



## Article

# A Robust TCPHD Filter for Multi-Sensor Multitarget Tracking Based on a Gaussian–Student’s t-Mixture Model

Shaoming Wei <sup>1</sup>, Yingbin Lin <sup>1</sup>, Jun Wang <sup>1,2,\*</sup>, Yajun Zeng <sup>1</sup>, Fangrui Qu <sup>1</sup>, Xuan Zhou <sup>1</sup> and Zhuotong Lu <sup>1</sup>

<sup>1</sup> School of Electronics and Information Engineering, Beihang University, Beijing 100191, China; shaoming.wei@buaa.edu.cn (S.W.); sy2102418@buaa.edu.cn (Y.L.); by1902004@buaa.edu.cn (Y.Z.); 20373482@buaa.edu.cn (F.Q.); xuanz@buaa.edu.cn (X.Z.); zy2302303@buaa.edu.cn (Z.L.)

<sup>2</sup> Hangzhou Innovation Institute, Beihang University, Hangzhou 310000, China

\* Correspondence: wangj203@buaa.edu.cn; Tel.: +86-10-8233-9203

**Abstract:** To realize multitarget trajectory tracking under non-Gaussian heavy-tailed noise, we propose a Gaussian–Student t-mixture distribution-based trajectory cardinality probability hypothesis density filter (GSTM-TCPHD). We introduce the multi-sensor GSTM-TCPHD (MS-GSTM-TCPHD) filter to enhance tracking performance. Conventional cardinality probability hypothesis density (CPHD) filters typically assume Gaussian noise and struggle to accurately establish target trajectories when faced with heavy-tailed non-Gaussian distributions. Heavy-tailed noise leads to significant estimation errors and filter dispersion. Moreover, the exact trajectory of the target is crucial for tracking and prediction. Our proposed GSTM-TCPHD filter utilizes the GSTM distribution to model heavy-tailed noise, reducing modeling errors and generating a set of potential target trajectories. Since single sensors have a limited field of view and limited measurement information, we extend the filter to a multi-sensor scenario. To tackle the issue of data explosion from multiple sensors, we employed a greedy approximation method to assess measurements and introduced the MS-GSTM-TCPHD filter. The simulation results demonstrate that our proposed filter outperforms the CPHD/TCPHD filter and Student’s t-based TCPHD filter in terms of accurately estimating the trajectories of multiple targets during tracking while also achieving improved accuracy and shorter processing time.

**Keywords:** TCPHD filter; heavy-tailed noise; multitarget tracking; multi-sensor



**Citation:** Wei, S.; Lin, Y.; Wang, J.; Zeng, Y.; Qu, F.; Zhou, X.; Lu, Z. A Robust TCPHD Filter for Multi-Sensor Multitarget Tracking Based on a Gaussian–Student’s t-Mixture Model. *Remote Sens.* **2024**, *16*, 506. <https://doi.org/10.3390/rs16030506>

Academic Editor: Andrzej Stateczny

Received: 12 December 2023

Revised: 18 January 2024

Accepted: 26 January 2024

Published: 28 January 2024



**Copyright:** © 2024 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (<https://creativecommons.org/licenses/by/4.0/>).

## 1. Introduction

### 1.1. Literature Review and Motivation

Target tracking is widely used in intelligent driving, robot identification, and defense. The observation platform employs an assortment of sensors, including millimeter-wave radar, lidar, and camera, to acquire data; applies filters to preserve pertinent information; and ultimately estimates the target’s state information [1].

Conventional techniques for multitarget tracking rely on data association; these include nearest-neighbor association, multi-hypothesis tracking (MHT), and joint probabilistic data association [2]. However, these algorithms are susceptible to combinatorial explosion because of the data association concept. Random finite set (RFS) theory can simultaneously reduce correlation operations and has a solid theoretical demonstration inside the Bayesian framework. RFS filters commonly used today include probabilistic hypothesis density (PHD), cardinality PHD (CPHD) [3], multi-objective multi-Bernoulli (MeMBer), and generalized labeled multi-Bernoulli (GLMB) [4]. Given the GLMB filter’s numerous prediction and update processes, it demands a lot of processing power. The PHD filter uses fewer computing resources [5]. In recent years, trajectory cardinality probability hypothesis density (TCPHD) filters have emerged to address this shortcoming by enabling trajectory generation based on CPHD [6], which provides trajectory representation with superior estimation accuracy and computational complexity. The TCPHD filter addresses the issue of the standard CPHD filter’s inability to infer trajectories by utilizing the set of trajectories

in the state variables instead of the set of targets. It employs IID-clustered multi-trajectory densities for the prediction and update steps, followed by KLD minimization after the update step. Compared with previous trajectory modeling methods used in CPHD filters, the TCPHD filter has a better theoretical foundation and trajectory estimation performance. It is also capable of updating the past state information of the trajectory [7]. The performance of filters dominated by Gaussian noise may rapidly deteriorate in the presence of outliers and non-Gaussian noise. This leads to poorer estimates due to model mismatches [8]. Therefore, the mainstream trend is to replace Gaussian models with robust estimators.

The Kalman filter is a technique that works under the assumption of Gaussian noise and tries to minimize root-mean-square errors. However, in practice, non-Gaussian noise is more common and frequently exhibits heavy-tailed characteristics that depart from the Gaussian distribution [9]. Non-Gaussian heavy-tailed noise is caused by the sensor's performance becoming unstable under intense target motion [10]. When the target moves rapidly, the state displays heavy-tailed non-Gaussian noise [11]; changes in the detected signal's reflection cause non-Gaussian heavy-tailed noise in the measurements [12]. Non-Gaussian heavy-tailed noise causes significant reductions in tracking performance and an increase in model mismatch, filter dispersion, and prior filter noise covariance modeling errors.

Research on target tracking with non-Gaussian noise can be divided into two categories: when the measurement noise is non-Gaussian and when both the process and measurement noise are non-Gaussian [13]. There are three mainstream methods for dealing with process noise, regardless of whether it is Gaussian or not: variational Bayes (VB) inference [10], the sequential Monte Carlo method [14], and Student's t-distribution Kalman filtering [15]. The VB inference method is an adaptive Kalman filtering algorithm that approximates the target state via VB inference by introducing auxiliary noise parameters. For instance, the covariance of the observed noise is subjected to an inverse gamma distribution or an inverse Weiser distribution, and the variational Bayesian method is used to jointly estimate the parameters of the target. The sequential Monte Carlo method utilizes particle filtering to estimate measurement noise outliers. It employs weighted particle approximation to represent the posterior probability of the target. However, it requires a sufficient number of particles, resulting in a significant increase in computational complexity and low efficiency in high-dimensional space. The Kalman filtering method used by Student's t-distribution model incorporates noise with heavy tail characteristics. Various nonlinear Kalman-filtering algorithms based on Student's t-distribution have been proposed, depending on the integration operation [16]. Although Student's t-distribution provides good tracking performance, it requires fixed-point iteration to calculate various parameters in variational Bayesian inference, which increases computational complexity. Other processing methods, such as Huber Kalman filtering and maximum correlation entropy criterion Kalman filtering, are also used by a small number of researchers [17–20]. In recent years, the GSTM distribution has maintained its characteristic of modeling outlier noise using Student's t-distribution while also utilizing the Gaussian distribution to improve computational efficiency [20]. Further research is required to investigate the application of the GSTM distribution in target tracking with non-Gaussian noise.

Currently, there is limited research on outlier noise in RFS multitarget tracking. Two types of approaches are commonly used: Student's t-distribution PHD filter and variational Bayesian inference. For the PHD filter method based on Student's t-distribution [21], Liu Z proposed a PHD filter algorithm, modeling heavy-tailed noise with Student's t-distribution, simultaneously expressing the same posterior probability density distribution as Student's t-distribution alone. Later, the method was extended to nonlinear scenarios [22]. For the multitarget tracking method based on variational Bayesian approximation, Yang J and Li W proposed using inverse gamma distribution to represent the variance in measurement noise and expressed the posterior probability distribution as inverse gamma distribution, used the VB method to estimate parameters, and proposed the implementation of PHD [23]. Yang J L also proposed a CBMeMber filter based on VB approximation [24]. Li W proposed modeling flicker noise with Student's t-distribution and provided a mixed implementation of a

Gaussian gamma distribution of the PHD filter [25]. The use of PHD filters in multitarget tracking has inherent drawbacks, such as the potential for large estimation errors and the inability to form trajectories. However, the TCPHD filter, which is based on CPHD, compensates for these drawbacks and has the added advantage of forming trajectories. This makes it a useful tool for multitarget tracking, particularly in the presence of outlier noise.

Tracking based on single sensors has the problems of limited field of view, measurement information, and instability, so the data fusion technology used in a multi-sensor is crucial [26]. Mahler first proposed the iterated corrector PHD (IC-PHD) filter, which sequentially processes information from different sensors [27]. In 2009, Mahler introduced the concept of the General PHD filter (G-PHD) for multi-sensor systems, and in 2015, Nannuru derived a rigorous theoretical proof and proposed a General CPHD filter [28]. In 2010, Mahler proposed the product multi-sensor PHD (PM-PHD) filter and the product multi-sensor CPHD (PM-CPHD) filter [29].

In the algorithm for multi-sensor multitarget tracking based on random finite sets, measurement partitioning is a crucial step. The algorithm for greedy approximation obtains the optimal measurement set for each target in sensor order while retaining the best measurement combination. Compared with Gibbs sampling, which assumes a finite number of measurements with higher weights as the final measurement partition result, the greedy partitioning method aims to preserve the measurement set that contains missed items when obtaining the measurement set for each target [30]. This ensures that the measurement partitioning results are in line with reality and are not affected by small likelihood weights, resulting in better tracking results. The greedy approximation algorithm is significant in multi-sensor and multitarget tracking scenarios where there is low probability and high maneuverability.

Based on the above research, multi-sensor multitarget tracking under heavy-tailed noise has important research significance and various application scenarios. The three key challenges include (1) generating the trajectory of many targets; (2) modeling non-Gaussian heavy-tailed noise to engage in iterative filter tracking [31]; and (3) leveraging the measurement data from multiple sensors to enhance tracking performance.

### 1.2. Our Contributions

The multi-sensor multitarget tracking problem under heavy-tailed noise has significant scientific value and promising application opportunities, according to the paper mentioned above. Consequently, we proposed the GSTM-TCPHD and MS-GSTM-TCPHD filters. The following are some of our approach's primary innovations:

In addition to introducing some related multitarget tracking contents based on a random finite set, this study proposes a new multitarget trajectory tracking method under non-Gaussian noise, which uses GSTM to model non-Gaussian noise and implements multitarget tracking based on the TCPHD filter. Simultaneously, a new multi-sensor multitarget tracking method is introduced based on the GSTM-TCPHD filter. This method generates trajectories under non-Gaussian noise, and the implementation of the MS-GSTM-TCPHD filter is provided. Then, the proposed filter is simulated and evaluated to verify its tracking performance.

### 1.3. Organization of the Article

This article is structured as follows: Section 2 provides background knowledge. Section 3 presents a theoretical derivation process for the GSTM-TCPHD filter. Section 4 shows the implementation of the filter. Section 5 introduces the implementation of the MS-GSTM-TCPHD filter. In Section 6, the proposed method is validated, and the results of the simulation run are demonstrated and analyzed. Section 7 is the conclusion of this article.

## 2. Background

### 2.1. Notation

Assume that the variable  $X = (t, x^{1:l})$  with  $x \in R^{n_x}$  is represented as a single trajectory variable, where

$t$  denotes the starting time;

$n_x$  is the dimension of the target state,  $x$ ;

$x^{1:l} = (x^1, \dots, x^l)$  denotes a set of target state sequences of length  $l$ .

At time  $k$ , the length of the trajectory is  $t + l - 1$ ; let  $I_k = \{(t, i) : 0 \leq t \leq k, 1 \leq i \leq k - t + 1\}$ . The trajectory space is represented as

$$\mathbb{T}_k = \uplus_{(t,i) \in I_k} \{t\} \times R^{i n_x} \tag{1}$$

where  $\uplus$  denotes the mutually exclusive set.

The binomial coefficients, permutation coefficients, and generalized Kronecker delta function are expressed as follows:

$$C_m^n = \frac{n!}{m!(n-m)!}, P_j^n = \frac{n!}{(n-j)!}, \delta_A[B] \triangleq \begin{cases} 1, & \text{if } A = B \\ 0, & \text{otherwise} \end{cases} \tag{2}$$

The inner product of two real-valued functions is expressed as follows:

$$\langle a, b \rangle = \int a(x)b(x)dx \tag{3}$$

A  $j$ -th order primitive symmetric function of a finite set,  $\mathcal{X}$ , of real numbers is expressed as follows:

$$e_j(\mathcal{X}) = \sum_{\sigma \subseteq \mathcal{X}, |\sigma|=j} \prod_{\xi \in \sigma} \xi \tag{4}$$

If  $p$  is the observed probability distribution function and  $q$  is used to approximate it [32], then the Kullback–Leibler divergence (KLD) between the two distributions can be calculated as follows:

$$D_{KL}(p||q) = \sum_{i=1}^N p(x_i) \cdot (\log p(x_i) - \log q(x_i)) = \sum_{i=1}^N p(x_i) \cdot \log \frac{p(x_i)}{q(x_i)} \tag{5}$$

### 2.2. Set Integral

Suppose that, in the trajectory space, the single trajectory variable,  $\pi(X)$ , and the trajectory RFS have the following representation:

$$\int_{\mathbb{T}} \pi(X)dX = \sum_{(t,i) \in I(k)} \int \pi(t, x^{1:i}) dx^{1:i} \tag{6}$$

$$\int \pi(\mathbf{X})\delta\mathbf{X} = \sum_{n=0}^{\infty} \frac{1}{n!} \int \pi(\{X_1, \dots, X_n\})dX_{1:n} \tag{7}$$

where

$\mathbf{X}_k = \{X_1, \dots, X_{N^k}\} \in \mathcal{F}(\mathbb{T}_k)$ ;

$\mathcal{F}(\mathbb{T}_k)$  denotes the set of all finite subsets in the trajectory space.

The cardinality distribution and PHD of the multi-trajectory density,  $\pi(\mathbf{X})$ , are expressed as

$$\rho_{\pi}(n) = \frac{1}{n!} \int \pi(\{X_1, \dots, X_n\})dX_{1:n} \tag{8}$$

$$D_{\pi}(X) = \int \pi(\{X\} \cup \mathbf{X})\delta\mathbf{X} \tag{9}$$

Assuming that there is a Poisson multi-trajectory density,  $\pi(\mathbf{X})$ , at time  $k$ , the target's multitarget density,  $\pi_k^T$ , is Poisson, and its PHD can be expressed as follows:

$$D_{\pi_k^T}(y) = \sum_{t=1}^k \int D_{\pi}(t, x^{1:k-t}, y) dx^{1:k-t} \tag{10}$$

### 2.3. TPHD and TCPHD Filter

The TPHD filter propagates a Poisson multi-trajectory density, its multi-trajectory density is expressed as follows:

$$p_k(\{X_1, \dots, X_{N^k}\}) = e^{-\lambda_k} \lambda_k^n \prod_{j=1}^{N^k} \dot{p}_k(X_j) \tag{11}$$

where  $\lambda_k$  represents the expected number of trajectories.

The TCPHD filter propagates IID cluster Poisson multi-trajectory density; its multi-trajectory density is expressed as follows:

$$p_k(\{X_1, \dots, X_{N^k}\}) = \rho_k(n) n! \prod_{j=1}^{N^k} \dot{p}_k(X_j) \tag{12}$$

where

$\dot{p}(X_j)$  is the density of single trajectories;

$\rho(n)$  is the cardinality distribution.

When the number of sets is distributed according to a Poisson distribution, the IID cluster random finite set degenerates into a Poisson random finite set. The PHD of the multi-trajectory densities of the Poisson RFS and the IID cluster RFS are

$$D_k(X) = \lambda_k \dot{p}_k(X), D_k(X) = \dot{p}_k(X) \sum_{n=0}^{\infty} n \rho_k(n) \tag{13}$$

At time  $k - 1$ , the density of multiple trajectories in the state space,  $\mathbf{X}_{k-1} = \{X_1, \dots, X_{N^{k-1}}\} \in \mathcal{F}(\mathbb{T}_{k-1})$ , and the density of multiple trajectories in the measurement space,  $Z_{k-1} = \{z_1, \dots, z_{M^{k-1}}\} \in \mathcal{F}(\mathbb{Z}_{k-1})$ , result in a Bayesian recursive representation as follows [3]

$$\pi_{k|k-1}(\mathbf{X}_k) = \int f(\mathbf{X}_k|\mathbf{X}_{k-1}) \pi_{k-1}(\mathbf{X}_{k-1}) \delta \mathbf{X}_{k-1} \tag{14}$$

$$\pi_k(\mathbf{X}_k) = \frac{\ell_k(Z_k|\mathbf{X}_k) \pi_{k|k-1}(\mathbf{X}_k)}{\int \ell_k(Z_k|\mathbf{X}_k) \pi_{k|k-1}(\mathbf{X}_k) \delta \mathbf{X}_k} \tag{15}$$

where

$f(\mathbf{X}_k|\mathbf{X}_{k-1})$  denotes the transfer density of the trajectory;

$\pi_{k|k-1}(\mathbf{X}_k)$  denotes the predicted multi-trajectory density at time  $k$ ;

$\ell_k(Z_k|\mathbf{X}_k)$  denotes the quantile likelihood function.

### 2.4. Multi-Sensor Observation Model

At time  $k$ , assuming the existence of  $s$  sensors, the set of measurements is denoted as  $Z_k^{1:s} = \{Z_k^1, \dots, Z_k^s\}$ . Let  $W \subseteq Z_{k+1}^{1:s}, |W|_j \leq 1, |W|_j = \left| \left\{ z \in W : z \in Z_{k+1}^j \right\} \right|$ , denoting that the subset  $W$  takes, at most, one measurement from the set of measurements of each sensor, and all belong to the same PHD component;  $\Omega$  denotes the set of all measure subsets of  $W$ . For a mutually exclusive subset of  $W$ , let  $V = Z_{k+1}^{1:s} \setminus (\cup_{i=1}^n W_i)$  represents the set of clutters generated by all sensor measurements [33]. The set of measurements can be divided into

$$P = \{W_1, W_2, \dots, W_{|P|-1}, V\} \quad (16)$$

$$\bigcup_{i=1}^{|P|-1} W_i \cup V = Z_{k+1}^{1:s}, W_i \cap W_j = \emptyset, W_i \cap V = \emptyset, W_i, W_j \in P, i \neq j \quad (17)$$

where

$P$  denotes the set of divisions;

$|P|$  denotes the number of partitions,  $P$ ;

$|P|_j$  is the measurement produced by the sensor,  $j$ , on the target and  $|P|_j = \sum_{i=1}^{|P|-1} |W_i|_j$ .

The number of measurements generated in sensor  $j$  that are considered to be clutter is  $m_j - |P|_j$ , and  $m_j$  is the number of measurements generated in the  $j$ -th sensor.

### 3. Proposed GSTM-TCPHD Filters

Compared with the CPHD filter, the TCPHD filter has a wider range of scenarios given its ability to track multiple targets and produce target trajectories; however, under non-Gaussian heavy-tailed noise, it lacks relevant theory [34]. In this section, we introduce the GSTM-TCPHD filter and provide a theoretical derivation based on the L-scan TPHPD and TCPHD filters [35].

#### 3.1. GSTM Distribution

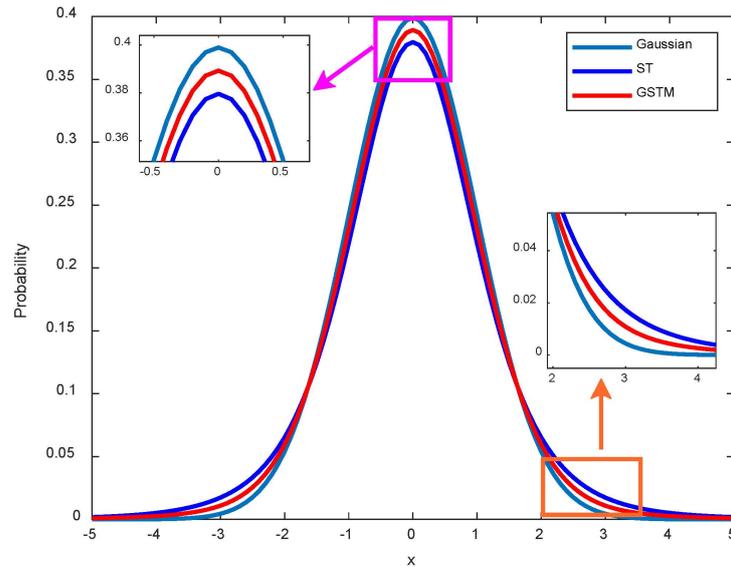
While non-Gaussian models are occasionally seen, Gaussian models account for the majority of noise [36]. Heavy tails are a defining feature of Student's t-distribution, better simulating the heavy tail phenomenon [37]. Assume that the random variable,  $x$ , follows Student's t-distribution with a mean,  $\mu$ ; a scale matrix,  $\Sigma$ ; and degrees of freedom,  $\nu$  [38], and denote its probability density function as  $St(x; \mu, \Sigma, \nu)$ . The GSTM distribution adaptively learns the mixture of Student's t-distribution and Gaussian probabilities to adjust to time-varying state and measurement noise.

Assume that the GSTM distribution with mixing probability,  $e$ , is represented as follows:

$$p(x) = eN(x; \mu, \Sigma) + (1 - e)St(x; \mu, \Sigma, \nu) \quad (18)$$

In this paper,  $Gst(x; \mu, \Sigma, \nu, e)$  is used to describe the Gaussian–Student's t-mixture distribution.

The heavy-tailed characteristic of Student's t-distribution is modulated by a degrees-of-freedom parameter, and the thickness on either side of the probability curve is correlated with the degrees of freedom. As the degree of freedom tends toward a Gaussian distribution, the more critical the heavy tail, and the smaller the heavy tail, the larger the degree of freedom [39]. The GSTM distribution uses the mixture probability parameter to adaptively adjust the probability of Student's t-distribution and the Gaussian distribution to match the true noise distribution, as shown in Figure 1. Figure 1 shows that there are differences between the three distributions near the center point. As the distance range increases, the difference between the three distributions narrows and then increases. This suggests that the GSTM distribution may perform better than the Gaussian distribution and Student's t-distribution, regardless of whether the situation is normal or abnormal.



**Figure 1.** Noise simulation distribution comparison.

### 3.2. Implementation of GSTM-TCPHD Filter

At time  $k - 1$ , the multitarget posterior probability hypothesis density,  $D_{k-1}(x)$ , and the posterior potential distribution,  $\rho_{k-1}$ , are known. The GSTM-TCPHD filter is predicted and updated as follows [40].

We make the following assumptions in the prediction step:

P1: Each target survives with probability  $p_S(x) = p_S$ ; moves to a new state with a transition density of  $g(\cdot)$ ; or dies with probability  $1 - p_S$ .

P2: The multitarget state at the next time step is the union of the surviving targets and new targets, which are born independently with a Poisson multitarget density,  $\beta(\cdot)$ .

P3: The state transition is represented as  $g(x^k | x^{k-1}) = Gst(x_k; Fx_{k-1}, Q, v_1, \tau)$ .

At time  $k - 1$ , multi-trajectory PHD and cardinality distribution are as follows:

$$D_{k|k-1}(X) = p_S(x^{i-1})f(x^i | x^{i-1})D_{k-1}(t, x^{1:i-1}) + \beta_k(X) \tag{19}$$

$$\rho_{k|k-1}(n) = \sum_{j=0}^n \rho_{\beta,k}(n-j) \sum_{l=j}^{\infty} C_l^j \rho_{k-1}(l) \frac{\langle p_S, D_{k-1}^\tau \rangle^j \langle 1 - p_S, D_{k-1}^\tau \rangle^{l-j}}{\langle 1, D_{k-1}^\tau \rangle^l} \tag{20}$$

where

$$D_{k|k-1}^\tau(x^i) = \sum_{t=1}^k \int D_{k|k-1}(t, x^{1:k-t+1}) dx^{1:k-t} \tag{21}$$

Assuming that the predicted TPHD,  $D_{k|k-1}(x)$ , and cardinality distribution,  $p_{k|k-1}$ , are known, the following assumptions are made in the update step [41].

U1: The probability of detection for each target is  $p_D$ , and each detection produces a measurement with density  $l(\cdot|x)$ , or the probability of missing a detection is  $1 - p_D(x)$ .

U2: Measurement  $z_k$  is the union of the target-generated measurements with the IID cluster density,  $c(\cdot)$ .

U3: The quantile likelihood function is expressed as  $l(z|x) = Gst(z; Hx, R, v_2, \pi)$ .

U4: The multi-trajectory density,  $\omega_k(\cdot)$ , is the IID cluster, RFS.

Given the predicted cardinality distribution,  $\rho_{k|k-1}(n)$ , and TPHD,  $D_{k-1}(t, x^{1:i-1})$ , at moment  $k$ , the update step of the TCPHD filter is as follows:

$$\rho_k(n) = \frac{Y_k^0 [D_{k|k-1}^\tau; Z_k] (n) \rho_{k|k-1}(n)}{\langle Y_k^0 [D_{k|k-1}^\tau; Z_k], \rho_{k|k-1} \rangle}, \tag{22}$$

$$D_k(X) = D_{k|k-1}(X) q_D(x^i) \times \frac{\langle Y_k^1 [D_{k|k-1}^\tau; Z_k], \rho_{k|k-1} \rangle}{\langle Y_k^0 [D_{k|k-1}^\tau; Z_k], \rho_{k|k-1} \rangle} + D_{k|k-1}(X) p_D(x^i) \times \sum_{z \in Z_k} \frac{l_k(z|x^i)}{\bar{c}(z)} \frac{\langle Y_k^1 [D_{k|k-1}^\tau; Z_k \setminus \{z\}], \rho_{k|k-1} \rangle}{\langle Y_k^0 [D_{k|k-1}^\tau; Z_k], \rho_{k|k-1} \rangle} \tag{23}$$

where

$q_{D,k}$  denotes the probability missing;  
 $\bar{c}(z)$  is the clutter density.

$$Y^u [D_{k|k-1}^\tau, Z_k] (n) = \sum_{j=0}^{\min(M^k, n-u)} (M^k - j)! \rho_c(M^k - j) \times P_{j+u}^n \frac{\langle q_D, D_{k|k-1}^\tau \rangle^{n-j-u}}{\langle 1, D_{k|k-1}^\tau \rangle^n} e_j(\Xi(D_{k|k-1}^\tau, Z_k)) \tag{24}$$

$$\Xi(D_{k|k-1}^\tau, Z_k) = \left\{ \int p_{D,k}(x^i) \frac{l_k(z|x^i)}{\bar{c}(z)} D_{k|k-1}^\tau(x^i) dx^i : z \in Z_k \right\} \tag{25}$$

### 3.3. VB Approximation for Posterior Probabilities

The GSTM state model’s parameters are more difficult to calculate given their inter-dependence. By breaking down the multivariate variables into products of individual variables, VB approximation is used to approximate the posterior probability density and provide an analytical solution to the posterior probability. Figure 2 displays the filter’s flowchart.

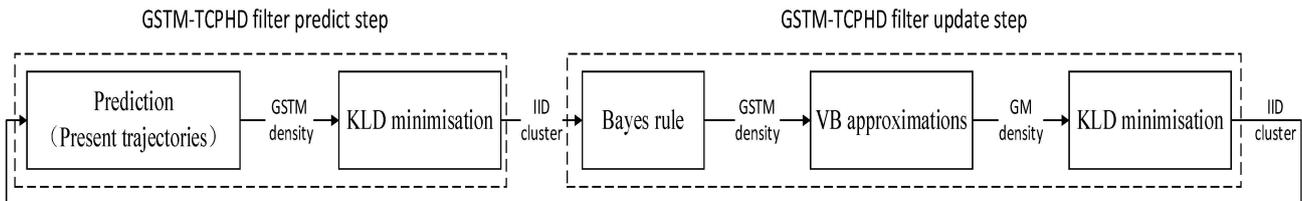


Figure 2. Filter flow block diagram.

The VB method uses an approximation of simply computed joint density,  $q(X, \theta)$ , to approximate the true joint a posteriori probability density,  $q(X, \theta|Z)$ , while considering the parameters to be independent of each other, computed as  $q(X, \theta) = q(X)q(\theta)$ . The errors on both sides of the equation are judged by the KL distance.

VB approximation of the solution equation for the posterior probability is expressed as follows:

$$\log q(\varphi) = E_{\theta^{(-\varphi)}} [\log p(X, \theta, Z)] + c_\varphi, \text{ s.t. } \int q(\varphi) d\varphi = 1 \tag{26}$$

where

- $\theta$  denotes the set consisting of state vectors and unknown parameters;
- $\varphi \in \theta$  is denoted as any element of set  $\theta$ ;
- $\theta^{(-\varphi)}$  denotes the elements remaining in set  $\theta$  after the removal of element  $\varphi$ ;
- $q(\varphi)$  denotes the approximate a posteriori density of  $\varphi$ ;

$E_x$  denotes the expectation of variable  $x$ ;  
 $c_\varphi$  represents a constant independent of  $\varphi$ .

The GSTM distribution models the noise with the parameter  $\{\xi_k, t_k, \tau_k\}$  describing the state noise and the parameter  $\{\lambda_k, y_k, \pi_k\}$  describing the measurement noise. Its joint probability density is as follows:

$$p(X, \theta, Z) = p(Z)g(x^k|x^{k-1})l(z|x) \tag{27}$$

The VB approximation to the parameter decomposition is expressed as follows:

$$q(X, \theta|Z) = q(x)q(\xi)q(t)q(\tau)q(\lambda)q(y)q(\pi) \tag{28}$$

#### 4. GSTM-TCPHD Filter Implementation under Single Sensor

The GSTM distribution is used to model the noise. In cases where the state noise and measurement noise in the GM-TCPHD filter mixture are non-Gaussian heavy-tailed distributions, the VB approximation is solved to update the state to a Gaussian mixture [42].

##### 4.1. Predict Process

At time  $k - 1$ , the PHD representation of a priori multi-trajectory density,  $\pi^{k-1}(\cdot)$ , is as follows

$$D_{\pi^{k-1}}(X) = \sum_{j=1}^{j^{k-1}} w_j^{k-1} \mathcal{N}(X; t_j^{k-1}, m_j^{k-1}, P_j^{k-1}) \tag{29}$$

where

$$t_j^{k-1} + i_j^{k-1} - 1 = k - 1;$$

$$i_j^{k-1} = \dim(m_j^{k-1})/n_x.$$

The PHD of the birth density,  $\beta^k$ , is

$$D_{\beta^k}(X) = \sum_{j=1}^{j_\beta^k} w_{\beta,j}^k \mathcal{N}(X; k, m_{\beta,j}^k, P_{\beta,j}^k) \tag{30}$$

where

$j_\beta^k \in \mathbb{N}$  is the number of components;

$w_{\beta,j}^k$  is the weight of the  $j$ -th component;

$m_{\beta,j}^k, P_{\beta,j}^k$  is the mean and covariance matrix.

The TPHPD is predicted as follows:

$$D_{S,k|k-1} = D_{\beta^k}(X) + p_S \sum_{j=1}^{j^{k-1}} w_j^{k-1} \mathcal{N}(x_{k|k-1}; t_j^{k-1}, m_{\omega,j}^k, P_{\omega,j}^k) \tag{31}$$

where

$$\dot{F}_j = \begin{bmatrix} 0_{1, i_j^{k-1}-1} & 1 \end{bmatrix} \otimes F \tag{32}$$

$$m_{\omega,j}^k = \left[ \left( m_j^{k-1} \right)^T, \left( \dot{F}_j m_j^{k-1} \right)^T \right]^T \tag{33}$$

$$P_{\omega,j}^k = \begin{bmatrix} P_j^{k-1} & P_j^{k-1} \dot{F}_j^T \\ \dot{F}_j P_j^{k-1} & \dot{F}_j P_j^{k-1} \dot{F}_j^T + Q \end{bmatrix} \tag{34}$$

Cardinality is predicted as follows:

$$\rho_{\omega^k}(m) = \sum_{j=0}^m \rho_{\beta^k}(m-j) \times \sum_{l=j}^{\infty} \binom{n}{j} \rho_{\pi^{k-1}}(n)(1-p_S)^{l-j} p_S^j \quad (35)$$

#### 4.2. Update Process

Suppose that the TPHD representation of the a priori density,  $\omega^k$ , in the update step is as follows:

$$D_{\omega^k}(X) = \sum_{j=1}^{J_{\omega}^k} w_{\omega,j}^k \mathcal{N}(X; t_k^{\omega,j}, m_k^{\omega,j}, P_k^{\omega,j}) \quad (36)$$

updated to

$$D_{\pi^k}(X) = \frac{\langle \Psi^1[w_{\omega}^k, \mathbf{z}^k], \rho_{\omega^k} \rangle}{\langle \Psi^0[w_{\omega}^k, \mathbf{z}^k], \rho_{\omega^k} \rangle} (1-p_D) D_{\omega^k}(X) + \sum_{z \in \mathbf{z}^k} \sum_{j=1}^{J_{\omega}^k} w_j(z) \mathcal{N}(X; t_k^{\omega,j}, m_j^k(z), P_j^k) \quad (37)$$

$$\rho_{\pi^k}(n) = \frac{\Psi^0[w_{\omega}^k, \mathbf{z}^k](n) \rho_{\omega^k}(n)}{\langle \Psi^0[w_{\omega}^k, \mathbf{z}^k], \rho_{\omega^k} \rangle} \quad (38)$$

$$\Psi^u[D_{k|k-1}^{\tau}, Z_k](n) = \sum_{j=0}^{\min(|\mathbf{z}^k|, n-u)} \binom{|\mathbf{z}^k|}{j} \rho_c(|\mathbf{z}^k| - j) \times \frac{(1-p_D)^{n-(j+u)}}{\langle 1, w_{\omega}^k \rangle^{j+u}} \frac{n!}{(n-j-u)!} \times e_j(\Lambda(w_{\omega}^k, \mathbf{z}^k)) \quad (39)$$

$$\Lambda(w_{\omega}^k, \mathbf{z}^k) = \left\{ \frac{p_D}{\bar{c}(z)} (w_{\omega}^k)^T q(z) : z \in \mathbf{z}^k \right\} \quad (40)$$

where

$\bar{c}(z)$  represents clutter density;

$w_j(z), q(z), m_j^k(z), P_j^k$  is the value after VB approximation.

The approximation of  $q(\theta)$  is obtained via iterative solving using the VB method by setting immobile points. The posterior density,  $q(\theta)$ , is updated to  $q^{i+1}(\theta)$  by calculating the expectation of the above equation using  $q^i(\theta_k^{(-\varphi)})$  at the  $i+1$  iteration. The joint PDF decomposition is expressed as follows:

$$p(\theta_k, Z_k) = p(Z_{k-1}) \left[ N(x_k; m_k^{\omega,j}, P_k^{\omega,j} / \xi_k) \right]^{(1-t_k)} \left[ N(x_k; m_k^{\omega,j}, P_k^{\omega,j}) \right]^{t_k} \times \left[ N(z_k; \dot{H}x_k, R_k / \lambda_k) \right]^{(1-y_k)} \left[ N(z_k; \dot{H}x_k, R_k) \right]^{y_k} \times G(\xi_k; \frac{\omega_k}{2}, \frac{\omega_k}{2}) G(\lambda_k; \frac{\nu_k}{2}, \frac{\nu_k}{2}) \tau_k^{t_k} (1-\tau_k)^{(1-t_k)} \pi_k^{y_k} (1-\pi_k)^{(1-y_k)} \times \text{Be}(\tau_k; g_0, 1-g_0) \text{Be}(\pi_k; e_0, 1-e_0) \quad (41)$$

where

$$\dot{H} = \left[ 0_{1, J_{\omega,j}^k - 1}, 1 \right] \otimes H;$$

$G$  and  $Be$  represent gamma and beta distributions;

$e_0$  and  $g_0$  are priori mixed probabilities.

At each iteration,  $\theta$  is updated sequentially. In order, update  $q^{i+1}(x_k)$  for Gaussian distribution, update  $q^{i+1}(t_k)$  and  $q^{i+1}(y_k)$  for Bernoulli distribution, update  $q^{i+1}(\xi_k)$  and  $q^{i+1}(\lambda_k)$  for Gamma distribution, update  $q^{i+1}(\tau_k)$  and  $q^{i+1}(\pi_k)$  for Beta distribution, and finally determine the state vector and covariance matrix.

The noise covariance,  $\tilde{R}_{k|k-1}^{(j)(n)}$ , and estimation error covariance,  $\tilde{P}_{k|k-1}^{(j)(n)}$ , are corrected during the process:

$$\tilde{P}_{k|k-1}^{(j)(n)} = \frac{P_k^{\omega,j}}{\mathbb{E}^{(n)}[t_{1,k}] + (1 - \mathbb{E}^{(n)}[t_{1,k}])\mathbb{E}^{(n)}[\zeta_{1,k}]} \quad (42)$$

$$\tilde{R}_{k|k-1}^{(j)(n)} = \frac{R_k}{\mathbb{E}^{(n)}[y_{2,k}] + (1 - \mathbb{E}^{(n)}[y_{2,k}])\mathbb{E}^{(n)}[\lambda_{2,k}]} \quad (43)$$

The procedure for updating the potential distribution is the same as for GM-TCPHD, but the weight computation is altered, and the weights are updated by utilizing the noise matrix's optimal value following VB approximation.

$$w_j(z) = \frac{p_D w_{\omega,j}^k q_k^{(j)(n+1)}(z) \langle \Psi^1[w_{\omega}^k, \mathbf{z}^k \setminus \{z\}], \rho_{\omega^k} \rangle}{\bar{c}(z) \langle \Psi^0[w_{\omega}^k, \mathbf{z}^k], \rho_{\omega^k} \rangle} \quad (44)$$

$$q_k^{(j)(n+1)}(\mathbf{z}_k) = \mathcal{N}\left(\mathbf{z}_k; \dot{H}_k m_k^{\omega,j}, \tilde{R}_{k|k-1}^{(j)(n)} + \dot{H}_k \tilde{P}_{k|k-1}^{(j)(n)} \dot{H}_k^T\right) \quad (45)$$

$$K_k^{(j)(n+1)} = \tilde{P}_{k|k-1}^{(j)(n)} \dot{H}_k^T \left( \dot{H}_k \tilde{P}_{k|k-1}^{(j)(n)} \dot{H}_k^T + \tilde{R}_{k|k-1}^{(j)(n)} \right)^{-1} \quad (46)$$

$$m_k^{(j)(n+1)} = m_k^{\omega,j} + K_k^{(j)(n+1)} \left( \mathbf{z}_k - \dot{H}_k m_k^{\omega,j} \right) \quad (47)$$

$$P_k^{(j)(n+1)} = \left[ I - K_k^{(j)(n+1)} \dot{H}_k \right] \tilde{P}_{k|k-1}^{(j)(n)} \quad (48)$$

## 5. GSTM-TCPHD Filter Implementation with a Multi-Sensor

For multitarget tracking, we employ the GSTM-TCPHD filter for non-Gaussian heavy-tailed noise systems [43]. Nevertheless, the tracking field of view is constrained with a single sensor, and there is a lot of clutter information and little target measurement information [44]. Consequently, multi-sensor scenarios must be studied; however, as the number of sensors increases, so do the measurements, which can lead to a combinatorial explosion. The corresponding PHD components are updated following the subset of measurement selections, and the greedy algorithm is utilized for both path selection and subset division [45].

By enhancing the algorithm from the previous section, the MS-GSTM-TCPHD filter is suggested in this section, along with an implementation. Below are the MS-GSTM-TCPHD filter's prediction and updating steps.

### 5.1. Prediction Process

Under assumptions P1–P4, MS-GSTM-TCPHD filter predictions are obtained, and the prediction process is the same as that of GSTM-TCPHD filters in single-sensor conditions.

At time  $k - 1$ , the posteriori density is as follows:

$$D_{k-1}(X) = \sum_{j=1}^{J^{k-1}} w_j^{k-1} \mathcal{N}\left(X; t_j^{k-1}, m_j^{k-1}, P_j^{k-1}\right) \quad (49)$$

The prediction is as follows:

$$\rho_{k|k-1}(m) = \sum_{j=0}^m \rho_{\beta^k}(m-j) \times \sum_{l=j}^{\infty} \binom{n}{j} \rho_{\pi^{k-1}}(n) (1-p_S)^{l-j} p_S^j \quad (50)$$

$$D_{k|k-1}(X) = D_{\beta^k}(X) + p_S \sum_{j=1}^{j^{k-1}} w_j^{k-1} \mathcal{N}(X; t_j^{k-1}, m_{\omega,j}^k, P_{\omega,j}^k) \tag{51}$$

### 5.2. Two-Step Greedy Approximation

We can use a two-step greedy approximation algorithm to solve the data association problem for multi-sensor data. The first step involves processing the subset of sensor measurements one at a time, calculating the scoring function for the current sensor measurement and the target, keeping the m measurements with the highest weights as the subset of measurements for the current target, and processing each subsequent subset of sensor measurements until the last sensor, as shown in Figure 3.

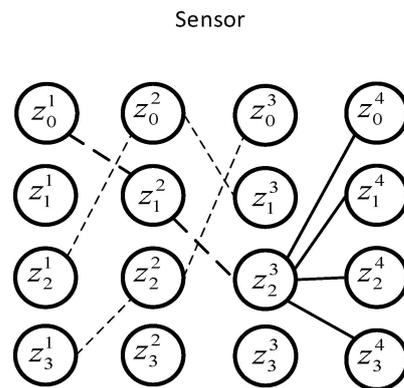


Figure 3. First step of the greedy algorithm.

For a subset of multi-sensing measures,  $W$ , the likelihood function is expressed as follows:

$$f(W|x) = \prod_{(j,l) \in T_W} \frac{p_d^j(x) h_j(z_l^j|x)}{c_j(z_l^j)} \prod_{(j,l) \notin T_W} (1 - p_d^j(x)) \tag{52}$$

where

$$T_W = \{(j, l) : z_l^j \in W\};$$

$p_d^j(\cdot)$  represents the detection probability of the  $j$ -th sensor;

$h_j$  denotes the measurement matrix of the  $j$ -th sensor.

The score function in the measurement subset,  $W$ , is expressed as follows:

$$dW = \frac{\int w(x) \left( \prod_{(i,l) \in T_W} p_d^i(x) h_i(z_l^i|x) \right) \prod_{j:(j,*) \notin T_W} q_d^j(x) dx}{\prod_{(i,l) \in T_W} c_i(z_l^i)} \tag{53}$$

where

$q_d^j(\cdot)$  represents the detection probability of the  $j$ -th sensor;

$dW$  is the ratio of the likelihood function of the subset of target-generated measurements,  $W$ , to the likelihood function of the subset of clutter-generated measurements.

In order to avoid splitting the measurement into distinct components, the subset of measures that yield Gaussian components is re-partitioned in the second step. The set of optimal paths is chosen by keeping the top few partitioning hypotheses with the highest weights, and the set of disjoint measures obtained in the first step is chosen in the order of Gaussian components. Figure 4 depicts the second step as follows.

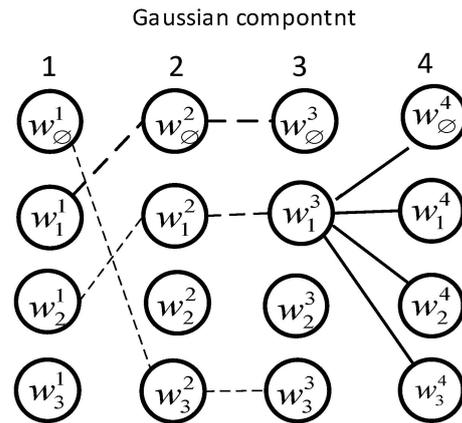


Figure 4. Second step of the greedy algorithm.

Given a partitioned set of measures,  $W_i$ , the weights are calculated as follows.

### 5.3. Update Process

Each partition,  $P = \{w^1, \dots, w^n\}$ , is created via greedy approximation, and each Gaussian component is updated using a partition,  $P$ . The updated cardinality distribution and TPHD can be expressed as follows:

$$\rho_{\pi^k}(n) = \frac{\Psi^0[w_{\omega}^k, \mathbf{z}^k](n)\rho_{\omega^k}(n)}{\langle \Psi^0[w_{\omega}^k, \mathbf{z}^k], \rho_{\omega^k} \rangle} \tag{54}$$

$$D_{\pi_s^k}(X) = \sum_{j=1}^k w_j^k \mathcal{N}(X; t_j^k, m_j^k, P_j^k) \tag{55}$$

The updated optimal value of the noise covariance matrix for the state is used in the VB approximation of the posterior state, which is performed as demonstrated in the preceding section as part of the update process. The difference is that the update of the weights in it has been changed, and the formula is as follows:

$$w_j(z) = \frac{p_D w_{\omega,j}^k q_k^{(j)(n+1)}(z) \langle \Psi^1[w_{\omega}^k, P_w^i \setminus \{w\}], \rho_{\omega^k} \rangle}{\tilde{c}(z) \langle \Psi^0[w_{\omega}^k, P_w^i], \rho_{\omega^k} \rangle} \tag{56}$$

## 6. Simulation

To compare the CPHD, TPHD, and TCPHD filters, the proposed GSTM-TCPHD filter is simulated. Table 1 contains the target parameters' designs, while Table 2 contains the TPHD filter's design. A linear 2D scene with multiple targets appearing, disappearing, omitted, and cluttered is observed. There are four birth regions at positions [1000, 1500], [250, 750], [-250, 1000], and [-1500, 250]. There are eight targets in the scenario, each with different start and end times. The equations of motion of the target are the CV model; the CV model is a linear model, which makes it convenient for implementing target-tracking algorithms and simplifying calculations. Its state vector is represented as  $x = [x, v_x, y, v_y]$ , where  $x, y$  and  $v_x, v_y$  are the position and velocity of the target. Its state transfer equation is as follows:

$$x_k = Fx_{k-1} + Gw \tag{57}$$

$$F = \begin{bmatrix} 1 & T & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & T \\ 0 & 0 & 0 & 1 \end{bmatrix}, G = \begin{bmatrix} \frac{T^2}{2} & 0 \\ T & 0 \\ 0 & \frac{T^2}{2} \\ 0 & T \end{bmatrix} \tag{58}$$

where

$w$  represents process noise;

$T$  represents the sampling interval of the sensor.

The initial process noise covariance matrix,  $Q$ , is as follows:

$$Q = \text{diag}\left(\left[GG^T\right]\right) \quad (59)$$

**Table 1.** Configuration of target parameters.

Parameter	Value
Degrees of freedom	10
State noise	5 m/s <sup>2</sup>
Measurement of mixing probabilities	0.7
State mixing probability	0.7
Measurement of heavy-tailed noise probability	0.10
State heavy-tailed noise probability	0.05
Distance error	15 m

**Table 2.** TPHD filtering parameters.

Parameter	Value
Detection probability	0.9
Survival probability	0.98
Clutter	40
Tracking period	1 s
Pruning threshold	$10 \times 10^{-5}$
Maximum number of PHDs	100
Trajectory scanning steps	5

The sensor converts its observations into a two-dimensional distance measurement between the target and the sensor. The measurement model is as follows:

$$z_j = H_j x + \varepsilon_j \quad (60)$$

where:

$\varepsilon_j$  represents the measurement noise of the  $j$ -th sensor.

$H_j$  represents the measurement matrix of the  $j$ -th sensor.

$$H_j = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 \end{bmatrix} \quad (61)$$

The measurement noise covariance matrix is expressed as follows:

$$R = r\mathbf{I}_2 \quad (62)$$

where  $r = 25$ . Heavy-tailed noise in the system is caused by reflections of signals and instability in the sensors. As a result, we incorporate heavy-tailed noise points into our measurements with a 10% probability and a value that is 100 times greater than the measurement's original value. The maneuver incorporates a heavy-tailed noise point with a probability of 5% and a value equal to 50 times the initial value that was predicted.

$$w_k \sim \begin{cases} N(0, Q), & w.p. 1 - p_{mo} \\ N(0, 100Q), & w.p. p_{mo} \end{cases} \quad (63)$$

$$\varepsilon_k \sim \begin{cases} N(0, R), & w.p. 1 - p_{po} \\ N(0, 50R), & w.p. p_{po} \end{cases} \quad (64)$$

where  $w.p.$  is the size of the probability, denoting heavy-tailed noise with probabilities  $p_{m0}$  and  $p_{p0}$ . Its model is shown in Figure 5.

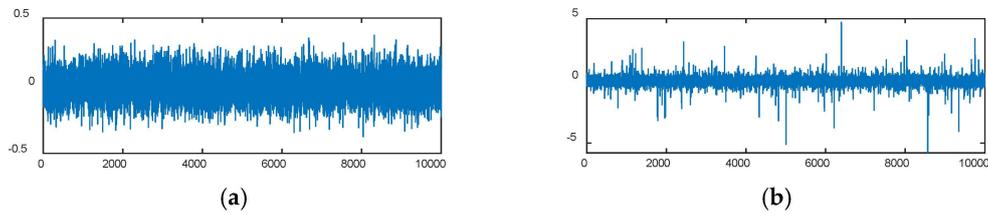


Figure 5. (a) Gaussian white noise sequence; (b) outlier noise sequence.

The optimal subpattern assignment (OSPA) distance was used to assess the tracking results, which is defined as

$$\bar{d}_p^{(c)}(\mathbf{X}, \mathbf{Y}) = \begin{cases} 0 & m = n = 0 \\ \left( \frac{1}{n} \left( \min_{\pi \in \Pi_{n=1}} \sum_{i=1}^m d^{(c)}(x_i, y_{\pi(i)})^p + c^p(n - m) \right) \right)^{\frac{1}{p}} & m \leq n \\ \bar{d}_p^{(c)}(\mathbf{Y}, \mathbf{X}) & m > n \end{cases} \quad (65)$$

where

$c$  is the truncation distance; the parameter is set to 100;

$p$  is the order, which is set to 1.

The accuracy of both target position and target number estimations are measured by the OSPA distance.

### 6.1. Single-Sensor Simulation

According to the scene description, the target moves with a constant velocity model. If outlier value noise is present, the target’s motion model will change. Figure 6 shows the result of the motion path. From the simulation of the target path, it can be seen that the target’s maneuvering has significant changes at certain times, indicating that the target has strong maneuvering at these times.

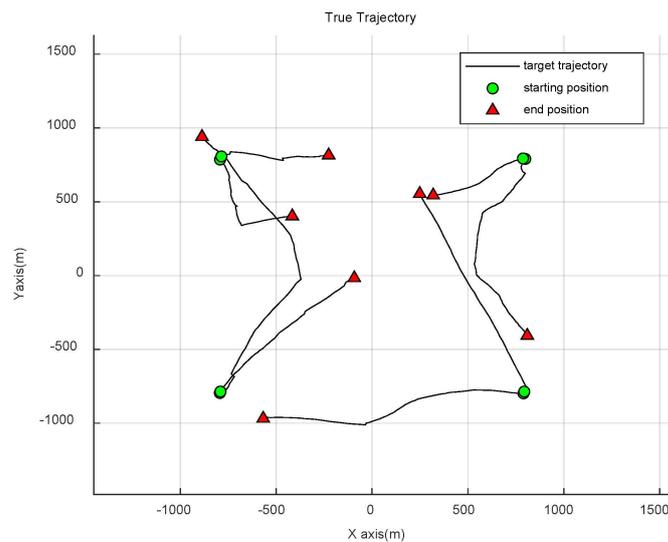


Figure 6. Path of target movement.

In the absence of heavy-tailed noise, the noise exactly follows a Gaussian distribution. Several filters were simulated against this background, and the results are shown in Figures 7 and 8. Then, we only introduced heavy-tailed noise into the measurements. The results of 100 Monte Carlo simulations are shown in Figures 9 and 10.

In the scenario where noise outliers are not considered, Figures 7 and 8 show that traditional methods can achieve accurate tracking. The number of targets and the OSPA distance simulated by the two strong maneuvering anti-outlier algorithms are relatively large. This is because the two filters deliberately increase the covariance calculation during the filtering process to improve the anti-noise performance, resulting in an increase in errors.

When process noise outliers are not considered, multiple algorithms can achieve stable tracking. Figure 9 illustrates that several algorithms have relatively small errors in estimating the number of targets. The GSTM-TCPHD algorithm has a slightly larger estimate of the number of targets than other methods, particularly when the number of targets increases and there is target extinction. Figure 10 shows that several algorithms have similar performance in estimating the OSPA distance.

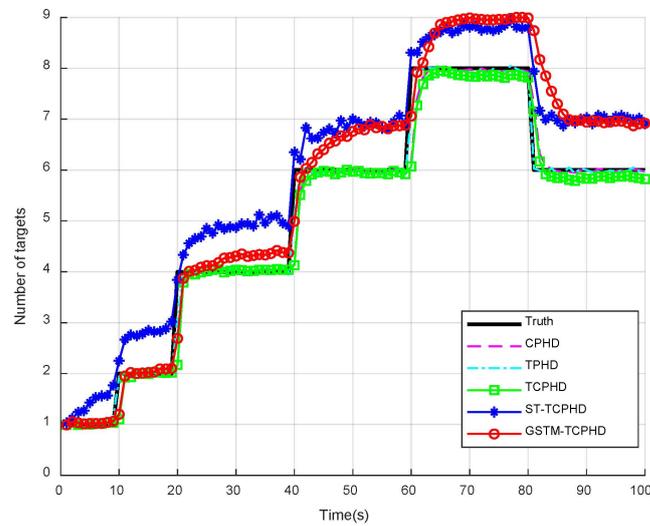


Figure 7. Estimation of the number without heavy-tailed noise.

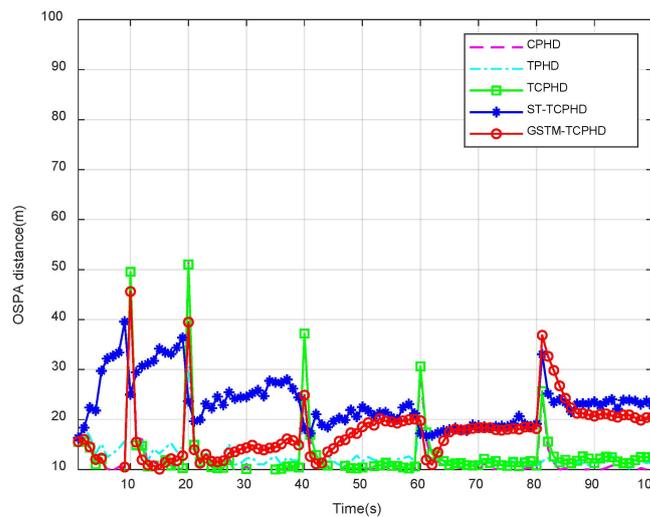
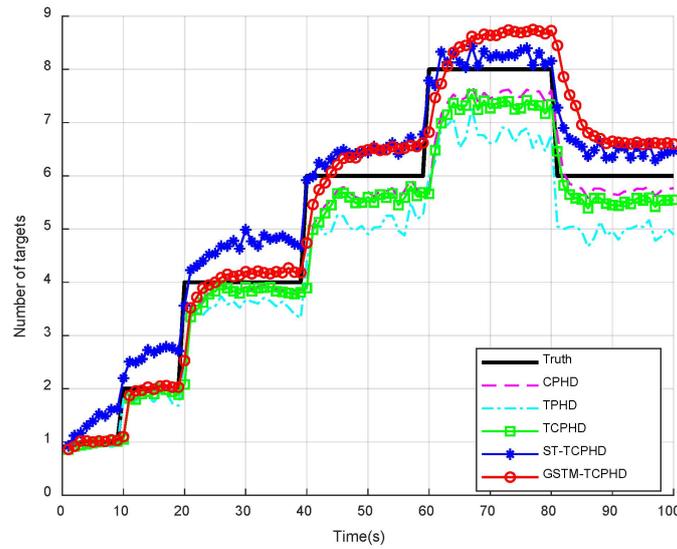
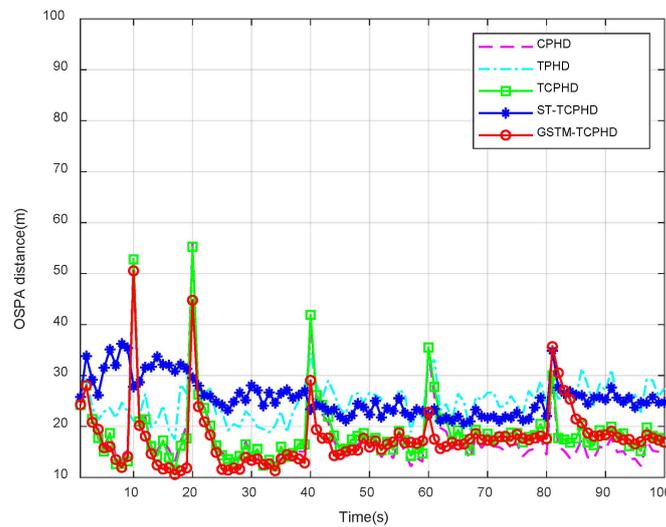


Figure 8. OSPA distance without heavy-tailed noise.



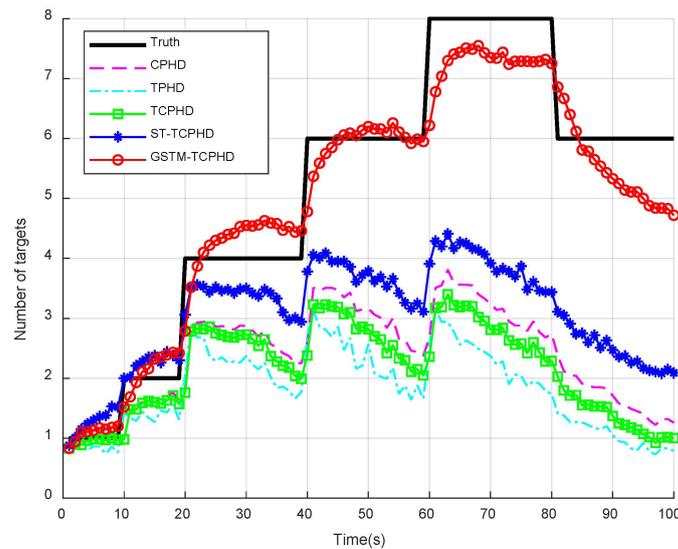
**Figure 9.** Estimation of the number in the measurement of heavy-tailed noise.



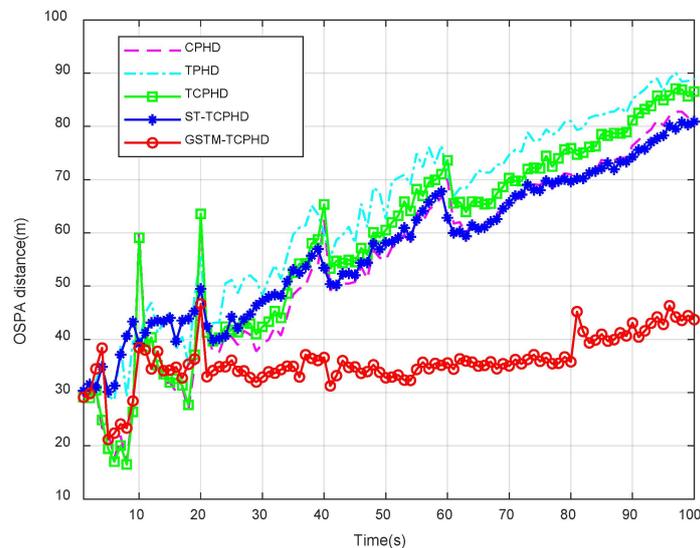
**Figure 10.** OSPA distance in the measurement of heavy-tailed noise.

Comparing the simulation results without considering outlier noise and without considering process noise outliers, when measurement noise outliers are added, the tracking number error and OSPA distance of several algorithms increase. The OSPA distance is composed of a distance error and a cardinality error. Therefore, when the estimated number of targets is similar, the improvement in OSPA distance performance mainly comes from the increased accuracy of target position estimation. This suggests that the proposed GSTM-TCPHD algorithm has higher position estimation accuracy in this scenario.

Simulations were conducted after adding heavy-tailed noise to the state and measurement noise. The results of 100 MC simulation runs are shown in Figures 11 and 12. The figures illustrate the relatively large error and gradual tendency toward divergence in the standard TPHD/TCPHD algorithm. The GSTM-TCPHD filter performs exceptionally well.



**Figure 11.** Estimation of the amount of heavy-tailed noise for state and measurement.



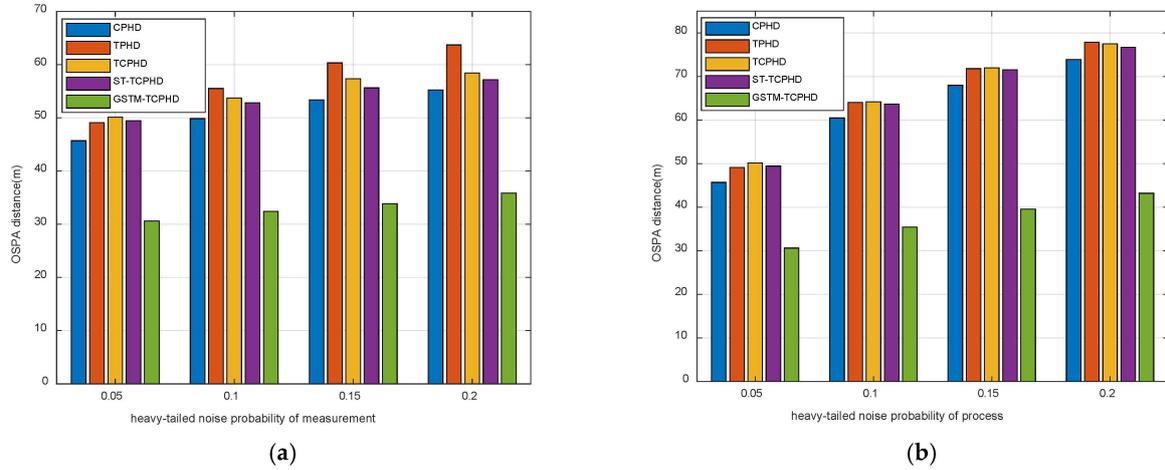
**Figure 12.** OSPA distance in heavy-tailed noise for state and measurement.

Figures 11 and 12 demonstrate that the GSTM-TCPHD proposed in this study outperforms the remaining filters in terms of overall performance, exhibiting the smallest tracking number error and OSPA distance error, as well as the most stable target tracking, when considering process outlier noise and measurement outlier noise. The tracking effects of the remaining filters tend to diverge with increasing tracking time.

Among the better-performing filters, the ST-TCPHD filter has a runtime of 21.91 s and an average OSPA distance of 53.19, while the GSTM-TCPHD filter has a runtime of 41.09 s and an average OSPA distance of 32.39. When comparing the simulation results while excluding outlier noise, only considering measurement outlier noise, and considering both process and measurement outlier noise, the addition of outlier noise leads to an increase in the number of target tracking errors and OSPA distance and a sharp decrease in the tracking effect. The proposed filter performs well in all three scenarios, particularly after the addition of outlier noise. It combines the characteristics of a short running time and high tracking accuracy.

The performance of the filter was further evaluated to investigate the impact of different measurement noises with the same process noise and different process noises with the same measurement noise. Figure 13 displays the simulation results. Figure 13

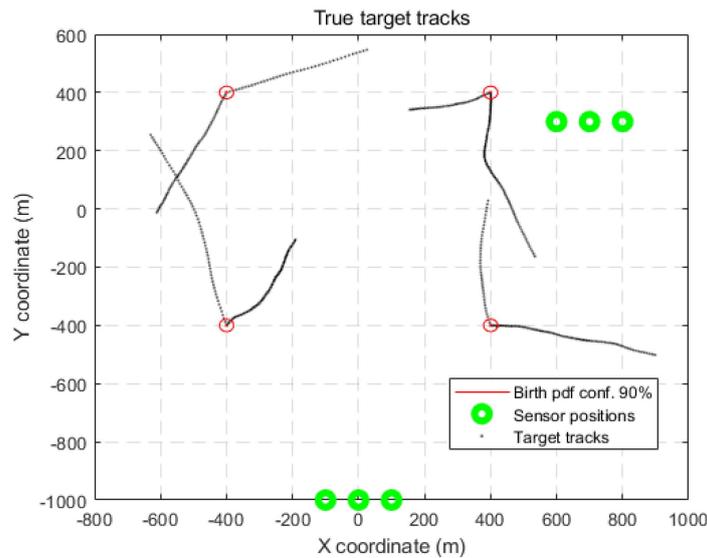
shows that process outlier noise has a greater impact on target-tracking accuracy than measurement outlier noise. OSPA distance is more sensitive to process noise when the probability of outlier noise changes. Our proposed method exhibits the smallest increase in OSPA distance and more stable tracking when the probability of outlier noise changes.



**Figure 13.** OSPA distance with different heavy-tailed noise probabilities: (a) state remains unchanged, and measurement noise has different heavy-tailed probabilities; (b) measurements remain unchanged, and state noise has different heavy-tailed probabilities.

### 6.2. Multi-Sensor Simulation

To explore the performance of the proposed MS-GSTM-TCPHD filter, six sensors are arranged in the scene. The sensor positions are  $[800, 300]$ ,  $[700, 300]$ ,  $[600, 300]$ ,  $[-100, 1000]$ ,  $[0, 1000]$ . The sensor location and the simulated primary path are shown in Figure 14, where the green hollow circle is the sensor location, and the black line represents the trajectory map of the primary simulation. The red circle is the born region. Each sensor has a certain probability of appearing as a quantitative wild value, and there is a total of eight target appearances and extinctions in the scene.



**Figure 14.** Sensor distribution and target motion.

The tracking performance of the proposed algorithm in a multi-sensor background was verified. Firstly, a simulation was conducted without considering outlier noise, and the results are shown in Figures 15 and 16. Secondly, a simulation was conducted when considering measurement outlier noise, and the results are shown in Figures 17 and 18.

Figures 15–18 demonstrate that the methods employed achieve complete tracking and reduce target-tracking error as measurement outlier noise increases. Table 3 shows a comparison of the tracking results of a single sensor and multi-sensors running CPHD filters, TCHHD filters, ST-TCPHD filters, and GSTM-TCPHD filters in terms of average OSPA distance and running time without considering outlier noise. Table 4 shows the same comparison without considering process outlier noise. The comparison of computational efficiency and accuracy demonstrates that our proposed method outperforms the others.

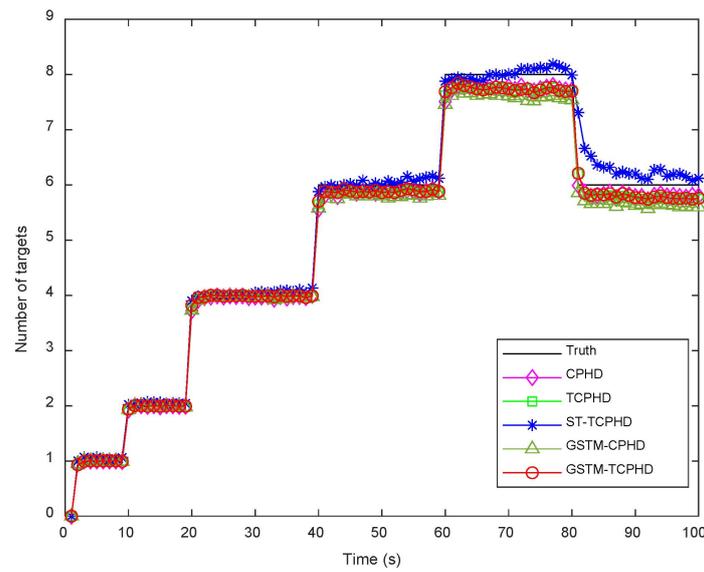


Figure 15. Number of tracking estimates without outlier noise under multiple sensors.

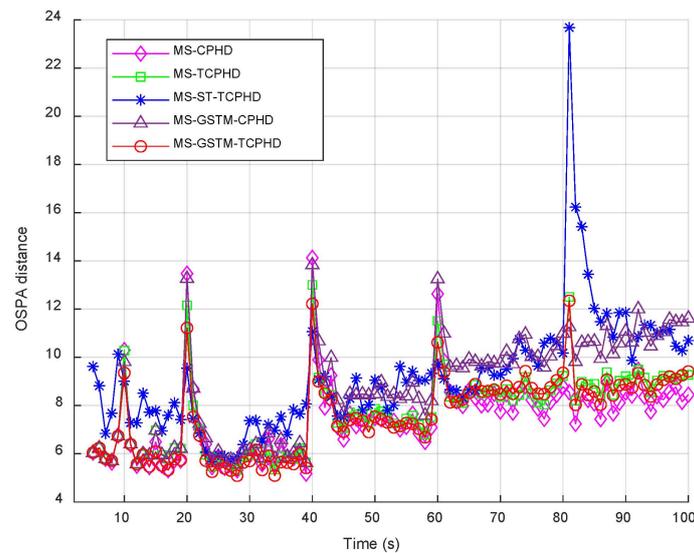


Figure 16. Tracking OSPA distance without outlier noise under multiple sensors.

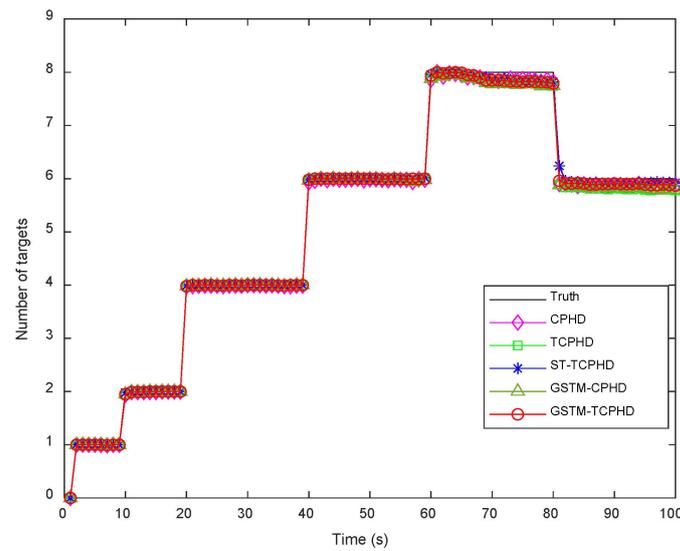


Figure 17. Number of tracking estimates without outlier process noise under multiple sensors.

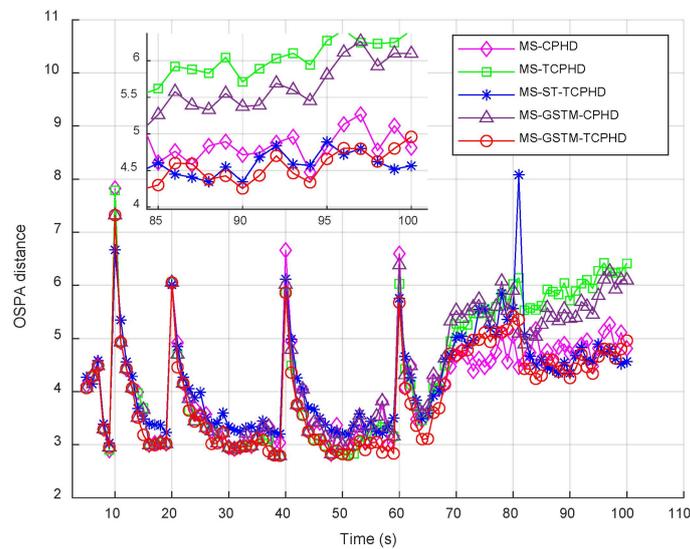


Figure 18. Tracking OSPA distance estimates without outlier process noise under multiple sensors.

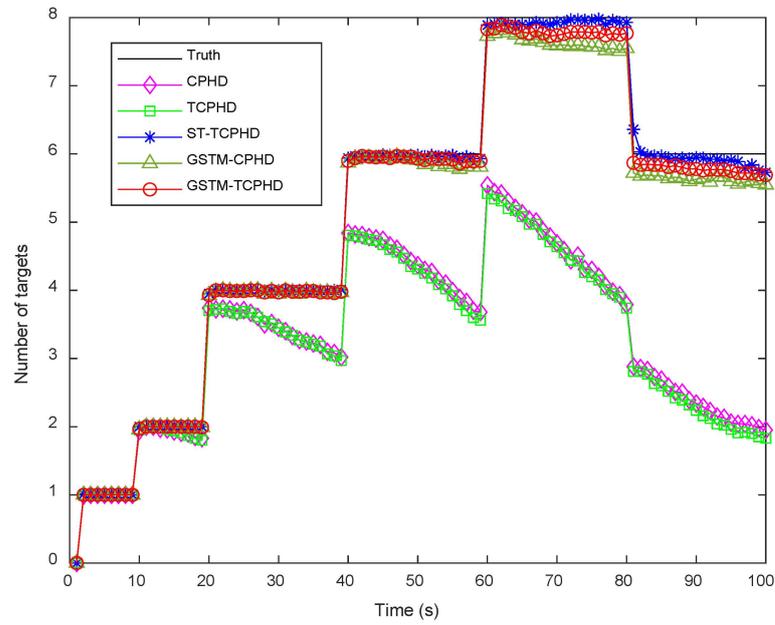
Table 3. OSPA and runtime with different numbers of sensors under Gaussian noise.

Filter	OSPA for MS	OSPA for SS	Time for MS	Time for SS
CPHD	3.68	11.84	2.27 s	0.68 s
TCPHD	4.03	12.42	4.37 s	0.29 s
ST-TCPHD	4.19	23.50	119.18 s	17.70 s
GSTM-TCPHD	3.88	28.89	27.81 s	24.11 s

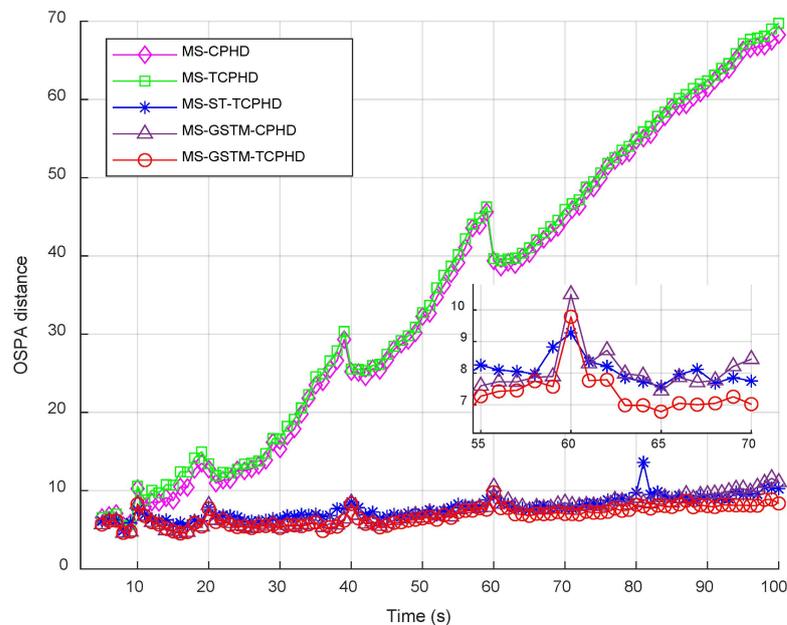
Table 4. OSPA and runtime with different numbers of sensors under outlier measurement noise.

Filter	OSPA for MS	OSPA for SS	Time for MS	Time for SS
CPHD	4.08	17.49	2.74 s	0.79 s
TCPHD	4.35	18.71	5.15 s	0.28 s
ST-TCPHD	4.21	25.54	152.07 s	12.15 s
GSTM-TCPHD	3.92	17.54	48.88 s	34.61 s

The results of 100 Monte Carlo simulations, considering the process outlier noise and measurement outlier noise, are shown in Figures 19 and 20. When considering both process outlier noise and measurement outlier noise simultaneously, the tracking error of traditional Gaussian noise filters increases sharply. Even under multiple sensors, tracking accuracy cannot be guaranteed. Compared with the operating results of a single sensor against the same background, a filter that can effectively track significantly reduces the OSPA distance, highlighting the advantages of multiple sensors.



**Figure 19.** Number of tracking estimates with multiple sensors with respect to the presence of process outlier noise and measurement outlier noise.



**Figure 20.** Tracking OSPA distance with multiple sensors with respect to the presence of process outlier noise and measurement outlier noise.

Finally, the effect of the number of sensors is shown in Figure 21. The effect of different outlier noise probabilities on the target-tracking results was further analyzed. The simulation results are shown in Figure 22.

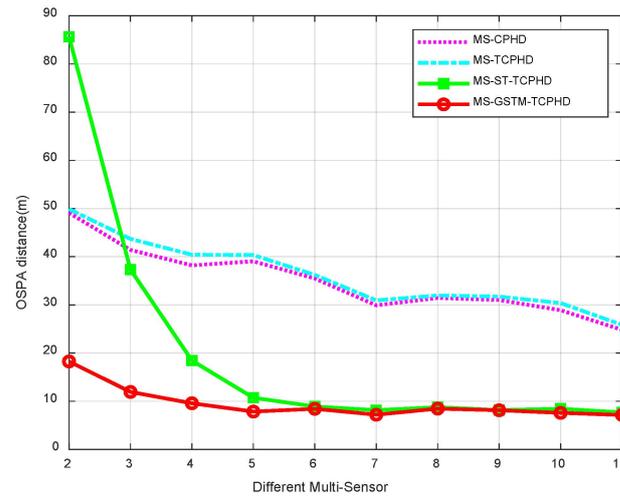


Figure 21. OSPA distance for different numbers of sensors.

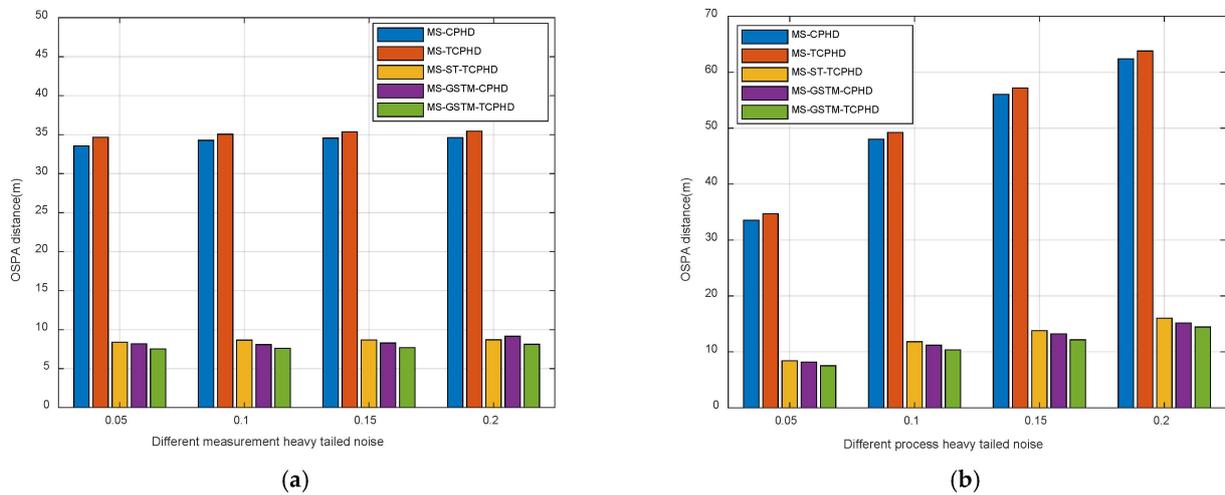


Figure 22. Multi-sensor OSPA distance with different heavy-tailed noise probabilities: (a) state remains unchanged, and measurement noise has different heavy-tailed probabilities; (b) measurements remain unchanged, and state noise has different heavy-tailed probabilities.

It has been verified that, by increasing the number of sensors, the error continues to decrease, and after reaching six sensors, the OSPA distance gradually converges to the limiting value. In a multi-sensor scenario, the change in the probability of outlier noise has little effect on the tracking results, and the stability is greater than that of a single sensor. The average OSPA distance and running time of the four methods are shown in Table 5. Even if the probability of outlier noise changes, the increase in OSPA distance is not significant. The resulting graph and table demonstrate that our proposed method enhances computational efficiency while maintaining tracking accuracy.

Table 5. OSPA and runtime with different numbers of sensors under outlier noise.

Filter	OSPA for MS	OSPA for SS	Time for MS	Time for SS
CPHD	34.71	51.77	2.74 s	0.65 s
TCPHD	35.55	55.47	5.15 s	0.19 s
ST-TCPHD	7.76	54.18	221.07 s	14.09 s
GSTM-TCPHD	6.75	32.52	48.66 s	41.91 s

## 7. Conclusions

In order to handle the multitarget tracking problem in the presence of non-Gaussian heavy-tailed noise, in this paper, we propose a GSTM-TCPHD filter for a single-sensor background and the MS-GSTM-TCPHD filter for a multi-sensor background. By utilizing trajectory sets to create multitarget trajectories, the TCPHD filter allows us to model non-Gaussian heavy-tailed noise with a GSTM distribution. The true posterior probability is then approximated using VB approximation. We derive the GSTM-TCPHD filter's implementation procedure under heavy-tailed noise scenarios. Furthermore, we create the MS-GSTM-TCPHD filter, which extends the GSTM-TCPHD filter to multi-sensor systems. We use a greedy approximation technique to ensure that no measurements with low-likelihood weights are missed in order to address the problem of data explosion in multi-sensor scenarios. This leads to a workable implementation of the MS-GSTM-TCPHD filter for multiple sensors.

We conducted Monte Carlo simulations of the filters to compare OSPA distances and cardinality errors. The simulation results show that the error of the CPHD/TPHD/TCPHD filter gradually increases in the presence of heavy-tailed noise, eventually leading to divergence. In contrast, the GSTM-TCPHD filter achieves stable tracking with minimal cardinality errors. Furthermore, compared with the GSTM-TCPHD filter, the MS-GSTM-TCPHD filter further reduces OSPA distance and exhibits improved tracking accuracy in multi-sensor scenarios. Additionally, we also compared filters using Student's t-distribution to model heavy-tailed noise. According to the results, our suggested filter operates more efficiently and requires less time to operate, all while exhibiting smaller OSPA distance and cardinality errors. To sum up, a greater variety of scenarios can be covered by our suggested filter.

**Author Contributions:** Conceptualization, S.W., Y.L. and J.W.; methodology, S.W., Y.L., J.W., F.Q. and Y.Z.; software, S.W. and Y.Z.; validation, S.W., Y.Z. and J.W.; formal analysis, S.W., Y.L., Y.Z., F.Q., X.Z. and Z.L.; investigation, S.W., Y.L., Y.Z., F.Q., X.Z. and Z.L.; resources, S.W., Y.Z., X.Z. and Z.L.; data curation, Y.L.; writing—original draft preparation, Y.L.; writing—review and editing, S.W., Y.L., Y.Z. and J.W.; visualization, Y.L.; supervision, Y.L.; project administration, Y.L.; funding acquisition, S.W. and J.W. All authors have read and agreed to the published version of the manuscript.

**Funding:** This research was funded by the National Natural Science Foundation of China (Grants No. 62171029 and No. 61671035).

**Data Availability Statement:** Data are contained within the article.

**Conflicts of Interest:** The authors declare no conflicts of interest.

## References

1. Li, H.; Zhu, J. Target Tracking Algorithm Based on Meanshift and Kalman Filter. *J. Beijing Inst. Technol.* **2019**, *28*, 365–370.
2. Bar-Shalom, Y.; Willett, P.K.; Tian, X. *Tracking and Data Fusion*; YBS Publishing: Storrs, CT, USA, 2011.
3. Mahler, R.P.S. *Advances in Statistical Multisource-Multitarget Information Fusion*; Artech House: New York, NY, USA, 2014.
4. Vo, B.T.; Vo, B.N.; Cantoni, A. The cardinality balanced multi-target multi-Bernoulli filter and its implementations. *IEEE Trans. Signal Process.* **2008**, *57*, 409–423.
5. Park, W.J.; Park, C.G. Distributed GM-CPHD Filter Based on Generalized Inverse Covariance Intersection. *IEEE Access* **2021**, *9*, 94078–94086. [[CrossRef](#)]
6. García-Fernández, Á.F.; Svensson, L.; Morelande, M.R. Multiple target tracking based on sets of trajectories. *IEEE Trans. Aerosp. Electron. Syst.* **2019**, *56*, 1685–1707. [[CrossRef](#)]
7. Wei, S.; Zhang, B.; Yi, W. Trajectory PHD and CPHD Filters With Unknown Detection Profile. *IEEE Trans. Veh. Technol.* **2022**, *8*, 8042–8058. [[CrossRef](#)]
8. Hou, L.; Lian, F.; Tan, S.; Xu, C.; de Abreu, G.T.F. Robust Generalized Labeled Multi-Bernoulli Filter for Multitarget Tracking With Unknown Non-Stationary Heavy-Tailed Measurement Noise. *IEEE Access* **2021**, *9*, 94438–94453. [[CrossRef](#)]
9. Saha, S. Noise Robust Online Inference for Linear Dynamic Systems. *arXiv* **2015**, arXiv:1504.05723.
10. Agamennoni, G.; Nieto, J.J.; Nebot, E.M. Approximate Inference in State-Space Models With Heavy-Tailed Noise. *IEEE Trans. Signal Process.* **2012**, *60*, 5024–5037. [[CrossRef](#)]

11. Lee, C.S.; Cho, H.C.; Jung, B.Y.; Ra, W.S. Adaptive Sequential Gaussian Sum Approximation for High-Resolution Radar Target Tracking with Non-Gaussian Multi-Detection. In Proceedings of the 2022 22nd International Conference on Control, Automation and Systems (ICCAS), Jeju, Republic of Korea, 27 November–1 December 2022; pp. 1654–1659.
12. García-Fernández, Á.F.; Ralph, J.; Horridge, P.; Maskell, S. A Gaussian Filtering Method for Multitarget Tracking With Nonlinear/Non-Gaussian Measurements. *IEEE Trans. Aerosp. Electron. Syst.* **2021**, *57*, 3539–3548. [[CrossRef](#)]
13. Gordon, N. A hybrid bootstrap filter for target tracking in clutter. *IEEE Trans. Aerosp. Electron. Syst.* **1997**, *33*, 353–358. [[CrossRef](#)]
14. Kitagawa, G. Monte Carlo filter and smoother for non-Gaussian nonlinear state space models. *J. Comput. Graph. Stat.* **1996**, *5*, 1–25.
15. Loxam, J.; Drummond, T. Student-t mixture filter for robust, real-time visual tracking. In *Computer Vision—ECCV 2008: 10th European Conference on Computer Vision, Marseille, France, October 12–18, 2008, Proceedings, Part III 10*; Springer: Berlin/Heidelberg, Germany, 2008; pp. 372–385.
16. Tao, Y.; Yau, S.S.T. Outlier-Robust Iterative Extended Kalman Filtering. *IEEE Signal Process. Lett.* **2023**, *30*, 743–747. [[CrossRef](#)]
17. Chen, B.; Principe, J.C. Maximum correntropy estimation is a smoothed MAP estimation. *IEEE Signal Process. Lett.* **2012**, *19*, 491–494. [[CrossRef](#)]
18. Yang, Y.; Luo, Y.; Qi, A.; Shi, G.; Miao, M.; Fan, J.; Ma, J.; Qi, Y. Huber Kalman Filter for Wi-Fi Based Vehicle Driver’s Respiration Detection. *IEEE Trans. Veh. Technol.* **2022**, *71*, 8933–8943. [[CrossRef](#)]
19. Tollkühn, A.; Particke, F.; Thielecke, J. Gaussian state estimation with non-gaussian measurement noise. In *2018 Sensor Data Fusion: Trends, Solutions, Applications (SDF)*; IEEE: Bonn, Germany, 2018; pp. 1–5.
20. Huang, Y.; Zhang, Y.; Li, N.; Chambers, J. A robust Gaussian approximate filter for nonlinear systems with heavy tailed measurement noises. In Proceedings of the IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), Shanghai, China, 20–25 March 2016; pp. 4209–4213.
21. Liu, Z.; Chen, S.; Wu, H.; He, R.; Hao, L. A Student’s t mixture probability hypothesis density filter for multi-target tracking with outliers. *Sensors* **2018**, *18*, 1095. [[CrossRef](#)] [[PubMed](#)]
22. Liu, Z.; Chen, S.; Wu, H.; Chen, K. Robust student’s t mixture probability hypothesis density filter for multi-target tracking with heavy-tailed noises. *IEEE Access* **2018**, *6*, 39208–39219. [[CrossRef](#)]
23. Yang, J.; Ge, H. Adaptive probability hypothesis density filter based on variational Bayesian approximation for multi-target tracking. *IET Radar Sonar Navig.* **2013**, *7*, 959–967. [[CrossRef](#)]
24. Yang, J.L.; Ge, H.W. An improved multi-target tracking algorithm based on CB-MeMBeR filter and variational Bayesian approximation. *Signal Process.* **2013**, *93*, 2510–2515. [[CrossRef](#)]
25. Li, W.; Jia, Y.; Du, J.; Zhang, J. PHD filter for multi-target tracking by variational Bayesian approximation. In Proceedings of the 52nd IEEE Conference on Decision and Control, Firenze, Italy, 10–13 December 2013; pp. 7815–7820.
26. Yi, W.; Li, S.; Wang, B. Computationally Efficient Distributed Multi-sensor Fusion with Multi-Bernoulli Filter. *IEEE Trans. Signal Process.* **2020**, *68*, 241–256. [[CrossRef](#)]
27. Mahler, R.P.S. Multitarget Bayes filtering via first-order multitarget moments. *IEEE Trans. Aerosp. Electron. Syst.* **2003**, *39*, 1152–1178. [[CrossRef](#)]
28. Nannuru, S.; Coates, M.; Rabbat, M.; Blouin, S. General solution and approximate implementation of the multisensor multitarget CPHD filter. In Proceedings of the 2015 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), South Brisbane, QLD, Australia, 19–24 April 2015; pp. 4055–4059.
29. Mahler, R. Approximate multisensor CPHD and PHD filters. In Proceedings of the 2010 13th International Conference on Information Fusion, Edinburgh, UK, 26–29 July 2010; pp. 1–8.
30. Sun, J.; Dai, B.; Zhang, Y. MS-LMB Filter for Stealthy Target Tracking Based on Greedy Measurement Partitioning Mechanism. *Trans. Beijing Inst. Technol.* **2022**, *42*, 1290–1298.
31. Huang, Y.; Zhang, Y.; Zhao, Y.; Chambers, J.A. A Novel Robust Gaussian–Student’s t Mixture Distribution Based Kalman Filter. *IEEE Trans. Signal Process.* **2019**, *67*, 3606–3620. [[CrossRef](#)]
32. Ji, S.; Zhang, Z.; Ying, S.; Wang, L.; Zhao, X.; Gao, Y. Kullback–Leibler Divergence Metric Learning. *IEEE Trans. Cybern.* **2022**, *4*, 2047–2058. [[CrossRef](#)] [[PubMed](#)]
33. Nannuru, S.; Blouin, S.; Coates, M.; Rabbat, M. Multisensor CPHD filter. *IEEE Trans. Aerosp. Electron. Syst.* **2016**, *52*, 1834–1854. [[CrossRef](#)]
34. Vo, B.N.; Ma, W.K. The Gaussian mixture probability hypothesis density filter. *IEEE Trans. Signal Process.* **2006**, *54*, 4091–4104. [[CrossRef](#)]
35. García-Fernández, Á.F.; Svensson, L. Trajectory PHD and CPHD filters. *IEEE Trans. Signal Process.* **2019**, *67*, 5702–5714. [[CrossRef](#)]
36. Vo, B.T.; Vo, B.N.; Hoseinnezhad, R.; Mahler, R.P.S. Robust Multi-Bernoulli Filtering. *IEEE J. Sel. Top. Signal Process.* **2013**, *7*, 399–409. [[CrossRef](#)]
37. Roth, M.; Özkan, E.; Gustafsson, F. A Student’s t filter for heavy tailed process and measurement noise. In Proceedings of the 2013 IEEE International Conference on Acoustics, Speech and Signal Processing, Vancouver, BC, Canada, 26–31 May 2013; pp. 5770–5774.
38. Kotz, S.; Nadarajah, S. *Multivariate T-Distributions and Their Applications*; Cambridge University Press: Cambridge, UK, 2004.
39. Youn, W.; Huang, Y.; Myung, H. Outlier-Robust Student’s-t-Based IMM-VB Localization for Manned Aircraft Using TDOA Measurements. *IEEE/ASME Trans. Mechatron.* **2020**, *25*, 1646–1658. [[CrossRef](#)]

40. Huang, Y.; Zhang, Y.; Wu, Z.; Li, N.; Chambers, J. A novel adaptive Kalman filter with inaccurate process and measurement noise covariance matrices. *IEEE Trans. Autom. Control.* **2018**, *63*, 594–601. [[CrossRef](#)]
41. Yi, W.; Li, G.; Battistelli, G. Distributed Multi-Sensor Fusion of PHD Filters With Different Sensor Fields of View. *IEEE Trans. Signal Process.* **2020**, *68*, 5204–5218. [[CrossRef](#)]
42. Liu, S.; Shen-tu, H.; Chen, H.; Peng, D.; Shi, Y. Asynchronous Multi-Sensor Fusion Multi-Target Tracking Method. In Proceedings of the 2018 IEEE 14th International Conference on Control and Automation (ICCA), Anchorage, AK, USA, 12–15 June 2018; pp. 459–463.
43. Li, T.; Wang, X.; Liang, Y.; Pan, Q. On Arithmetic Average Fusion and Its Application for Distributed Multi-Bernoulli Multitarget Tracking. *IEEE Trans. Signal Process.* **2020**, *68*, 2883–2896. [[CrossRef](#)]
44. Movassagh, S.; Fatehi, A.; Sedigh, A.K.; Shariati, A. Kalman Filter Fusion With Smoothing for a Process With Continuous-Time Integrated Sensor. *IEEE Sens. J.* **2023**, *23*, 7279–7287. [[CrossRef](#)]
45. Clark, D.E. Multi-Sensor Network Information for Linear-Gaussian Multi-Target Tracking Systems. *IEEE Trans. Signal Process.* **2021**, *69*, 4312–4325. [[CrossRef](#)]

**Disclaimer/Publisher’s Note:** The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.