{k|k,d}\}_{i=1}^{|Z_d||Z_k|J_{k|k-1}}$
```

3.3. Weights Correction

In the above section, we get the intensities that are re-updated with OOSM. These two intensities are re-updated using the same measurement Z_d, which is different from Equation (39). Weights of the intensities need to be corrected, and corrected weights are given as follows:

Prior target:

$$\omega_{k|k-1,d}^{(i)} = \frac{p_{D,k}\omega_{d|k-1}^{(i)} q_d^{(i)}(z)}{\kappa_k(z) + p_{D,k}(\sum_{j=1}^{J_{k|k-1}} \omega_{d|k-1}^{(j)} q_{d|k-1}^{(j)}(z) + \sum_{l=J_{k|k-1}+1}^{|Z_k|J_{k|k-1}} \omega_{k|d}^{(l)} q_{d|k}^{(l)}(z))}. \tag{44}$$

Posterior target:

$$\omega_{k|k,d}^{(i)} = \frac{p_{D,k}\omega_{k|d}^{(i)} q_d^{(i)}(z)}{\kappa_k(z) + p_{D,k}(\sum_{j=1}^{J_{k|k-1}} \omega_{d|k-1}^{(j)} q_{d|k-1}^{(j)}(z) + \sum_{l=J_{k|k-1}+1}^{|Z_k|J_{k|k-1}} \omega_{d|k}^{(l)} q_{d|k}^{(l)}(z))}, \tag{45}$$

where

$$q_{d|k-1}^{(j)}(z) = \mathcal{N}(z; H_d m_{d|k-1}^{(j)}, H_d P_{d|k-1}^{(j)} H_d^T + R_d), \tag{46}$$

$$q_{d|k}^{(l)}(z) = \mathcal{N}(z; H_d m_{d|k}^{(l)}, H_d P_{d|k}^{(l)} H_d^T + R_d). \tag{47}$$

For completeness, the pseudo-code for GM-PHD using OOSM is given in Algorithm 3.

Algorithm 3 Weight correction

```
Input $\{w_{k|k-1}^{(i)}, m_{k|k-1}^{(i)}, P_{k|k-1}^{(i)}\}_{i=1}^{J_{k|k-1}} \cup \{w_{k|k}^{(i)}, m_{k|k}^{(i)}, P_{k|k}^{(i)}\}_{i=1}^{|Z_k|J_{k|k-1}} \cup \{\omega_{k|k-1,d}^{(j)}, m_{k|k-1,d}^{(i)}, P_{k|k-1,d}^{(i)}\}_{i=1}^{|Z_d|J_{k|k-1}} \cup \{w_{k|k,d}^{(i)}, m_{k|k,d}^{(i)}, P_{k|k,d}^{(i)}\}_{i=1}^{|Z_d||Z_k|J_{k|k-1}}$
Step 1. Weight Correction
for $l = 0, 1, ..., |Z_d| - 1$ do
  for $i = 1, 2, ..., J_{k|k-1}$ do
    $\omega_{k|k-1,d}^{(lJ_{k|k-1}+i)} = \frac{\omega_{k|k-1,d}^{(i)}}{\kappa_k(z)+\sum_{i=1}^{|Z_d|J_{k|k-1}} \omega_{k|k-1,d}^{(i)}+\sum_{i=1}^{|Z_k||Z_d|J_{k|k-1}} \omega_{k|k,d}^{(i)}}$
  end for
  for $i = 1, 2, ..., |Z_k|J_{k|k-1}$ do
    $\omega_{k|k,d}^{(l|Z_k|J_{k|k-1}+i)} = \frac{\omega_{k|k,d}^{(i)}}{\kappa_k(z)+\sum_{i=1}^{|Z_d|J_{k|k-1}} \omega_{k|k-1,d}^{(i)}+\sum_{i=1}^{|Z_k||Z_d|J_{k|k-1}} \omega_{k|k,d}^{(i)}}$
  end for
end for
Output $(1 - p_{D,d})^2 w_{k|k-1}^{(i)}, m_{k|k-1}^{(i)}, P_{k|k-1}^{(i)}\}_{i=1}^{J_{k|k-1}} \cup \{(1 - p_{D,d}) w_{k|k}^{(i)}, m_{k|k}^{(i)}, P_{k|k}^{(i)}\}_{i=1}^{|Z_k|J_{k|k-1}} \cup \{\omega_{k|k-1,d}^{(j)}, m_{k|k-1,d}^{(i)}, P_{k|k-1,d}^{(i)}\}_{i=1}^{|Z_d|J_{k|k-1}} \cup \{w_{k|k,d}^{(i)}, m_{k|k,d}^{(i)}, P_{k|k,d}^{(i)}\}_{i=1}^{|Z_d||Z_k|J_{k|k-1}}$
```

Algorithm 4 RFS-based multi-target tracking using OOSM

```
Input
$\{w_k^{(i)}, m_k^{(i)}, P_k^{(i)}\}_{i=1}^{J_{k|k}} = \{w_{k|k-1}^{(i)}, m_{k|k-1}^{(i)}, P_{k|k-1}^{(i)}\}_{i=1}^{J_{k|k-1}} \cup \{w_{k|k}^{(i)}, m_{k|k}^{(i)}, P_{k|k}^{(i)}\}_{i=1}^{|Z_k|J_{k|k-1}}$, and the out of sequence measurement $Z_d$
$i = 0$
Step 1. Backward prediction
for $j = 1, 2, ..., J_{k|k-1}$ do
    $\omega_{d|k-1}^{(j)} = p_{S,d}\omega_{k|k-1}^{(j)}, m_{d|k-1}^{(j)} = F_{d,k}m_{k|k-1}^{(j)}$
    $P_{d|k-1}^{(i)} = F_{d,k}[P_{k|k-1}^{(i)} + P_{k,d|k-1}^{ww,(i)} - P_{k,d|k-1}^{xw,(i)} - (P_{k,d|k-1}^{xw,(i)})^T]F_{d,k}^T$
end for
for $j = 1, ..., |Z_k|J_{k|k-1}$ do
    $\omega_{d|k}^{(j)} = p_{S,d}\omega_{k|k}^{(j)}, m_{d|k}^{(j)} = F_{d,k}m_{k|k}^{(j)}$
    $P_{d|k}^{(i)} = F_{d,k}[P_{k|k}^{(i)} + P_{k,d|k}^{ww,(i)} - P_{k,d|k}^{xw,(i)} - (P_{k,d|k}^{xw,(i)})^T]F_{d,k}^T$
end for
Step 2. PHD update components
for $j = 1, 2, ..., J_{k|k-1}$ do
    $P_{k,d|k-1}^{xz,(i)} = (P_{k|k-1} - Q_{k,d})F_{d,k}^T H_d^T$
    $S_{d|k-1}^{(i)} = H_d P_{d|k-1}^{(i)} H_d^T + R_d$
    $K_{k,d|k-1}^{(i)} = P_{k,d|k-1}^{xz,(i)}(S_{d|k-1}^{(i)})^{-1}$
end for
for $j = J_{k|k-1} + 1, ..., |Z_k|J_{k|k-1}$ do
    $P_{k,d|k}^{xw,(i)} = Q_{k,d} - P_{k|k-1}^{(i)} H_k^T (S_k^{(i)})^{-1} H_k Q_{k,d}$
    $P_{k,d|k}^{xz,(i)} = (P_{k|k}^{(i)} - P_{k,d|k}^{xw,(i)})F_{d,k}^T H_d^T$
    $S_{d|k}^{(i)} = H_d P_{d|k}^{(i)} H_d^T + R_d$
    $K_{k,d|k}^{(i)} = P_{k,d|k}^{xz,(i)}(S_{d|k}^{(i)})^{-1}$
end for
Step 3. Re-update using OOSM
for $l = 0, 1, ..., |Z_d| - 1$ do
    for $j = 1, 2, ..., J_{k|k-1}$ do
        $\omega_{k|k-1,d}^{(lJ_{k|k-1}+j)} = p_{D,d}\omega_{d|k-1}^{(j)} N(z^{(l+1)}; H_d m_{d|k-1}^{(j)}, S_{d|k-1}^{(j)})$
        $m_{k|k-1,d}^{(lJ_{k|k-1}+j)} = m_{k|k-1}^{(j)} + K_{k,d|k-1}^{(j)}(z - H_d m_{d|k-1}^{(j)})$
        $P_{k|k-1,d}^{(lJ_{k|k-1}+j)} = P_{k|k-1}^{(j)} - P_{k,d|k-1}^{xz,(j)}(S_{d|k-1}^{(j)})^{-1}(P_{k,d|k-1}^{xz,(j)})^T$
    end for
    for $j = 1, 2, ..., |Z_k|J_{k|k-1}$ do
        $\omega_{k|k,d}^{(l|Z_k|J_{k|k-1}+j)} = p_{D,d}\omega_{d|k}^{(j)} N(z^{(l+1)}; H_d m_{d|k}^{(j)}, S_{d|k}^{(j)})$
        $m_{k|d}^{(l|Z_k|J_{k|k-1}+j)} = m_{k|k}^{(j)} + K_{k,d|k}^{(i)}(z - H_d m_{d|k}^{(i)})$
        $P_{k|d}^{(l|Z_k|J_{k|k-1}+j)} = P_{k|k}^{(j)} - P_{k,d|k}^{xz,(j)}(S_{d|k}^{(j)})^{-1}(P_{k,d|k}^{xz,(j)})^T$
    end for
end for
Step 4. Weight Correction
for $l = 0, 1, ..., |Z_d| - 1$ do
    for $j = 1, 2, ..., J_{k|k-1}$ do
        $\omega_{k|k-1,d}^{(lJ_{k|k-1}+j)} = \frac{\omega_{k|k-1,d}^{(j)}}{\kappa_k(z) + \sum_{i=1}^{|Z_d|J_{k|k-1}} \omega_{k|k-1,d}^{(i)} + \sum_{i=1}^{|Z_k||Z_d|J_{k|k-1}} \omega_{k|k,d}^{(i)}}$
    end for
    for $j = 1, 2, ..., |Z_k|J_{k|k-1}$ do
        $\omega_{k|k,d}^{(l|Z_k|J_{k|k-1}+j)} = \frac{\omega_{k|k,d}^{(j)}}{\kappa_k(z) + \sum_{i=1}^{|Z_d|J_{k|k-1}} \omega_{k|k-1,d}^{(i)} + \sum_{i=1}^{|Z_k||Z_d|J_{k|k-1}} \omega_{k|k,d}^{(i)}}$
    end for
end for
Output $\{(1 - p_{D,d})w_{k|k-1}^{(i)}, m_{k|k-1}^{(i)}, P_{k|k-1}^{(i)}\}_{i=1}^{J_{k|k-1}} \cup \{(1 - p_{D,d})w_{k|k}^{(i)}, m_{k|k}^{(i)}, P_{k|k}^{(i)}\}_{i=1}^{|Z_k|J_{k|k-1}} \cup \{\omega_{k|k-1,d}^{(j)}, m_{k|k-1,d}^{(i)}, P_{k|k-1,d}^{(i)}\}_{i=1}^{|Z_d|J_{k|k-1}} \cup \{w_{k|k,d}^{(i)}, m_{k|k,d}^{(i)}, P_{k|d}^{(i)}\}_{i=1}^{|Z_d||Z_k|J_{k|k-1}}$
```

4. Simulations

In this section, the distance which is used to measure the performance of multi-target tracking is introduced first. Then another two algorithms are given to compare with the proposed algorithm.

Finally, several simulation scenarios are given to prove the effectiveness of our proposed algorithm. And simulation results show that the proposed GM-PHD using OOSM has almost the same performance as the GM-PHD using in-sequence measurement (ISM).

4.1. Distance

It is also an important issue to measure the tracking performance properly, which directly leads to the choice of the tracking algorithm. Improper criterion will result in tracking accuracy that cannot meet the requirement. Root mean square error (RMSE) is used to measure the distance error between estimate position and real position in a single-target tracking situation. Analogously, in a multi-target tracking scenario, RMSE is also an indicator of tracking performance. However, there is more criterion than that. For example, the accurate rate of the track associate is also an important indicator, but it does not work because the GM-PHD filter based on RFS avoids using the data association technique. Thus, Mahler [32] and Vo [33] proposed three general criteria to measure the effectiveness of multi-target tracking. The essence of these criteria for multi-target tracking performance is to measure the distance between the real state set and estimate state set.

Optimal subpattern assignment (OSPA) distance is an improvement of Wasserstein distance. The definition of OSPA distance is given as follows:

$$\bar{d}_p^{(c)}(X,Y) = \begin{cases} (\frac{1}{n}(\min_{\pi\in\Pi_n}\sum_{i=1}^{|X|} d^{(c)}(x_i, y_{\pi_{(i)}})^p + c^p(|Y|-|X|)))^{\frac{1}{p}}, & m \leq n \\ \bar{d}_p^{(c)}(Y,X), & m > n \\ 0, & m = n = 0 \end{cases}$$

$$\bar{d}_\infty^{(c)}(X,Y) = \begin{cases} 0, & m \leq n \\ \min_{\pi\in\Pi_n}\max_{1\leq i\leq n}\bar{d}^{(c)}(x_i, y_{\pi_{(i)}}), & m = n \\ 0, & m \neq n, \end{cases}$$

where $d^{(c)} = \min(c, ||x-y||)$, $||\cdot||$ represents the Euler norm, c is the truncation distance, $|X|$ is the cardinality of set X, and Π_n represents all permutations of $\{0, 1, \cdots, n\}$. The OSPA distance can be divided into location error and cardinality error which, respectively, correspond to first item and second item of $\bar{d}_p^{(c)}(X,Y)$. Thus, it can evaluate the performance of multi-target filtering more comprehensively. It also serves the shortcomings of Wasserstein distance. And it has definition even when one of the two sets is empty or all the two sets are empty. Parameter c and p affect the weights of cardinality error and location error. How to choose the values of c and p can be refer to [33]. In general, the OSPA distance is currently recognized as an indicator of multi-target filtering.

4.2. Compared Algorithm

To prove the effectiveness of the proposed algorithm, two other algorithms are given here for comparison.

The first is GM-PHD using ISM. Under this circumstance, the order of arrival of two sensors at the fusion center is the same as that of sampling. Then, the sequential fusion of asynchronous measurement based on the GM-PHD filter is performed in fusion center. In another situation, the OOSM is discarded directly. Estimation of targets' states is performed using only in sequential measurements. This situation is called Frame Drop here. The representations of these two situations are shown in Figures 4 and 5.

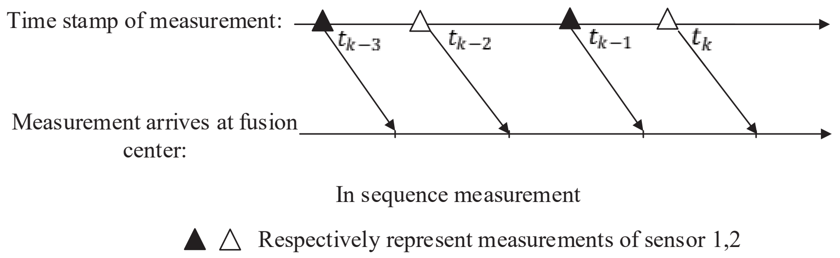

Time stamp of measurement:
t_{k-3}
t_{k-2}
t_{k-1}
t_k
Measurement arrives at fusion center:
In sequence measurement
▲ △ Respectively represent measurements of sensor 1,2

Figure 4. Measurements arrive at fusion center in sequence.

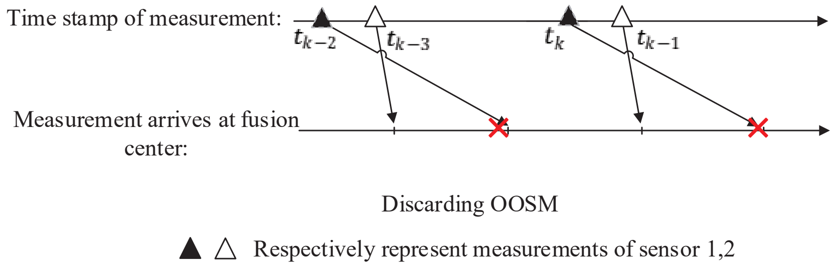

Time stamp of measurement:
t_{k-2}
t_{k-3}
t_k
t_{k-1}
Measurement arrives at fusion center:
Discarding OOSM
▲ △ Respectively represent measurements of sensor 1,2

Figure 5. Discard OOSM when it arrives.

4.3. Simulations Results

The simulation scenario designed here is in the 2-D plane. In the detection area of $[-800, 800]$ m $\times[-800, 800]$ m , there are four targets appearing, all of which adopt the constant velocity (CV) model. We recall the dynamic equation as follows:

$$x_{k+1} = F_k x_k + w_k,$$

when the motion of target is CV, the transition matrix and the covariance of process noise are given as follows, respectively:

$$F_k = \begin{bmatrix} 1 & T & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & T \\ 0 & 0 & 0 & 1 \end{bmatrix}, Q_k = q_v^2 \begin{bmatrix} T^3/3 & T^2/2 & 0 & 0 \\ T^2/2 & T & 0 & 0 \\ 0 & 0 & T^3/3 & T^2/2 \\ 0 & 0 & T^2/2 & T \end{bmatrix}$$

where $q_v = 1(m/s^2)$, T is sampling time, and we set it to 5s. Each target has survival probability $p_{S,k} = 0.99$, and each target is detected with probability $p_{D,k} = 0.98$. The observation model has been given in Equation (3). The measurement matrix and covariance of measurement noise are given as follows:

$$H = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 \end{bmatrix}, R_k = \begin{bmatrix} \sigma_x^2 & 0 \\ 0 & \sigma_x^2 \end{bmatrix}$$

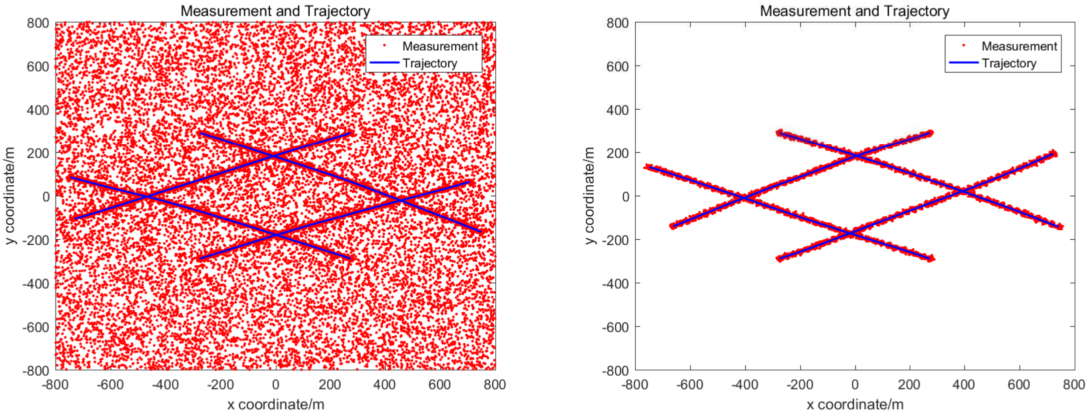
where $\sigma_x = \sigma_y = 5$ m . Measurements are immersed in clutter, and the clutter can be modeled as a Possion RFS K_k with intensity:

$$\kappa_k(z) = \lambda_c V u(z),$$

where $u(z)$ is the uniform density over the surveillance area, V is the volume of the surveillance region, and λ_c is the average number of clutter returns per unit volume.

The algorithm uses the Gaussian component pruning method in the GM-PHD filter. Pruning threshold is $T_w = 10^{-5}$, merging threshold is $U = 4\ m$, and the maximum number of Gaussian component is $J_{max} = 200$. Gaussian component with weights over 0.5 is considered as the estimation of state of target.

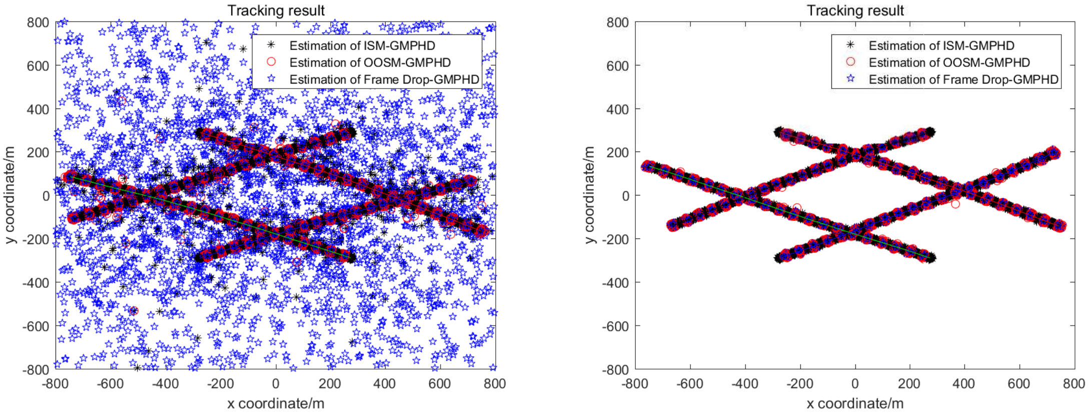
Simulation results are averaged by 100 Monte Carlo experiments. The real trajectory of each simulation is the same while measurements are regenerated each time. The real trajectory of targets and measurements of 100 simulations are shown in Figure 6. The tracking results of three algorithms are shown in Figure 7. It is obvious that the tracking performance of GM-PHD using OOSM and GM-PHD using ISM is nearly the same, while the performance of Frame Drop GM-PHD has decreased significantly.



(a) In the presence of clutter.

(b) In the absence of clutter.

Figure 6. Measurements and trajectories of multi-target.

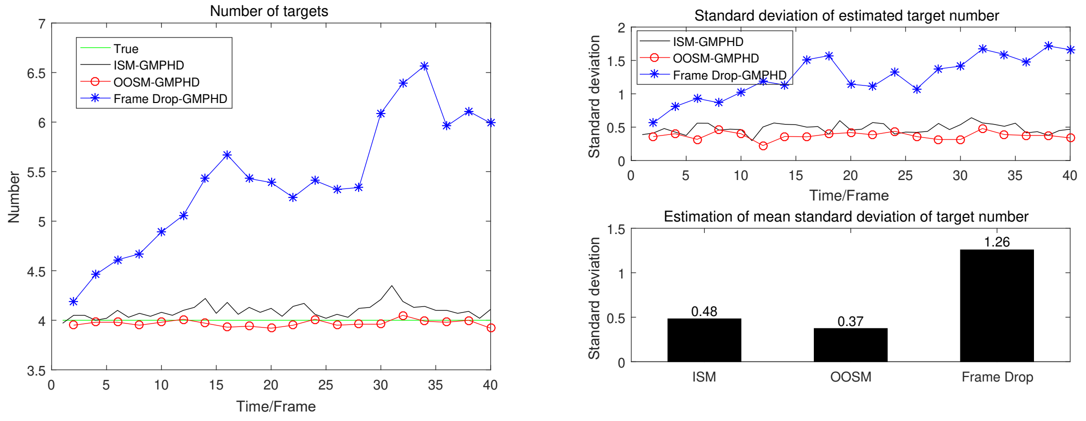


(a) In the presence of clutter.

(b) In the absence of clutter.

Figure 7. Tracking results of multi-target.

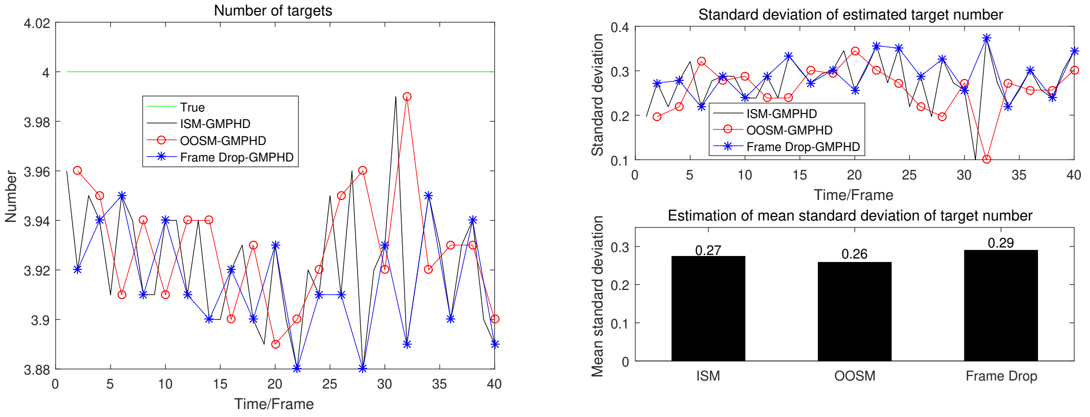
A detailed analysis of the performance of tracking algorithms is given below. The real number of targets at different times and the number of targets estimated by different algorithms in the presence of clutter are given in Figure 8a. And the estimated number and real number in the absence of clutter is given in Figure 9a.



(a) Estimated number and real number of targets.

(b) Standard deviation of estimated number.

Figure 8. Estimated number and standard deviation in the presence of clutter.



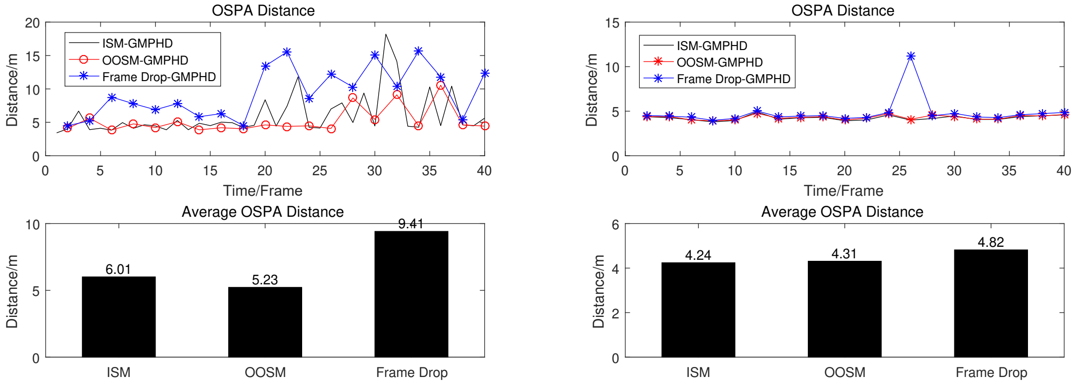
(a) Estimated number and real number of targets.

(b) Standard deviation of estimated number.

Figure 9. Estimated number and standard deviation in the absence of clutter.

From Figure 8a we can see that in the presence of clutter, the result of target number estimation of GM-PHD using OOSM is even more accurate than GM-PHD using ISM. Meanwhile, the estimation of the number of GM-PHD discarding OOSM has a great deviation. And as time goes by, the algorithm even begin to diverge. The more detailed data can be seen in Figure 8b. From Figure 9, we can see that these three algorithms have almost the same performance in the estimation of number.

The simulation results of OSPA distance used to measure the results of multi-target tracking are also given as follows.



(a) Optimal subpattern assignment (OSPA) distance in presence of clutter.

(b) OSPA distance in absence of clutter.

Figure 10. OSPA distance between real state and estimated state.

From Figure 10b, we can see that under OSPA distance criteria, our proposed algorithm which can process OOSM has almost the same performance as that of GM-PHD using ISM in the absence of clutter. However, as we all know, the clutter exists in most tracking scenario. From Figure 10a, we can see that our proposed algorithm using OOSM has better performance than GM-PHD using ISM in the presence of clutter. At the same time, in the presence of clutter situation, the performance of GM-PHD has decreased evidently when OOSMs are discarded directly. It is easy to understand that discarding OOSM directly leads to a reduction of information and consequently results in performance degradation. Compared with GM-PHD using ISM, GM-PHD using OOSM can be seen as doubling the sampling interval while the sampling information of each sample double. Due to targets are all adopting CV model, the maneuverability is not strong. Therefore, compared with the increase of information obtained by each sampling, the increase of sampling interval has little effect on the estimation accuracy. Thus, our proposed algorithm which re-updates intensity using OOSM outperforms GM-PHD using ISM in performance.

Table 1. Tracking performance comparison of different algorithms.

Algorithm **OSPA Distance (m)**	**ISM**	**OOSM**	**Frame Drop**
clutter = 0	4.24	4.31	4.82
clutter = 4	6.01	5.23	9.41

Where clutter = 0 represents there is no clutter in the tracking region, and clutter = 4 represents that the average number of clutter in tracking region is 4 at each sampling time. The detailed data is shown in Table 1 which can adequately demonstrate the effectiveness of our proposed algorithm.

5. Conclusions

The issue of OOSM appears frequently in tracking systems. To better use the OOSM in multi-target tracking scenarios, a novel algorithm was designed in this paper. Firstly, the GM-PHD filter is introduced, and then the B1 algorithm which can solve the OOSM problem in a single-target tracking scenario is also given. Based on this, we used OOSM to re-update the posterior intensity of GM-PHD. Under the framework of the Kalman filter, we derived closed-form recursion for the intensity which is re-updated by the OOSM. Implementation of GM-PHD using OOSM was also given. Finally, several simulations were given to show that the tracking performance of our proposed algorithm is almost the same as that of GM-PHD using ISM in the absence of clutter, and the tracking performance of our proposed algorithm is even better than GM-PHD using ISM in the presence of clutter. This strongly proves the effectiveness of our algorithm.

Author Contributions: Conceive the main idea, M.L. and T.H.; design the simulation, R.Z. and S.Z.; performe the simulation, M.L. and T.H.; write the paper, T.H.

Funding: This work was supported by the National Natural Science Foundation of China (General Program) under Grant 61673345 and the Aeronautical Science Foundation of China under Grant 20172076003.

Conflicts of Interest: The authors declare no conflict of interest.

References

1. Bar-Shalom, Y. *Tracking and Data Association*; Academic Press: Boston, MA, USA, 1987.
2. Blackman, S.S. *Multiple-Target Tracking with Radar Applications;* Artech House, Inc.: Norwood, MA, USA, 1986.
3. Reid, D. An algorithm for tracking multiple targets. *IEEE Trans. Autom. Control* **1979**, *24*, 843–854. [CrossRef]
4. Blackman, S.S. Multiple hypothesis tracking for multiple target tracking. *IEEE Aerosp. Electron. Syst. Mag.* **2004**, *19*, 5–18. [CrossRef]
5. He, S.; Shin, H.S.; Tsourdos, A. Joint probabilistic data association filter with unknown detection probability and clutter rate. *Sensors* **2018**, *18*, 269.
6. Coraluppi, S.; Carthel, C.; Coon, A. An MHT approach to multi-sensor passive sonar tracking. In Proceedings of the 2018 21st International Conference on Information Fusion (FUSION), Cambridge, UK, 10–13 July 2018; pp. 480–487.
7. Chong, C.Y.; Mori, S.; Reid, D.B. Forty years of multiple hypothesis tracking-a review of key developments. In Proceedings of the 2018 21st International Conference on Information Fusion (FUSION), Cambridge, UK, 10–13 July 2018; pp. 452–459.
8. Mahler, R.P. Multitarget Bayes filtering via first-order multitarget moments. *IEEE Trans. Aerosp. Electron. Syst.* **2003**, *39*, 1152–1178. [CrossRef]
9. Goodman, I.R.; Mahler, R.P.; Nguyen, H.T. *Mathematics of Data Fusion*; Springer Science & Business Media: Heidelberg, Germany, 2013; Volume 37.
10. Vo, B.N.; Singh, S.; Doucet, A. Sequential Monte Carlo methods for multitarget filtering with random finite sets. *IEEE Trans. Aerosp. Electron. Syst.* **2005**, *41*, 1224–1245.
11. Vo, B.N.; Ma, W.K. The Gaussian mixture probability hypothesis density filter. *IEEE Trans. Signal Process.* **2006**, *54*, 4091–4104. [CrossRef]
12. Chen, Y.; Sheng, X.; Guo, L.; Li, P.; Hao, H.; Yin, J. Asynchronous Multisonar Data Integration Approach Used to Detect Sparse Targets. *IEEE Access* **2019**, *7*, 37908–37917. [CrossRef]
13. Chen, X.; Li, Y.; Li, Y.; Yu, J. PHD and CPHD Algorithms Based on a Novel Detection Probability Applied in an Active Sonar Tracking System. *Appl. Sci.* **2018**, *8*, 36. [CrossRef]
14. Tanzmeister, G.; Steyer, S. Spatiotemporal alignment for low-level asynchronous data fusion with radar sensors in grid-based tracking and mapping. In Proceedings of the 2016 IEEE International Conference on Multisensor Fusion and Integration for Intelligent Systems (MFI), Baden-Baden, Germany, 19–21 September 2016; pp. 231–237.
15. Hunde, A.; Ayalew, B. Automated multi-target tracking in public traffic in the presence of data association uncertainty. In Proceedings of the 2018 Annual American Control Conference (ACC), Milwaukee, WI, USA, 27–29 June 2018; pp. 300–306.
16. Zhang, Z.; Fu, K.; Sun, X.; Ren, W. Multiple target tracking based on multiple hypotheses tracking and modified ensemble kalman filter in multi-sensor fusion. *Sensors* **2019**, *19*, 3118. [CrossRef]
17. He, S.; Shin, H.S.; Tsourdos, A. Distributed joint probabilistic data association filter with hybrid fusion strategy. *IEEE Trans. Instrum. Meas.* **2019**, 1–15. [CrossRef]
18. Li, G.; Yi, W.; Jiang, M.; Kong, L. Distributed fusion with PHD filter for multi-target tracking in asynchronous radar system. In Proceedings of the 2017 IEEE Radar Conference (RadarConf), Seattle, WA, USA, 8–12 May 2017; pp. 1434–1439.
19. Hamid Rezatofighi, S.; Milan, A.; Zhang, Z.; Shi, Q.; Dick, A.; Reid, I. Joint probabilistic data association revisited. In Proceedings of the IEEE International Conference on Computer Vision, Santiago, Chile, 13–16 December 2015; pp. 3047–3055.
20. Kim, D.Y.; Vo, B.N.; Vo, B.T.; Jeon, M. A labeled random finite set online multi-object tracker for video data. *Pattern Recognit.* **2019**, *90*, 377–389. [CrossRef]

21. Gostar, A.K.; Hoseinnezhad, R.; Bab-Hadiashar, A. Multi-Bernoulli sensor-selection for multi-target tracking with unknown clutter and detection profiles. *Signal Process.* **2016**, *119*, 28–42. [CrossRef]
22. Sheng, X.; Chen, Y.; Guo, L.; Yin, J.; Han, X. Multitarget tracking algorithm using multiple GMPHD filter data fusion for sonar networks. *Sensors* **2018**, *18*, 3193. [CrossRef] [PubMed]
23. Úbeda-Medina, L.; García-Fernández, Á.F.; Grajal, J. Adaptive auxiliary particle filter for track-before-detect with multiple targets. *IEEE Trans. Aerosp. Electron. Syst.* **2017**, *53*, 2317–2330. [CrossRef]
24. Martino, L.; Elvira, V.; Camps-Valls, G. Group Importance Sampling for particle filtering and MCMC. *Digital Signal Process.* **2018**, *82*, 133–151. [CrossRef]
25. Crisan, D.; Miguez, J. Nested particle filters for online parameter estimation in discrete-time state-space Markov models. *Bernoulli* **2018**, *24*, 3039–3086. [CrossRef]
26. Martino, L.; Read, J.; Elvira, V.; Louzada, F. Cooperative parallel particle filters for online model selection and applications to urban mobility. *Digital Signal Process.* **2017**, *60*, 172–185. [CrossRef]
27. Bar-Shalom, Y. Update with out-of-sequence measurements in tracking: exact solution. *IEEE Trans. Aerosp. Electron. Syst.* **2002**, *38*, 769–777. [CrossRef]
28. Blackman, S.; Popoli, R. *Design and Analysis of Modern Tracking Systems (Artech House Radar Library)*; Artech House: Boston, MA, USA, 1999.
29. Bar-Shalom, Y.; Li, X.R. *Multitarget-Multisensor Tracking: Principles and Techniques*; University of Connecticutl: Storrs, CT, USA, 1995.
30. Zhang, S.; Bar-Shalom, Y. Out-of-sequence measurement processing for particle filter: Exact Bayesian solution. *IEEE Trans. Aerosp. Electron. Syst.* **2012**, *48*, 2818–2831. [CrossRef]
31. Li, M.; Yi, W.; Yang, Q.; Kong, L. An Efficient Particle Filter for the OOSM Problem in Nonlinear Dynamic Systems. In Proceedings of the 2018 21st International Conference on Information Fusion (FUSION), Cambridge, UK, 10–13 July 2018; pp. 1890–1895.
32. Hoffman, J.R.; Mahler, R.P. Multitarget miss distance via optimal assignment. *IEEE Trans. Syst. Man Cybern.* **2004**, *34*, 327–336. [CrossRef]
33. Schuhmacher, D.; Vo, B.T.; Vo, B.N. A consistent metric for performance evaluation of multi-object filters. *IEEE Trans. Signal Process.* **2008**, *56*, 3447–3457. [CrossRef]