Robust Generalized Labeled Multi-Bernoulli Filter and Smoother for Multiple Target Tracking using Variational Bayesian

Peng Li, Wenhui Wang, Junda Qiu*, Congzhe You, and Zhenqiu Shu
School of Computer Engineering, Jiangsu University of Technology,
Changzhou, 213001, China
[e-mail: lipengjiangnan@163.com]
*Corresponding author: Junda Qiu

Received March 31, 2021; revised January 19, 2022; accepted March 8, 2022; published March 31, 2022

Abstract

Multiple target tracking mainly focuses on tracking unknown number of targets in the complex environment of clutter and missed detection. The generalized labeled multi-Bernoulli (GLMB) filter has been shown to be an effective approach and attracted extensive attention. However, in the scenarios where the clutter rate is high or measurement-outliers often occur, the performance of the GLMB filter will significantly decline due to the Gaussian-based likelihood function is sensitive to clutter. To solve this problem, this paper presents a robust GLMB filter and smoother to improve the tracking performance in the scenarios with high clutter rate, low detection probability, and measurement-outliers. Firstly, a Student-T distribution variational Bayesian (TDVB) filtering technology is employed to update targets' states. Then, The likelihood weight in the tracking process is deduced again. Finally, a trajectory smoothing method is proposed to improve the integrative tracking performance. The proposed method are compared with recent multiple target tracking filters, and the simulation results show that the proposed method can effectively improve tracking accuracy in the scenarios with high clutter rate, low detection rate and measurement-outliers. Code is published on GitHub.

Keywords: GLMB, target tracking, trajectory smoothing, TDVB.

http://doi.org/10.3837/tiis.2022.03.009 **ISSN : 1976-7277**

1. Introduction

The significant difficulty of multiple target tracking (MTT) is to track an unknown number of targets in the complex scenarios with strong interference, missed detection, and clutters. In the past decade, MTT technology has been widely used in various fields, such as radar networks [31], vehicle tracking [32], robot vision [33], etc. The most important theories in MTT mainly include the random finite set (RFS) [1] and the labeled RFS (LRFS) [2]. A lot of work has been done based on these two theories, which make RFS and LRFS become hot MTT frameworks in recent years.

Based on the Finite Set Statistics (FISST), Mahler presented the RFS theory and its first-order moment, i.e., the probability hypothesis density (PHD) filter [1]. The most important implementation of the PHD framework is the Gaussian mixture PHD (GM-PHD) filter [3], which simplifies the difficulty of integral solution of a PHD by the Gaussian mixture method. The Poisson multi-Bernoulli mixture (PMBM) filter was proposed in [4], which is the closed-form solution for Poisson birth model and can provide data association result. The PMBM model has been combined with Gamma Gaussian inverse Wishart (GGIW) model and applied to extended target tracking [26]. Other important RFS based tracking methods include cardinalized PHD (CPHD) [5], particle PHD [6], etc.

However, previous studies have found that the RFS based filters cannot provide track association results, which greatly limits the application of RFS based filters. Recently, some RFS methods have been proposed to solve this problem, such as [4] and [27]. Another important solution is the LRFS method [2], which establishes a strict mathematical model for the trajectory. The important implementations of LRFS are delta GLMB (δ-GLMB) [7] and labeled multi-Bernoull (LMB) [8] filters, which can achieve more accurate tracking result than PHD filters and provide target trajectories. Therefore, the GLMB/LMB filtering technologies are state-in-art tracking frameworks in recent years. Other important discussions on GLMB/LMB can be found in [9]-[17]. Compared with RFS based filters, LRFS based filters achieve better tracking accuracy, but need more computational cost. Therefore, [18] presented an efficient GLMB (E-GLMB) filter to reduce the computational complexity using the Gibbs sampling method. The E-GLMB filter is an important work in recent years and has shown superior performance in MTT. However, in complex scenarios, such as scenarios with high clutter rate, high missed detection probability, and measurement-outliers, the performance of the E-GLMB filter will decrease significantly. Note that the measurement-outliers are generated by the sudden interference of the sensor at a certain time, i.e. the mathematical expectation of the measurement noise increases at that time.

Tracking in complex scenarios is an important problem of MTT. Particle filtering technology is a method to solve the complex or nonlinear tracking problem, and it has been discussed in GLMB [23] and PHD [24] filters. However, due to the large computational cost, it is difficult to implement a particle filter in real scenarios. Student-T distribution filtering technology is an efficient method to handle the complex linear scenarios. Compared with the Gaussian distribution, the Student-T distribution achieves better performance in dealing with outliers, thus it can improve the tracking accuracy in complex scenarios. The Student-T distribution filtering technology is a hot topic in recent years [28]-[29] and Student-T-distribution-based PHD filters have been derived in [25].

This paper proposes a Student-T distribution variational Bayesian E-GLMB (TDVB-E-GLMB) filter and a multi-target smoother. The TDVB [19] is a robust filtering technology compared with Bayesian filtering framework, thus the proposed method can improve the

tracking performance in complex scenarios. For example, when the E-GLMB filter is applied to vehicle tracking, the strong light, fog and other environmental factors may lead to a large number of clutter in the radar system, resulting in the decline of tracking accuracy. The proposed method use the TDVB framework to improve the robustness of the filtering process, thus the influence of clutter on accuracy will be reduced. Moreover, a corresponding smoothing method is proposed for GLMB framework to improve the overall tracking accuracy. The contributions of this work are summarized as follows:

(1)This work applies TDVB filtering technology [19] to the E-GLMB filter. A new likelihood function is derived to replace the Gaussian likelihood. Therefore, the proposed TDVB-E-GLMB filter can significantly improve the tracking accuracy in complex scenarios.

(2)A multi-target TDVB smoothing method is proposed for GLMB framework. This method considers the missed detected targets and pseudo targets based on trajectories provided by the GLMB framework. Therefore, the proposed TDVB smoother can significantly improve the performance of a GLMB filter.

Code is published on Github [22]. Further analysis of the proposed TDVB-E-GLMB filter can be done through this code.

The paper is organized as follows. Section II introduces the E-GLMB filter, variational Bayesian inference, and TDVB filtering technology. Section III introduces the proposed TDVB-E-GLMB filter. Section IV introduces the proposed multi-target smoothing method for a GLMB filter. Section V shows the simulation results. Section VI contains our conclusions.

2. Background

2.1 Problem Statements

In general, the system function of the tracking process is described as

$$x_k = f(x_{k-1}, v_{k-1}), \tag{1}$$

$$z_k = \tau(x_k, w_k), \tag{2}$$

where $f(\cdot)$ denotes the state model and $\tau(\cdot)$ denotes the observation model. v_{k-1} and w_k are corresponding noises. In a standard RFS or LRFS filter, the state and observation models are usually defined as Gaussian model, due to the simple mathematical form and low computational complexity. Also, the state updating process is a Bayesian recursive process. Therefore, the state model and measurement model can be written as

$$x_k \mid x_{k-1} \sim \mathrm{N}\left(f(x_{k-1}), Q_k\right), \tag{3}$$

$$z_k \mid x_k \sim \mathrm{N}\left(\tau(x_k), R_k\right). \tag{4}$$

However, the Gaussian model also leads to the lack of robustness in complex scenarios such as low detection rate, high clutter rate, and measurement-outliers. The measurement-outliers is the measurement at a certain scan when expectation of measurement error increases by the strong interference. For example, the expectation of measurement noise is usually R_0, but the expectation increases to a large R_1 at the time of strong interference. The measurement obtained by the sensor system with R_1 as the expected error calls measurement-outlier.

In the complex scenarios, using Gaussian model to describe the measurement generation will lead to the lack of robustness. Compared with Gaussian distribution, the Student-T is a distribution with lower peak and higher tail, thus it is easier to handle outliers. In [25] and [28]-[29], the Student-T distribution was employed to handle the measurement-outliers and

the results show its better performance. However, the Student-T distribution can only perform well for measurement-outliers, but it is not robust to other complex scenarios. Moreover, compared with the Bayesian recursive process, the VB recursive process has more robustness. Therefore, the combination of Student-T distribution and VB process will greatly improve the tracking robustness, and this work has been discussed in [19].

In this work, we employ the TDVB model proposed in [19] to modify the E-GLMB filter. The state and measurement models are defined as the Gaussian model with the random auxiliary variable, i.e. (18)-(19). Note that this measurement model allows for measurement-outliers. The state updating model is modified by TDVB filtering technology. The proposed new likelihood is calculated by KL divergence, i.e. formula (35)-(37).

2.2 The E-GLMB Filter

The E-GLMB filter uses the Gibbs sampling and joint prediction-update approaches to reduce the high computational cost of the δ-GLMB filter. To achieve the goal of joint prediction and update, the E-GLMB filtering density is given by

$$\pi_{Z_+}(\mathrm{X}) \propto \Delta(\mathrm{X}) \sum_{\xi, I_+, \theta_+} w_{Z_+}^{(\xi, I_+, \theta_+)} \delta_{I_+}\left[L(\mathrm{X})\right]\left[p_{Z_+}^{(\xi,\theta_+)}\right]^{\mathrm{X}}, \tag{5}$$

where, ξ denotes the history of association. θ denotes the association map. $L(\mathrm{X})$ denotes the label set and I denotes the set of object labels. $\delta_{I_+}\left[L(\mathrm{X})\right]$ means that if the new label set $I_+ = L(\mathrm{X})$, the value is 1, otherwise, 0. $p_{Z_+}^{(\xi,\theta_+)}$ is the component function with its weight $w_{Z_+}^{(\xi, I_+, \theta_+)}$. X is the target state, and exponential notation f^{X} means that $\prod_{x\in\mathrm{X}} f(x)$. $\Delta(\mathrm{X})$ means that $\delta_{|X|}\left[\left|L(\mathrm{X})\right|\right]$, where $|\cdot|$ denotes the number of elements in a set. Each part of the formula (5) is given by

$$\begin{aligned} w_{Z_+}^{(\xi, I_+, \theta_+)} = {} & 1_{\Theta_+(I_+)}(\theta_+)\left[1-\bar{P}_S^{(\xi)}\right]^{I-I_+}\left[\bar{P}_S^{(\xi)}\right]^{I\cap I_+} \\ & \times\left[1-r_{B,+}\right]^{\mathrm{B}_+-I_+} r_{B,+}^{\mathrm{B}_+\cap I_+}\left[\bar{\psi}_{Z_+}^{(\xi,\theta_+)}\right]^{I_+} \end{aligned}, \tag{6}$$

$$\bar{P}_S^{(\xi)}(h) = \left\langle p^{(\xi)}(\cdot, h), P_S(\cdot, h)\right\rangle, \tag{7}$$

$$\bar{\psi}_{Z_+}^{(\xi,\theta_+)}(h_+) = \left\langle \bar{p}_+^{(\xi)}(\cdot, h_+), \psi_{Z_+}^{(\theta_+(h_+))}(\cdot, h_+)\right\rangle, \tag{8}$$

$$\begin{aligned} \bar{p}_+^{(\xi)}(x_+, h_+) = {} & 1_{C_+}(h_+)\frac{\left\langle P_S(\cdot, h_+) f_+(x_+ \mid \cdot, h_+), p^{(\xi)}(\cdot, h_+)\right\rangle}{\bar{P}_S^{(\xi)}(h_+)} \\ & + 1_{\mathrm{B}_+}(h_+)\, p_{B,+}(x_+, h_+) \end{aligned}, \tag{9}$$

where, $I \in F(C), I_+ \in F(C_+), \theta_+ \in \Theta_+$. $1_{C_+}(h_+)$ means that if $h_+ \subset C_+$, the value is 1, otherwise, 0. B denotes the track label space for target birth. $r_{B,+}$ is probability that a new object with label is born. $\psi_{Z_+}^{(\theta_+(h_+))}(\cdot, h_+)$ is a likelihood function which is defined in [18, section II.A]. $P_S(\cdot, h)$ denotes the target survival probability. In the specific implementation process, the GLMB component is a Gaussian mixture form

$$p_{Z_+}^{(\xi,\theta_+)}(x, h \mid Z_+) = \sum_{i=1}^{J^{(\xi)}(h)} \frac{w_{Z_+,i}^{(\xi,\theta)}(h)}{\eta_{Z_+}^{(\xi,\theta)}(h)} \mathrm{N}\left(x; m_{Z_+,i}^{(\xi,\theta)}(h), P_{Z_+,i}^{(\xi,\theta)}(h)\right), \tag{10}$$

where, $\mathrm{N}(\cdot)$ denotes the Gaussian distribution. $m_{Z_+,i}^{(\xi,\theta)}(h)$ and $P_{Z_+,i}^{(\xi,\theta)}(h)$ denote the Gaussian mean and covariance matrix. $w_{Z_+,i}^{(\xi,\theta)}(h)$ is the weight of the Gaussian mixture. $\eta_{Z_+}^{(\xi,\theta)}(h)$ denotes the normalization.

2.3 The Variational Bayesian Inference

The key method of the VB technology is to use a function $\Psi(Z)$ to approximate the posterior probability density $g(Z \mid X)$ by computing the Kullback-Leibler (KL) divergence, i.e.,

$$\ln g(X) = \mathrm{KL}(\Psi \| g) + L(\Psi). \tag{11}$$

Assuming that $\Psi(Z) = \prod_i \Psi(z_i)$, then $L(\Psi)$ can be written as

$$\begin{aligned} L(\Psi) = &-\mathrm{KL}\left(\Psi(z_j) \| \Psi^*(z_j)\right) \\ &+ \prod_{i:i\neq j} H\left(\Psi(z_j)\right) - const \end{aligned}. \tag{12}$$

Obviously, the function $L(\Psi)$ will maximize when $-\mathrm{KL}\left(\Psi(z_j) \| \Psi^*(z_j)\right) = 0$, thus we obtain

$$\Psi(z_j) = \Psi^*(z_j) = \frac{\exp\left(E_{i\neq j}\left[\ln g(Z, X)\right]\right)}{\text{normalize constant}}. \tag{13}$$

As a result, $\Psi(Z)$ can be approximated to $g(Z \mid X)$.

2.4 The TDVB Filtering Technology

Compared with the Gaussian distribution, the Student-T distribution achieves better performance in dealing with complex scenarios with strong interference. Also, the variational Bayesian processes can deal with nonlinear cases by iteration. Therefore, the Student-T distribution variational Bayesian filtering technology is a robust approach to handle non-linear scenarios, especially in measurement-outlier scenarios.

According to [19], the system model is given by

$$x_k \mid x_{k-1} \sim \mathrm{N}\left(f(x_{k-1}), Q_k\right), \tag{14}$$

$$z_k \mid x_k \sim \mathrm{St}\left(\tau(x_k), R_k, \nu\right), \tag{15}$$

where, x_k denotes the target state. z_k denotes the measurement. $\mathrm{St}(\cdot)$ denotes the Student-T distribution. $f(\cdot)$ and $\tau(\cdot)$ are dynamic and observation models, respectively. Q_k and R_k are process and measurement noise, respectively. ν is the degree of freedom of Student-T distribution. The probability density of Student-T distribution can be written as

$$\mathrm{St}\left(\tau(x_k), R_k, \nu\right) \propto \left(1 + \frac{1}{\nu}\aleph^T R_k^{-1} \aleph\right)^{-\frac{\nu+d}{2}}, \tag{16}$$

$$\aleph = \left(z_k - \tau(x_k)\right), \tag{17}$$

where, d is the dimensions of physical space. The formula (16) can be approximately expressed as a hierarchical Gaussian form by employs an auxiliary random variable λ_k, thus formula (15) become

$$z_k \mid x_k, \lambda_k \sim \mathrm{N}\left(\tau(x_k), \frac{1}{\lambda_k} R_k\right), \tag{18}$$

$$\lambda_k \sim \text{Gamma}\left(\nu/2, \nu/2\right), \tag{19}$$

where, $\text{Gamma}(\cdot)$ denotes the Gamma distribution. Since (14) is a standard Gaussian form, the state prediction steps can use a standard Kalman filtering technology. According to (18)-(19), the Student-T distribution is approximated to a Gaussian distribution with the random auxiliary variable, thus the sate updating can be given by

$$m_k = m_{k|k-1} + K_k\left(z_k - \mu_k\right), \tag{20}$$

$$P_k = P_{k|k-1} + K_k S_k K_k^T, \tag{21}$$

where,

$$\mu_k = \int \tau\left(x_k\right) \mathrm{N}\left(x_k \mid m_{k|k-1}, P_{k|k-1}\right) dx_k, \tag{22}$$

$$\begin{aligned} S_k = &\int \left(\tau\left(x_k\right) - \mu_k\right)\left(\tau\left(x_k\right) - \mu_k\right)^T \\ &\times \mathrm{N}\left(x_k \mid m_{k|k-1}, P_{k|k-1}\right) dx_k + \frac{1}{\lambda_k} R_k \end{aligned}, \tag{23}$$

$$\begin{aligned} K_k = &\int \left(x_k - m_{k|k-1}\right)\left(\tau\left(x_k\right) - \mu_k\right)^T \\ &\times \mathrm{N}\left(x_k \mid m_{k|k-1}, P_{k|k-1}\right) dx_k \cdot S_k^{-1} \end{aligned}, \tag{24}$$

$$\begin{aligned} \overline{\gamma}_k = tr\Bigg(&\int \left(z_k - \tau\left(x_k\right)\right)\left(z_k - \tau\left(x_k\right)\right)^T \\ &\times \mathrm{N}\left(x_k \mid m_k, P_k\right) dx_k R_k^{-1}\Bigg) \end{aligned}, \tag{25}$$

$$\lambda_k = \left(\nu + d\right)/\left(\nu + \overline{\gamma}_k\right), \tag{26}$$

where, $tr(\cdot)$ denotes the trace of the matrix. d is the dimensions of physical space. Note that, formulas (20)-(26) represent an iterative process using VB filtering technology. Therefore, given the number of iterations, formulas (20)-(21) will converge through a changing λ_k.

3. The TDVB-E-GLMB Filter

The use of Gaussian component leads to the decrease of E-GLMB filter′s performance in the complex scenarios. This section proposed an implementation approach using TDVB filtering technology to enhance the robustness of the E-GLMB filer.

3.1 The Key Method

A GLMB filter uses the Kalman filtering technology to update the kinematical states of components, and uses the Gaussian distribution to calculate the likelihood. Then, this information will be employed to update the track table. The form of a track table is shown in **Fig. 1**. $I^{(h)}$ is the set of labels, $w^{(h)}$ is the weight of a track, and $p^{(h)}$ denotes the densities. In the δ-GLMB filter, all the target kinematical states are stored in another memory, but the E-GLMB filter does not need to store all of them by joint prediction and update. The density $p^{(h)}$ is related to the method of the filtering technology for updating target kinematical state, and its physical meaning is the spatial likelihood of the measurement.

$I^{(1)}$	$I^{(2)}$	···	$I^{(h)}$
$w^{(1)}$	$w^{(2)}$	···	$w^{(h)}$
$p^{(1)}$	$p^{(2)}$	···	$p^{(h)}$

Fig. 1. Example of track component.

In a standard GLMB filter, $p^{(h)}$ is the Gaussian mixture form and the details can be found in [7, section IV.B]. However, it is found in the simulation that the robustness of Gaussian mixture form is insufficient in complex scenarios, which leads to a significant decline in tracking accuracy. Therefore, the key method of proposed approach is to employ the more robust TDVB filtering technology to update the target kinematical state. Then, deriving a corresponding $p^{(h)}$ to improve the accuracy of the track table. Note that the proposed method is derived as a linear filter, and therefore can only be guaranteed to work for linear state transition and measurement models.

3.2 Implementation of the TDVB-E-GLMB Filter

Since the main contribution of this work is to implement the TDVB filtering technology and propose a new density, the trajectory density is the same as that in [18]. The system equations of the proposed method are (14)-(15). According to (14), the state prediction of TDVB approach is the Gaussian form, thus the state prediction step of the proposed TDVB-E-GLMB filter is the same as in [7, section V]. According to (15)-(17) and VB filtering technology, the density $p^{(h)}$ can be written in the form of Student-T distribution, i.e.

$$\begin{aligned} p_k^{(h)}(x,l \mid Z) &= \sum_{i=1}^{J} \frac{w_{k|k,i}^{(h)}(l)}{\eta_{k|k}^{(h)}(l)} \mathrm{St}\left(m_{k|k,i}^{(h)}(l), R_k + P_{k|k,i}^{(h)}(l), v\right) \\ &\approx \sum_{i=1}^{J} \frac{w_{k|k,i}^{(h)}(l)}{\eta_{k|k}^{(h)}(l)} \mathrm{N}\left(m_{k|k-1,i}^{(h)}(l), \frac{1}{\lambda_k} R_k + P_{k|k-1,i}^{(h)}(l)\right) \\ &\quad \times \mathrm{Gamma}\left(\lambda_k \mid \frac{v}{2}, \frac{v}{2}\right) \end{aligned}, \tag{27}$$

where,

$$m_{k|k,i}^{(h)}(l) = \begin{cases} m_{k|k-1,i}^{(h)}(l) + K_k u_k^{(h)}, & if\ \theta(l) > 0 \\ m_{k|k-1,i}^{(h)}(l), & if\ \theta(l) = 0 \end{cases}, \tag{28}$$

$$P_{k|k,i}^{(h)}(l) = \begin{cases} P_{k|k-1,i}^{(h)}(l) + K_k S_k^{-1} K_k, & if\ \theta(l) > 0 \\ P_{k|k-1,i}^{(h)}(l), & if\ \theta(l) = 0 \end{cases}, \tag{29}$$

where, according to (22)-(24), the innovation parameters can be calculated by

$$u_k^{(h)} = z_k - H_k m_{k|k-1,i}^{(h)}(l), \tag{30}$$

$$K_k = \left(P_{k|k-1,i}^{(h)}(l) \cdot H_k^T\right) S_k^{-1}, \tag{31}$$

$$S_k = H_k P_{k|k-1,i}^{(h)}(l) H_k^T + \frac{1}{\bar{\lambda}_k} R_k, \tag{32}$$

where, H_k is the observation matrix. The auxiliary random variable $\bar{\lambda}_k$ can be obtained by

$$\bar{\lambda}_k = (v+d) \Big/ \left(v + \mathrm{tr}\left\{ \left(u_k^{(h)} \left(u_k^{(h)} \right)^T \right) R_k^{-1} \right\} \right). \tag{33}$$

Note that, formulas (28)-(32) need to be iterated to make $\bar{\lambda}_k$ converge. $p_k^{(h)}(x, l \mid Z)$ can be calculated by

$$w_{k|k,i}^{(h)}(l) = \begin{cases} \dfrac{p_D w_{k|k-1,i}^{(h)}(l) q_{k,i}^{(h)}(z_k, \lambda_k, l)}{\kappa(z_k, l)}, & if\ \theta(l) > 0 \\ 1 - p_D, & if\ \theta(l) = 0 \end{cases}, \tag{34}$$

where, p_D is the target missed detection probability. $\kappa(z_k, l)$ is the density of clutters. $w_{k|k-1,i}^{(h)}(l)$ is the prediction weight and it is equal to $w_{k-1|k-1,i}^{(h)}(l)$ (or birth weight when $w_{k-1|k-1,i}^{(h)}(l)$ does not exist). $q_{k,i}^{(h)}(z_k, \lambda_k, l)$ is the measurement likelihood. Since formulas (28)-(33) are TDVB form, $q_{k,i}^{(h)}(z_k, \lambda_k, l)$ will be the same as that in [19], i.e.

$$q_{k,i}^{(h)}(z_k, \lambda_k, l) \approx \phi_{k,i}^{(h)}(z_k, \lambda_k, l) \varphi_{k,i}^{(h)}(\lambda_k, l). \tag{35}$$

where,

$$\begin{aligned} \varphi_{k,i}^{(h)}(\lambda_k, l) \propto \exp\Bigg(& -\frac{1}{2} \bar{\lambda}_k \mathrm{tr}\left\{ \left(u_k^{(h)} \left(u_k^{(h)} \right)^T \right) R_k^{-1} \right\} \\ & + \left(\frac{v+d}{2} - 1 \right) \ln \bar{\lambda}_k - \frac{v \bar{\lambda}}{2} \Bigg) \end{aligned}, \tag{36}$$

$$\begin{aligned} & \phi_{k,i}^{(h)}(z_k, \lambda_k, l) \propto \\ & \exp\Bigg(-\frac{1}{2} \lambda_k \left(z_k - H_k m_{k|k,i}^{(h)}(l) \right)^T R_k^{-1} \left(z_k - H_k m_{k|k,i}^{(h)}(l) \right). \\ & \quad -\frac{1}{2} \left(\varepsilon_k^{(h)} \right)^T \left(P_{k|k-1,i}^{(h)}(l) \right)^{-1} \varepsilon_k^{(h)} \Bigg) \end{aligned} \tag{37}$$

The process of computing GLMB parameters

$$\left\{ \left(I_+^{(h_+)}, w_+^{(h_+)}, p_+^{(h_+)} \right) \right\}_{h_+=1}^{H_+}$$

is the same as the E-GLMB filter. Since it will take a long time to introduce the updating method of these trajectory parameters and this work has not contributed to this part, the specific formulas will not be introduced in detail in this work. The details can be found in [18, section 3 E].

In summary, this section improve the update steps of target kinematical state by formulas (28)-(33). Also, a new likelihood function is given by formulas (35)-(37). The code was published on GitHub [22] to help readers understand full details. The proof process of formula (36) and (37) can be found in the Appendix. In the next subsection, an effective multi-target TDVB smoother will be introduced.

4. The Multi-target TDVB Smoother

Smoothing the estimated state can effectively improve the performance of a tracker. The GLMB is an LRFS based tracking framework and can provide accurate trajectory information, thus it is feasible to use a smoothing method to improve the tracking performance. In this

section, we proposed an MTT smoothing method based on TDVB smoothing technology.

4.1 The Key Method

The proposed smoothing method is based on the work in [19] and [20] (which can only be used in single target scenarios). The MTT scenario is different from a single target scenario, because it has missed detection, pseudo-track and data association problems. Therefore, we modify the single target smoother and apply it to the GLMB filtering framework.

The method in [19] and [20] is to smooth the estimated state after filtering. However, in MTT scenarios, the estimated state is obtained by the multiple hypotheses filtering process, which means that the estimated state is greatly affected by other targets, missed detection, and clutters. **Fig. 2** shows an example in which the accuracy of multiple hypothesis filtering is reduced due to the closely-spaced targets. The gray arrow represents the movement direction. The forks and the dots represent the measurements produced by two targets at different scans, respectively. In the process of multiple hypothesis tracking, the measurement of two targets will affect the tracking accuracy. Further considering the influence of missed detection and clutters, as a result, the state estimation accuracy of filtering two targets will be lower than that of filtering a single target.

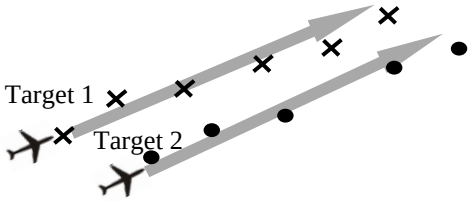


Fig. 2. Example of closely-spaced targets. Forks and the dots denote measurements of the two targets, respectively.

Therefore, in MTT scenarios, our view is not to smooth the filtered target state, but to make full use of the measurement information collected in the filtering and track association process. Our key method is: 1) storing the measurement set used to update the track during the tracking process, thus we obtain the measurement set of each target; 2) filtering the measurement set of each target, thus more accurate state estimation results can be obtained; 3) smoothing the state using TDVB smoothing technology. An example of a measurement set is all the forks in **Fig. 2**.

The details of the application of the proposed idea in a GLMB filter will be introduced in the next sub-section. Note that the proposed method is not only suitable for the GLMB filtering framework, but also to other filters that can provide target tracks.

4.2 The TDVB smoother for the GLMB filter

The standard GLMB filter can provide the labels and states, but not the corresponding measurement set of each target. Therefore, measurement storing steps should be included in the state update part. Formulas (23)-(27) are the steps of updating the state using a measurement z_k, thus a measurement memory $\hat{Z}_k^{(h)}$ of the h-th track can be included in these steps as $\hat{Z}_k^{(h)} \leftarrow z_k$. When a GLMB component is deleted due to its small weight, the memory also needs to be deleted. As a result, at time k, all tracks output by the GLMB filter will have a corresponding measurement set $\hat{Z}_k$. Note that $\hat{Z}_k$ is only a memory in the GLMB filtering

process and does not affect other steps of the GLMB filter.

At time k, the measurement memory can be written as

$$\left\{\left\{z_k^{(h)}\right\}_{k=D_{\min}^{(h)}}^{D_{\max}^{(h)}}\right\}_{h=1}^{N_{traj}},$$

where, $D_{\min}^{(h)}$ and $D_{\max}^{(h)}$ are the beginning and ending time interval of the h track, respectively. N_{traj} is the number of all the tracks that have appeared. $z_k^{(h)}$ denotes the corresponding measurement used to update the component states at time k. **Fig. 3** shows an example where $N_{traj}=2$. The memory stores the measurements corresponding to the two tracks. Note that if no measurement is used to update the component at some scan (i.e. it is a missed detection component), then $z_k^{(h)}=\varnothing$.

k	1	2	3	4	5	6	7	8
$h=1$	$z_1^{(1)}$	$z_2^{(1)}$	$z_3^{(1)}$	$z_4^{(1)}$	$z_5^{(1)}$	$z_6^{(1)}$		
$h=2$			$z_3^{(2)}$	$z_4^{(2)}$	$\varnothing$	$z_6^{(2)}$	$z_7^{(2)}$	$z_8^{(2)}$

Fig. 3. Example of the measurement memory.

For each measurement set

$$\left\{z_k^{(h)}\right\}_{k=D_{\min}^{(h)}}^{D_{\max}^{(h)}},$$

the first step is to determine whether the track is generated from the clutters. Given a number N_{pse}, the trajectory should be deleted if

$$\left|\left\{z_k^{(h)}\right\}_{k=D_{\min}^{(h)}}^{D_{\max}^{(h)}}\right| \le N_{pse}, \tag{38}$$

otherwise, the new target states of the track can be calculated by

$$m_{k|k}'^{(h)}=\begin{cases} m_{k|k-1}'^{(h)}+K_k u_k, \text{ if } z_k^{(h)}\neq\varnothing \\ m_{k|k-1}'^{(h)}, \text{ if } z_k^{(h)}=\varnothing \end{cases}, \tag{39}$$

$$P_{k|k}'^{(h)}=\begin{cases} P_{k|k-1}'^{(h)}+K_k S_k^{-1}K_k^T, \text{ if } z_k^{(h)}\neq\varnothing \\ P_{k|k-1}'^{(h)}, \text{ if } z_k^{(h)}=\varnothing \end{cases}, \tag{40}$$

where, $m_{k|k}'^{(h)}$ and $P_{k|k}'^{(h)}$ are the new states calculated by the measurement set of the h-th track. The calculation approach of innovation parameters $u_k^{(h)}$, K_k, and S_k is the same as that in (25)-(27). The formulas (39)-(40) consider the missed detection. Therefore, formulas (38)-(40) consider the scenarios of the pseudo target and missed detection. Note that, the initialization of formulas (39)-(40) is determined by the state provided by the GLMB filter, i.e.

$$m_{D_{\min}^{(h)}}'^{(h)}=\Xi\left(\left\{m_k^{(h)}\right\}\right), \tag{41}$$

$$P_{D_{\min}^{(h)}}'^{(h)}=P_0^{(h)}, \tag{42}$$

where, $\Xi(\cdot)$ are defined as the first non-empty element in a set. In fact, (41) is to deal with the situation that $m_{D_{\min}^{(h)}}^{(h)}$ provided by a GLMB filter is an empty set. $P_0^{(h)}$ is the initial error matrix of the birth component of the GLMB filter.

The initialization of the proposed smoother is

$$m_{S,D_{\max}^{(h)}}^{\prime(h)} = m_{D_{\max}^{(h)}}^{\prime(h)}, \tag{43}$$

$$P_{S,D_{\max}^{(h)}}^{\prime(h)} = P_{D_{\max}^{(h)}}^{\prime(h)}. \tag{44}$$

The smoothing process is

$$m_{S,k|k}^{\prime(h)} = m_{k|k}^{\prime(h)} + G\left(m_{S,k+1|k}^{\prime(h)} - m_{k+1|k}^{\prime(h)}\right), \tag{45}$$

$$P_{S,k|k}^{\prime(h)} = P_{k|k}^{\prime(h)} - G\left(P_{k+1|k}^{\prime(h)} - P_{S,k+1|k}^{\prime(h)}\right)G^T, \tag{46}$$

$$G = P_{k|k}^{\prime(\ell)} F_k^T \left(P_{k+1|k}^{\prime(h)}\right)^{-1}. \tag{47}$$

Given a number of iterations N_{ite}, the proposed TDVB smoother can be implemented to a GLMB filter. A smoother is not a real-time filter, thus a smoothing window should be set. In fact, the size of the window will affect the real-time performance and smoothing accuracy. Therefore, different smoothing windows should be selected according to different scenarios. The proposed method is based on the assumption that the window is large enough, thus the case of small window is not discussed. Research about this will be carried out in the future. Discussion on window size selection will be held in the next work. The pseudo-code is shown in **Table 1** and the MATLAB code can be found in [22].

Table 1. Pseudo Code of TDVB Smoother

INPUT: $\left\{z_k^{(h)}\right\}_{k=D_{\min}^{(h)}}^{D_{\max}^{(h)}}$, $\Xi\left(\left\{m_k^{(h)}\right\}\right)$, $P_0^{(h)}$, N_{pse}, and N_{ite}.

OUPUT: $m_{S,k|k}^{\prime(h)}$, and $P_{S,k|k}^{\prime(h)}$.

Delete pseudo-tracks using the inequality (38);
Initialize $\bar{\lambda}_k = 1$;
for i = 1 to N_{ite}
    Initialize the state using (41)-(42);
    for k = $D_{\min}^{(h)}$ to $D_{\max}^{(h)}$
        Compute the new target states using (39)-(40) and (25)-(27);
    end for
    Initialize the smoother using (43)-(54);
    for k = $D_{\max}^{(h)} - 1$ to $D_{\min}^{(h)}$
        Smooth the states using (45)-(47);
    end for
    for k = $D_{\min}^{(h)}$ to $D_{\max}^{(h)}$
        Compute the auxiliary random variable using (28);
    end for
end for

5. Simulation Result

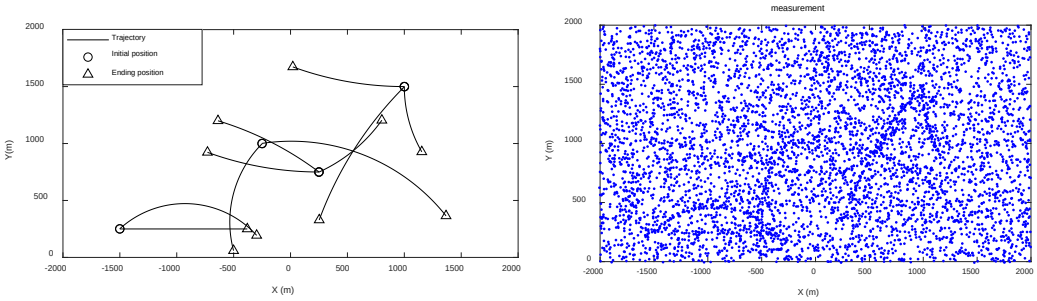
In this section, the proposed methods are compared with the PHD filter [1], CPHD filter [5], and E-GLMB filter [18] with different parameters. The simulation data are generated according to [18], which simulates radar measurements. The target trajectories contain various complex cases are shown in **Fig. 4. (a)**, and the simulated data of 100 scans (clutter rate is 50 and $p_D = 0.75$) is shown in **Fig. 4. (b)**. The Scenario1 is a low interference scenario with low clutter rate, high detection probability, and no measurement-outlier. The Scenario2 is a complex scenario with a high cluter rate, low detection probability, and measurement-outlier. The section 5.3 shows the performance under different parameters.

The accuracy of each approach is judged by the OSPA metric [21], which does not consider the track association accuracy, thus it can be used to judge the performance of PHD and CPHD filters. Moreover, the OSPA(2) metric [30], which considers the track association accuracy, is used to more accurately judge the performance of GLMB filters. The simulation environment is as follows: *Software: MATLAB 2019(a); Computer system: Windows 10; CPU: Intel(R) Core(TM) i7-6700HQ; RAM: 8GB.* The specific tracking parameters are shown in TABLE II.

Table 2. The Parameters of Different Scenarios

	Meaning	Scenario1	Scenario2
Invariant parameters	Surveillance volume (m^2)	$\mathcal{S}=4000\times 2000$	
	Scanning interval (s)	$T_s = 1$	
	Survival probability	$p_S = 0.99$	
	Process noise	$Q_k = \mathrm{diag}([1,0,1,0])$	
	Measurement noise	$R_k = \mathrm{diag}([10^2, 10^2])$	
	Freedom of Student-T	$v = 10$	
	Pseudo track threshold	$N_{pse} = 1$	
	VB iteration times	$N_{ite} = 10$	
Birth	Weight	$w_0 = 0.01$	
	Kinematical state	$m_0 = [u_{x,0}, 0, u_{y,0}, 0]^T$	
	Covariance matrix	$P_0 = \mathrm{diag}([100,100,100,100])$	
Interference	Outlier noise		$\Re_k = \mathrm{diag}([50^2, 50^2])$
	Outlier probability	0	10%
	Clutter rate (per scan)	20	50
	Detection probability	$p_D = 0.95$	$p_D = 0.75$

In the table, $u_{x,0}$ and $u_{y,0}$ denote the coordinates of X and Y axes, respectively. The outlier noise $\Re_k$ with probability 10% means that: the measurement noise R_k of each target per scan has a 10% probability of being replaced by $\Re_k$.



(a) (b)

Fig. 4. Target trajectories and simulated data.

5.1 Results of Scenario1

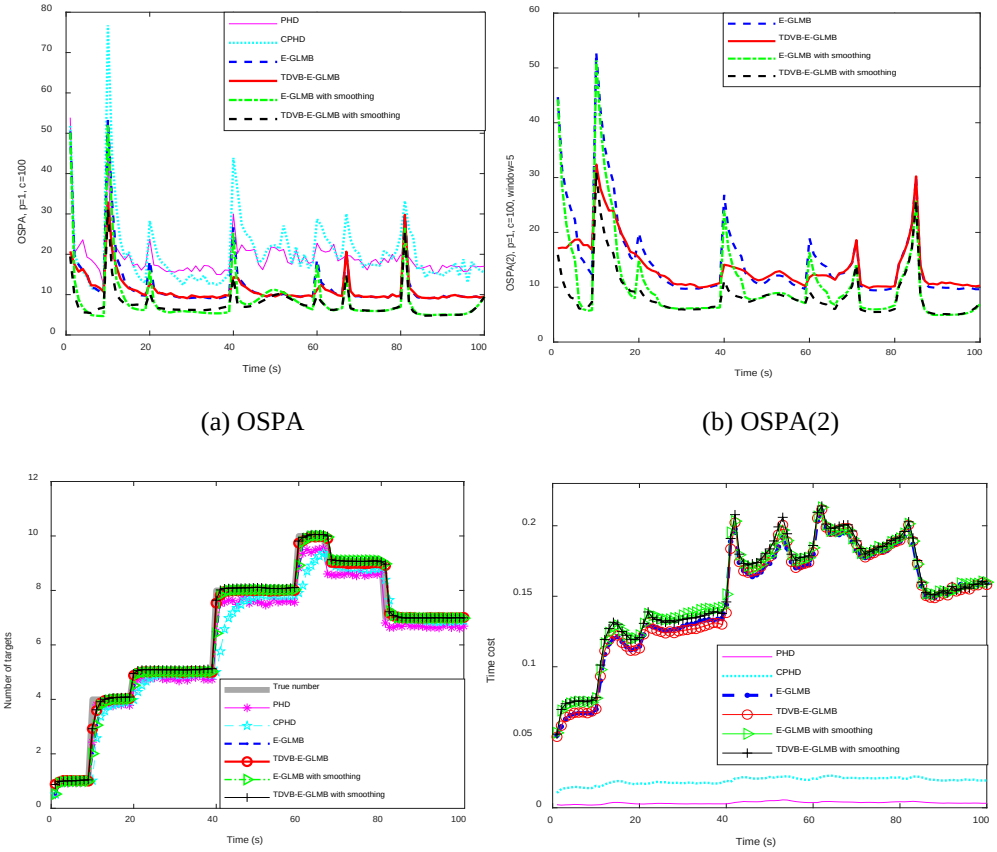
In this subsection, the performance of each approach is judged by average result of 200 Monte Carlo runs.

Fig. 5. (a) shows the average OSPA results. The values of PHD and CPHD filters are obviously larger than GLMB filters, thus the tracking accuracy of a GLMB filter is higher than that of PHD and CPHD filters.

Fig. 5. (b) shows the average OSPA(2) results. The values of proposed TDVB-E-GLMB filter are significantly lower than that of E-GLMB filter when the data reaches the peaks. These data peaks are caused by the crossing and birth of targets, thus the proposed TDVB-E-GLMB filter is more robust in complex cases. However, the values of TDVB-E-GLMB are slightly higher than E-GLMB sometimes. It means that although the proposed method has strong robustness, the track association accuracy is slightly lower than the E-GLMB filter in ideal tracking environment. Moreover, the proposed smoother can significantly improve the accuracy of the two GLMB filters, and the smoothing results for TDVB-J-GLMB achieve the best performance.

Fig. 5. (c) shows the average estimation results of the number of targets. It can be seen that the GLMB trackers achieve high accuracy in this scenario, but the accuracy of PHD and CPHD filters is insufficient.

Fig. 5. (d) shows the average time costs. The values of the PHD and CPHD filters are significantly lower than GLMB filters. Therefore, the PHD and CPHD are low precision high speed filters. The values of proposed TDVB-E-GLMB are slightly lower than E-GLMB filter sometimes. In fact, the use of TDVB filtering technology will increase the updating computation time of each component. However, the improvement of the filtering accuracy leads to the decrease of the number of components, thus the overall time cost is slightly lower than that of the E-GLMB filter. Moreover, the proposed smoother only slightly increases the time cost of a GLMB filter (the corresponding values in the figure are obtained by filtering time + smoothing time). Therefore, the proposed smoother can greatly improve the filtering accuracy with minimal computational cost.



(a) OSPA (b) OSPA(2)

(c) Number of target (d) Time cost

Fig. 5. Average performance of 200 Monte Carlo runs in Scenario1.

5.2 Results of Scenario2

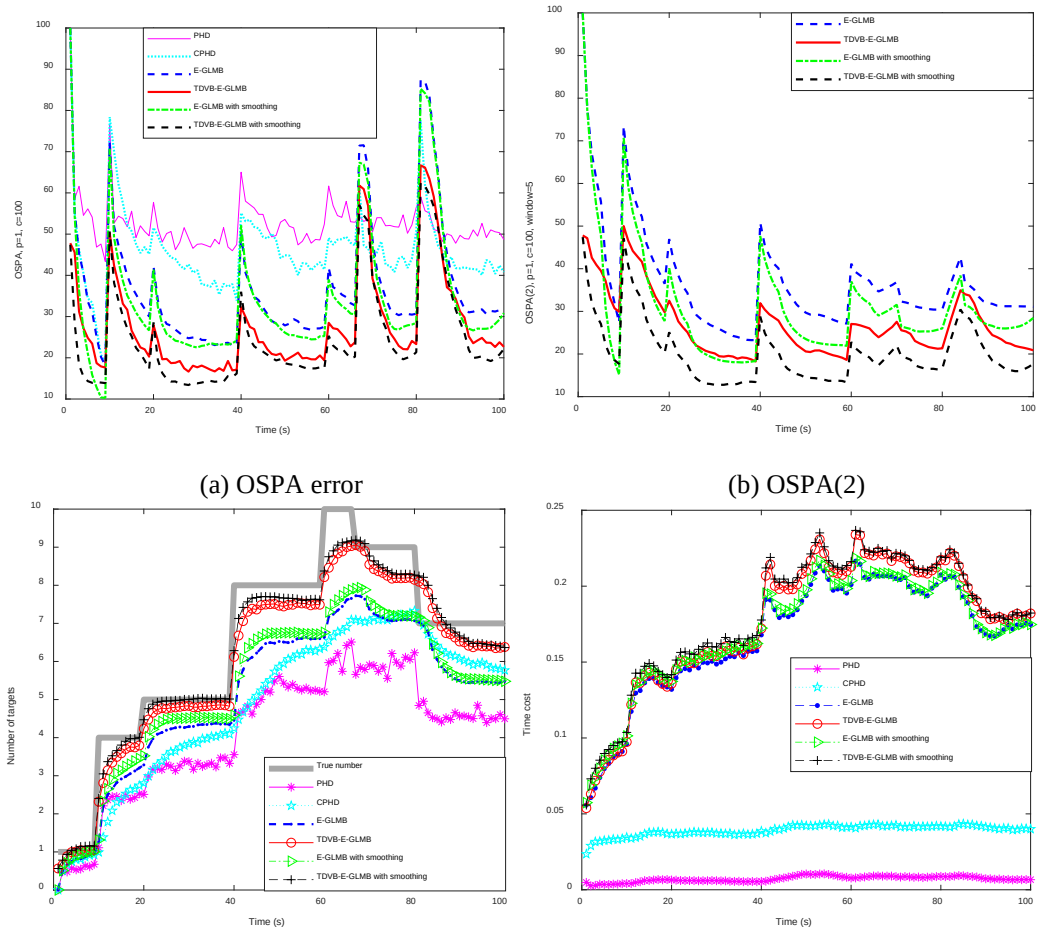
Scenario2 is a complex scenario, thus it can judge the robustness of different approaches. The performances are judged by the average results of 200 Monte Carlo runs.

Fig. 6. (a) shows the average OSPA results. The GLMB filters are still better than PHD and CPHD filters in tracking accuracy.

Fig. 6. (b) shows the average OSPA(2) results. The values of the TDVB-E-GLMB filter are obviously lower than that of E-GLMB, which means that the proposed method has strong robustness in complex scenarios. Moreover, the proposed smoother achieves the best performance. Therefore, the proposed smoother can handle complex scenarios as well.

Fig. 6. (c) shows the average estimation results of the number of targets. Compared with **Fig. 5. (b)**, the complex tracking environment significantly affects the estimation accuracy of all approaches. However, the values estimated by the proposed TDVB-E-GLMB filter and smoother are closer to the real values.

Fig. 6. (d) shows the average time cost. As can be seen, the computational cost of proposed TDVB-E-GLMB filter and smoother are slightly higher than the standard E-GLMB filter.



(a) OSPA error (b) OSPA(2)

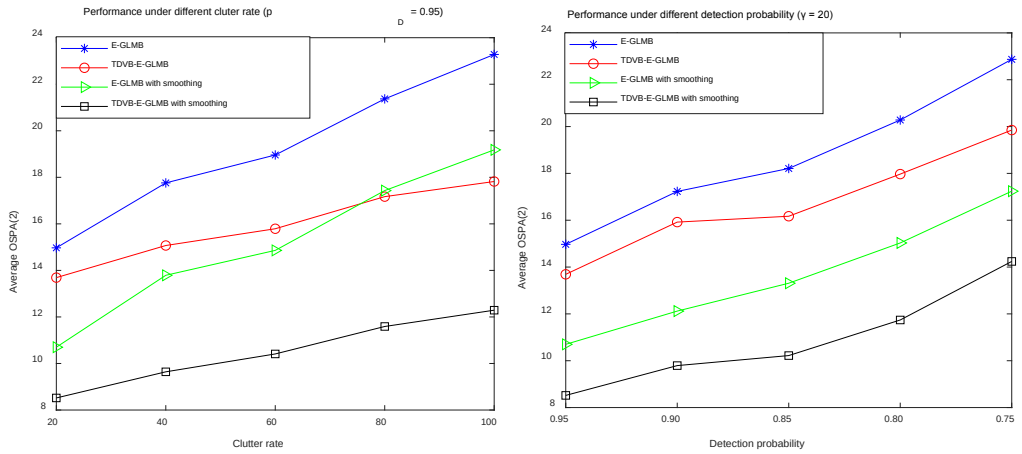
(c) Number of targets (d) Time cost

Fig. 6. Average performance of 200 Monte Carlo runs in Scenario2.

5.3 Performance Under Different Parameters

In this scenario, we show the average OSPA(2) performance of 200 Monte Carlo runs with different parameters.

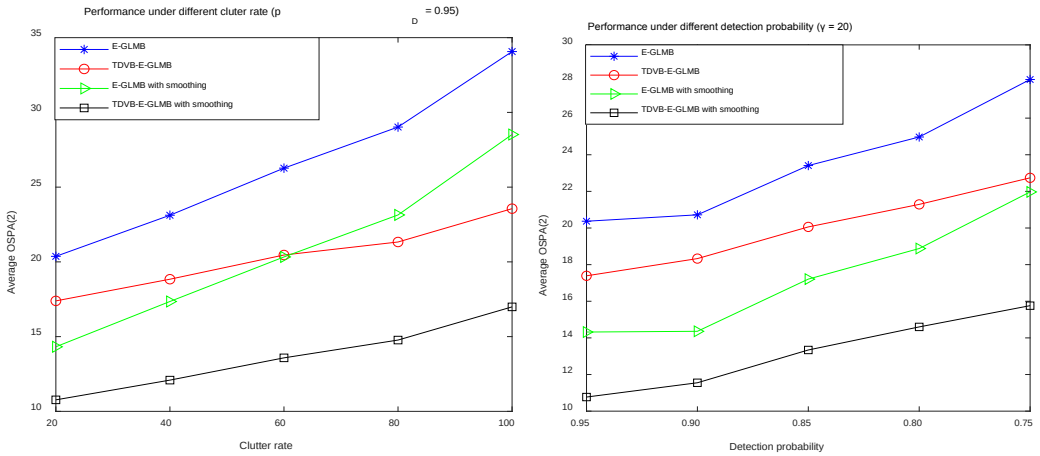
Fig. 7 shows the performance of different clutter rate and detection probability in the scenarios without measurement-outliers. It can be seen that the OSPA(2) values of the TDVB-E-GLMB filter is slightly lower than that of the E-GLMB filter. However, with the increase of the clutter rate (or the decrease of detection probability), the performance advantage of the proposed TDVB-E-GLMB filter becomes more obvious. It shows that the proposed algorithm has strong robustness under different parameters. Also, the proposed TDVB-E-GLMB filter with our smoothing approach achieves the best performance.



(a) $p_D = 0.95$ (b) $\gamma = 20$

Fig. 7. Average OSPA(2) of 200 Monte Carlo runs with different parameters in the scenarios without measurement-outlier.

Fig. 8 shows the average performance in the scenarios with measurement-outliers. Compared with **Fig. 4**, the performance advantage of the proposed TDVB-E-GLMB filter and the corresponding smoother is more obvious. Therefore, in the case of various parameters of different scenarios, the proposed approaches can effectively deal with measurement-outliers.



(a) $p_D = 0.95$ (b) $\gamma = 20$

Fig. 8. Average OSPA(2) of 200 Monte Carlo runs with different parameters in the scenarios where the probability of measurement-outlier is 10%.

6. Conclusion

In this paper, we propose a TDVB-E-GLMB filter and a corresponding smoother for tracking multiple targets. The first contribution is that the TDVB filtering technology is used in the E-GLMB filter to improve the accuracy and robustness. The second contribution is that we proposed a smoother for the GLMB filtering framework to improve the state estimation

accuracy. The simulation results show that the performances of the proposed TDVB-E-GLMB filter and corresponding smoother are better than PHD, CPHD, and E-GLMB filters in various complex tracking scenarios. Note that, the proposed smoother uses the track information to obtain the measurement memory, and uses this memory for smoothing. Therefore, in theory, the proposed smoother can be applied to other filters which can provide tracks, rather than only to the GLMB filter.

In the future, we plan to apply the Student-T distribution to extended target tracking filters, such as [26].

Appendix

The KL divergence between $q_{k,i}^{(h)}(z_k,\lambda_k,l)$ and $\phi_{k,i}^{(h)}(z_k,\lambda_k,l)\varphi_{k,i}^{(h)}(\lambda_k,l)$ can be written as

$$\begin{aligned} KL\left(\phi_{k,i}^{(h)}(z_k,\lambda_k,l)\varphi_{k,i}^{(h)}(\lambda_k,l)\,\|\,q_{k,i}^{(h)}(z_k,\lambda_k,l)\right)= \\ \times\iint\phi_{k,i}^{(h)}(z_k,\lambda_k,l)\varphi_{k,i}^{(h)}(\lambda_k,l) \\ \times\ln\left(\frac{\phi_{k,i}^{(h)}(z_k,\lambda_k,l)\varphi_{k,i}^{(h)}(\lambda_k,l)}{q_{k,i}^{(h)}(z_k,\lambda_k,l)}\right)\mathrm{d}x_k\,\mathrm{d}\lambda_k \end{aligned} \quad . \tag{48}$$

According to formula (13), $\phi_{k,i}^{(h)}(z_k,\lambda_k,l)$ and $\varphi_{k,i}^{(h)}(\lambda_k,l)$ can be written as

$$\ln\left(\varphi_{k,i}^{(h)}(\lambda_k,l)\right)=E_{x_k}\left(\ln\left(q_{k,i}^{(h)}(z_k,\lambda_k,l)\right)\right)+\text{const}\,, \tag{49}$$

$$\ln\left(\phi_{k,i}^{(h)}(z_k,\lambda_k,l)\right)=E_{\lambda_k}\left(\ln\left(q_{k,i}^{(h)}(z_k,\lambda_k,l)\right)\right)+\text{const}\,. \tag{50}$$

According to (27), we obtain

$$\begin{aligned} q_{k,i}^{(h)}(z_k,\lambda_k,l)\propto\mathrm{N}\left(x_k,R_k/\lambda_k\right) \\ \times\mathrm{N}\left(m_{k|k-1,i}^{(h)}(l),P_{k|k-1,i}^{(h)}(l)\right)\mathrm{Gamma}\left(\lambda_k\,|\,v/2,v/2\right) \end{aligned}\,. \tag{51}$$

Take formula (51) into (49), we obtain

$$\begin{aligned} \ln\left(\varphi_{k,i}^{(h)}(\lambda_k,l)\right)\propto E_{x_k}\left(\ln\left(\mathrm{N}\left(x_k,R_k/\lambda_k\right)\right)\right. \\ +\ln\left(\mathrm{N}\left(m_{k|k-1,i}^{(h)}(l),P_{k|k-1,i}^{(h)}(l)\right)\right) \\ \left.+\ln\left(\mathrm{Gamma}\left(\lambda_k\,|\,v/2,v/2\right)\right)\right)+\text{const} \end{aligned}\quad . \tag{52}$$

Obviously,

$$\begin{aligned} E_{x_k}\left(\ln\left(\mathrm{Gamma}\left(\lambda_k\,|\,v/2,v/2\right)\right)\right)= \\ \left(\frac{v+d}{2}-1\right)\ln\bar{\lambda}_k-\frac{v\bar{\lambda}_k}{2}\,, \end{aligned} \tag{53}$$

$$E_{x_k}\left(\ln\left(\mathrm{N}\left(x_k,R_k/\lambda_k\right)\right)\right)=-\frac{1}{2}\bar{\lambda}_k\,\mathrm{tr}\left\{\left(u_k^{(h)}\left(u_k^{(h)}\right)^T\right)R_k^{-1}\right\}. \tag{54}$$

As a result, we obtain

$$\begin{aligned} \varphi_{k,i}^{(h)}(\lambda_k,l)\propto\exp\left(-\frac{1}{2}\bar{\lambda}_k\mathrm{tr}\left\{\left(u_k^{(h)}\left(u_k^{(h)}\right)^T\right)R_k^{-1}\right\}\right. \\ \left.+\left(\frac{v+d}{2}-1\right)\ln\bar{\lambda}_k-\frac{v\bar{\lambda}}{2}\right) \end{aligned}\,, \tag{55}$$

where, $u_k^{(h)}$ is the same as that in formula (30).

Similarly, $\phi_{k,i}^{(h)}(z_k,\lambda_k,l)$ can be written as

$$\begin{aligned}\ln\left(\phi_{k,i}^{(h)}(z_k,\lambda_k,l)\right)\propto & E_{\lambda_k}\left(\ln\left(\mathrm{N}\left(x_k,R_k/\lambda_k\right)\right)\right)\\ &+\ln\left(\mathrm{N}\left(m_{k|k-1,i}^{(h)}(l),P_{k|k-1,i}^{(h)}(l)\right)\right)\\ &+\ln\left(\mathrm{Gamma}\left(\lambda_k\mid v/2,v/2\right)\right)+\mathrm{const}\end{aligned}. \tag{56}$$

Here,

$$\begin{aligned}&E_{\lambda_k}\left(\ln\left(\mathrm{N}\left(x_k,R_k/\lambda_k\right)\right)\right)=\\ &-\frac{1}{2}\lambda_k\left(z_k-H_k m_{k|k,i}^{(h)}(l)\right)^T R_k^{-1}\left(z_k-H_k m_{k|k,i}^{(h)}(l)\right),\end{aligned} \tag{57}$$

$$\begin{aligned}&E_{\lambda_k}\left(\ln\left(\mathrm{N}\left(m_{k|k-1,i}^{(h)}(l),P_{k|k-1,i}^{(h)}(l)\right)\right)\right)=\\ &-\frac{1}{2}\left(\varepsilon_k^{(h)}\right)^T\left(P_{k|k-1,i}^{(h)}(l)\right)^{-1}\varepsilon_k^{(h)},\end{aligned} \tag{58}$$

where,

$$\varepsilon_k^{(h)}=m_{k|k,i}^{(h)}(l)-m_{k|k-1,i}^{(h)}(l), \tag{59}$$

therefore,

$$\begin{aligned}&\phi_{k,i}^{(h)}(z_k,\lambda_k,l)\propto\\ &\exp\left(-\frac{1}{2}\lambda_k\left(z_k-H_k m_{k|k,i}^{(h)}(l)\right)^T R_k^{-1}\left(z_k-H_k m_{k|k,i}^{(h)}(l)\right),\right.\\ &\left.-\frac{1}{2}\left(\varepsilon_k^{(h)}\right)^T\left(P_{k|k-1,i}^{(h)}(l)\right)^{-1}\varepsilon_k^{(h)}\right)\end{aligned} \tag{60}$$

Acknowledgement

This paper is supported by National Natural Science Foundation of China (No.62002142, No.61902160); Natural Science Foundation of Jiangsu Province (No.BK20201057); Natural Science Foundation of Universities in Jiangsu Province (No.20KJD520002).

References

[1] R. Mahler, “Multi-target Bayes filtering via first-order multi-target moments,” *IEEE Transactions on Aerospace and Electronic Systems*, vol. 39, no. 4, pp. 1152-1178, 2003. Article (CrossRef Link)

[2] B. T. Vo, and B. N. Vo, “Labeled random finite sets and multi-object conjugate priors,” *IEEE Transactions on Signal Processing*, vol. 61, no. 13, pp. 3460-3475, 2013. Article (CrossRef Link)

[3] B. N. Vo, and W. K. Ma, “The Gaussian mixture probability hypothesis density filter,” *IEEE Transactions on Signal Processing*, vol. 54, no. 11, pp. 4091-4104, 2006. Article (CrossRef Link)

[4] W. Aftab and L. Mihaylova, "A Learning Gaussian Process Approach for Maneuvering Target Tracking and Smoothing," *IEEE Transactions on Aerospace and Electronic Systems*, vol. 57, no. 1, pp. 278-292, 2021. Article (CrossRef Link)

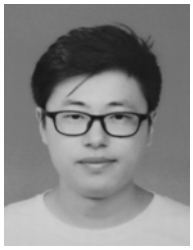
[5] R. Mahler, B. T. Vo, and B. N. Vo, “CPHD filtering with unknown clutter rate and detection profile,” *IEEE Transactions on Signal Processing*, vol. 59, no. 8, pp. 3497-3513, 2011. Article (CrossRef Link)

[6] T. Li, F. Hlawatsch, “A distributed particle-PHD filter using arithmetic-average fusion of Gaussian mixture parameters,” *Information Fusion*, vol.73, no.4, pp.111-124, 2021. Article (CrossRef Link)

[7] B. N. Vo, B. T. Vo, and D. Phung, “Labeled random finite sets and the Bayes multi-target tracking filter,” *IEEE Transactions on Signal Processing*, vol. 62, no. 24, pp. 6554-6567, 2014. Article (CrossRef Link)

[8] S. Reuter, B. T. Vo, B. N. Vo, and K. Dietmayer, "The labelled multi-Bernoulli filter," *IEEE Transactions on Signal Processing*, vol. 62, no. 12, pp. 3246-3260, 2014. Article (CrossRef Link)
[9] Z. -X. Liu, J. Gan, J. -S. Li and M. Wu, "Adaptive δ-Generalized Labeled Multi-Bernoulli Filter for Multi-Object Detection and Tracking," *IEEE Access*, vol. 9, pp. 2100-2109, 2021. Article (CrossRef Link)
[10] J. Zhao, R. Gui, X. Dong, "A new measurement association mapping strategy for DOA tracking," *Digital Signal Processing*, vol.118, 2021. Article (CrossRef Link)
[11] C. Cao and Y. Zhao, "An Efficient Implementation of the Multiple-Model Generalized Labeled Multi-Bernoulli Filter for Track-Before-Detect of Point Targets Using an Image Sensor," *IEEE Transactions on Aerospace and Electronic Systems*, vol. 57, no. 6, pp. 4416-4432, 2021. Article (CrossRef Link)
[12] C. Cao, and Y. Zhao, "A multiple-model generalized labeled multi-Bernoulli filter based on blocked Gibbs sampling for tracking maneuvering targets," *Signal Processing*, vol.186, 2021. Article (CrossRef Link)
[13] C. Do, T. Nguyen, and H. Nguyen, "Robust multi-sensor generalized labeled multi-Bernoulli filter," *Signal Processing*, vol.192, 2022. Article (CrossRef Link)
[14] L. Hou, F. Lian, S. Tan, C. Xu and G. T. F. de Abreu, "Robust Generalized Labeled Multi-Bernoulli Filter for Multitarget Tracking With Unknown Non- Stationary Heavy-Tailed Measurement Noise," *IEEE Access*, vol. 9, pp. 94438-94453, 2021. Article (CrossRef Link)
[15] S. Zhu, B. Yang, and S. Wu, "Measurement-driven multi-target tracking filter under the framework of labeled random finite set," *Digital Signal Processing*, vol.112, 2021. Article (CrossRef Link)
[16] X. Ru, Y. Chi, and W. Liu, "A Detection and Tracking Algorithm for Resolvable Group with Structural and Formation Changes Using the Gibbs-GLMB Filter," *Sensors*, vol.20, no.12, 2020. Article (CrossRef Link)
[17] C. Cao, Y. Zhao, X. Pang, and et al, "An efficient implementation of multiple weak targets tracking filter with labeled random finite sets for marine radar," *Digital Signal Processing*, vol.101, 2020. Article (CrossRef Link)
[18] B. N. Vo, B. T. Vo, and H. Hoang, "An efficient implementation of the generalized labeled multi-Bernoulli filter," *IEEE Transactions on Signal Processing*, vol. 65, no. 8, pp. 1975-1987, 2017. Article (CrossRef Link)
[19] R. Piche, S. Sarka, and J. Hartikainen, "Recursive outlier-robust filtering and smoothing for nonlinear systems using the multivariate Student-T distribution," in *Proc. of the 2012 IEEE International Workshop on Machine Learning for Signal Processing*, Santander, Spain, pp. 1-8, 2012. Article (CrossRef Link)
[20] K. Granström, and J. Bramstång, "Bayesian smoothing for the extended object random matrix model," *IEEE Transactions on Signal Processing*, vol. 67, no. 14, pp. 3732-3742, 2019. Article (CrossRef Link)
[21] D. Schuhmacher, B. T. Vo, and B. N. Vo, "A consistent metric for performance evaluation of multi-object filters," *IEEE Transactions on Signal Processing*, vol. 56, no. 8, pp. 3447–3457, 2008. Article (CrossRef Link)
[22] P. Li, "The MATLAB code of TDVB-J-GLMB filter and smoother,". [Online]. Available: https://github.com/PengLi6868/TDVB-J-GLMB-filter-and-smoother.git
[23] N. Chen, H. Ji, and Y. Ga, "Efficient box particle implementation of the multi-sensor GLMB filter in the presence of triple measurement uncertainty," *Signal Processing*, vol. 162, pp. 307-316, 2019. Article (CrossRef Link)
[24] P. Feng, W. Wang, S. M. Naqvi and J. Chambers, "Adaptive Retrodiction Particle PHD Filter for Multiple Human Tracking," *IEEE Signal Processing Letters*, vol. 23, no. 11, pp. 1592-1596, Nov. 2016. Article (CrossRef Link)
[25] Z. LIU, S. CHEN, H. WU, and et al, "A Student's T mixture probability hypothesis density filter for multi-target tracking with outliers," *Sensors*, vol. 18, no. 4, p.1095, 2018. Article (CrossRef Link)

[26] K. Granström, M. Fatemi, and L. Svensson, "Poisson Multi-Bernoulli Mixture Conjugate Prior for Multiple Extended Target Filtering," *IEEE Transactions on Aerospace and Electronic Systems*, vol. 56, no. 1, pp. 208-225, 2020. Article (CrossRef Link)
[27] Y. Sung and P. Tokekar, "GM-PHD Filter for Searching and Tracking an Unknown Number of Targets With a Mobile Sensor With Limited FOV," *IEEE Transactions on Automation Science and Engineering*, pp. 1-13, 2021. Article (CrossRef Link)
[28] X. Lai, G. Zhu, J. Chambers, “A fuzzy adaptive extended Kalman filter exploiting the Student's t distribution for mobile robot tracking,” *Measurement Science and Technology*, vol. 32, no. 10, 2021. Article (CrossRef Link)
[29] T. Zhang, J. Wang, L. Zhang and L. Guo, "A Student's T-Based Measurement Uncertainty Filter for SINS/USBL Tightly Integration Navigation System," *IEEE Transactions on Vehicular Technology*, vol. 70, no. 9, pp. 8627-8638, 2021. Article (CrossRef Link)
[30] M. Beard, B. T. Vo, and B. Vo, "Performance Evaluation for Large-Scale Multi-Target Tracking Algorithms," in *Proc. of 21st International Conference on Information Fusion (FUSION)*, Cambridge, pp. 1-5, 2018. Article (CrossRef Link)
[31] J. Yan, J. Dai, W. Pu, H. Liu and M. Greco, "Target Capacity Based Resource Optimization for Multiple Target Tracking in Radar Network," *IEEE Transactions on Signal Processing*, vol. 69, pp. 2410-2421, 2021. Article (CrossRef Link)
[32] Y. Xia, and et al., "Learning-Based Extended Object Tracking Using Hierarchical Truncation Measurement Model With Automotive Radar," *IEEE Journal of Selected Topics in Signal Processing*, vol. 15, no. 4, pp. 1013-1029, 2021. Article (CrossRef Link)
[33] H. Kim, K. Granström, L. Gao, G. Battistelli, S. Kim and H. Wymeersch, "Joint CKF-PHD Filter and Map Fusion for 5G Multi-cell SLAM," in *Proc. of 2020 IEEE International Conference on Communications (ICC)*, pp. 1-6, 2020. Article (CrossRef Link)

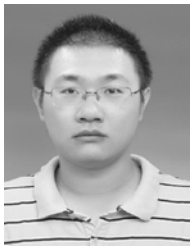


PENG LI was born in 1989. He received his B.S. degree in mathematics and applied mathematics from Langfang Normal University. He received his Ph.D. degree in Control Science and Engineering from Jiangnan University. Currently, he works at Jiangsu University of Technology. His research interests include target tracking and signal processing.

JUNDA QIU was born in 1990. He received his Ph.D. degree in Jiangnan University in 2019. Currently, he works at Jiangsu University of Technology. His research interests include decision making and target

WENHUI WANG was born in 1990. She received her B.A. degree in Shandong University of Arts. She received her M.A. degree in Design from Jiangnan University. Currently, she is studying for her doctorate at Tongming University and works at Jiangsu University of Technology. Her research interests include visual communication and target tracking.

CONGZHE YOU was born in 1989. He received his Ph.D. degree in Jiangnan University in 2017. Currently, he works at Jiangsu University of Technology. His research interests include pattern recognition and artificial intelligence.



ZHENQIU SHU received his Ph.D. degree in Nanjing University of Science and Technology. He was a visiting scholar with the School of Information and Communication Technology, Griffith University in 2014. He is currently an associate Professor with Jiangsu University of Technology. His research interests include patter recognition and machine learning.