蘭州理工大學

科研成果汇总

学　　号：	211081101005
研 究 生：	赵子文
导　　师：	陈辉 教授
研究方向：	信息融合
论文题目：	鲁棒自适应目标跟踪及其智能优化策略研究
学　　科：	控制理论与控制工程
学　　院：	自动化与电气工程学院
入学时间：	2021 年 9 月

目录

1. 检索报告 1

2. **Zhao Ziwen**, Chan Hui*, Zhang Wenxu. Adaptive elliptical extended object tracking based on deep reinforcement learning (SCI) 4

3. **Zhao Ziwen**, Chan Hui*, Zhang Wenxu. Reinforcement learning based hyperparameter optimization for irregular shape extended object tracking (SCI) 21

4. **Zhao Ziwen**, Chan Hui*. Multi-Maneuvering Target Tracking Based on a Gaussian Process (SCI) 44

5. **赵子文**, 陈辉*, 连峰, 张光华. 厚尾噪声条件下的学生t泊松多伯努利混合滤波器 (EI) 67

6. **赵子文**, 陈辉*, 连峰, 张光华, 张文旭. 高斯过程认知学习的多机动目标泊松多伯努利混合滤波器 (EI, 已录用) 79

7. 赵子文, 陈辉*. 基于高斯过程的概率假设密度滤波多机动目标跟踪方法, 发明专利 (已授权, 专利号: ZL 2023 1 1238475.X) 91

图书馆　　　　文献检索报告

兰州理工大学图书馆　　　　报告编号：**R2025-1224**

机构：兰州理工大学 电气工程与信息工程学院

姓名：赵子文　**[211081101005]**

著者要求对其在国内外学术出版物所发表的科技论著被以下数据库收录情况进行查证。

检索范围：

- 科学引文索引（Science Citation Index Expanded）：1900年-2025年
- 工程索引（Engineering Index）：1884年-2025年

检索结果：

检索类型	数据库	年份范围	总篇数	第一作者篇数
SCI-E 收录	SCI-EXPANDED	2024 - 2025	3	3
EI 收录	EI-Compendex	2024 - 2025	4	4

委托人声明：

本人委托兰州理工大学图书馆查询论著被指定检索工具收录情况，经核对检索结果，附件中所列文献均为本人论著，特此声明。

作　　者（签字）：赵子文

完成人（签字）：徐春园

完　成　日　期　：2025年10月30日

完成单位（盖章）：兰州理工大学图书馆信息咨询与学科服务部

（本检索报告仅限校内使用）

图书馆

文献检索报告
SCI-E 收录

SEARCH

兰州理工大学图书馆

报告编号：R2025-1224 SCI-E 收录

数据库：科学引文索引 (Science Citation Index Expanded) 时间范围：2024年至2025年	作者姓名：赵子文 作者单位：兰州理工大学 电气工程与信息工程学院	检索人员：徐春园 检索日期：2025年10月30日

检索结果：被 SCI-E 收录文献 3 篇

#	作者	地址	标题	来源出版物	文献类型	入藏号
1	**Zhao, ZW**; Chen, H	[Zhao, Ziwen; Chen, Hui] Lanzhou Univ Technol, Sch Elect & Informat Engn, Lanzhou 730050, Peoples R China.	Multi-Maneuvering Target Tracking Based on a Gaussian Process	***SENSORS*** 2024, 24 (22): 7270.	J Article	WOS:001366108200001
2	**Zhao, ZW**; Chen, H; Zhang, WX	[Zhao, Ziwen; Chen, Hui; Zhang, Wenxu] Lanzhou Univ Technol, Sch Elect Engn & Informat Engn, Lanzhou 730050, Peoples R China.	Reinforcement learning based hyperparameter optimization for irregular shape extended object tracking	***PHYSICA SCRIPTA*** 2025, 100 (7): 075247.	J Article	WOS:001520278800001
3	**Zhao, ZW**; Chen, H; Zhang, WX	[Zhao, Ziwen; Chen, Hui; Zhang, Wenxu] Lanzhou Univ Technol, Sch Elect Engn & Informat Engn, Lanzhou 730050, Peoples R China.	Adaptive elliptical extended object tracking based on deep reinforcement learning	***MEASUREMENT SCIENCE AND TECHNOLOGY*** 2025, 36 (5): 056116.	J Article	WOS:001478023200001
					合计	3

图书馆

文献检索报告
EI 收录

兰州理工大学图书馆

报告编号：R2025-1224 EI 收录

数据库：工程索引 (Engineering Index) 时间范围：2024年至2025年	作者姓名：赵子文 作者单位：兰州理工大学 电气工程与信息工程学院	检索人员：徐春园 检索日期：2025年10月30日

检索结果：被 EI 收录文献 4 篇

#	作者	地址	标题	来源出版物	文献类型	入藏号
1	**Zhao, Ziwen**; Chen, Hui; Zhang, Wenxu	School of Electrical Engineering and Information Engineering, Lanzhou University of Technology, Lanzhou	Reinforcement learning based hyperparameter optimization for irregular shape extended object tracking	***Physica Scripta*** 2025, 100 (7): 075247.	Journal article (JA)	20252818731052
2	**Zhao, Ziwen**; Chen, Hui; Zhang, Wenxu	School of Electrical Engineering and Information Engineering, Lanzhou University of Technology, Lanzhou	Adaptive elliptical extended object tracking based on deep reinforcement learning	***Measurement Science and Technology*** 2025, 36 (5): 056116.	Journal article (JA)	20251818344079
3	**Zhao, Ziwen**; Chen, Hui	School of Electrical and Information Engineering, Lanzhou University of Technology, Lanzhou	Multi-Maneuvering Target Tracking Based on a Gaussian Process (Open Access)	***Sensors*** 2024, 24 (22): 7270.	Journal article (JA)	20244817453074
4	**Zhao, Zi-Wen**; Chen, Hui; Lian, Feng; Zhang, Guang-Hua	School of Electrical and Information Engineering, Lanzhou University of Technology, Gansu, Lanzhou; School of Automation Science and Engineering, Xi'an Jiaotong University, Shaanxi, Xi'an	A Student's t Poisson multi-Bernoulli mixture filter in the presence of heavy-tailed noise 厚尾噪声条件下的学生t泊松多伯努利混合滤波器	***Kongzhi Lilun Yu Yingyong/Control Theory and Applications*** 2024, 41 (9): 1598-1609.	Journal article (JA)	20244517327054
					合计	4

Measurement Science
and Technology

100 year anniversary
Measurement Science and Technology

PURPOSE-LED
PUBLISHING™

PAPER

Adaptive elliptical extended object tracking based on deep reinforcement learning

To cite this article: Ziwen Zhao *et al* 2025 *Meas. Sci. Technol.* **36** 056116

View the article online for updates and enhancements.

IOP Publishing Measurement Science and Technology

Meas. Sci. Technol. 36 (2025) 056116 (16pp) https://doi.org/10.1088/1361-6501/adce1e

Adaptive elliptical extended object tracking based on deep reinforcement learning

Ziwen Zhao, Hui Chen* and Wenxu Zhang

School of Electrical Engineering and Information Engineering, Lanzhou University of Technology,
Lanzhou 730050, People's Republic of China

E-mail: chenh@lut.edu.cn, ziw@lut.edu.cn and wenxu_zhang@foxmail.com

Received 4 December 2024, revised 9 April 2025
Accepted for publication 17 April 2025
Published 28 April 2025

Abstract

To overcome the challenges arising from the estimation of elliptical shape parameters (orientation and semi-axis length) and unknown multiplicative noise covariance (MNC) in extended object tracking (EOT), this study proposes an innovative adaptive EOT method based on the soft actor-critic deep reinforcement learning (DRL) algorithm. Multiplicative noise, as a complex uncertainty factor, significantly affects the tracking performance of extended objects (EO), especially the recognition of shapes. Improper processing of MNC may lead to unstable parameter estimation or significant bias. The remarkable development of artificial intelligence, especially DRL, has provided a strong drive for intelligent decision making of optimal estimation. Through the combination of DRL and recursive Kalman filter, the estimation accuracy of EO orientation and elliptic axis length can be jointly improved, thus optimizing EOT performance to a large extent. This approach enhances the adaptability of the EOT algorithm through interactive learning between the agent and the environment with online adaptive tuning of the MNC and elliptical parameters. The simulation results show the superiority of the method compared with the existing methods, with significant improvement in estimation accuracy and robustness. In various test scenarios, the method can estimate the state and shape parameters of the EO more accurately, in which, for the shape estimation accuracy of the EO, the Gaussian Wasserstein distance of the proposed algorithm is optimized by 20.3%, 23.7%, and 18.9%, respectively, compared with the existing optimal method in different experimental scenarios.

Keywords: extended object tracking, deep reinforcement learning, unknown multiplicative noise covariance, elliptical parameters

1. Introduction

Extended object tracking (EOT) refers to the estimation of the kinematic state and shape parameters of an object in a radar or sensor system by monitoring and processing multiple measurements in real time and is a critical technology in radar system. In fact, traditional object tracking methods usually assume that the object only produces a single measurement and is therefore considered a point object or small object. Notably, with the rapid development of high-resolution sensing technology, the object presents a large physical size and provides rich information about its shape through multiple measurements. This type of tracking task is often referred to as 'EOT,' where the key is to simultaneously estimate the object's kinematic (e.g. position, velocity) as well as its shape characteristics (e.g. orientation angle, axis length). This lays a solid foundation for the development of EOT, and significantly expands the prospect of the application of EOT in complex environment. As an innovation in tracking

[*] Author to whom any correspondence should be addressed.

technology in recent years, EOT has been applied in areas such as autonomous driving [1–3], marine surveillance [4–6], and drone navigation [7, 8]. These applications improve the system's tracking accuracy and improve object identification and classification capabilities.

Some classical EOT methods, such as those based on random hypersurface model [9] and Gaussian process [10], have garnered widespread acclaim for their superior estimation performance. These techniques excel in scenarios with dense measurements, enabling effective estimation of complex irregular object shapes, including star-convex [11] and non-star-convex [12] geometries. However, their performance significantly deteriorates in environments with sparse measurements, making them less viable for practical use. Consequently, approximating an object as an ellipse in sparse measurement environments becomes a more reasonable and pragmatic approach, simplifying the estimation process while maintaining efficient identification of the basic contour features of the object.

In previous research, various methods have been proposed to address the problem of elliptical EOT. In [13], the symmetric positive definite random matrix was proposed to model the extended state of the object. However, this approach did not consider the potential impact of sensor errors, which could lead to overestimating the extended state. Subsequently, literature [14] improved upon this method by incorporating sensor errors and adjusting the measurement likelihood function accordingly. Despite this improvement, introducing additive terms into covariance presents a challenge for easily managed conjugate priors. To address these issues, literature [15] proposed a new measurement update method that considered measurement noise, using variable decibels Bayesian (VB) technology to derive approximate measurement updates. Although the performance is improved compared with the previous methods, the VB method still introduces obvious kinematic state errors when there is a large prediction error. In [16], an approximate Bayesian method was proposed to address unresolved scattering centers of extended or group objects, combining a measurement noise model to estimate the object's kinematic and extended state. However, all these methods assume linear measurements even though actual system often exhibit non-linearity. In [17], a method was proposed for linearizing nonlinear measurements for extended object (EO) or group object tracking, providing a more realistic approach to handle the inherent non-linearity measurement system. Although these methods are capable of accurately estimating the size and orientation of an ellipse, it is often difficult to distinguish between uncertainties in the semi-axis and the orientation of the ellipse. This is because when using the inverse Wishart distribution, the uncertainty in both the ellipse orientation and the semi-axis length is encoded in a single scalar degree of freedom. Consequently, new methods have emerged in recent years aimed at explicitly estimating the ellipse shape parameters of EOT.

In [18], a novel method for elliptical EOT was proposed, which parametrizes the ellipse and introduces multiplicative error term to establish the relationship between measurements and elliptical parameters. Recent research has addressed the problem of tracking ellipses with fixed but unknown axis lengths and varying orientations using the expectation-maximization (EM) approach [19]. This method estimates the object's kinematic state and shape parameters within a unified framework. However, existing methods still have limitations, such as sensitivity to initial conditions and lack of flexibility. These limitations may lead to suboptimal performance in practical applications.

In addition, existing algorithms for elliptical EOT either ignore the influence of measurement noise or only consider additive measurement noise but do not consider the impact of multiplicative noise (MN) on the state estimation of EO. In practical applications, MN is widespread and more challenging than additive noise due to its nonzero mean and the difficulty of determining mean and covariance. This uncertainty poses significant challenges to state estimation and seriously affects the accuracy of filters. Currently, the primary methods for solving the unknown and time-varying MN covariance (MNC) problem include adaptive filtering, with VB-based methods being the research focus. In [20], an unknown MNC is often modeled as an inverse Gamma or Wishart distribution. For this, two adaptive Kalman filters (KF) are derived using the VB technique to estimate unknown MNC. In another study, multiplicative and additive noise are jointly treated, and the VB method is employed to estimate the covariance of both types of noise simultaneously [21]. However, these methods are only tailored to the problem of unknown MNC in point object tracking. In EOT, the unknown MNC not only introduces bias to the motion state estimation of the EO but also further interferes with the accurate estimation of the elliptical shape parameters through the error propagation of the state estimation. Specifically, the unknown MNC leads to covariance estimation distortion decreasing the estimation accuracy of shape parameters and triggering filtering instability; in addition, the bias of kinematic state estimation may lead to shape misfit interfering with morphology perception and subsequent decision making; meanwhile, during the linearization process, the bias of the linearization point selection is amplified in iteration, significantly increasing the estimation error and decreasing the tracking accuracy. Although some progress has been made in related research in point object tracking, there is no effective solution for the unknown MNC problem in EOT. Therefore, tracking EO in complex environment remains a major challenge.

Recently, the application of deep reinforcement learning (DRL) [22] in object tracking has provided new insights for adaptive adjustment of elliptical parameters and unknown MNC. This approach combines deep learning (DL) with reinforcement learning (RL), enabling an agent to acquire environmental states and reward signals through interactive learning. Consequently, the agent learns how to take action to achieve optimal objectives. The DRL demonstrates superior performance in complex decision-making problems and has been widely applied in robot control [23–25] and autonomous driving [26–28].

Several recent studies have applied DRL to the radar object tracking domain [29–31]. For instance, the DRL algorithm was used for adaptive covariance tuning of the KF, which improves the accuracy of state estimation and location in autonomous driving [32]. Moreover, another study used a proximal policy optimization algorithm to adaptively estimate the process noise covariance of the KF, improving tracking estimation for highly maneuverable drone object [33]. However, previous DRL algorithms have faced limitations in multitasking learning, handling continuous action spaces, and maintaining stability. For these challenges, the recently proposed soft actor-critic (SAC) algorithm addressed these shortcomings [34]. The SAC algorithm innovatively extends the traditional actor-critic framework by introducing entropy regularization, which significantly improves the exploration ability of strategies. The introduction of entropy regularization not only motivates the strategies to maintain a high degree of randomness while pursuing high rewards but also effectively prevents the intelligence from falling into local optimal solutions in the early learning stage by encouraging the strategies to be uniformly distributed in the action space. Compared with other DRL algorithms, the SAC algorithm has significant robustness, stability, and learning efficiency advantages [34, 35]. Entropy regularization enhances the exploration of strategies and improves their robustness. By encouraging the diversity of strategies, SAC enables intelligence to maintain a high level of adaptability in the face of changing environments and, thus, better cope with uncertainties.

The combination of SAC and KF has been studied more. For example, the literature [36] proposes the RL-EKF framework that utilizes SAC to train the EKF filter gain, which solves the problem of the traditional method relying on manual adjustment of the filter gain and improves the navigation accuracy in a complex ocean dynamic environment. Literature [37] combines KF, EKF, and interacting multiple model to optimize communication power and accuracy in remote traffic state estimation using SAC for efficient autonomous driving. Literature [38] proposed a framework combining KF and DRL to predict the state of dynamic objects using KF and optimize UAV path planning by the SAC algorithm to solve the navigation problem of autonomous UAVs in non-static environments. However, none of the existing studies have addressed the EOT domain. In EOT tasks, the kinematic state and shape parameters of the object usually change, and the high uncertainty of the environment creates additional challenges for the EOT. Unlike existing methods, the SAC algorithm can dynamically adjust the MNC and elliptical shape parameters according to the current environment state to effectively respond to the EO state and shape changes. In this way, SAC provides a highly effective means to optimize the filtering parameters in EOT, resulting in a more accurate estimation of the EO state and shape. Therefore, it is of great practical significance and urgent that the SAC algorithm be used to adjust the parameters dynamically and optimize the tracking performance in complex EOT tasks. And, adaptive estimation of unknown MNC and EO parameters using DRL algorithms remains an innovative and challenging research topic.

This paper proposes a novel approach that leverages the SAC algorithm to address adaptive estimation challenges in elliptical EOT with unknown MNC and elliptical shape parameters (SAC-EOT). First, the measurement likelihood affected by MN is modeled as a Gaussian mixture distribution representing additive and MN. The SAC algorithm is used to adaptively adjust the unknown and time-varying MNC through the interaction between the agent and the tracking environment. Subsequently, the recursive KF is utilized to estimate the state of EO. The SAC algorithm then adaptively adjusts the elliptical shape parameters to account for uncertainties in the orientation and axis lengths, thereby reducing errors introduced during the linearization of the shape parameter measurement equation. To achieve this, a novel reward function of the entropy reward [29] is used. This function is the sum of the state estimate and the shape estimate of the entropy state. Finally, simulation experiments and comparisons with existing algorithms demonstrate the effectiveness of the proposed algorithm. The results show that the proposed SAC-EOT algorithm achieves higher estimation accuracy and robustness than traditional algorithms.

2. Problem background

2.1. Elliptical EO model

In the tracking of elliptical EO, the following shape parameters are considered in light of elliptical motion orientation and axis lengths

$$e_k = \left[\theta_k, b_k^1, b_k^2\right]^{\mathrm{T}} \tag{1}$$

where θ_k denotes the orientation of the elliptical, representing the angle between the major axis and the positive orientation of the x-axis, while b^1 and b^2 respectively denote the major and minor axes of the elliptical. Through these parameters, precise information regarding the shape of the elliptical EO can be determined. Figure 1 illustrates the parameterized model of the elliptical EO.

The kinematic parameter of the elliptical EO at time k can be represented as

$$x_k = \left[x_{k,x}, x_{k,y}, \dot{x}_{k,x}, \dot{x}_{k,y}\right]^{\mathrm{T}} \tag{2}$$

where $(x_{k,x}, x_{k,y})$ indicating the centroid position and $(\dot{x}_{k,x}, \dot{x}_{k,y})$ indicating the velocity.

2.2. Measurement and kinematic model

At time k, the EO generates a different number of independent two-dimensional Cartesian detections, denoted as $Z_k = \left\{z_k^{(i)}\right\}_{i=1}^{n_k}$, each measurement $z_k^{(i)}$ originating from its corresponding measurement source $y_k^{(i)}$. Unlike traditional EOT, the measurement source here is influenced by both Gaussian multiplicative measurement noise $m_k \sim \mathcal{N}(\bar{m}_k, \delta_k)$ and additive measurement noise $v_k \sim \mathcal{N}(0, R_k)$. Therefore, the measurement equation can be represented as

$$z_k^{(i)} = m_k^{(i)} H_k x_k + M_k h_k^{(i)} + v_k^{(i)} \tag{3}$$

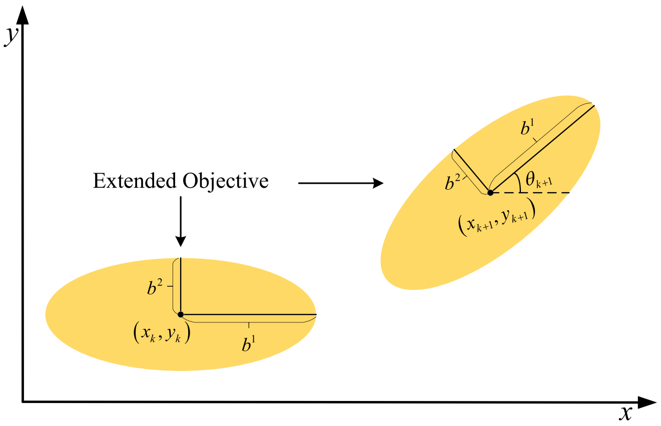


Figure 1. Parameterization model of the EO.

where $H_k = \begin{bmatrix} I_2 & 0 \end{bmatrix}$ denotes the observation matrix, and

$$M_k = \begin{bmatrix} \cos\theta & -\sin\theta \\ \sin\theta & \cos\theta \end{bmatrix} \begin{bmatrix} b^1 & 0 \\ 0 & b^2 \end{bmatrix} \tag{4}$$

representing the orientation and size of the EO, $h_k^{(i)} \sim \mathcal{N}\left(0, \sigma^h\right)$ is a multiplicative error term that establishes the relationship between the measurement source and the object shape, and $\sigma^h = 1/4 \times I_2$.

Due to the introduction of MN in the measurements, the measurement likelihood exhibits non-Gaussian characteristic. To effectively characterize this non-Gaussian characteristic, a mixture of Gaussian distributions (MG) is adopted to model the measurement likelihood

$$p\left(z_k^{(i)}|x_k, e_k\right) = \mathcal{N}\left(z_k^{(i)}; 0, \tilde{R}_k\right) + \mathcal{N}\left(z_k^{(i)}; \bar{m}_k H_k x_k, M_k \sigma^h M_k^{\mathrm{T}} + \breve{R}_k\right) \tag{5}$$

where $\tilde{R}_k$ and $\breve{R}_k$ respectively representing the covariance of multiplicative and additive measurement noise

$$\tilde{R}_k = \delta_k H_k D_k H_k^{\mathrm{T}} \tag{6}$$

$$\breve{R}_k = \bar{m}_k^2 H_k P_{k|k} H_k^{\mathrm{T}} + R_k \tag{7}$$

$$D_k = E\left[x_k x_k^{\mathrm{T}}\right]. \tag{8}$$

Here $P_{k|k}$ denotes error matrix. Considering the following discrete-time linear motion state models

$$x_{k+1} = F_k x_k + w_k \tag{9}$$

$$e_{k+1} = F_k^e e_k + w_k^e \tag{10}$$

where F_k and F_k^e represent the transition matrix of state and shape, respectively, $w_k \sim \mathcal{N}\left(0, P_r^w\right)$ and $w_k^e \sim \mathcal{N}\left(0, P_e^w\right)$ represent the Gaussian process noise.

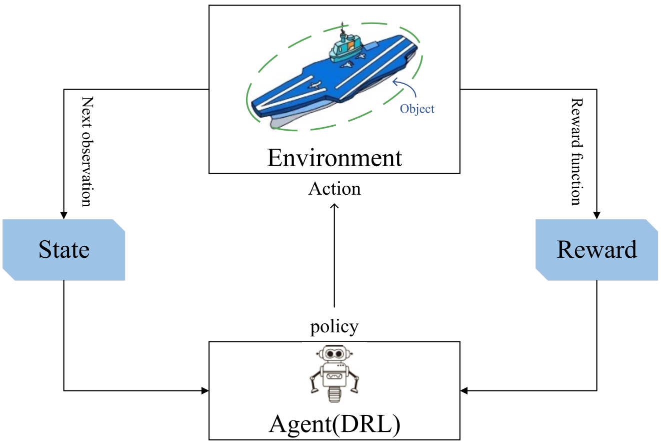


Figure 2. Structure of the agent-environment interaction.

2.3. DRL

RL can be defined as a Markov decision process (MDP) that learns policies through interaction with the environment to maximize cumulative rewards. DRL integrates DL techniques into RL and approximates the value and policy functions using deep neural networks, significantly improving the performance of reinforcement learning algorithm. Figure 2 illustrates the structure of the agent-environment interaction. By defining the parameters appropriately, the problem of adjusting the EOT parameters can be formulated within the DRL framework:

Environment: It is a tracking system for EO, comprising a radar surveillance area and EOT filtering process. EO moves within the surveillance area, and sensors receive measurements of it from the environment. In addition, the agent learns action policies based on information from the environment and obtains new actions through action policies.

State space S: The input state for the SAC algorithm is defined as the motion and extended state of the object. These two components provide comprehensive information about the environment and the task, enabling the agent to make decisions and learn to adapt to the environment. Therefore, at each time step, the input state is $S = \{x_k, y_k, \dot{x}_k, \dot{y}_k, e_k\}$.

Action space A: Considering the practical application scenarios of EOT, the action space includes unknown MNC distribution parameter, elliptical direction angle and axis lengths, which are adjusted to directly affect the state of the elliptical EO. This action space is a continuous interval, and the agent adapts to the dynamic changes in the object state by selecting different actions in this space.

Reward r: For the problem of tracking elliptical EO, the algorithm's estimation performance can be evaluated based on the covariance of estimation errors in the motion state and shape. Here, the evaluation metric can utilize the entropy state at time k, defined as

$$es_k = |\cdot|. \tag{11}$$

Therefore, the reward function design for the SAC-EOT algorithm is as follows:

$$r_k = \log\left(1+\left|es^x_{k-1}-es^x_k\right|\right)\text{sign}\left(es^x_{k-1}-es^x_k\right) + \log\left(1+\left|es^e_{k-1}-es^e_k\right|\right)\text{sign}\left(es^e_{k-1}-es^e_k\right) \tag{12}$$

where es^x is the entropy of motion state, es^e is the entropy of shape and $\text{sign}(\cdot)$ denotes the signum function. When es at time k is less than at time $k-1$, it indicates a smaller uncertainty in the state estimation, resulting in a positive reward. Conversely, it receives a negative reward. This design effectively drives the agent to learn and achieve the desired behavior.

3. SAC for EOT

3.1. System model

The adaptive elliptical EOT method proposed in this paper mainly consists of two core components: the estimation of the EO state based on the recursive KF and the adjustment of the unknown MNC, elliptical orientation and semi-axis length based on the SAC algorithm. The key to this method lies in describing EOT as an MDP and then utilizing SAC to adaptively adjust parameters, thereby enhancing the accurate estimation of EO position and shape in complex environments.

To adjust the parameters in EOT, the following objective function is defined as

$$O(\Theta) = \sum_{k=1}^{T}\kappa(x_k,\Theta) \tag{13}$$

where T represents the number of steps in which the agent interacts with the environment, κ is evaluated at time k based on the state x_k and the elliptical parameters, Θ represents the vector of parameters to be adjusted. Then, maximizing the expected value of (13) yields the optimal parameter vector

$$\arg\max_{\Theta}\hat{O}(\Theta) \tag{14}$$

where $\hat{O}(\Theta) := E[O(\Theta)]$.

Next, the process of how SAC is used to adjust EOT parameters is explained. First, based on the current state $s_k \in S$ and a tractable policy π_ϕ(ϕ is the parameter of the policy network), a specific action a_k can be determined, i.e. $a_k \sim \pi_\phi(a_k|s_k)$. This action space includes the EOT parameters of the current training step. Then, these parameters are substituted into the EOT algorithm to obtain the next state $s_{k+1}\sim p(s_{k+1}|s_k,a_k)$ and the corresponding reward $r_k(s_k,a_k)$. Since SAC utilizes an experience replay buffer, this state transition set is stored in the buffer, i.e. $B \leftarrow B\cup\{s_k,a_k,r(s_k,a_k),s_{k+1}\}$, where B is the previously sampled state and action distribution or replay buffer. Therefore, the objective function for training the soft Q-value function is given by

$$O_Q(\vartheta) = E_{(s_k,s_k)\sim B}\left[\frac{1}{2}\left(Q_\vartheta(s_k,a_k)-\hat{Q}_\vartheta(s_k,a_k)\right)^2\right] \tag{15}$$

where ϑ is the parameter of the Q-value network and

$$\hat{Q}_\vartheta(s_k,a_k) = r(s_k,a_k)+\gamma E_{s_{k+1}\sim p}\left[V_{\bar{\Phi}}(s_{k+1})\right]. \tag{16}$$

The policy parameters are learned by

$$O_\pi(\phi) = E_{s_k\sim B}\left[D_{\text{KL}}\left(\pi_\phi(\cdot|s_k)\left\|\frac{\exp(Q_\vartheta(s_k,\cdot))}{Z_\vartheta(s_k)}\right.\right)\right]. \tag{17}$$

Here, the policy is reparameterized using neural network transformation

$$a_k = f_\phi(\upsilon_k;s_k) \tag{18}$$

where f_ϕ represents a neural network transformation, υ_k represents input noise. Therefore, the optimal policy for SAC is

$$\pi^* = \arg\max_\pi E_{s_k,a_k\sim\pi(\cdot|s_k)}\left[\sum_k\gamma^k r(s_k,a_k)+\beta U(\pi(\cdot|s_k))\right] \tag{19}$$

where γ^k is the discount factor for future rewards, $U(\cdot)$ represents entropy and β represents the temperature coefficient, determining the relative importance of entropy to reward. The structure of the SAC algorithm for EOT is illustrated in figure 3. In this structure, the policy network generates action based on the current state, using a fully connected feedforward neural network with inputs as state vectors of the object's kinematic state and shape parameters. The network contains two hidden layers of 256 neurons with the ReLU activation function; the output layer generates the mean and standard deviation of the actions, the standard deviation is ensured to be non-negative by the soft plus function, and the reparameterization trick is used to achieve the sampling of the continuous action space. The inputs of the Q-network are the state vectors and action vectors spliced together with the two fully connected layers of the same structure, and the final output is a scalar to assess the action quality. The target network is aligned with the Q-network through soft updates to improve training stability. Based on this structure, the agent determines the action a_k based on the current state s_k and the policy network, and the action space covers EOT-related parameters such as the unknown MNC distribution parameter, elliptical direction angle, and axis length. By substituting these parameters into the EOT algorithm, the intelligent agent can compute the next state s_{k+1} and reward r_k, and the state transfer information is stored in the experience replay buffer. By sampling data from the buffer, the training network and the policy network continuously adjust the parameters to realize the effective application of the SAC algorithm in EOT. This process continuously improves the performance of the EOT by iteratively adjusting and optimizing the parameters.

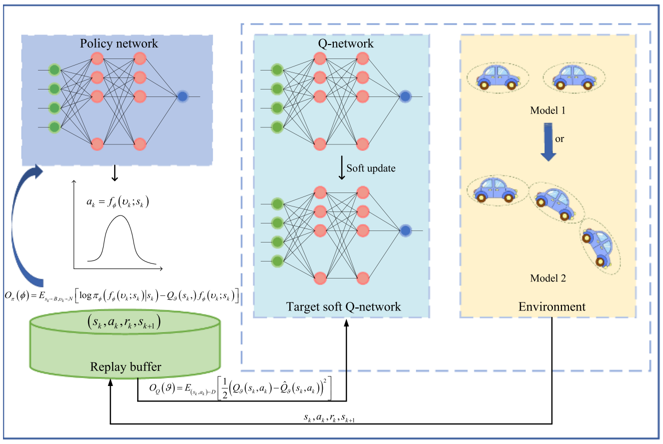

Policy network
$a_k = f_\phi(\upsilon_k; s_k)$
$O_\pi(\phi) = E_{s_k\sim B,\upsilon_k\sim N}\left[\log\pi_\phi\left(f_\phi(\upsilon_k;s_k)|s_k\right) - Q_\vartheta(s_k,)f_\phi(\upsilon_k;s_k)\right]$
(s_k, a_k, r_k, s_{k+1})
Replay buffer
Q-network
Soft update
Target soft Q-network
Model 1
or
Model 2
Environment
$O_Q(\vartheta) = E_{(s_k,a_k)\sim D}\left[\frac{1}{2}\left(Q_\vartheta(s_k,a_k) - \hat{Q}_\vartheta(s_k,a_k)\right)^2\right]$
s_k, a_k, r_k, s_{k+1}

Figure 3. The structure of SAC for EOT.

3.2. *Adjust unknown MNC and elliptical parameters*

In order to acquire the unknown MNC, assume that the measurement likelihood follows the MG model, as shown in (5). However, obtaining the state estimate directly from the measurement likelihood is not feasible. Therefore, the VB method needs to be employed. According to Bayes' theorem and variational approximation, $q(x_k) = \mathcal{N}\left(x_k; \hat{x}_{k|k}^{(i)}, P_{k|k}^{r(i)}\right)$ can be used to approximate $q(x_k)$, the derivation of variational Bayesian approximation is shown in the appendix. Using the update equation of the KF and the measurement received at time k, denoted as $z_k^{(i)}$, the update of the motion state can be represented as

$$\hat{x}_{k|k}^{(i)} = \hat{x}_{k|k-1}^{(i-1)} + P_{k|k}^{ry(i)}\left(P_{k|k}^{y(i)}\right)^{-1}\left(z_k^{(i)} - \bar{m}_k H_k \hat{x}_{k|k-1}^{(i-1)}\right) \tag{20}$$

$$P_{k|k}^{r(i)} = P_{k|k-1}^{r(i-1)} - \bar{m}_k P_{k|k}^{ry(i)}\left(P_{k|k}^{y(i)}\right)^{-1}\left(P_{k|k}^{ry(i)}\right)^{\mathrm{T}} \tag{21}$$

where $P_{k|k}^{y(i)}$ represents the measurement covariance and $P_{k|k}^{ry(i)}$ represents the cross-covariance between the measurement and the motion state. With the measurement noise affected by MN, a new challenge arises in finding the analytical solution for $P_{k|k}^{y(i)}$ and $P_{k|k}^{ry(i)}$.

In the context of EOT, due to the presence of the shape matrix $M_{k|k}$ in the measurement equation (3), it is necessary to linearize $M_{k|k}h_k^{(i)}$ at $\hat{e}_{k|k}^{(i-1)}$($\hat{\cdot}_{k|k}^{(i-1)}$ is the evaluation at $\hat{p}_{k|k}^{(i-1)}$) using the shape parameter e_k, such that

$$\begin{aligned} M_{k|k}h_k^{(i)} \approx & \underbrace{\hat{M}_{k|k}^{(i-1)} h_k^{(i)}}_{\mathrm{I}} \\ & + \underbrace{\begin{bmatrix} \left(h_k^{(i)}\right)^T & \widehat{C}_{1k|k}^{(i-1)} \\ \left(h_k^{(i)}\right)^T & \widehat{C}_{2k|k}^{(i-1)} \end{bmatrix}\left(e_{k|k} - \hat{e}_{k|k}^{(i-1)}\right)}_{\mathrm{II}} \end{aligned} \tag{22}$$

where C_1 and C_2 are the Jacobian matrices of M's first and second rows, respectively, given by

$$C_1 = \frac{\partial M^1}{\partial e} = \begin{bmatrix} -b_{a_k^*}^1 \sin\theta_{a_k^*} & \cos\theta_{a_k^*} & 0 \\ -b_{a_k^*}^2 \cos\theta_{a_k^*} & 0 & -\sin\theta_{a_k^*} \end{bmatrix} \tag{23}$$

$$C_2 = \frac{\partial M^2}{\partial e} = \begin{bmatrix} b_{a_k^*}^1 \cos\theta_{a_k^*} & \sin\theta_{a_k^*} & 0 \\ -b_{a_k^*}^2 \sin\theta_{a_k^*} & 0 & \cos\theta_{a_k^*} \end{bmatrix}. \tag{24}$$

Here, $\left[\theta_{a_k^*}, b_{a_k^*}^1, b_{a_k^*}^2\right]$ can be obtained from the training according to the elliptical parameters of the previous time, $M^1 = \begin{bmatrix} b_{a_k^*}^1 \cos\theta_{a_k^*} & -b_{a_k^*}^2 \sin\theta_{a_k^*} \end{bmatrix}$, $M^2 = \begin{bmatrix} b_{a_k^*}^1 \sin\theta_{a_k^*} & b_{a_k^*}^2 \cos\theta_{a_k^*} \end{bmatrix}$. Additionally, I and II are uncorrelated, and the covariance of $M_{k|k}h_k^{(i)}$ can be approximated by the sum of P^{I} and P^{II}, i.e.

$$P^{\mathrm{I}} = \hat{M}_{k|k}^{(i-1)} P^h \left(\hat{M}_{k|k}^{(i-1)}\right)^{\mathrm{T}} \tag{25}$$

$$P^{\mathrm{II}} = tr\left\{P_k^{p^{(i-1)}}\left(\widehat{C}_{nk|k}^{(i-1)}\right)^{\mathrm{T}} P^h \widehat{C}_{mk|k}^{(i-1)}\right\}, \quad m,n \in \{1,2\}. \tag{26}$$

The cross-covariance and covariance are given by

$$P_{k|k}^{ry(i)} = \bar{m}_k P_{k|k}^{r(i-1)} H_k^{\mathrm{T}} \tag{27}$$

$$P_{k|k}^{y(i)} = \bar{m}_k^2 H_k P^{r(i-1)} H_k^{\mathrm{T}} + P^{\mathrm{I}} + P^{\mathrm{II}} + \bar{R}_k \tag{28}$$

$$\bar{R}_k = \delta_k H_k M_k H_k^{\mathrm{T}} + \bar{m}_k^2 H_k P_{k|k-1} H_k^{\mathrm{T}} + R_k. \tag{29}$$

Similarly, letting $\eta = \delta_k$ and substituting (22) into (21) can obtain $q(\delta_k) = IG\left(\delta_k; \xi_{k|k}, \zeta_{k|k}\right)$, where

$$\xi_{k|k} = \xi_{k|k-1} + 1/2 \tag{30}$$

$$\zeta_{k|k} = \rho^* \zeta_{k|k-1} + 1/2tr[\mathrm{A}] \tag{31}$$

$$\delta_k = \frac{\zeta_{k|k}}{\xi_{k|k}} \tag{32}$$

where

$$\begin{aligned} \mathrm{A} = & \left[\left(z_k^{(i)} - \bar{m}_k H_k \hat{x}_{k|k}^{(i)}\right)(D_k)^{-1}\left(z_k^{(i)} - \bar{m}_k H_k \hat{x}_{k|k}^{(i)}\right)^{\mathrm{T}}\right. \\ & \left. + \bar{m}_k^2 H_k P_{k|k}^{r(i)} H_k^{\mathrm{T}}\right]. \end{aligned} \tag{33}$$

3.3. Update elliptical parameters

Shape update: To update the shape parameters of the elliptical, one needs to construct a pseudo-measurement using the 2-fold Kronecker product (Kp) (see appendix). Then, based on the KF update equation and the pseudo-measurement $\mathcal{Z}_k^{(i)}$, the shape parameters are updated as

$$e_{k|k}^{(i)} = \hat{e}_{k|k-1}^{(i-1)} + M_{k|k}^{e\mathcal{Z}(i)} \left(M_{k|k}^{\mathcal{Z}(i)}\right)^{-1} \left(\mathcal{Z}_k^{(i)} - \bar{\mathcal{Z}}_k^{(i)}\right) \tag{34}$$

$$P_{k|k}^{e(i)} = P_{k|k}^{e(i-1)} - P_{k|k}^{e\mathcal{Z}(i)} \left(P_{k|k}^{\mathcal{Z}(i)}\right)^{-1} \left(P_{k|k}^{e\mathcal{Z}(i)}\right)^{\mathrm{T}} \tag{35}$$

where

$$P_{k|k}^{\mathcal{Z}(i)} = F\left(P_{k|k}^{z(i)} \otimes P_{k|k}^{z(i)}\right)\left(F + \tilde{F}\right)^{\mathrm{T}} \tag{36}$$

$$P_{k|k}^{e\mathcal{Z}(i)} = P_{k|k}^{e(i-1)} \left(\hat{J}_{k|k}^{(i-1)}\right)^{\mathrm{T}} \tag{37}$$

$$J = \begin{bmatrix} 2M_1 P^h C_1 \\ 2M_2 P^h C_2 \\ M_1 P^h C_2 + M_2 P^h C_1 \end{bmatrix}. \tag{38}$$

Here, $M_{k|k}^{\mathcal{Z}(i)}$ denotes the covariance of the pseudo-measurement, $M_{k|k}^{e\mathcal{Z}(i)}$ is the covariance between the pseudo-measurement and the shape parameters. The derivation of (47) can be found in [18]. The covariance matrix of the measurement is denoted by

$$P_k^{z(i)} = \begin{bmatrix} p_{11} & p_{12} \\ p_{12} & p_{22} \end{bmatrix}. \tag{39}$$

The $\bar{\mathcal{Z}}_k^{(i)}$ represents the expected ith predicted pseudo-measurement and

$$\bar{\mathcal{Z}}_k^{(i)} = F\mathrm{vect}\left\{P_{k|k}^{z(i)}\right\} \tag{40}$$

$$\mathrm{vect}\left\{P_{k|k}^{z(i)}\right\} = [p_{11}\ p_{12}\ p_{12}\ p_{22}]^{\mathrm{T}}. \tag{41}$$

Based on the motion state and shape update, the reward function of the SAC-EOT algorithm at time k can be expressed explicitly as

$$\begin{aligned} r_k = {} & \log\left(1 + \left| es_{k-1}\left|P_{k-1}^{r(i)}\right| - es_k\left|P_k^{r(i)}\right|\right|\right) \\ & \times \mathrm{sign}\left(es_{k-1}\left|P_{k-1}^{r(i)}\right| - es_k\left|P_k^{r(i)}\right|\right) \\ & + \log\left(1 + \left| es_{k-1}\left|P_{k-1}^{e(i)}\right| - es_k\left|P_k^{e(i)}\right|\right|\right) \\ & \times \mathrm{sign}\left(es_{k-1}\left|P_{k-1}^{e(i)}\right| - es_k\left|P_k^{e(i)}\right|\right). \end{aligned} \tag{42}$$

The SAC algorithm interacts with the environment of EOT, obtains corresponding reward values based on various selected actions, and then selects the action with the maximum reward value as the optimal action, thereby accommodating the uncertainty in the elliptical parameters and unknown and time-varying MNC.

Algorithm 1. SAC-EOT.

```
Input: initial state of kinematic $x_0$, elliptical parameter $e_0$, SAC parameters $B$, $O_Q$, $O_\pi$
Output: Optimized state estimates $\hat{x}_{k|k}$, $\hat{e}_{k|k}$
Initialization parameters: $S, A, r_k, Q(s,a;O_Q), \pi(a|s;O_\pi)$
for each time step $k = 1$ to $T$ do
    for each iteration $t = 1$ to $N$ do
        Obtaining measurement $Z_k$ from sensors using (3)
        Selection of action $a_k$ according to the current strategy
        Measurement Update:
        Calculation of pseudo-measurements $\mathcal{Z}_k^{(i)}$ using (55)
        Update the state $\hat{x}_{k|k}^{(i)}$ and covariance $P_{k|k}^{e(i)}$ using (20) and (21)
        Update the elliptical parameter $e_{k|k}^{(i)}$ and covariance $P_{k|k}^{e(i)}$ using (34) and (35)
        Calculation of reward $r_k$ under (42)
        Storing Experiences to the replay buffer $B$
        Sample replayed experience from the $B$
        Update the Critic network $O_Q$ using (15)
        Update the Actor network $O_\pi$ using (17)
        Update target network parameter $\overline{O}_Q \leftarrow \tau O_Q + (1-\tau)O_Q$
    end for
    Time Update:
    Compute the time update of the state $\hat{x}_{k|k+1}^{(0)}$, $P_{k|k+1}^{r(0)}$ using (43) and (44)
    Compute the time update of the elliptical parameters $\hat{e}_{k|k+1}^{(0)}$, $P_{k|k+1}^{e(0)}$ using (45) and (46)
end for
```

3.4. Time update

Since the evolution of the kinematic state of the elliptical EO is linear, the time update process can be performed using the standard KF time update, i.e.

$$\hat{x}_{k|k+1}^{(0)} = F_k^r \hat{x}_{k|k}^{(n_k)} \tag{43}$$

$$P_{k|k+1}^{r(0)} = F_k^r P_{k|k}^{r(n_k)} (F_k^r)^{\mathrm{T}} + P_r^w \tag{44}$$

where $(\cdot)_{k|k+1}^{(0)}$ denotes the estimation at time k. Similarly, the time update of the elliptical parameters are as follows

$$\hat{e}_{k|k+1}^{(0)} = F_k^e \hat{e}_{k|k}^{(n_k)} \tag{45}$$

$$P_{k|k+1}^{e(0)} = F_k^e P_{k|k}^{e(n_k)} (F_k^e)^{\mathrm{T}} + P_e^w. \tag{46}$$

The pseudo-code of the proposed algorithm is shown in algorithm 1 and the source code is available at the following URL: https://github.com/kadlfj/drl-for-eot.

4. Simulation

This section sets three different simulation scenarios, and the proposed algorithm is evaluated using simulated data. To illustrate the effectiveness of the proposed algorithm

(Algorithm2) clearly, it is compared with MN-MEM-EKF-EOT (Algorithm1), MN-SOEKF-EOT (Algorithm3), MN-EKF-EOT (Algorithm4), and MN-EM-EOT (Algorithm5). Additionally, two evaluation metrics are adopted to assess the performance of different algorithms. Firstly, the root mean square error (RMSE) is used to evaluate the estimation accuracy of the object's motion state. Secondly, the Gaussian Wasserstein distance (GWD) [39] is employed to evaluate the comprehensive performance of the object, including both position and extension. The GWD is defined as

$$\mathrm{GWD}^2 = \|x_1 - x_2\|_2^2 + tr\left(\Sigma_1 + \Sigma_2 - 2\left(\sqrt{\Sigma_1}\Sigma_2\sqrt{\Sigma_1}\right)^{1/2}\right). \tag{47}$$

Here, x_1 and x_2 represent the elliptical's centroid position, while Σ_1 and Σ_2 denote the positive definite symmetric shape matrices.

4.1. Experimental results

4.1.1. Scenario 1. Considering tracking an elliptical EO (modeled after the Nimitz-class aircraft carrier [14]) with the major axis $l_1 = 170\,\mathrm{m}$, minor axis $l_2 = 40\,\mathrm{m}$. The object moves in a straight line and the motion orientation keeps as $\pi/3$. The sampling interval is $T_\mathrm{s} = 10\,\mathrm{s}$. The number of measurements at each time is drawn from a Poisson distribution with a mean of 8. The object starts from the origin and moves at a constant speed of $50\,\mathrm{km\,h^{-1}}$, with fixed but unknown axis lengths.

For all algorithms, the initial motion state is set to be $x_0 = [100, 100, 10, -17]^\mathrm{T}$, the covariance is $P_0^r = \mathrm{diag}\,(900, 900, 16, 16)$, the process noise covariance is $\mathrm{diag}\,(100, 100, 1, 1)$, the MN $m_k \sim \mathcal{N}\,(\bar{m}_k, \sigma_k)$ with mean $\bar{m}_k = 2.5$ and covariance $\sigma_k = 2 + 0.05\cos\,(\pi k/T_\mathrm{s})$, and the additive measurement noise covariance $R_k = \mathrm{diag}\,(2000, 80)$. For MN-MEM-EKF-EOT, The prior covariance of the shape parameters is $C_0^p = \mathrm{diag}\left(0.2, 20^2, 20^2\right)$, the process noise covariance of the shape parameters is $C_\mathrm{p}^w = \mathrm{diag}\,(0.5, 0.001, 0.001)$. For MN-EM-EOT, the iteration threshold is $\delta = 10^{-6}$, and the algorithm iterates 10 times, setting the initial variance of the orientation angle vector $P_0^s = \mathrm{diag}\left(0.5^2, 0.5^2\right)$ and the covariance of the shape process noise $Q_\mathrm{s} = \mathrm{diag}\left(0.05^2, 0.05^2\right)$. For MN-EKF-EOT, setting the process noise covariance $\tilde{C}_\mathrm{p}^w = \mathrm{diag}\,(0.5, 60, 60)$ for the shape variable, and for MN-SOEKF-EOT, setting $\bar{C}_\mathrm{p}^w = \mathrm{diag}\,(0.04, 0.5, 0.5)$.

Figure 4 shows the tracking estimation results for the trajectory and shape of the EO. In the linear motion scenario, despite the influence of multiplicative measurement noise, various algorithms can still accurately estimate the position and shape of the ellipse. However, enlarged local views show preliminary differences between the algorithms. In particular, the SAC-EOT algorithm outperforms others in evaluating the axis length and motion orientation of the ellipse. Next, a numerical analysis is performed to compare the performance differences between the algorithms further.

Figures 5–8 show the GWD and RMSE results for 500 MC simulations. These numerical results illustrate the significant advantages of the proposed algorithm, especially in the initial tracking phase. The algorithm provides more accurate estimates of EO motion and shape information with faster convergence and better stability. This advantage is particularly evident in the average GWD distance shown in figure 5, as the SAC-EOT algorithm provides superior performance in estimating position, orientation, and velocity, as shown in figures 6–8. The SAC-EOT algorithm effectively overcomes the effects of time-varying axis lengths and unknown MNC, significantly improving the overall estimation performance.

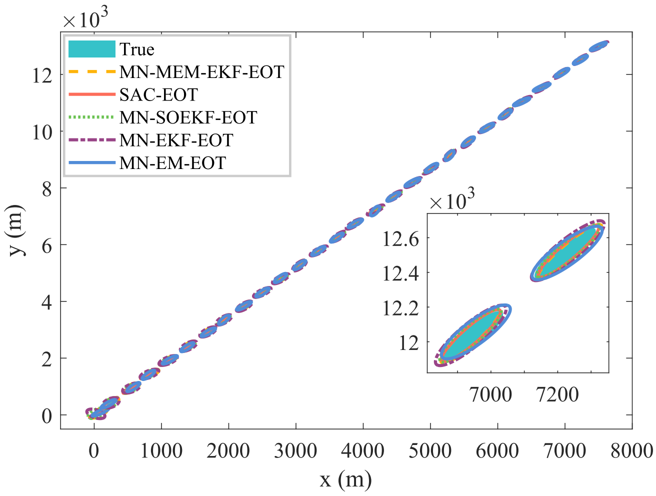


Figure 4. The estimation of the EOT (single Monte Carlo (MC) run).

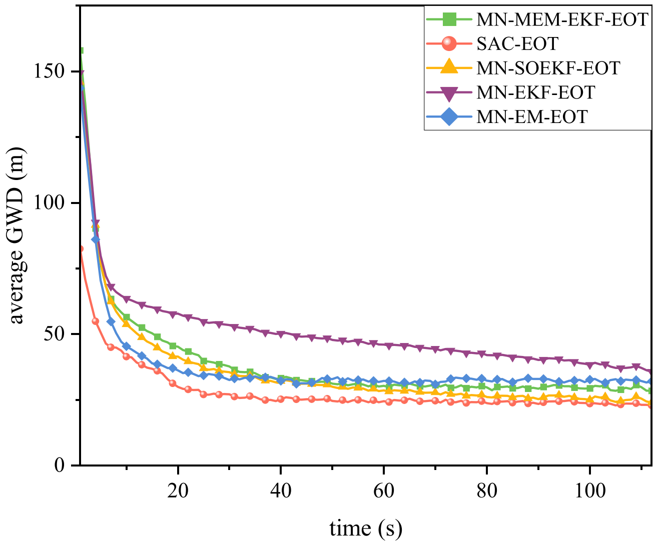


Figure 5. The average GWD for EOT.

Figure 9 explicitly illustrates the effectiveness of several algorithms with different numbers of measurements. As shown, the number of measurements significantly impacts on the accuracy of EOT estimates, with a higher number of measurements resulting in improved estimation accuracy.

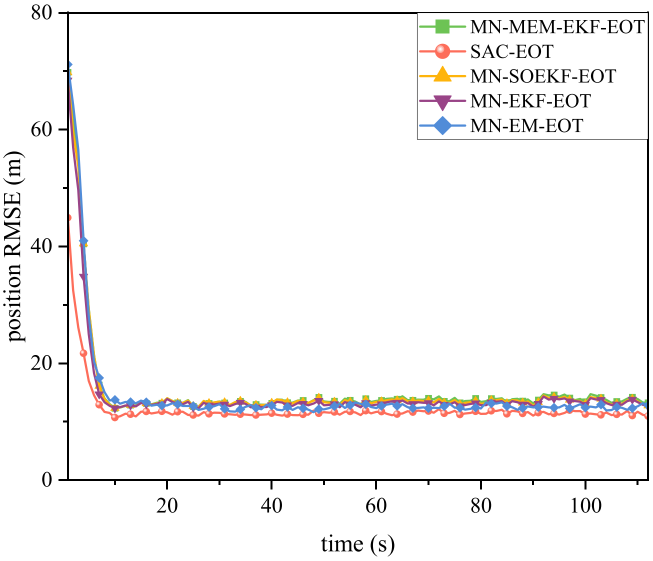


Figure 6. The RMSE of the EO position.

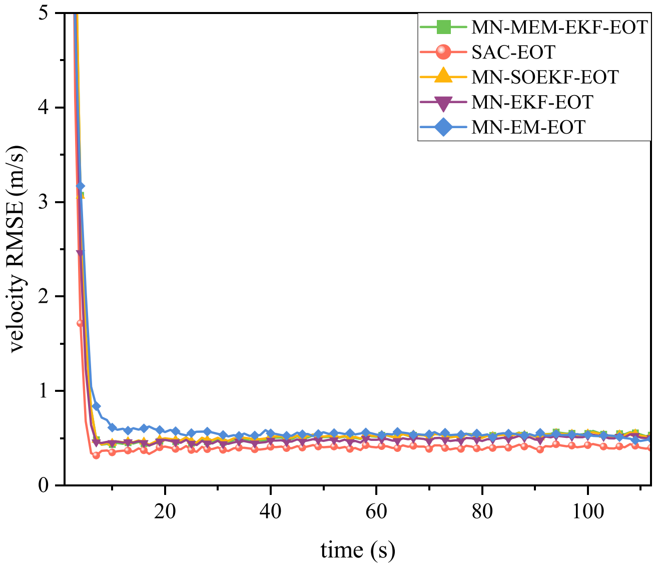


Figure 8. The RMSE of the EO velocity.

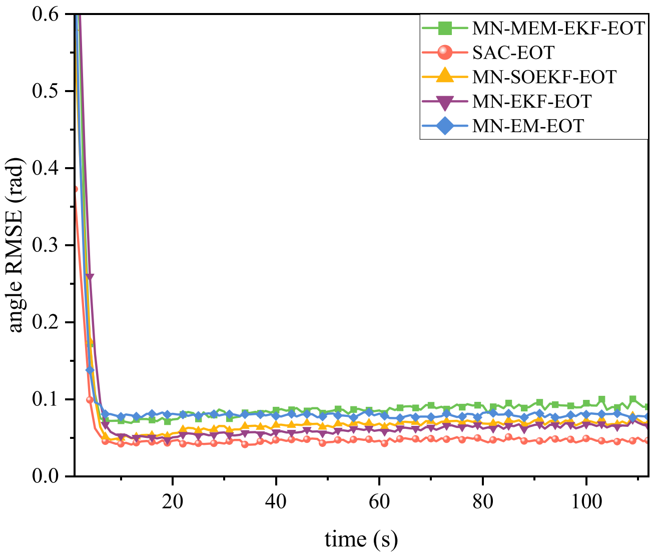


Figure 7. The RMSE of the EO orientation angle.

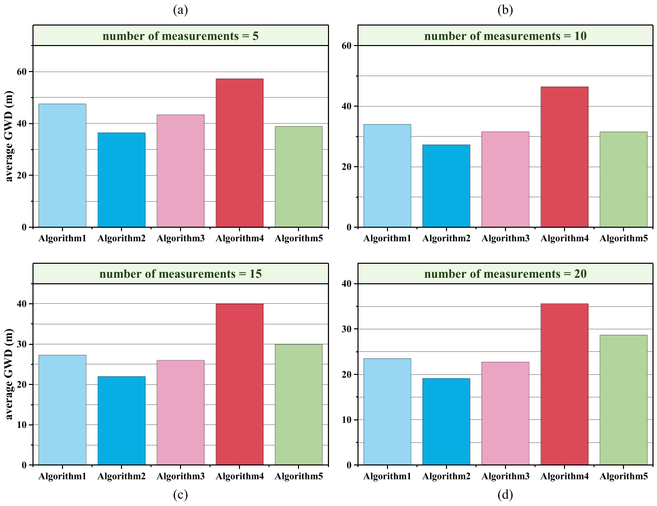


Figure 9. The average GWD under different number of measurements.

Under various measurement conditions, the proposed SAC-EOT algorithm consistently outperforms other algorithms in estimation performance, especially under low measurement conditions.

Then, to evaluate the convergence characteristics of different algorithms more intuitively, the analysis is carried out by observing the trend of GWD and calculating its relative rate of change[1], and the specific results are presented in table 1. The proposed algorithm can quickly converge to a stable state in the shortest time step from table 1. This result shows the advantage of the algorithm in terms of convergence, which can adapt to environmental changes and achieve a stable tracking effect faster than the other comparative algorithms. The MN-EKF-EOT algorithm is second only to the proposed algorithm regarding convergence speed. However, the MN-EM-EOT algorithm has the slowest convergence speed.

4.1.2. Scenario 2. Considering an EO with the same shape in Scenario 1, its trajectory consists of a straight path with one 90° turn and two 45° turns. The initial motion orientation is 0°. For the MN-MEM-EKF-EOT algorithm, the prior covariance of shape parameters $C_0^p = \operatorname{diag}\left(0.4, 20^2, 20^2\right)$ and the process noise covariance of shape parameters $C_{\mathrm{p}}^w = \operatorname{diag}\left(0.05, 0.001, 0.001\right)$. For MN-EM-EOT,

[1] For each set of data $\{v_1, v_2, \ldots, v_N\}$, define the relative rate of change of adjacent time steps: $r_i = |v_{i+1} - v_i| / v_i,\ i = 1, 2, \ldots, N-1$. If 5 consecutive r_j in a given interval $i, i+1, \ldots, i+4$ satisfy $r_j < 0.1$, the end time step at which this occurs first is noted as $i+4+1$, i.e. the algorithm reaches convergence at the step $i+5$. If five consecutive intervals less than 10% cannot be found in $N-1$ neighboring time steps, the algorithm is considered not to have converged yet for a given sequence length.

Table 1. Comparative analysis of scenario 1.

Algorithm	convergence time step (s)
MN-MEM-EKF-EOT	11
SAC-EOT	**9**
MN-SOEKF-EOT	11
MN-EKF-EOT	10
MN-EM-EOT	12

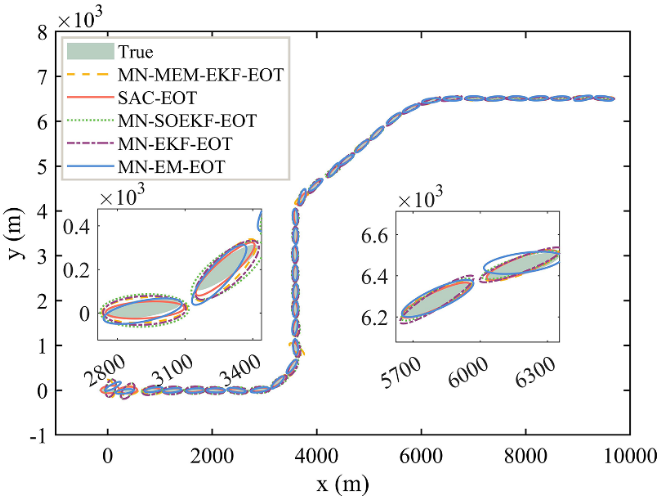


Figure 10. The estimation of the EOT (single MC run).

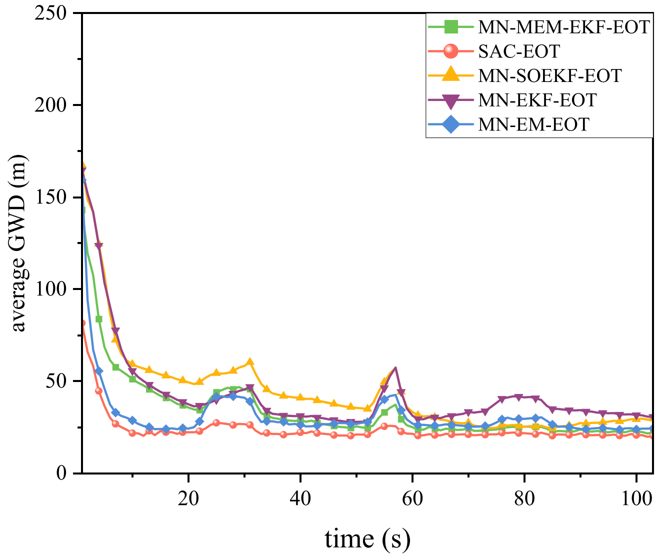


Figure 11. The average GWD for EOT.

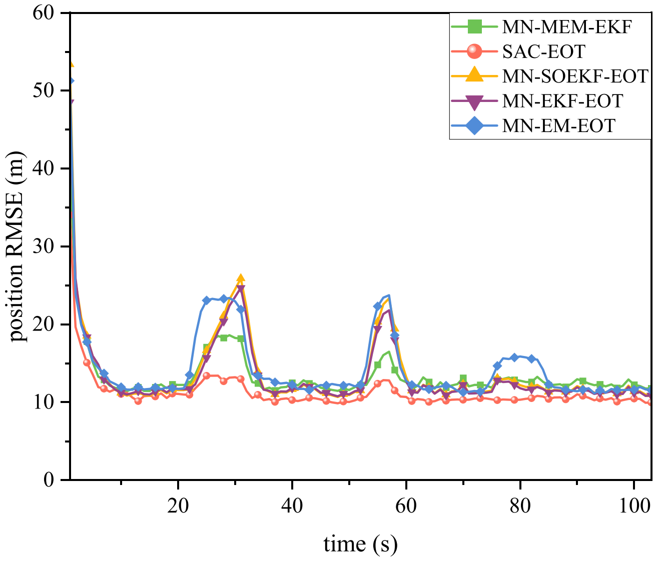


Figure 12. The RMSE of the EO position.

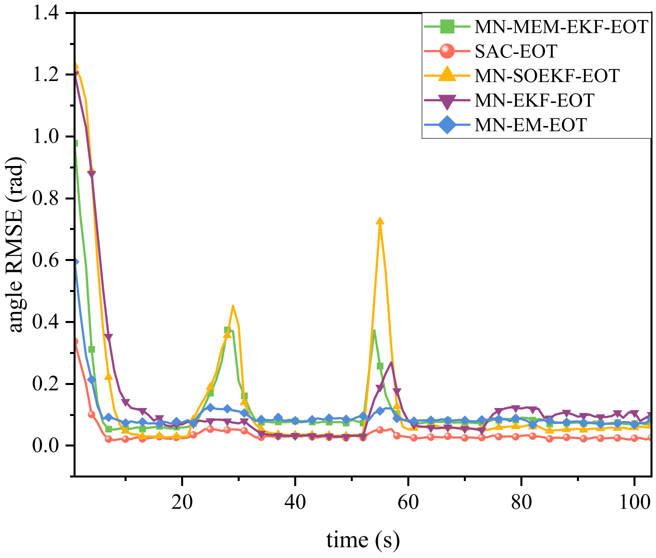


Figure 13. The RMSE of the EO orientation angle.

$Q_s = \mathrm{diag}\left(0.25^2, 0.25^2\right)$ is specified. The remaining initial parameters are identical to those in scenario 1.

Figure 10 illustrates the elliptical EOT and estimation result in scenario with different motion models, including zoomed views at particular moments. The results show that the proposed method more accurately estimates the state of the EO during maneuvering than other algorithms.

Figures 11–14 visually compare the performance differences between algorithms using 500 MC simulations and various evaluation metrics. Figure 11 shows that the proposed SAC-EOT algorithm has better convergence in estimating the shape of the EO, achieving faster and more accurate state estimation. In addition, the SAC-EOT algorithm provides more stable estimation performance during object maneuvering, with the estimation error less influenced by changes in the object's motion state. In contrast, the performance of other algorithms is significantly degraded, and more obvious biases in EO shape estimation are observed, especially in the MN-SOEKF-EOT and MN-EKF-EOT algorithms.

Figures 12–14 confirm the effectiveness of the proposed SAC-EOT algorithm. The superior performance of the SAC-EOT algorithm in estimating the EO shape is demonstrated by its more accurate estimation of the center of gravity position, orientation angle, and velocity. Figure 12 shows the estimation

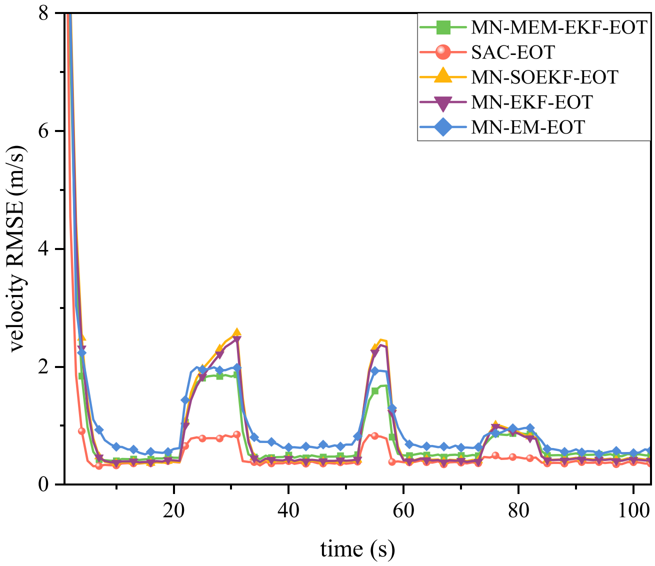


Figure 14. The RMSE of the EO velocity.

of the centroid position during object maneuvering. The SAC-EOT algorithm shows a more stable centroid position estimate with less influence from object maneuvering. In contrast, the MN-EM-EOT algorithm significantly increases the position estimation error in each maneuvering, indicating lower stability under dynamic condition. Figure 13 shows a similar trend. As the estimation error for the speed of the EO increases, the position estimation error also increases accordingly. Figure 14 illustrates the error in estimating the motion orientation of the EO. Accurate orientation estimation is crucial for precise state estimation, especially during changes in motion orientation. The SAC-EOT algorithm shows lower error in orientation estimation for all three turns, while the MN-MEM-EKF-EOT and MN-SOEKF-EOT algorithms show less stable orientation estimation. These results show that the SAC-EOT algorithm outperforms other algorithms regarding stability and accuracy in complex environments, further supporting its superiority in EOT.

The improved performance of SAC-EOT is attributed to several key factors: (1) The SAC-EOT algorithm leverages an experienced pool to store rich historical data. This data is crucial because it allows the algorithm to learn optimal strategies over time. The pool of experience helps eliminate uncertainty in estimating motion orientation and axis length by leveraging previous information to make more informed decision. (2) Through interactive learning, the SAC-EOT algorithm can adaptively adjust the parameters of MNC and elliptical shape, thereby significantly improving its robustness. (3) The SAC-EOT algorithm has a faster convergence speed than other methods, enabling faster adaptation to changing conditions.

Table 2 shows the average values of various evaluation metrics for different EOT algorithms under Scenario 2. From the table, it can be seen that the SAC-EOT algorithm significantly outperforms the other algorithms on these metrics. The MN-EM-EOT and MN-MEM-EKF-EOT algorithms have their strengths and weaknesses, with MN-EM-EOT showing a clear advantage in GWD. The performance of the MN-SOEKF-EOT and MN-EKF-EOT algorithms is comparable but generally falls behind. These results further confirm the superior performance of the SAC-EOT algorithm on EOT tasks and demonstrate its robustness and accuracy across various evaluation metrics. The SAC-EOT algorithm effectively handles the uncertainties caused by MN and elliptic parameters variations, which provides a key advantage for adaptive EOT estimation. Consequently, the SAC-EOT algorithm performs exceptionally well in dynamic and complex environments, highlighting its significant practical value for real-world application.

Table 2. Comparative analysis of scenario 2.

Algorithm	GWD (m)	position RMSE (m)	velocity RMSE (m)	angle RMSE (rad)
MN-MEM-EKF-EOT	34.28	13.38	0.92	0.11
SAC-EOT	**23.88**	**11.27**	**0.61**	**0.04**
MN-SOEKF-EOT	44.17	13.71	0.98	0.14
MN-EKF-EOT	42.71	13.46	0.96	0.13
MN-EM-EOT	31.31	14.44	1.08	0.09

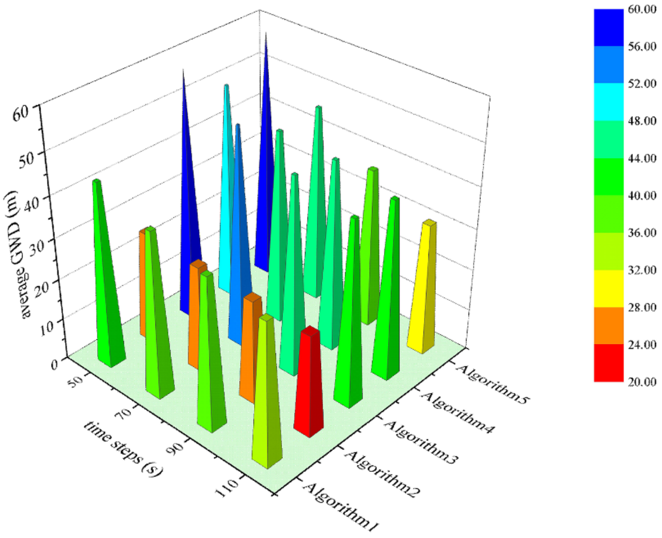


Figure 15. The average GWD at different time steps.

Figures 15 and 16 respectively validate the effectiveness of the proposed algorithm at different time steps and varying numbers of measurements. In figure 15, with additive measurement noise assumed as, experiments under various time steps are conducted to test the estimation performance of different algorithms. Different shapes and colors in the figure represent different GWD values. The results show that the SAC-EOT algorithm has more reliable performance than others.

Figure 16 illustrates the performance of different algorithms with different numbers of measurements. The experimental results show that the SAC-EOT algorithm always performs best regardless of the number of measurements and

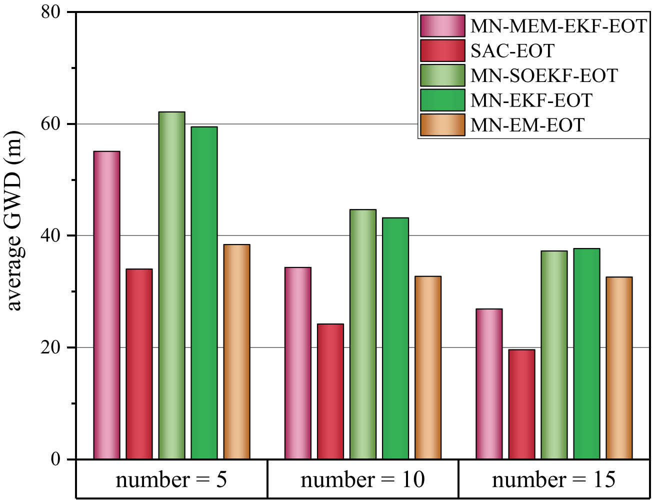


Figure 16. The average GWD under different number of measurements.

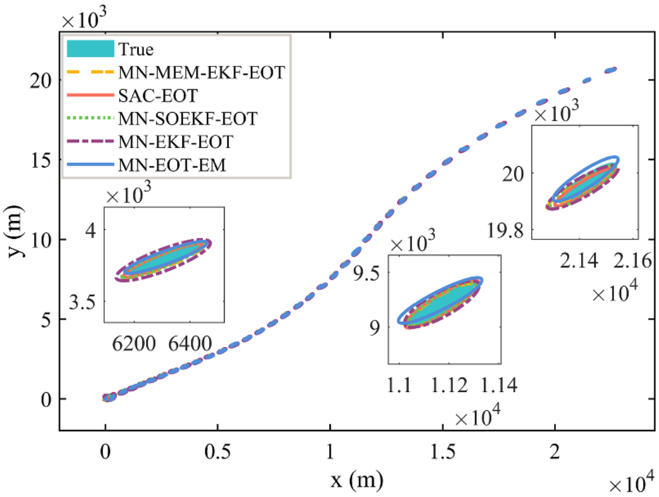


Figure 17. The estimation of the EOT (single MC run).

Table 3. Comparative analysis of scenario 2.

Algorithm	convergence time step (s)
MN-MEM-EKF-EOT	11
SAC-EOT	**10**
MN-SOEKF-EOT	12
MN-EKF-EOT	13
MN-EM-EOT	11

significantly outperforms other algorithms. This indicates that the SAC-EOT algorithm is adaptable to changes in the number of measurements. This adaptability is critical for EOT tasks because the number of available measurements can fluctuate due to sensor resolution, environmental conditions, or changes in object dynamics. The results further confirm the significant advantages of the SAC-EOT algorithm in EOT, especially under complex and dynamic environmental conditions. The SAC-EOT algorithm maintains highly accurate estimates across different measurement conditions and effectively adapts to uncertainty and variability. This feature provides reliable support for EOT in practical applications and ensures precise and stable tracking performance even under changing measurement conditions.

Table 3 compares the convergence time steps of the algorithms in Scenario 2. The algorithm convergence in Scenario 2 changes due to the rotation during the object's movement. For this reason, the time step at which the algorithms reached the convergence condition for the first time was counted. As shown in table 3, the proposed SAC-EOT algorithm maintains the fastest convergence speed despite the changes in the tracking environment affecting the convergence speed of the algorithms, which further validates its excellent adaptability in different environments. In contrast, MN-EM-EOT and MN-MEM-EKF-EOT are slightly inferior in terms of convergence, while MN-EKF-EOT has the slowest convergence rate. This result highlights the efficiency advantage of the proposed algorithm in complex dynamic environment.

4.1.3. Scenario 3.

In practical engineering applications, the motion of the EO is usually accompanied by a change in velocity and a change in shape. Therefore, in this scenario, the EO is accelerated with an acceleration of 0.25 m s^{-2}. The EO maintains the initial motion direction of 30° in time steps 1–30, and then, it makes a gradual steering in time steps 31–70, gradually shifting from 30° to 60°, and then finally restoring to the initial direction again in the remaining time steps. In addition, the EO undergoes a shape enlargement mutation at step 41, in which its axis length is mutated to 150% of the original, and a shape reduction mutation at step 81, in which the axis length is changed to 50% of the original. The scenarios in real engineering are simulated by such settings to verify the performance of the proposed algorithm further. The rest of the initial parameters are the same as those in Scenario 1.

Figure 17 demonstrates the motion trajectory of the EO, the effect of different algorithms on the tracking estimation of the EO, and the local magnification of the EO estimated at the original size, zoomed in, and zoomed out. Figure 18 shows the axis length variation of the EO. The GWD of the EOT estimation is shown in figure 19. From figure 19, it can be seen that abrupt changes in the shape of the EO cause a degradation in the estimation performance of the algorithms. At the moment of sudden shape change, the estimation errors of several algorithms have a significant increase, but with iterations, they all recover to a stable state. Specifically, it can be seen from the change of GWD that the proposed algorithm has the smallest change when the object shape changes abruptly, which shows strong adaptability to the sudden change of shape. It can also recover to a stable state quickly after the sudden shape change, maintaining the optimal estimation performance. The performance of the other algorithms is relatively weak, especially the MN-EKF-EOT algorithm, which is less capable of adapting to such changes.

Figures 20–22 show the RMSE of the different algorithms for estimating the EO's position, orientation, and velocity tracking, respectively. From these sets of comparisons, it

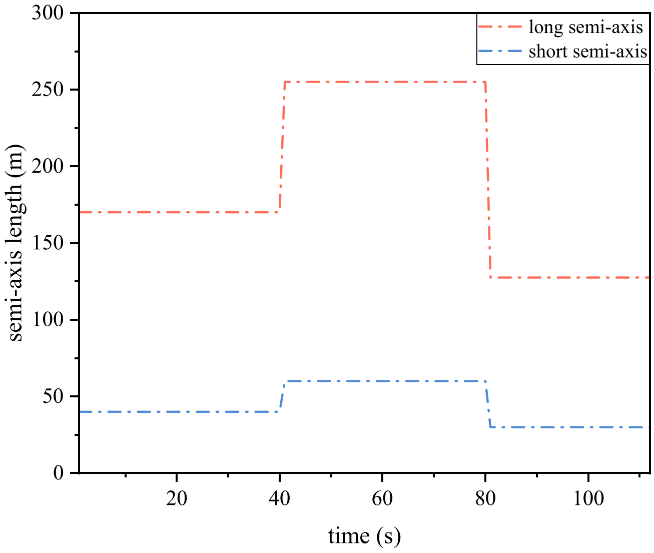


Figure 18. The axis length change of the EO.

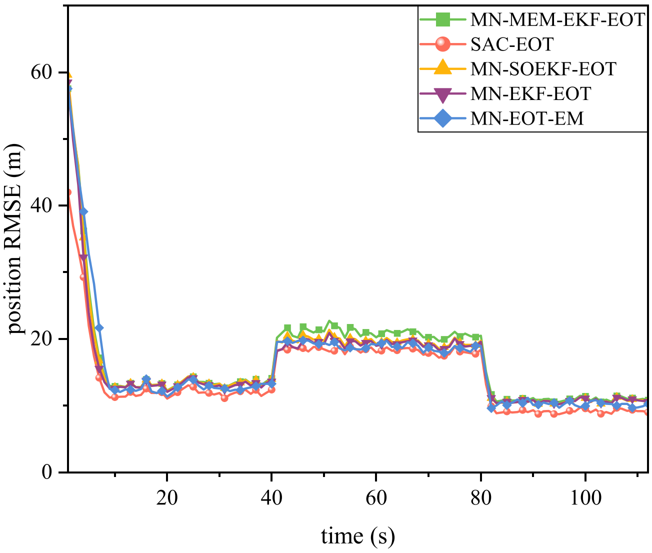


Figure 20. The RMSE of the EO position.

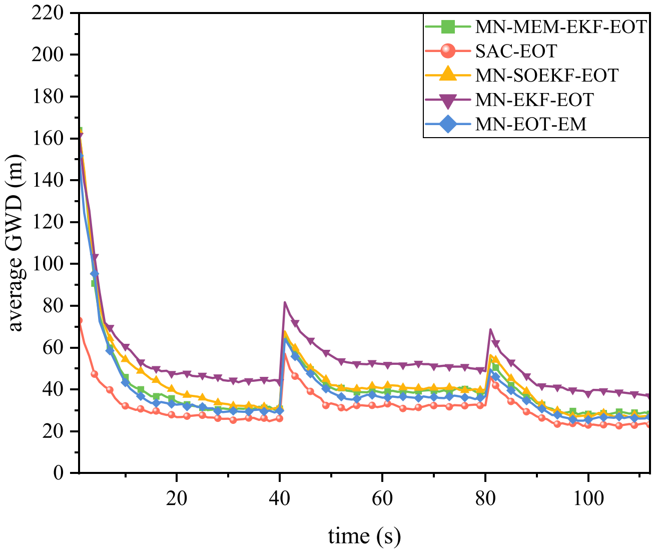


Figure 19. The average GWD for EOT.

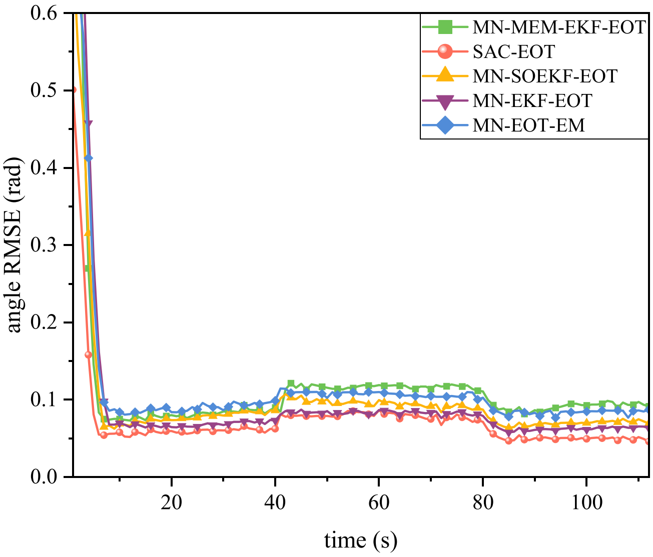


Figure 21. The RMSE of the EO orientation angle.

can be seen that the sudden change in the shape of the EO also causes estimation errors in its position, orientation, and velocity. An in-depth analysis of the curve changes in the figures shows that the proposed SAC-EOT algorithm consistently maintains superior performance in this dynamic tracking environment, both at the moment of the abrupt shape change and during the continuous tracking process after the shape change. This further demonstrates that SAC-EOT is superior to other conventional algorithms in its ability to adapt to such dynamic changes.

The EOT technology is crucial in practical engineering applications, especially in tracking objects such as aircraft, ships, and vehicles. For example, in traffic monitoring on urban road, vehicles as EOs, the accurate tracking of their shapes and kinematic states is crucial for accident warning and other aspects. The SAC-EOT algorithm proposed in this paper shows great potential in this field. Aiming at the differences in size and shape of different types of vehicles (e.g. cars, trucks, etc), SAC-EOT is able to accurately estimate the shape of a vehicle and achieve stable tracking by effectively processing sensor measurements. Especially in complex traffic environment, such as trajectory changes due to road curves and MN due to disturbances in the surrounding environment, SAC-EOT can effectively address these challenges and maintain high tracking accuracy. This makes the SAC-EOT algorithm highly feasible for practical engineering applications, especially for object tracking in dynamic and complex environment.

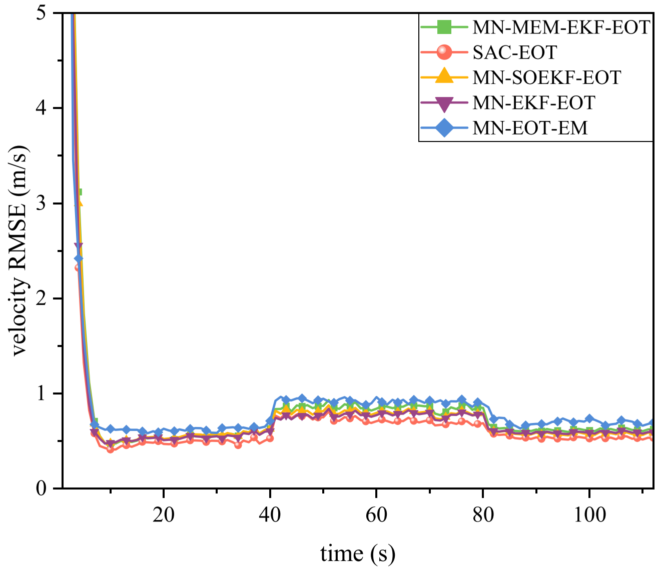


Figure 22. The RMSE of the EO velocity.

5. Conclusion

This paper proposes a novel adaptive elliptical EOT method based on DRL to address the challenges in EOT due to the unknown MNC and the uncertainty of the elliptical shape parameters. The proposed method exploits the interaction between the agent and the tracking environment and allows the algorithm to continuously adjust parameters based on the rewards received from the environment. This adaptive approach improves the accuracy and adaptability of EOT in complex environments. In order to verify the effectiveness of the proposed SAC-EOT method, three simulation scenarios are designed, compared, and analyzed with existing EOT algorithms. In the experiments, the proposed method is compared with the classical algorithms such as MN-MEM-EKF-EOT, MN-SOEKF-EOT, MN-EKF-EOT, and MN-EM-EOT, and RMSE is used to measure the accuracy of position, velocity, and direction angle estimation of the EO, and GWD evaluates the overall estimation error of the EO. Experimental results show that the SAC-EOT algorithm performs better in estimating the centroid and shape of the EO compared to traditional elliptical EOT algorithms. The SAC-EOT algorithm achieves higher estimation accuracy and exhibits greater robustness in the presence of MN and dynamic environmental changes. Future work will extend this method to address additional challenges in EOT. Possible research directions include multi-EO or non-elliptical EO.

Data availability statement

The data cannot be made publicly available upon publication because no suitable repository exists for hosting data in this field of study. The data that support the findings of this study are available upon reasonable request from the authors.

Acknowledgments

This work is supported by the National Natural Science Foundation of China (62163023, 61873116, 62366031, 62363023), the Gansu Provincial Basic Research Innovation Group of China (25JRRA058), the Central Government's Funds for Guiding Local Science and Technology Development of China (25ZYJA040), the Gansu Provincial Key Talent Project of China (2024RCXM86) and the Gansu Provincial Special Fund for Military-Civilian Integration Development of China.

Appendix

A.1. Variational Bayesian approximation

According to Bayes' theorem

$$p\left(\Xi_k \middle| z_{1:k}^{(i)}, \Lambda_k\right) = \frac{p\left(z_{1:k}^{(i)} | \Xi_k\right) p\left(\Xi_k | \Lambda_k\right)}{p\left(z_{1:k}^{(i)} | \Lambda_k\right)} \tag{48}$$

where $\Xi_k = \{x_k, \sigma_k, \bar{R}_k\}$, $\bar{R}_k = \tilde{R}_k + \breve{R}_k$ and $\Lambda_k = \left(\xi_{k|k}, \zeta_{k|k}\right)$. As obtaining an analytical solution for $p\left(\Xi_k \middle| z_{1:k}^{(i)}, \Lambda_k\right)$ directly is not feasible, utilize variational approximation to derive it, yields

$$\log q(\eta) = E_{\Xi_k(-\eta)}\left[\log p\left(z_{1:k}^{(i)} | \Xi_k\right) p\left(\Xi_k | \Lambda_k\right)\right]. \tag{49}$$

Subsequently, $p\left(z_{1:k}^{(i)} | \Xi_k\right) p\left(\Xi_k | \Lambda_k\right)$ can be decomposed as

$$\begin{aligned} p\left(z_{1:k}^{(i)} | \Xi_k\right) p\left(\Xi_k | \Lambda_k\right) &= \mathcal{N}\left(x_k; \hat{x}_{k|k-1}, P_{k|k-1}\right) \\ &\times IG\left(\delta_k; \xi_{k|k-1}, \zeta_{k|k-1}\right) p\left(z_{1:k-1}^{(i)}\right) \mathcal{N}\left(z_k^{(i)}; \bar{m}_k H_k x_k, \tilde{R}_k\right) \\ &\times \mathcal{N}\left(z_k^{(i)}; \bar{m}_k H_k x_k, \breve{R}_k\right). \end{aligned} \tag{50}$$

Here, the MNC is modeled as a random variable obeying an inverse gamma (IG) distribution, i.e. $p\left(\delta_k | z_{1:k-1}\right) = \mathrm{IG}\left(\delta_k; \xi_{k|k-1}, \zeta_{k|k-1}\right)$, and

$$\xi_{k|k-1} = \rho^* \xi_{k-1|k-1} \tag{51}$$

$$\zeta_{k|k-1} = \rho^* \zeta_{k-1|k-1} \tag{52}$$

where ρ^* is the parameter obtained by DRL training. Let $\eta = x_k$, substituting (53) into (52), can get

$$\begin{aligned} \log q(x_k) \propto \mathcal{N}\left(z_k^{(i)}; \bar{m}_k H_k x_k, \tilde{R}_k\right) \mathcal{N}\left(z_k^{(i)}; \bar{m}_k H_k x_k, \breve{R}_k\right) \\ \times \mathcal{N}\left(x_k; \hat{x}_{k|k}, P_{k|k}^r\right). \end{aligned} \tag{53}$$

A.2. 2-fold Kronecker product

The 2-fold Kp($\otimes$) of a two-dimensional vector $z = [z_1\ z_2]^{\mathrm{T}}$ is denoted as

$$z \otimes z = \left[z_1^2\ z_1 z_2\ z_2 z_1\ z_2^2\right]. \tag{54}$$

The pseudo-measurement is given by

$$\mathcal{Z}_k^{(i)} = F\left(\left(z_k^{(i)} - \bar{z}_k^{(i)}\right) \otimes \left(z_k^{(i)} - \bar{z}_k^{(i)}\right)\right) \tag{55}$$

where

$$F = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 1 & 0 & 0 \end{bmatrix} \text{ or } \tilde{F} = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 1 & 0 \end{bmatrix} \tag{56}$$

which is used to remove duplicate terms in the 2-fold Kp.

ORCID iDs

Ziwen Zhao https://orcid.org/0000-0001-7298-5599
Hui Chen https://orcid.org/0000-0002-1102-9912

References

[1] Dahal P, Mentasti S, Arrigoni S, Braghin F, Matteucci M and Cheli F 2023 Extended object tracking in curvilinear road coordinates for autonomous driving *IEEE Trans. Intell. Veh.* **8** 1266–78

[2] Xia Y, Wang P, Berntorp K, Svensson L, Granström K, Mansour H, Boufounos P and Orlik P V 2021 Learning-based extended object tracking using hierarchical truncation measurement model with automotive radar *IEEE J. Sel. Top. Signal Process.* **15** 1013–29

[3] Haag S, Duraisamy B, Govaers F, Fritzsche M, Dickmann J and Koch W 2021 Extended object tracking assisted adaptive multi-hypothesis clustering for radar in autonomous driving domain *2021 21st Int. Radar Symp. (IRS)* pp 1–10

[4] Liu J, Wang Z, Cheng D, Chen W and Chen C 2022 Marine extended target tracking for scanning radar data using correlation filter and Bayes filter jointly *Remote Sens.* **14** 5937

[5] Joshi S K, Baumgartner S V and Krieger G 2022 Tracking and track management of extended targets in range-Doppler using range-compressed airborne radar data *IEEE Trans. Geosci. Remote Sens.* **60** 1–20

[6] Fowdur J S, Baum M and Heymann F 2019 Tracking targets with known spatial extent using experimental marine radar data *2019 22th Int. Conf. on Information Fusion (FUSION)* (IEEE) pp 1–8

[7] Barbary M, Hafez A S, El-azeem M H A, Soliman T, Elbanna A E, Shalaby M, Saad W M and Mabrouk M 2022 Extended drones tracking from ISAR images with Doppler effect and orientation based robust sub-random matrices algorithm *IEEE Trans. Veh. Technol.* **71** 12648–66

[8] Guerlin L, Pannetier B, Rombaut M and Derome M 2020 Study on group target tracking to counter swarms of drones *Proc. SPIE* **11423** 8–27

[9] Baum M and Hanebeck U D 2014 Extended object tracking with random hypersurface models *IEEE Trans. Aerosp. Electron. Syst.* **50** 149–59

[10] Wahlstrom N and Ozkan E 2015 Extended target tracking using Gaussian processes *IEEE Trans. Signal Process.* **63** 4165–78

[11] Yang D, Guo Y, Yin T and Lin B 2023 Cost-effective Gaussian processes based extended target tracking *IEEE Trans. Aerosp. Electron. Syst.* **59** 8282–96

[12] Zea A, Faion F, Baum M and Hanebeck U D 2016 Level-set random hypersurface models for tracking nonconvex extended objects *IEEE Trans. Aerosp. Electron. Syst.* **52** 2990–3007

[13] Koch J W 2008 Bayesian approach to extended object and cluster tracking using random matrices *IEEE Trans. Aerosp. Electron. Syst.* **44** 1042–59

[14] Feldmann M, Franken D and Koch W 2011 Tracking of extended objects and group targets using random matrices *IEEE Trans. Signal Process.* **59** 1409–20

[15] Orguner U 2012 A variational measurement update for extended target tracking with random matrices *IEEE Trans. Signal Process.* **60** 3827–34

[16] Lan J and Li X R 2016 Tracking of extended object or target group using random matrix: new model and approach *IEEE Trans. Aerosp. Electron. Syst.* **52** 2973–89

[17] Lan J and Li X R 2019 Extended-object or group-target tracking using random matrix with nonlinear measurements *IEEE Trans. Signal Process.* **67** 5130–42

[18] Yang S and Baum M 2019 Tracking the orientation and axes lengths of an elliptical extended object *IEEE Trans. Signal Process.* **67** 4720–9

[19] Li M, Lan J and Li X R 2023 Tracking of elliptical object with unknown but fixed lengths of axes *IEEE Trans. Aerosp. Electron. Syst.* **59** 6518–6233

[20] Yu X and Meng Z 2024 Robust Kalman filters with unknown covariance of multiplicative noise *IEEE Trans. Autom. Control.* **69** 1171–8

[21] Yu X and Li J 2022 Adaptive Kalman filtering for recursive both additive noise and multiplicative noise *IEEE Trans. Aerosp. Electron. Syst.* **58** 1634–49

[22] Arulkumaran K, Deisenroth M P, Brundage M and Bharath A A 2017 Deep reinforcement learning: a brief survey *IEEE Signal Process. Mag.* **34** 26–38

[23] Li Y, Feng Q, Zhang Y, Peng C, Ma Y, Liu C, Ru M, Sun J and Zhao C 2024 Peduncle collision-free grasping based on deep reinforcement learning for tomato harvesting robot *Comput. Electron. Agric.* **216** 108488

[24] Chai R, Niu H, Carrasco J, Arvin F, Yin H and Lennox B 2024 Design and experimental validation of deep reinforcement learning-based fast trajectory planning and control for mobile robot in unknown environment *IEEE Trans. Neural Netw. Learn. Syst.* **35** 5778–92

[25] Shahid A A, Piga D, Braghin F and Roveda L 2022 Continuous control actions learning and adaptation for robotic manipulation through reinforcement learning *Auton. Robot.* **46** 483–98

[26] Du Y, Chen J, Zhao C, Liao F and Zhu M 2023 A hierarchical framework for improving ride comfort of autonomous vehicles via deep reinforcement learning with external knowledge *Comput.-Aided Civ. Infrastruct. Eng.* **38** 1059–78

[27] Shi H, Chen D, Zheng N, Wang X, Zhou Y and Ran B 2023 A deep reinforcement learning based distributed control strategy for connected automated vehicles in mixed traffic platoon *Transp. Res. C* **148** 104019

[28] Chen J, Li S E and Tomizuka M 2022 Interpretable end-to-end urban autonomous driving with latent deep reinforcement learning *IEEE Trans. Intell. Transp. Syst.* **23** 5068–78

[29] Zhu P, Liang J, Luo Z and Shen X 2023 Cognitive radar target tracking using intelligent waveforms based on reinforcement learning *IEEE Trans. Geosci. Remote Sens.* **61** 5107315

[30] Luo H, Luo Y, Han B and Zeng M 2022 A learning-based noise tracking method of adaptive Kalman filter for UAV positioning *2022 IEEE 25th Int. Conf. on Intelligent Transportation Systems (ITSC)* (IEEE) pp 440–5

[31] Tang Y, Hu L, Zhang Q and Pan W 2021 Reinforcement learning compensated extended Kalman filter for attitude estimation *2021 IEEE/RSJ Int. Conf. on Intelligent Robots and Systems (IROS)* (IEEE) pp 6854–9

[32] Gu J, Li J and Tei K 2022 A reinforcement learning approach for adaptive covariance tuning in the Kalman filter *2022 IEEE 5th Advanced Information Management, Communicates, Electronic and Automation Control Conf. (IMCEC)* (IEEE) pp 1569–74

[33] Alhadhrami E, Seghrouchni A E F, Barbaresco F and Zitar R A 2023 Drones tracking adaptation using reinforcement learning: proximal policy optimization *2023 24th Int. Radar Symp. (IRS)* pp 1–10

[34] Haarnoja T, Zhou A, Abbeel P and Levine S 2018 Soft actor-critic: off-policy maximum entropy deep reinforcement learning with a stochastic actor *Proc. 35th Int. Conf. on Machine Learning (PMLR)* pp 1861–70

[35] Haarnoja T *et al* 2019 Soft actor-critic algorithms and applications (arXiv:1812.05905)

[36] Geng S, Lin X, Pang Z and Hao C 2023 Position estimation for autonomous underwater vehicles (AUV) docking based on reinforcement learning *2023 38th Youth Academic Annual Conf. of Chinese Association of Automation (YAC)* pp 820–5

[37] Peng B, Xie Y, Seco-Granados G, Wymeersch H and Jorswieck E A 2022 Communication scheduling by deep reinforcement learning for remote traffic state estimation with Bayesian inference *IEEE Trans. Veh. Technol.* **71** 4287–300

[38] Marino F, Meymand A Z and Guglieri G 2023 Navigation in non-static environments with autonomous drones: a Kalman filter reinforcement learning approach *2023 Aerospace Europe Conf.* pp 1–13

[39] Yang S, Baum M and Granström K 2016 Metrics for performance evaluation of elliptic extended object tracking methods *2016 IEEE Int. Conf. on Multisensor Fusion and Integration for Intelligent Systems (MFI)* (IEEE) pp 523–8

Physica Scripta

RECEIVED
1 January 2025

REVISED
1 June 2025

ACCEPTED FOR PUBLICATION
18 June 2025

PUBLISHED
30 June 2025

PAPER

Reinforcement learning based hyperparameter optimization for irregular shape extended object tracking

Ziwen Zhao , Hui Chen* and Wenxu Zhang

School of Electrical Engineering and Information Engineering, Lanzhou University of Technology, Lanzhou 730050, People's Republic of China

* Author to whom any correspondence should be addressed.

E-mail: ziw@lut.edu.cn, chenh@lut.edu.cn and wenxu_zhang@foxmail.com

Keywords: extended object tracking, gaussian process, reinforcement learning, hyperparameter optimization

Abstract

The great leap forward in the development of artificial intelligence technology has brought a new prospect to optimal filtering technology. Extended object tracking (EOT), a well-known research hotspot, is optimized using reinforcement learning (RL) in this paper. Generally, in estimating star-convex extended objects using Gaussian process (GP), the selection of hyperparameters significantly impacts the estimation outcomes. Typically, *a priori* or predetermined set of fixed values constitutes these hyperparameters before the GP of learning commence. However, improper choices in hyperparameter selection may lead to suboptimal performance in practical shape estimation. Addressing this concern, this study introduces a hyperparameter optimization approach for GP-based EOT. This approach incorporates RL via Q-learning (QL) into GP of the extended object model, guiding the agent towards an adaptive hyperparameter strategy through a well-designed reward function. Consequently, this method effectively optimizes the selection of hyperparameters, progressively improving the performance of the multi-feature estimation, such as object trajectory and shape in EOT. Simulation experiments conducted across multiple tracking scenarios demonstrate that the proposed algorithm outperforms traditional GP and random hypersurface algorithms in estimation precision.

1. Introduction

The field of radar object tracking is divided into two primary categories: point object tracking and extended object tracking (EOT). Traditional point object tracking simplifies objects to single points, typically acquiring only one measurement per object and overlooks the shape or contour estimation. However, the advent of modern high-resolution radar sensors has changed this paradigm. Consequently, more than treating objects as singular points is required. The EOT emerges to address the limitation of the point object hypothesis by utilizing and fusing multiple object measurements from sensors, enabling a joint estimation of an object's motion and extension states. This approach has garnered widespread applications in diverse domains, such as autonomous driving [1–3], object detection and recognition [4–6], and classification [7–9]. As the urgent demand escalates for the precise joint estimation of object motion and shape in engineering systems, research into EOT technology has evolved into a focal point and a significant challenge within object tracking.

In recent years, the landscape of EOT has witnessed the emergence of several methodologies. Predominantly utilized approaches encompass the random matrix model (RMM) [10], the random hypersurface model (RHM) [11], and the Gaussian process (GP) [12]. Each method presents distinct characteristics and applicability within EOT frameworks. The RMM approximates an extended object's contour as an elliptical and utilizes a random symmetric positive definite matrix to represent the shape information [10]. Since [10] did not account for the measurement noise covariance in RMM, Feldmann *et al* proposed a modified RMM aimed at improving the applicability of RMM in a more complex and uncertain environment [13]. Additionally, [14] considers the orientation of the ellipse while estimating its size. [15] uses multiple ellipses to estimate the extended object.

[16] employs ellipses with time-varying orientation angles within the RMM framework to represent the extent of the object. However, its adaptability might encounter limitations when dealing with irregular shapes, such as those resembling star-convex contours. Furthermore, the RHM utilizes a Fourier series expansion to depict the object contour, modeling the shape of the extended object through expansion coefficients, rendering it more adept at describing detailed and complex shapes. [17] introduces a universal RHM modeling approach for tracking maneuvering extended objects, addressing the challenge of modeling complex motion models within RHM. In [18], a novel logarithmic likelihood formulation is introduced to address the difficulties caused by the scaling factor of measurement equations in particle weight updates within the RHM particle filter. In [19], a minimum cosine distance-based RHM method is proposed to address the shape estimation divergence caused by the potential negative values of radial functions during object maneuvers. However, representing irregular shapes with limited Fourier series coefficients might yield an inevitably coarse depiction. Within the RHM EOT framework, this might hinder the ability to delineate the shape of extended objects with higher precision.

GP can also be utilized to describe complex irregular shapes. The GP is a formidable tool in modeling random processes, representing distributions over sets of continuous random variables [20]. Its strength lies in adeptly capturing uncertainty and showcasing exceptional performance in flexible data fitting and predictive capabilities. Compared to the RHM, GP models offer increased flexibility, leading to more comprehensive and accurate state estimation and prediction. The inherent adaptability of GP to variations in object shape and dynamics grants them broader applicability across diverse scenarios. This heightened flexibility positions GP as a potent instrument for addressing the complexities inherent in EOT, particularly in accommodating diverse shapes and dynamic changes. Recent research has continuously explored the application of the GP method in various diverse scenarios. For instance, in [21], GP was applied to high-resolution automotive radar object tracking, aiming to accurately estimate the shape of vehicles. In [22], a GP-based method for axis-symmetric EOT is proposed, aiming to simultaneously estimate both the motion state and extension of the object using a GP model. [23] have employed the GP model to track unknown-shaped objects using three-dimensional point cloud data and proposed a GP-based EOT method with input uncertainty in polar coordinates. In [24], a cost-effective GP-probabilistic data association algorithm is introduced to tackle EOT in heavily cluttered environments. Additionally, there is research exploring the applicability of the GP method in other application scenarios [25]. However, parameter selection presents a challenge when constructing GP models, especially concerning hyperparameters like length scale and variance. The choice of hyperparameters significantly impacts the model's performance, underscoring the importance of selecting them appropriately to enhance model effectiveness. Furthermore, hyperparameters govern the model's complexity and enhance its generalization ability. Therefore, selecting suitable hyperparameters is essential for developing high-performance GP learning models. Despite GP's inherent flexibility and robustness, their estimation accuracy remains susceptible, albeit to a certain extent, to the nuances embedded within the chosen hyperparameters. The selection of these intricate parameters necessitates a blend of expertise and prior knowledge, with improper choices potentially culminating in a deterioration of estimation performance. Notably, within the current landscape, there is no universally applicable method for adaptively selecting hyperparameters for the GP model.

The emergence and widespread application of reinforcement learning (RL) have provided a new avenue and method for the adaptive optimization design in GP. This study proposes a Q-learning GP (QL-GP) method to improve EOT estimation performance. As a classical RL algorithm, the QL trains an agent to make decisions in unknown environments [26–31]. Its widespread application stems from its simple implementation, independence from environment models, and capacity to learn optimal strategies across various domains. The algorithm guides behavior by learning the value of state-action pairs and resolves decision problems by continuously updating the Q-value table. The interaction between the agent and the environment enables QL to continually explore and learn, aiming to make decisions that maximize the expected reward for the agent.

Some researchers have employed the RL method for adaptive parameter optimization in object tracking. Particularly in the navigation domain, some relevant research findings have emerged. For instance, [32] addressed the inadequacy of estimation in navigation applications due to uncertainty in noise covariance in extended Kalman filters (EKF), thereby enhancing the accuracy of spacecraft autonomous navigation. It combined the QL algorithm with the EKF (QL-EKF) and proposed a noise covariance adaptive method based on the QL algorithm. [33] introduced an improved QL algorithm called the dynamic grid-based QL EKF (DG-QL-EKF). This algorithm utilizes new ε-greedy and DG strategies for exhaustive search, matching more suitable noise covariance, to overcome the decrease in estimation performance caused by inappropriate heuristics in the QL-EKF method. Moreover, [34] suggests using the QL algorithm for the autonomous navigation of spacecraft to adaptively select appropriate objects by observing the orientations of known objects. This method enables the star tracker to autonomously select objects during the observation cycle, thereby improving the estimation performance of the EKF. [35] proposes an improved QL-EKF (IQ-EKF) method for accurately estimating the motion state of spatial objects in autonomous optical navigation. This approach improves the update rules of QL to enhance learning efficiency and leverages measurement data obtained from star cameras. Combining the

IQ-EKF algorithm with measurement data achieves higher navigation accuracy. Besides autonomous navigation, the QL method shows promising prospects in various domains. [36] proposes the QL Kalman filter (QL-KF) algorithm, which applies QL to adapt the noise process of KF when maneuvering objects that switch motion modes. Although RL methods have been widely applied in object tracking, most existing approaches focus on adaptive selection of filter noise covariance in point object tracking, and they have yet to be extensively applied in EOT. In particular, more research is still needed on utilizing RL for the adaptive selection of hyperparameters in GP models for EOT. The urgency to address this issue is apparent. Solving this problem will not only provide more convenience and flexibility in the selection of hyperparameters for GP models but will also propel further advancements in the field of EOT.

This study proposes a novel method for adaptively selecting hyperparameters in the GP model for star-convex EOT. The primary contributions are as follows:

(1) Combining the QL algorithm in RL with GP-EOT, a QL-GP algorithm that can interact with the environment is constructed, which solves the core problem of adjusting hyperparameters dependent on experience in the traditional GP-EOT model, and realizes the adaptive selection of hyperparameters.

(2) The intersection over union (IoU) measure [37] is adopted as the reward function, providing a clear quantitative objective for hyperparameter optimisation by maximising cumulative rewards, effectively enhancing the model's tracking estimation accuracy.

(3) The hausdorff distance [38] is introduced as an evaluation metric to precisely measure the accuracy of extended object shape estimation from a shape-matching perspective, offering a more rigorous standard for algorithm performance validation.

(4) Comparative experiments in challenging tracking scenarios, such as shape change and manoeuvring motion, demonstrate that the QL-GP algorithm outperforms traditional methods in extended object shape estimation accuracy, validating its effectiveness and superiority.

Overall, the QL-GP algorithm, through the RL mechanism, equips the GP model with the ability to autonomously adapt to dynamic environments, demonstrating advantages in hyperparameter optimisation efficiency, estimation accuracy, and robustness in complex scenarios. This provides a highly practical solution for star-convex EOT, overcoming the limitations of manual hyperparameter tuning and advancing the practical application of the GP in complex object tracking.

2. Background

This section initiates by defining the motion model for extended object, serving as a foundational framework. Subsequently, it introduces and elucidates the fundamental concepts underlying star-convex extended object, along with an associated measurement model.

2.1. Object motion model

In two-dimensional space, the equation describing the motion of an extended object is expressed as follows

$$x_{k+1} = Fx_k + w_k \tag{1}$$

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

where F represents the transition matrix for the object's motion, x_k denotes the object state, w_k signifies the process noise with a mean of 0, and Q_k stands for the covariance matrix. In EOT, augmenting the object's extension state into the object state space model yields the augmented extended object state, given by

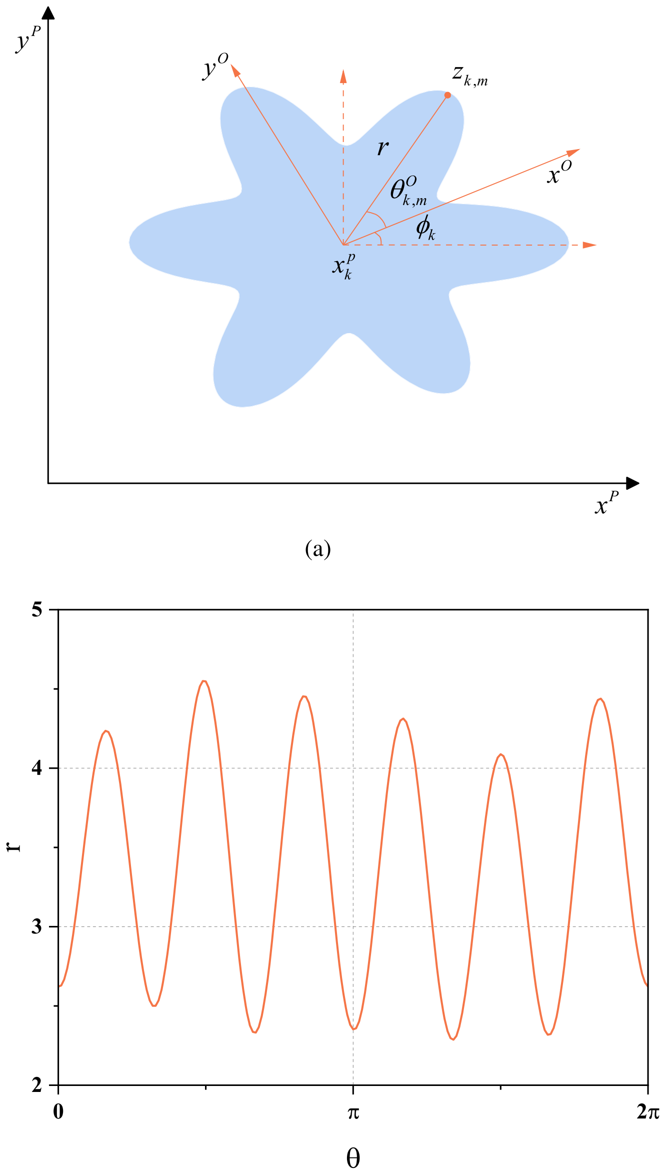
$$x_k = \left[\bar{x}_k^b \quad x_k^e\right]^{\mathrm{T}} \tag{3}$$

where $\bar{x}_k^b$ represents the motion state of the object, encompassing the object's position $[x_k,\, y_k]^{\mathrm{T}}$, velocity $[\dot{x}_k,\, \dot{y}_k]^{\mathrm{T}}$, and the direction ϕ_k, x_k^e denotes the extended state of the object, symbolized as $x_k^e = [f(\theta_{k,m}),\ldots,f(\theta_{k,M})]^{\mathrm{T}}$. Within this representation, $\theta_{k,m}$ signifies the angles of contour points in the local coordinate system aligned with the object direction, while $f(\theta_{k,m})$ represents the corresponding unknown radius function for M contour points. Due to the typically unknown nature of the radius function $f(\theta_{k,m})$, it becomes necessary to employ a GP model to learn and characterize it.

2.2. Object measurement model

In this study, irregularly shaped extended object is represented using a star-convex, exemplified in figure 1(a). When modeling irregularly shaped extended objects, the concept of star-convex emerges as a more versatile tool.

Additionally, the surface measurement model of the star-convex extended object is described as follows: Assuming that the measurement values of the extended object originate from within its region, an appropriate representation of the measurement model emerges as



(a)

(b)

Figure 1. (a) Star-convex object contour (b) Radial function $r = f(\theta)$.

$$z_{k,m} = x_k^p + s_{k,m}p(\theta_{k,m})f(\theta_{k,m}) + e_{k,m} \tag{4}$$

where x_k^p and $\{z_{k,m}\}_{m=1}^{n_k}$ respectively represent the position of the object at time k and the collection of n_k measurements obtained at that time; $\{\theta_{k,m}\}_{m=1}^{n_k}$ signifies the angles corresponding to n_k measurements; $e_{k,m} \sim \mathcal{N}(0,\ R_k)$ represents measurement noise following a Gaussian distribution with a mean of 0 and covariance of R_k; $s_{k,m} \in [0,\ 1]$ denotes the scaling factor. For the surface measurement model of the object, the squared scaling factor follows a uniform distribution, i.e., $s_{k,m}^2 \sim U[0,\ 1]$, the mean and covariance of $s_{k,m}$ specified as [12]

$$c_s = E(s_{k,m}) = \frac{2}{3} \tag{5}$$

$$\sigma_s^2 = E(s_{k,m}^2) - (E(s_{k,m}))^2 = \frac{1}{18} \tag{6}$$

and $p(\theta_{k,m})$ indicates a direction vector

$$p(\theta_{k,m}) \triangleq \begin{bmatrix} \cos(\theta_{k,m}) \\ \sin(\theta_{k,m}) \end{bmatrix} \tag{7}$$

For each measurement $z_{k,m}$, there exists a corresponding angle $\theta_{k,m}^O$. This angle can also be described using the object azimuth state in the local object coordinate system, i.e.

$$\theta^{O}_{k,m}(x^{p}_{k}, \phi_{k}) = \angle(z_{k,m} - x^{p}_{k}) - \phi_{k} \tag{8}$$

The corresponding relationship is illustrated in figure 1.

In this way, $\theta^{O}_{k,m}(x^{p}_{k}, \phi_{k})$ can be employed within (4) to describe the relationship between the state and the measurements.

$$z_{k,m} = x^{p}_{k} + s_{k,m}p(\theta^{P}_{k,m}(x^{p}_{k}))f(\theta^{O}_{k,m}(x^{p}_{k}, \phi_{k})) + e_{k,m} \tag{9}$$

where

$$\theta^{P}_{k,m}(x^{p}_{k}) = \angle(z_{k,m} - x^{p}_{k}) \tag{10}$$

$$p(\theta^{P}_{k,m}(x^{p}_{k})) = \frac{z_{k,m} - x^{p}_{k}}{\|z_{k,m} - x^{p}_{k}\|} \tag{11}$$

Utilizing the GP state-space description [12], the measurement equation can be written as

$$\begin{aligned} z_{k,m} &= x^{p}_{k} + s_{k,m}p(\theta^{P}_{k,m}(x^{p}_{k})) \\ &\quad \times [H^{f}_{k}(\theta^{O}_{k,m}(x^{p}_{k}, \phi_{k}))x^{e}_{k} + e^{f}_{k,m}] + e_{k,m} \\ &= x^{p}_{k} + s_{k,m}p(\theta^{P}_{k,m}(x^{p}_{k}))H^{f}_{k}(\theta^{O}_{k,m}(x^{p}_{k}, \phi_{k}))x^{e}_{k} \\ &\quad + s_{k,m}p(\theta^{P}_{k,m}(x^{p}_{k}))e^{f}_{k,m} + e_{k,m} \\ &= h_{k,m}(x_{k}) + \bar{e}_{k,m} \end{aligned} \tag{12}$$

where

$$h_{k,m}(x_{k}) = x^{p}_{k} + s_{k,m}p(\theta^{P}_{k,m}(x^{p}_{k}))H^{f}_{k}(\theta^{O}_{k,m}(x^{p}_{k}, \phi_{k}))x^{e}_{k} \tag{13}$$

$$\bar{e}_{k,m} = s_{k,m}p(\theta^{P}_{k,m}(x^{p}_{k}))e^{f}_{k,m} + e_{k,m} \tag{14}$$

$$\bar{e}_{k,m} \sim \mathcal{N}(0, R_{k,m}) \tag{15}$$

$$R_{k,m} = p(\theta^{P}_{k,m}(x^{p}_{k}))R^{f}_{k,m}p(\theta^{P}_{k,m}(x^{p}_{k}))^{\mathrm{T}} + R_{k} \tag{16}$$

$$R^{f}_{k,m} = R^{f}(\theta^{O}_{k,m}(x^{p}_{k}, \phi_{k})) \tag{17}$$

It's worth noting that the new measurement noise covariance $R_{k,m}$ encompasses the original measurement noise covariance R_{k}. To address the nonlinearity in the measurement equations, a Cubature Kalman filter (CKF) was utilized for linearization. The specific implementation details can be found in appendix.

3. Gaussian process model

3.1. Gaussian process

In measurement modeling, GP is frequently employed to represent stochastic functions within models, aiming to describe the uncertainty in measurements more precisely. Key components include the mean function and the kernel function. The deterministic nature of these two elements determines the overall structure of the GP. Specifically, the definitions of the mean function and the kernel function are as follows

$$c(\mu) = E[f(\mu)] \tag{18}$$

$$k(\mu, \mu') = E[(f(\mu) - c(\mu))(f(\mu') - c(\mu'))^{\mathrm{T}}] \tag{19}$$

where $c(\mu)$ represents the mean function of the function $f(\mu)$, $k(\mu, \mu')$ signifies the kernel function, and μ represents the input, and the GP is defined as $f(\mu) \sim gp(c(\mu), k(\mu, \mu'))$.

The GP is an extension of multivariate Gaussian distributions. Therefore, for finite-dimensional inputs $\mu_{1}, \ldots, \mu_{N}$, their function values also follow a normal distribution

$$\begin{bmatrix} f(\mu_{1}) \\ \vdots \\ f(\mu_{N}) \end{bmatrix} \sim \mathcal{N}(c, K) \tag{20}$$

where

$$c = \begin{bmatrix} c(\mu_{1}) \\ \vdots \\ c(\mu_{N}) \end{bmatrix}, K = \begin{bmatrix} k(\mu_{1}, \mu_{1}) & \cdots & k(\mu_{1}, \mu_{N}) \\ \vdots & & \vdots \\ k(\mu_{N}, \mu_{1}) & \cdots & k(\mu_{N}, \mu_{N}) \end{bmatrix} \tag{21}$$

3.2. Q-learning Gaussian process

The QL-GP model learns the distribution of a function using known data and utilizes this distribution for prediction. Considering the measurement model

$$z_k = f(\mu_k) + e_k \tag{22}$$

where μ_k represents the training input and $e_k \sim \mathcal{N}(0,\ R)$ denotes measurement noise. Then, using a set of input values and measured values, denoted as $\mu \triangleq [\mu_1,\ldots,\mu_N]^{\mathrm{T}}$ and $z \triangleq [z_1,\ldots,z_N]$ respectively, to learn the function values $f \triangleq [f(\mu_1^f),\ldots,f(\mu_{N^f}^f)]^{\mathrm{T}}$ for other inputs $\mu^f \triangleq [\mu_1^f,\ldots,\mu_N^f]^{\mathrm{T}}$. Similarly, the joint distribution of measurements z and function values f is obtained as

$$\begin{bmatrix} z \\ f \end{bmatrix} \sim \mathcal{N}\left(0, \begin{bmatrix} K(\mu,\ \mu)|_{s,a} + I_N \otimes\ R & K(\mu,\ \mu^f)|_{s,a} \\ K(\mu^f,\ \mu)|_{s,a} & K(\mu^f,\ \mu^f)|_{s,a} \end{bmatrix}\right) \tag{23}$$

where

$$K(\mu,\ \mu^f)|_{s,a} = \begin{bmatrix} k(\mu_1,\ \mu_1^f)|_{s,a} & \cdots & k(\mu_1,\ \mu_N^f)|_{s,a} \\ \vdots & & \vdots \\ k(\mu_N,\ \mu_1^f)|_{s,a} & \cdots & k(\mu_N,\ \mu_N^f)|_{s,a} \end{bmatrix} \tag{24}$$

and s and a respectively denote the state and action of QL; $k(\cdot,\cdot)|_{s,a}$ represents the covariance function corresponding to different QL state-action pairs, where each state-action pair corresponds to different hyperparameters, $K(\cdot,\cdot)|_{s,a}$ represents the matrix constructed from $k(\cdot,\cdot)|_{s,a}$.

For the joint Gaussian distribution in (23), its conditional distribution is given by

$$p(f|\ z) = \mathcal{N}(Az,\ \textstyle\sum) \tag{25}$$

where

$$A = K(\mu^f,\ \mu)|_{s,a} K_y^{-1} \tag{26}$$

$$\sum = K(\mu^f,\ \mu^f)|_{s,a} - K(\mu^f,\ \mu)|_{s,a} K_y^{-1} K(\mu,\ \mu^f)|_{s,a} \tag{27}$$

$$K_y = K(\mu,\ \mu)|_{s,a} + I_N \otimes R \tag{28}$$

In the EOT, a challenge arises when measurements may not be available in batches. To address this issue, efficient online inference needs to be implemented recursively. Therefore, by continuously applying Bayes' theorem, one can obtain

$$\begin{aligned} p(f|\ z_{1:N}) &\propto p(z_N \mid f,\ z_{1:N-1}) p(f|\ z_{1:N-1}) \\ &\propto \underbrace{p(z_k \mid f,\ z_{1:k-1}) p(f|\ z_{1:k-1}) \cdots p(f)}_{\propto p(f|z_{1:k})} \end{aligned} \tag{29}$$

This leads to the following recursion

$$p(f|\ z_{1:k}) \propto p(z_k \mid f,\ z_{1:k-1}) \times p(f|\ z_{1:k-1}) \tag{30}$$

Assuming f is independent of all past measurements, then

$$p(z_k \mid f,\ z_{1:k-1}) \approx p(z_k \mid f) \tag{31}$$

Combining (23), the joint distribution of the measurement and function f is given by

$$\begin{bmatrix} z_k \\ f \end{bmatrix} \sim \mathcal{N}\left(0, \begin{bmatrix} k(\mu_k,\ \mu_k)|_{\tilde{s},\tilde{a}} + R & K(\mu_k,\ \mu^f)|_{\tilde{s},\tilde{a}} \\ K(\mu^f,\ \mu_k)|_{\tilde{s},\tilde{a}} & K(\mu^f,\ \mu^f)|_{\tilde{s},\tilde{a}} \end{bmatrix}\right) \tag{32}$$

where $\tilde{s} = s_k$ and $\tilde{a} = a_k$. Similar to (25)-(28), the likelihood and initial prior can be expressed as

$$p(z_k \mid f) = \mathcal{N}(z_k;\ H_k^f f,\ R_k^f) \tag{33}$$

$$p(f) = \mathcal{N}(0,\ P_0^f) \tag{34}$$

where

$$H^f(\mu^k) = K(\mu_k,\ \mu^f)|_{\tilde{s},\tilde{a}} [K(\mu^f,\ \mu^f)|_{\tilde{s},\tilde{a}}]^{-1} \tag{35}$$

$$\begin{aligned} R^f(\mu^k) = {} & k(\mu^k,\ \mu^k)|_{\tilde{s},\tilde{a}} + R - K(\mu_k,\ \mu^f)|_{\tilde{s},\tilde{a}} \\ & \times [K(\mu^f,\ \mu^f)|_{\tilde{s},\tilde{a}}]^{-1} K(\mu^f,\ \mu_k)|_{\tilde{s},\tilde{a}} \end{aligned} \tag{36}$$

$$P_0^f = K(\mu^f,\ \mu^f)|_{s=s_0,a=a_0} \tag{37}$$

Utilizing the likelihood function, recursive regression can be computed through Kalman filtering

$$x_{k+1}^{f} = x_{k}^{f} \tag{38}$$

$$z_k = H^f(\mu^k)x_k^f + e_k^f \tag{39}$$

$$x_0^f \sim \mathcal{N}(0,\, P_0^f) \tag{40}$$

where the state is $x_k^f = f = [f(\mu_1^f),\ldots,f(\mu_{N^f}^f)]^{\mathrm{T}}$, $e_k^f \sim \mathcal{N}(0,\, R^f(\mu^k))$.

4. Q-learning gaussian process extended object model

4.1. Mean function and covariance function

When using the QL-GP model to describe the shape of an extended object, the input μ represents the polar angle θ, while the output $f(\mu) = r$ corresponds to the radius. This relationship is illustrated in figure 1(b).

Then, the unknown radius of the extended object can be represented using the mean function of the GP, as follows

$$f(\theta) \sim gp(r,\, k(\theta,\, \theta')|_{s,a}) \tag{41}$$

where $r \sim \mathcal{N}(0,\, \sigma_{r,s,a}^2)$, integrating over r yields $f(\theta) \sim gp(0,\, k(\theta,\, \theta')|_{s,a} + \sigma_{r,s,a}^2)$, whose covariance contains a constant term $\sigma_{r,s,a}^2$, indicating the global effect of the randomness of the mean radius on all angles. Notably, this term is independent of the measurement noise, which has been modeled in the measurement equation (4) by $e_{k,m} \sim \mathcal{N}(0,\, R_k)$. The kernel function containing the mean function $k_t(\theta,\, \theta')|_{s,a}$ is a critical component of the GP, determining the correlation between two points. The most commonly used kernel function is the squared exponential kernel [20]

$$k_t(\theta,\, \theta')|_{s=\{\sigma_{f,a}, l_a, \sigma_{r,a}\},a} = \sigma_{f,a}^2 e^{-\frac{2\sin^2\left(\frac{|\theta-\theta'|}{2}\right)}{l_a^2}} + \sigma_{r,a}^2 \tag{42}$$

where $\sigma_{f,a}$, l_a, and $\sigma_{r,a}$ are referred to as the hyperparameters of the QL-GP under different action.

4.2. Adaptive selection of hyperparameters

Experimental comparisons have revealed that selecting hyperparameters in GP significantly impacts the estimation of extended object shape. More than appropriate hyperparameter choices can lead to decreased performance in shape estimation, even resulting in inaccurate estimation. Typically, empirical values are used to set these hyperparameters, determining the optimal values by comparing the model's estimation errors under different parameter settings. Studies across various experimental scenarios have shown that the optimal selection of hyperparameters varies [12, 22–24, 39, 40]. However, this empirical selection method undoubtedly increases the workload, necessitating numerous experiments for comparison. Encouragingly, the rise of the RL in recent years has offered a promising new approach by enabling autonomous exploration and learning. This avenue holds promise for addressing this issue.

The QL algorithm is employed within the QL-GP method to select hyperparameters autonomously. The QL, a RL algorithm, is primarily utilized to train agents to make decisions in unknown environments. To date, it remains widely employed across various research domains. The core of the QL algorithm lies in learning a method called the Q-value function, which assesses the desirability of taking a specific action in a given state. In QL, an agent attempts actions in the environment and receives feedback as a reward or a penalty. Subsequently, the agent updates the Q-values based on this feedback. These Q-values are stored in a table called the Q-table. Through iterative trial and error, the agent continually refines and updates the Q-table, aiming to converge its values as closely as possible to the actual Q-values, thereby delineating the optimal strategy.

To specify the RL framework of the hyperparameter optimization problem, the problem is modeled as a Markov decision process (MDP) defined as a quintuple (S, A, P, R, γ), where:

- state space S: the state space consists of discrete hyperparameter configurations organized as a 6 × 6 grid (shown in figure 2).
- action space A: represents the transfer of hyperparameters from the current state to neighboring states in the grid (shown in figure 2).
- state transition probability P: state transfers are deterministic, defined by the grid structure, and interact with the environment through a reward function.
- reward function R: reward is based on IoU, and the larger the reward, the higher the accuracy of the shape estimation.

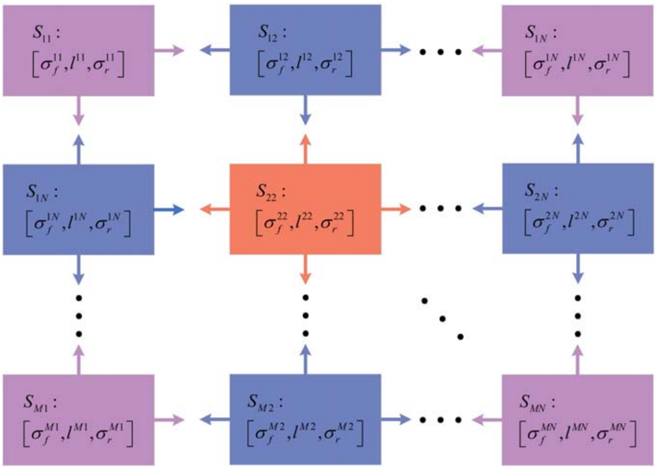

S_{11} :
$[\sigma_f^{11}, l^{11}, \sigma_r^{11}]$
S_{12} :
$[\sigma_f^{12}, l^{12}, \sigma_r^{12}]$
S_{1N} :
$[\sigma_f^{1N}, l^{1N}, \sigma_r^{1N}]$
S_{1N} :
$[\sigma_f^{1N}, l^{1N}, \sigma_r^{1N}]$
S_{22} :
$[\sigma_f^{22}, l^{22}, \sigma_r^{22}]$
S_{2N} :
$[\sigma_f^{2N}, l^{2N}, \sigma_r^{2N}]$
S_{M1} :
$[\sigma_f^{M1}, l^{M1}, \sigma_r^{M1}]$
S_{M2} :
$[\sigma_f^{M2}, l^{M2}, \sigma_r^{M2}]$
S_{MN} :
$[\sigma_f^{MN}, l^{MN}, \sigma_r^{MN}]$

Figure 2. State-action correspondence.

- discount factor γ: it is used to balance immediate rewards with future rewards and to encourage long-term optimal hyperparameter selection.

Based on this MDP framework, the equation governing the update of Q-values is expressed as follows

$$Q(s, a) \leftarrow Q(s, a) + \alpha\left[r_e + \gamma \max_{a'} Q(s', a') - Q(s, a)\right] \tag{43}$$

where $Q(s, a)$ denotes the Q-value for the current state-action pair; s' and a' represent the next state and action respectively, $Q(s', a')$ is the corresponding Q-value; α signifies the learning rate, γ represents the discount factor, and r_e denotes the obtained reward. The goal of QL is to maximize cumulative rewards by selecting hyperparameter configurations that optimize object shape estimation. Deterministic state transfer simplifies the exploration process, but ensures the environmental relevance of learning through the dynamic interaction of IoU rewards with the tracking environment.

According to (25)-(28), using GP to model the Q-value function can obtain the mean and covariance as follows

$$\hat{Q}(s, a) = K(\mu^f, \mu_k)|_{s,a} K_y^{-1} z \tag{44}$$

$$\begin{aligned} \operatorname{cov}(s, a) = {} & k(\mu_k, \mu_k)|_{s,a} + R \\ & - K(\mu_k, \mu^f)|_{s,a} K_y^{-1} K(\mu^f, \mu_k)|_{s,a} \end{aligned} \tag{45}$$

Then, using the estimated $\hat{Q}(s, a)$ instead of $Q(s, a)$, can obtain

$$\hat{Q}(s, a) \leftarrow \hat{Q}(s, a) + \alpha\left[r_e + \gamma \max_{a'} \hat{Q}(s', a') - \hat{Q}(s, a)\right] \tag{46}$$

where (44) uses the GP model to predict the initial estimate of the Q-value of the current state-action pair as the starting point for iterative updating; the core learning mechanism still follows the standard QL, and gradually optimizes the Q-table through the updating rule in (46) until it converges to the optimal policy.

To forge a link between the QL algorithm and the adaptive selection of GP hyperparameters, delineating sets of states and actions becomes imperative, representing diverse states and feasible actions. Visualizing all states in a tabular format, wherein each cell embodies a distinct state, becomes plausible. Each state encapsulates diverse values of GP hyperparameters, correlated with executable actions. This tabulated array of state-action pairs lays the foundation for the QL algorithm. Denoting $S = [S_{11},\ldots,S_{MN}]$ as the set of states, $S_{MN} = [\sigma_f^{MN}, l^{MN}, \sigma_r^{MN}]$ as the hyperparameters contained in each state, and $A = [up, down, left, right]$ as the set of actions. The relationship between each state and action is illustrated in figure 2, with arrows indicating the direction of action transitions for each state.

As depicted in figure 2, the selection of a state corresponds to acquiring specific values for the hyperparameters. Depending on various actions, the current state progresses to the subsequent state, thereby generating new hyperparameter values. Each transition from one state to another is coupled with a reward, assessing the action's quality. Through the QL algorithm's iterative learning and exploratory mechanism, as the Q-table converges towards the optimum, it identifies the state-action pair exhibiting the maximum reward. The

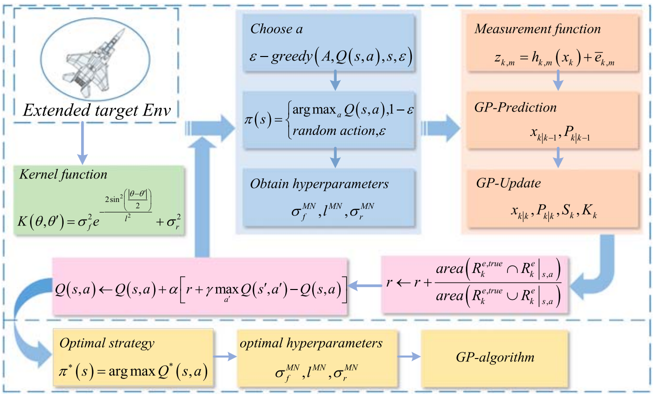

Extended target Env
Kernel function
$K(\theta,\theta')=\sigma_f^2 e^{-\frac{2\sin^2\left(\frac{|\theta-\theta'|}{2}\right)}{l^2}}+\sigma_r^2$
Choose a
$\varepsilon-greedy(A,Q(s,a),s,\varepsilon)$
$\pi(s)=\begin{cases}\arg\max_a Q(s,a),1-\varepsilon\\ random\ action,\varepsilon\end{cases}$
Obtain hyperparameters
$\sigma_f^{MN},l^{MN},\sigma_r^{MN}$
Measurement function
$z_{k,m}=h_{k,m}(x_k)+\bar{e}_{k,m}$
GP-Prediction
$x_{k|k-1},P_{k|k-1}$
GP-Update
$x_{k|k},P_{k|k},S_k,K_k$
$Q(s,a)\leftarrow Q(s,a)+\alpha\left[r+\gamma\max_{a'}Q(s',a')-Q(s,a)\right]$
$r\leftarrow r+\frac{area\left(R_k^{e,true}\cap R_k^e|_{s,a}\right)}{area\left(R_k^{e,true}\cup R_k^e|_{s,a}\right)}$
Optimal strategy
$\pi^*(s)=\arg\max Q^*(s,a)$
optimal hyperparameters
$\sigma_f^{MN},l^{MN},\sigma_r^{MN}$
GP-algorithm

Figure 3. Algorithm structure diagram.

corresponding action signifies the prevailing optimal set of GP hyperparameters, ultimately amplifying the performance of GP models in accurately estimating extended object shapes.

The crux of the intelligent decision-making process lies in achieving maximum cumulative rewards, warranting the definition of a pertinent reward function for effective feedback. In this investigation, which is focused on extended object shape estimation, an appropriate choice entails employing the IoU measure as a reward function. The IoU measure, a prevalent evaluation metric in computer vision, primarily assesses object similarities and is defined as

$$IoU(R_k^{e,true},\ R_k^e|_{s,a}) = \frac{area(R_k^{e,true} \cap R_k^e|_{s,a})}{area\left(R_k^{e,true} \cup R_k^e|_{s,a}\right)} \tag{47}$$

where $R_k^{e,true}$ represents the coverage area of the EOT and $R_k^e|_{s,a}$ is the estimated shape coverage area at different state-action pairs. Typically, a higher IoU value signifies greater object resemblance, while a lower value indicates the opposite. Thus, leveraging the IoU measure as a feedback mechanism, a larger cumulative reward denotes superior shape estimation performance, signifying the corresponding set of hyperparameter values as the optimal choice. The structure of the GP extended object model's adaptive algorithm based on the QL algorithm is depicted in figure 3. The pseudo-code of the proposed algorithm is shown in algorithm 1.

Algorithm 1. QL-GP

```
Input: state grid $S = \{S_{11},\ldots,S_{MN}\}$, action set $A = \{up,\ down,\ left,\ right\}$, learning rate $\alpha$, discount factor $\gamma$, exploration rate $\varepsilon$
Output: optimized Q-table $Q(s,\ a)$, adaptive hyperparameter strategy $\pi^*(s)$
Initialization parameters: $Q(s,\ a) = 0$, for all $s \in S, a \in A$
for each iteration $t = 1$ to $T$do
    Initialize object state $x_0$, GP model with default hyperparameters $s_0$
    for each time step $k = 1$ to $N$do
        Obtaining measurement $Z_k$ from sensors using (4)
        Choose action $a_k$ via $\varepsilon$-greedy
        Estimate object state $\tilde{x}_{k|k}$ using CKF (appendix)
        Calculate reward $r_k$ using (47)
        Observe current state $s_k$
        Execute $a_k \rightarrow$ transition to $s_{k+1}$
        Update Q-table using (46)
        Update kernel function using (42)
        $s_k = s_{k+1}$
    end for
end for
return $Q(s,\ a)$, $\pi^*(s) = \arg\max Q(s,\ a)$
```

Although the proposed hyperparameter optimisation problem shares similarities with structured multi-armed bandits in terms of deterministic state transitions and grid structures, it exhibits distinct advantages in the following aspects: Firstly, by introducing sequential actions such as up and down within the state grid (as shown in figure 2), the agent can leverage the local coherence of adjacent configurations for efficient exploration. Secondly, the dynamic reward function based on IoU (equation (47)) closely reflects the time-varying characteristics of object shapes and motion models, and, combined with QL's iterative update strategy, enables the capture and optimisation of long-term cumulative rewards in sequential decision-making. Lastly, the inherent scalability of the MDP framework not only supports the incorporation of stochastic state transitions and high-dimensional state representations but also provides flexibility for integrating additional environmental uncertainties into a unified optimisation process in future work.

To balance algorithm accuracy and operational efficiency, the proposed method divides the hyperparameter optimisation process into two phases: offline training and online execution. During the offline training phase, comprehensive interaction with diverse simulated scenarios (encompassing various object shapes and motion models) enables the learning of a near-optimal hyperparameter selection strategy. In the online execution phase, this strategy is directly applied without further environmental interaction, requiring only a table lookup or lightweight forward computation to obtain the hyperparameter configuration for the current state, thus meeting real-time requirements. This training-execution architecture relies on a reasonable but necessary assumption: that the online environment exhibits a degree of consistency with the training environment in terms of object shape variation patterns, motion models, and measurement noise statistical characteristics.

5. Experiment analysis

A series of simulation experiments were conducted to substantiate the efficacy of the proposed GP hyperparameter optimization algorithm. These experiments involved comprehensive comparisons with both the particle swarm optimization of the GP model (PSO-GP), the traditional GP model [12], and RHM [11], where the PSO-GP is a method that utilizes the traditional PSO algorithm [41] to optimize the hyperparameters of the GP model. Varied simulation scenarios were defined to depict and compare the experimental outcomes thoroughly. Given the intrinsic requirement in EOT for concurrent estimation of the object's position and shape, distinctive methodologies were employed to assess these two parameters. To evaluate the estimation accuracy of the object's centroid position, the root mean square error (RMSE) served as the metric, aligning with methodologies commonly utilized in analogous studies

$$RMSE(x^{es}, x^{p}) = \sqrt{\frac{1}{N}\sum_{k=1}^{N} \|x_k^{es} - x_k^{p}\|_2} \tag{48}$$

where x_k^{es} and x_k^{p} respectively denote the estimated and true positions at time k. Additionally, the evaluation of extended object shape estimation is conducted using the hausdorff distance. The hausdorff distance is a metric used to measure the similarity between two sets of points, describing the distance between two point sets. Its definition is as follows

$$H(D, G) = \max(h(D, G), h(G, D)) \tag{49}$$

where $D = [d_1,\ldots,d_i]$ and $G = [g_1,\ldots,g_j]$ represent two sets of points.

$$h(D, G) = \max_{d\in D}\min_{g\in G} \|d - g\|_2 \tag{50}$$

$$h(G, D) = \max_{g\in G}\min_{d\in D} \|g - d\|_2 \tag{51}$$

In the definition of the hausdorff distance, $H(D, G)$ is referred to as the two-way hausdorff distance, while $h(D, G)$ and $h(G, D)$ are termed as one-way hausdorff distance. $H(D, G)$ measures the maximum mismatch between two sets of points.

Different shapes of extended objects were obtained for the simulation experiments by approximating the shapes of some natural objects. Specifically, two typical natural objects were chosen, depicted in figure 4, alongside their corresponding approximated shapes. Figure 5 exhibits the approximated shapes derived from the natural objects with their respective measurements. Following this, validation is carried out across multiple simulated scenarios utilizing these distinct extended shapes. The subsequent analysis focused on assessing the effect of adaptively selecting hyperparameters on EOT performance. This evaluation encompassed an analysis based on various performance evaluation metrics from 100 Monte Carlo (MC) experiments.

5.1. Simulation scenario 1

In scenario 1, two different shapes are considered. Both shapes of the extended objects follow a nearly constant velocity (CV) motion model. The object's motion state transition equation is given by (1), where

$$F = \begin{bmatrix} F^{b} & 0 \\ 0 & F^{e} \end{bmatrix} \tag{52}$$

$$Q_k = \begin{bmatrix} Q_k^{b} & 0 \\ 0 & Q_k^{e} \end{bmatrix} \tag{53}$$

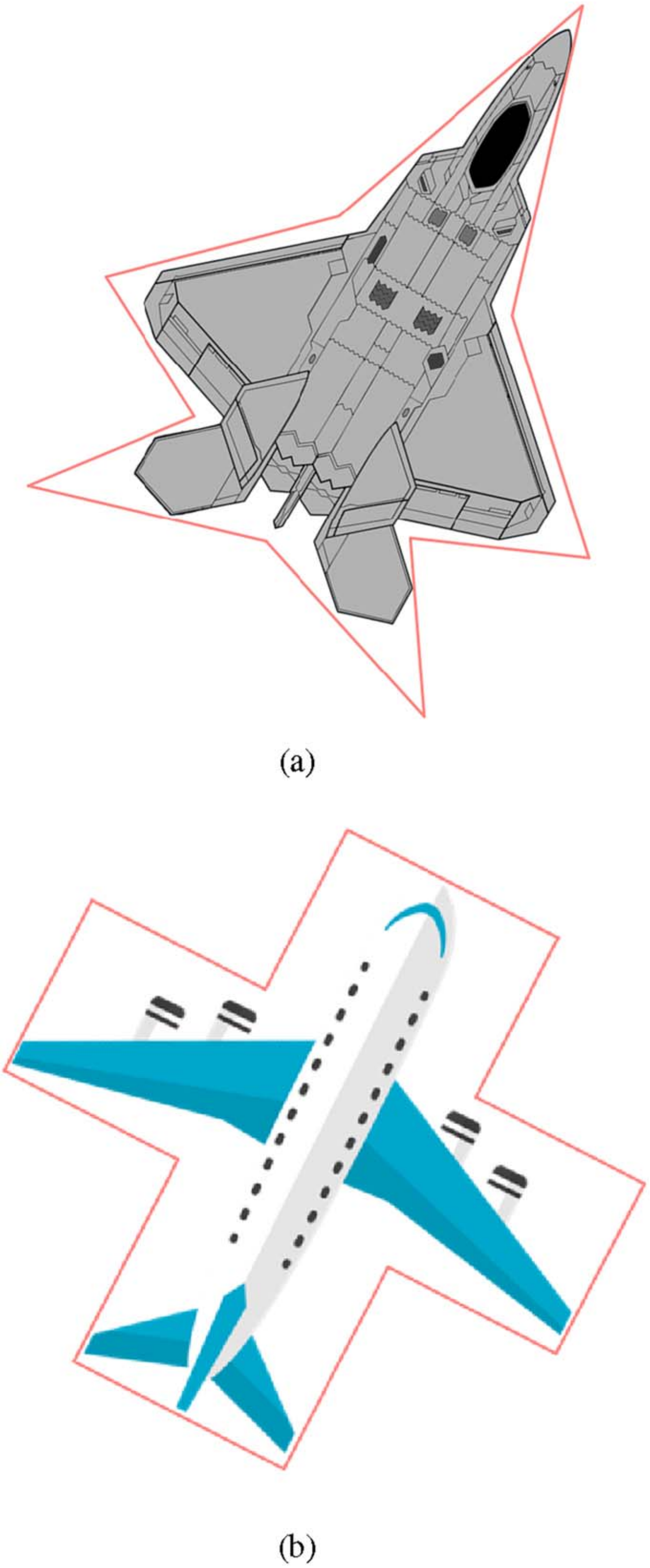
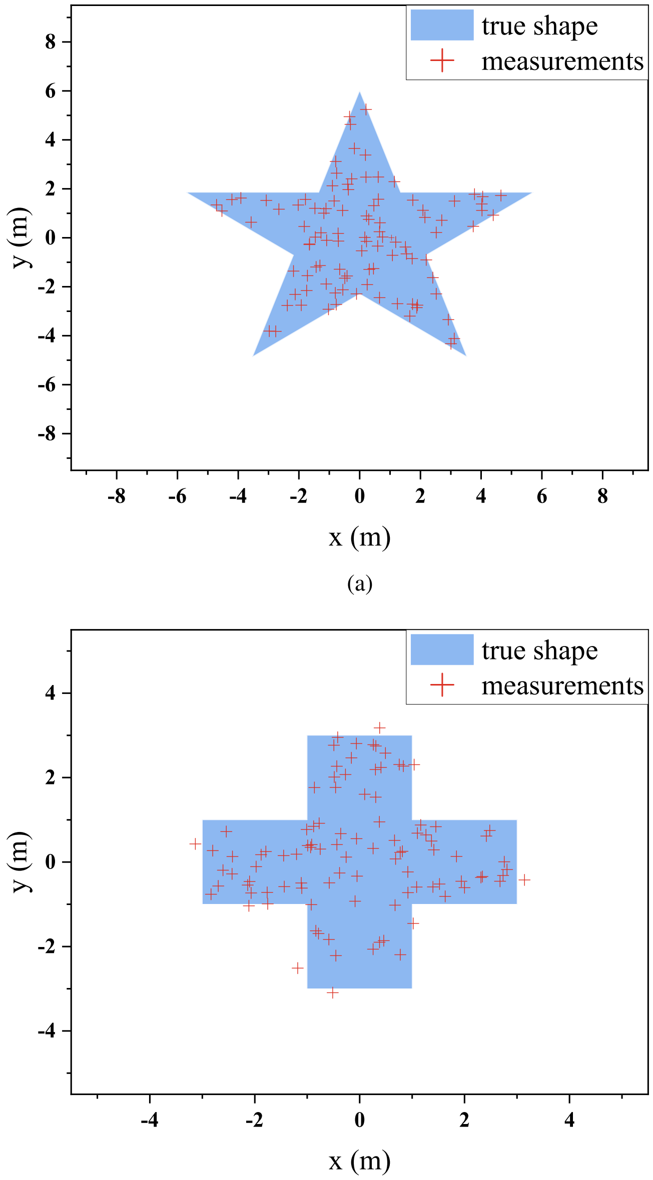


Figure 4. Two typical objects along with their shape approximations: (a) approximated pentagon shape (b) approximated cross shape.

$$F^b = \begin{bmatrix} I_2 & TI_2 \\ 0_2 & I_2 \end{bmatrix},\ F^e = e^{-\tau T} I \tag{54}$$

$$Q^b = \sigma^2 \begin{bmatrix} \frac{T^3}{3} I_2 & \frac{T^3}{2} I_2 \\ \frac{T^3}{2} I_2 & TI_2 \end{bmatrix},\ Q^e = (1 - e^{-2\tau T}) K(\mu^f, \mu^f) \tag{55}$$



(a)

(b)

Figure 5. Two typical objects along with their shape approximations: (a) approximated pentagon shape (b) approximated cross shape.

where $T = 1s$ represents the sampling period, and the experiment spans a total of 50 sampling instances, $I_{2\times 2}$ is denotes the identity matrix; $\tau = 1 \times 10^{-4}$ denotes the forgetting factor [12]; $\sigma = 0.1\text{m s}^{-2}$ signifies the standard deviation of the process noise. The initial state of the object motion is $\bar{x}_k^b = [-200,\ 400,\ -8,\ 8]^{\mathrm{T}}$. The number of measurements generated by each object follows a Poisson distribution with a mean of 20, and the scaling factor $s_{k,m} \sim \mathcal{N}\left(s_{k,m}, \frac{2}{3}, \frac{1}{18}\right)$. The standard GP model has parameters set as $\sigma_f = 0.5$, $l = 2\pi$, $\sigma_r = 0.2$. For the QL-GP model, a table search method is utilized to explore suitable hyperparameters values. Based on the standard GP model's parameter setting, the range for hyperparameters is set as $0.001\sigma_f \leqslant \sigma_f^{MN} \leqslant 10\sigma_f$, $0.001\sigma_r \leqslant \sigma_r^{MN} \leqslant 10\sigma_r$, $0.001l \leqslant l^{MN} \leqslant 10l$. Through experimental evaluation, balancing algorithm complexity and performance, the search table size depicted in figure 2 is configured as 6×6, ensuring an enhanced object estimation performance while reducing the computational burden. In the QL algorithm, $\alpha = 0.1$, $\gamma = 0.9$, $\varepsilon = 0.1$ are designated parameters, where ε is referred to as the exploration coefficient.

Figure 6 delineates specific aspects of the pentagram-shaped EOT under the CV motion model. Figure 6(a) portrays the object's trajectory, encapsulating its initiation and termination points. Figure 6(b) is an enlarged view of partial extended state estimation. The magnified views of the extended object estimation at 35s and 50s, displayed in figures 6(c)–(d), specifically focus on estimating the extended object's shape. While all four methods can outline the contours of the pentagram-shaped extended object, the QL-GP algorithm performs better in capturing finer details than the other methods. Figures 6(c)–(d) illustrate that the shape estimation algorithm

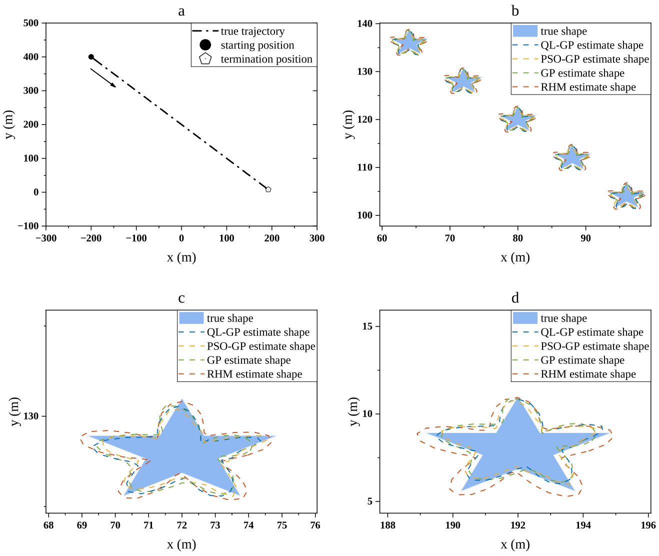


Figure 6. The tracking results of a single MC experiment for the pentagram-shaped object (a) displays the object's motion trajectory, (b) extension estimates, (c) extension estimate at 35s, (d) extension estimate at 50s.

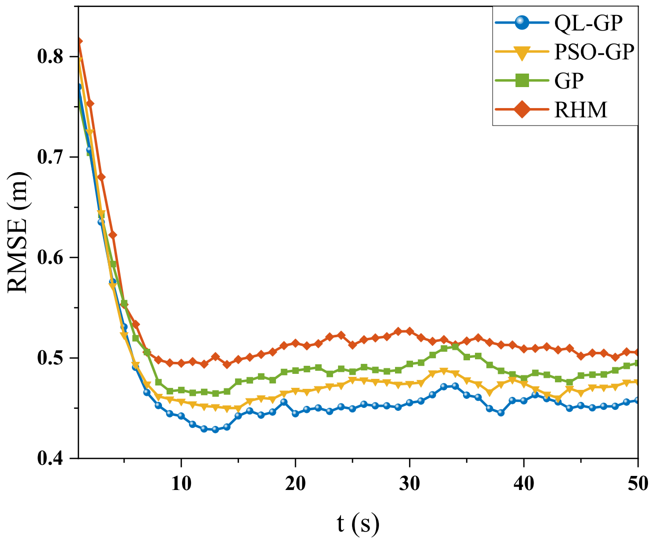


Figure 7. RMSE of centroid estimation.

based on QL-GP progressively exhibits more prominent effectiveness over time and algorithm iterations in estimating the shape of the extended object.

To further scrutinize the distinctions among different algorithms, numerical analyses were utilized to evaluate the centroid and shape estimation performance of the extended objects, as illustrated in figures 7 and 8. These graphical depictions present RMSE and hausdorff distance data. The graphical outcomes notably highlight the superior performance of the QL-GP algorithm in both centroid and shape estimations relative to PSO-GP, GP, and RHM. Specifically, while GP's estimation performance trails behind PSO-GP and QL-GP, RHM demonstrates the least effective performance among these algorithms.

A more complex scenario-transforming a pentagram into a cross shape-is employed to distinguish differences between algorithms better. Estimating shape variations is more challenging than estimating a single shape. The initial state of the object is $\bar{x}_k^b = [0,\ 0,\ 20,\ 8]^{\mathrm{T}}$. Figure 9 illustrates the details of estimating the transformation from a pentagram to a cross shape under the CV motion model. Figure 9(b) displays measurements at the moment of shape transformation, while figures 9(c)–(d) depicts the estimation results of the shape at that moment. It can be observed that, before the shape transformation, the three algorithms based on the GP model performed well in shape estimation. However, after the shape transformation, none of the four algorithms immediately adapted to this change, resulting in suboptimal shape estimation. Importantly, from

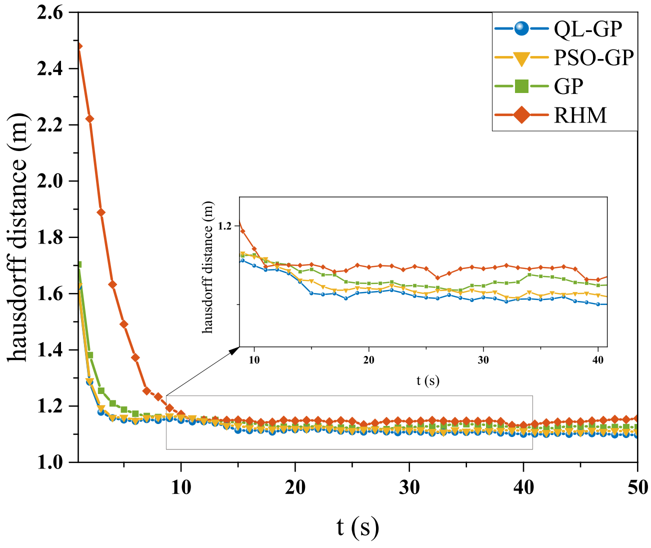


Figure 8. Hausdorff distance for shape estimation.

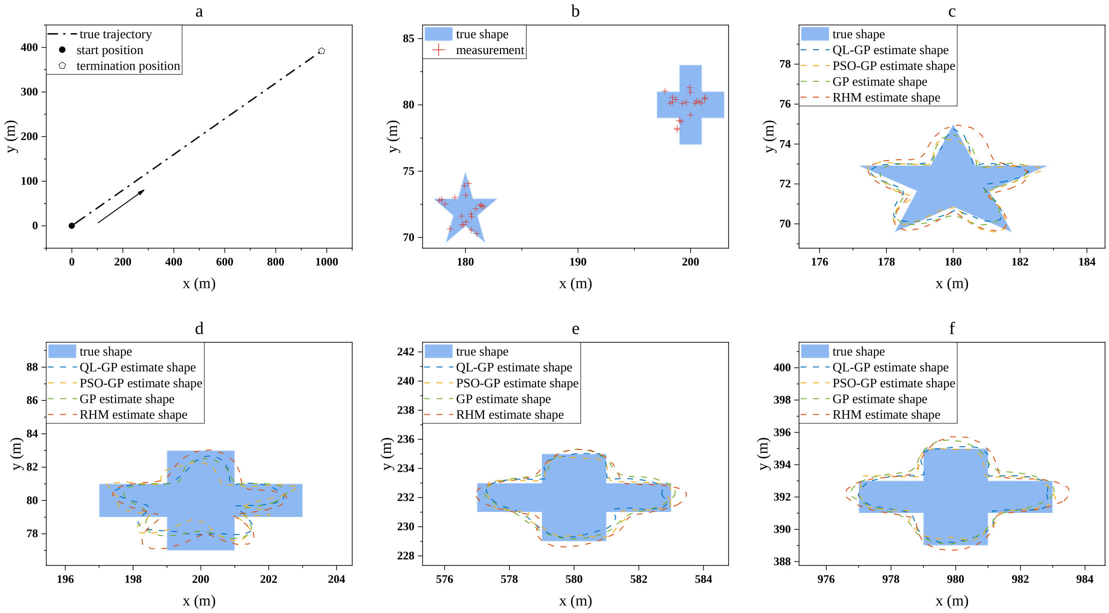


Figure 9. The tracking results of a single MC experiment for the pentagram to cross-shaped object (a) displays the object's motion trajectory, (b) measurements, (c) extension estimate at 10s, (d) extension estimate at 11s, (e) extension estimate at 30s, (f) extension estimate at 50s.

figures 9(e)–(f), it is evident that, with algorithm iterations, the QL-GP method approached the actual shape more closely during the contour estimation process, followed by the PSO-GP, both outperforming the traditional GP and RHM algorithms. This result further demonstrates the effectiveness of the QL-GP algorithm in adapting to scenarios involving changes in the shape of extended object.

Figures 10 and 11 intuitively depict the performance differences among different algorithms using RMSE and hausdorff distance, respectively. It can be observed from the figures that sudden changes in object shape pose higher demands on algorithm performance. At the moment of shape change, both centroid RMSE and hausdorff distance exhibit significant errors. An important observation is that the QL-GP algorithm demonstrates superior performance, with better estimation capabilities than PSO-GP, GP, and RHM algorithms. It is noteworthy that after the adaptive parameter adjustment in the QL algorithm, there is a noticeable performance improvement, emphasizing the crucial role of parameter selection in extended object shape estimation, with QL-based method outperforming PSO-based method.

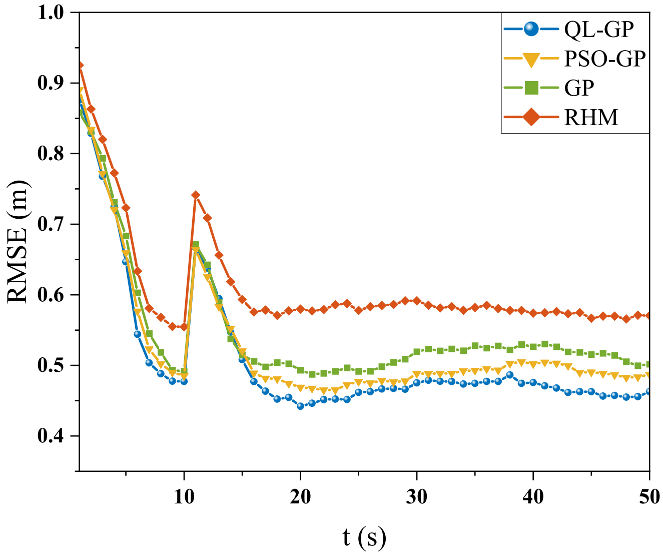


Figure 10. RMSE of centroid estimation.

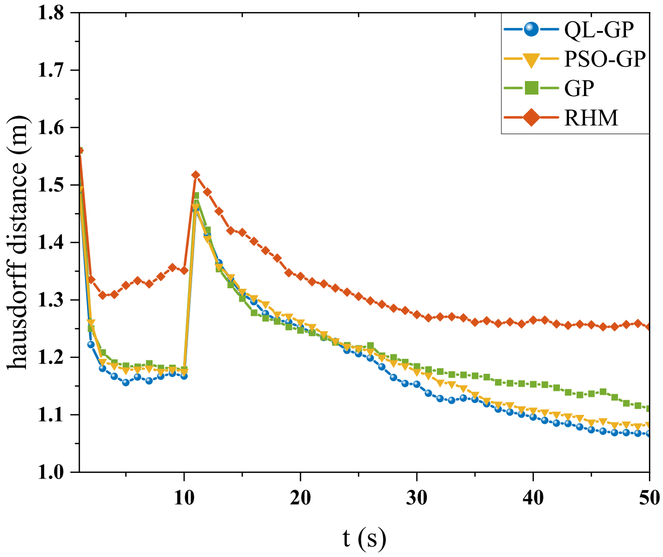


Figure 11. Hausdorff distance for shape estimation.

5.2. Simulation scenario 2

In this scenario, the validation of different algorithms still employs the cross and pentagon shapes. However, these shapes exhibit distinct motion patterns this time, augmenting the complexity and estimation difficulty. The transition matrix has undergone alterations despite maintaining the same motion equation for the object as in Scenario 1. For linear motion, the transition matrix remains consistent with Scenario 1, that is, $F_{CV}^{b} = F^{b}$. In the case of approximate coordinated turn (CT) motion for the pentagram-shaped object's movement, the transition matrix is described as

$$F_{CT}^{b} = \begin{bmatrix} 1 & 0 & \frac{\sin wT}{w} & -\frac{1-\cos wT}{w} \\ 0 & 1 & \frac{1-\cos wT}{w} & \frac{\sin wT}{w} \\ 0 & 0 & \cos wT & -\sin wT \\ 0 & 0 & \sin wT & \cos wT \end{bmatrix} \tag{56}$$

where the object follows a CT mode with a turn rate of $w = \pi/10$ rad/s from 11s to 20s and similarly from 31s to 40s, but with a turn rate of $w = -\pi/10$ rad/s, and in the CV model for the remaining sampling times. The initial

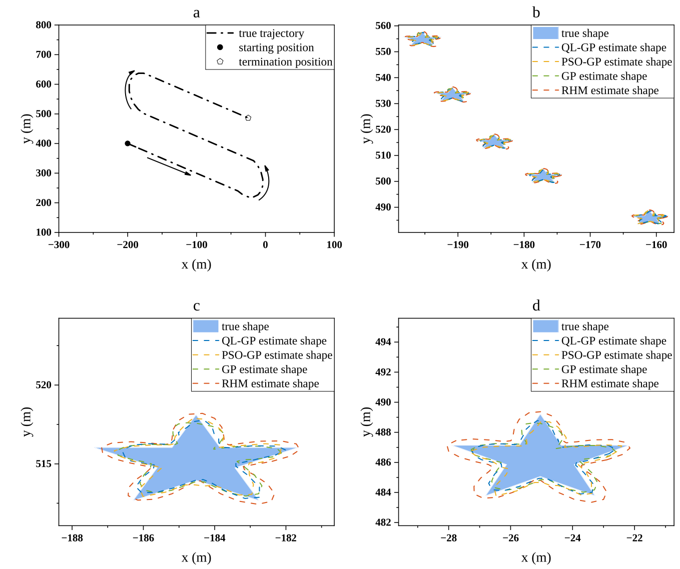


Figure 12. The tracking results of a single MC experiment for the pentagram-shaped object (a) displays the object's motion trajectory, (b) extension estimates, (c) extension estimates at 32s, (d) extension estimates from 50s.

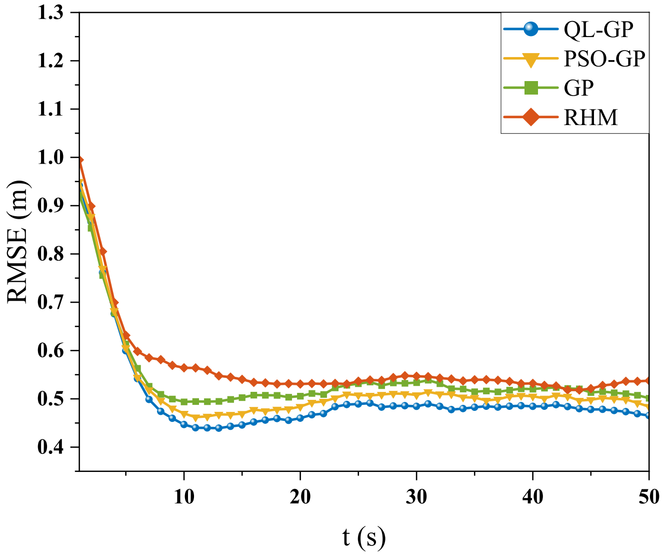


Figure 13. RMSE of centroid estimation.

state of the object motion is $\bar{x}_k^b = [-200,\ 400,\ 16,\ -16]^{\mathrm{T}}$. The settings for other parameters related to the object's motion and the parameters for the QL algorithm remain consistent with Scenario 1.

Figures 12(a)–(b) respectively illustrate the motion trajectory of the object and a locally magnified view during the second turning process of the object. Figures 12(c)–(d) illustrates detailed tracking estimation results, emphasizing the consistently superior performance of the QL-GP algorithm in extended object shape estimation. The figures show that whether the object is turning or moving in a straight line, the QL-GP algorithm provides better estimates of the object shape compared to PSO-GP, GP, and RHM. Additionally, figure 13 presents the RMSE of the center point estimates, indicating that the center point estimates of QL-GP are more accurate compared to PSO-GP, GP, and RHM. Figure 14 illustrates the hausdorff distance of shape estimation, indicating that QL-GP outperforms PSO-GP, GP, and RHM in shape estimation. Although the PSO-GP algorithm also optimizes GP hyperparameters, the optimization effect of QL-GP is more outstanding.

In the subsequent experiment, a different motion scenario is designed for the cross-shaped object. In particular, the object follows a CT mode with a turn rate of $w = \pi/18$ rad/s from 11s to 28s and similarly from 41s to 50s. In addition, as the centroid of the extended object moves with a certain turning rate, the expansion

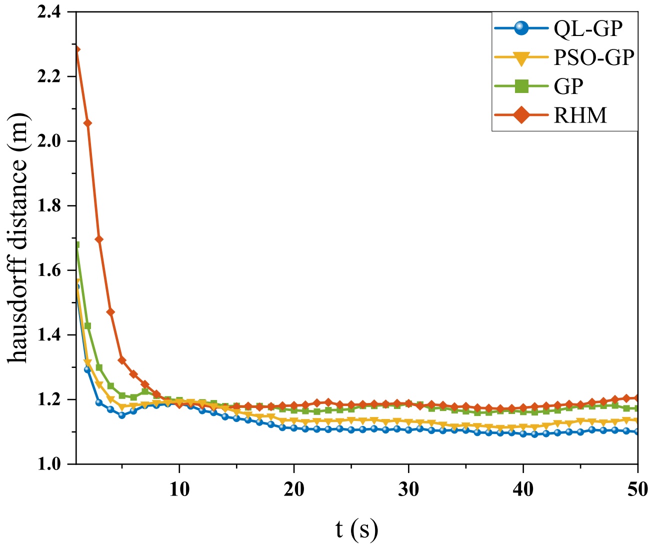


Figure 14. Hausdorff distance for shape estimation.

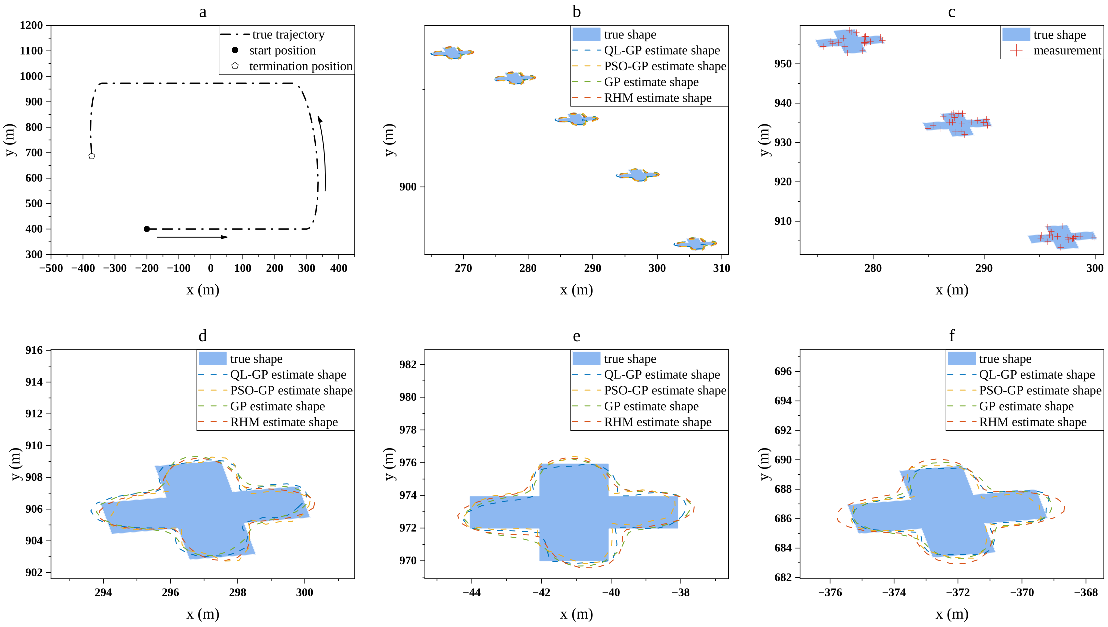


Figure 15. The tracking results of a single MC experiment for the cross-shaped object (a) displays the object's motion trajectory, (b) extension estimates, (c) measurements, (d) extension estimates at 25s, (e) extension estimates at 35s, (f) extension estimates at 50s.

state of the object also undergoes rotation at the same turning object, posing a greater challenge for the estimation of EOT.

Figure 15 illustrates the details of the cross-shaped object tracking estimation. Figures 15(b)–(c) is local enlargements and measurement generation during the object turning process. Figures 15(d)–(f) depicts the effects of different algorithms on shape estimation during both the turning and straight-line motion of the object. It can be observed that the performance of PSO-GP is improved compared to GP, but there is still a gap compared to QL-GP. QL-GP exhibits the best estimation performance, particularly in delineating the details of the extended object shape.

In figures 16 and 17, the RMSE of extended object centroid estimation and the hausdorff distance for shape evaluation are presented. A detailed examination of the figures reveals whether QL-GP consistently demonstrates more minor errors in centroid or shape estimation, which establishes its superior performance compared to PSO-GP, GP, and RHM. Notably, during the initial rotation of the object, both centroid RMSE and hausdorff distance exhibit an increasing trend, suggesting an impact of the object's rotation on the estimation.

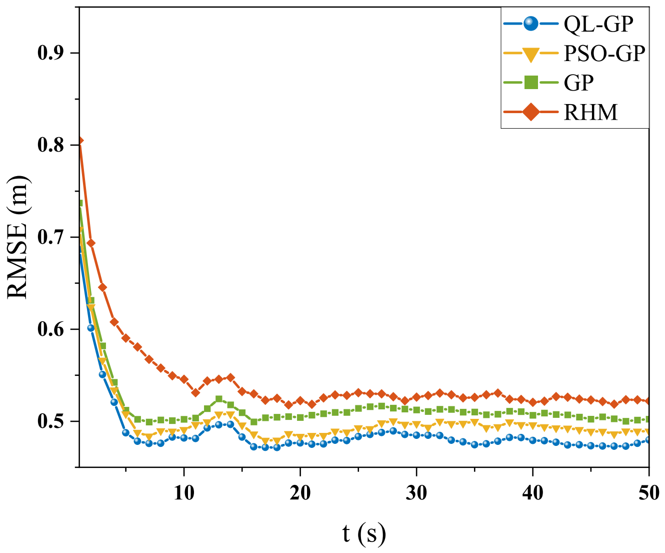


Figure 16. RMSE of centroid estimation.

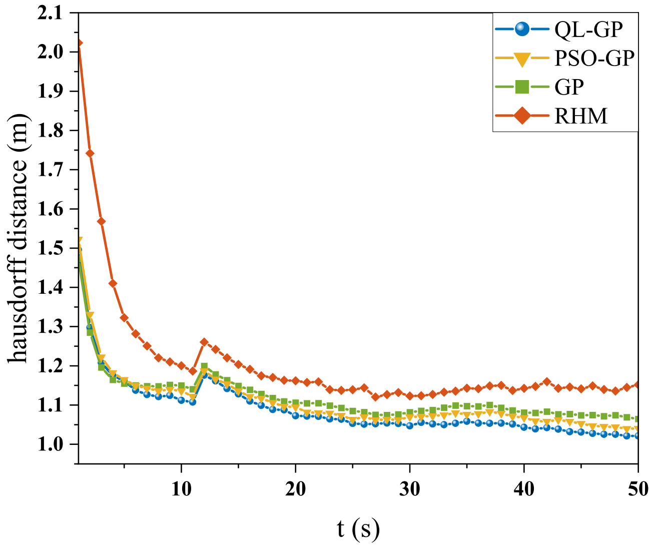


Figure 17. Hausdorff distance for shape estimation.

However, all four algorithms converge swiftly, delivering precise estimates of the object's centroid and extended state. QL-GP, in particular, maintains superior performance, underscoring its robust adaptability across diverse environments.

The heatmap depicted in figure 18 showcases the action-state selection heatmap of the QL-GP algorithm under the scenario of an extended object following a CV model within a pentagon-shaped trajectory. Each heatmap component denotes the reward value associated with selecting a particular action within a specific state, with distinct reward values represented by varying colors. Shades of red correspond to higher reward values, while shades of purple indicate lower reward values. Analyzing the heatmap makes it apparent which states and actions result in the highest rewards, thereby identifying the optimal set of GP hyperparameters for achieving superior performance in EOT.

Subsequently, two experiments are conducted using the CV motion model: (1) an ablation study was performed against a background of the cross to assess the impact of different hyperparameters on the estimation of extended object shapes; (2) an investigation into the effect of different measurement quantities on the shape estimation of QL-GP, PSO-GP and GP models is conducted against a background of pentagrams. First, the impact of hyperparameters on shape estimation was evaluated, as shown in figure 19.

Figure 19 illustrates the consequences of fixing one hyperparameter while adapting the other two using the QL-GP algorithm, offering insights into the individual effects of hyperparameters on shape estimation. In the

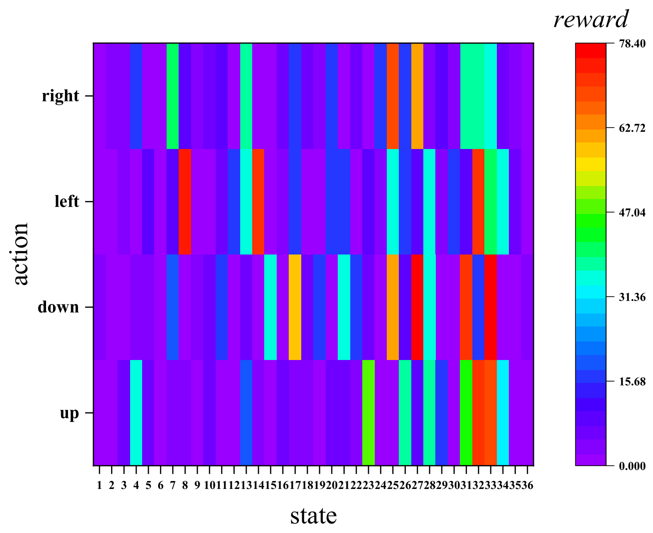


Figure 18. QL-GP action-state selection heatmap.

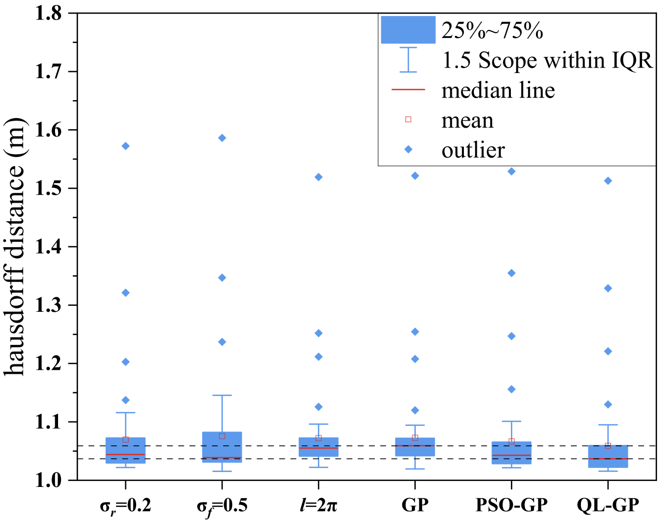


Figure 19. Influence of various hyperparameters on shape estimation.

figure, the two dashed lines, top to bottom, represent the numerical values of means and medians of QL-GP algorithm. It's evident from the graph that fixing $\sigma_r = 0.2$ or fixing $\sigma_f = 0.5$ while adapting the other parameter using the QL algorithm significantly affects the shape estimation. Compared to the standard GP model, performance appears to be superior when $\sigma_r = 0.2$, indicated by the substantial decrease in both median and mean values. However, while $\sigma_f = 0.5$ shows the smallest median, its mean value for hausdorff distance is the largest. Fixing $l = 2\pi$ and adapting σ_r and σ_f using the QL algorithm barely affects the results of shape estimation, closely resembling the standard GP model. This underscores the substantial impact of l on extended object shape estimation. However, to achieve the best estimation performance, it remains crucial to simultaneously optimize all three hyperparameters, as seen in both PSO-GP and QL-GP. Notably, QL-GP exhibits superior performance compared to PSO-GP, highlighting the effectiveness of the proposed algorithm.

Finally, a comparative analysis is conducted between the QL-GP, PSO-GP, and GP models to estimate the shape of the extended object with varying measurement numbers. Figure 20 showcases the mean of hausdorff distance in shape estimation for QL-GP, PSO-GP, and GP models under different measurement numbers. Observing the data in the graph reveals that, with fewer measurements, QL-GP, PSO-GP, and GP models exhibit significant errors in estimating the shape of the extended object. However, as the number of measurements increases, the performance of all three models improves. Particularly noteworthy is that the performance of the PSO-GP model consistently outperforms the GP model, displaying smaller hausdorff distances, while QL-GP shows superior estimation, with the smallest mean hausdorff distance. This highlights the advantage of the

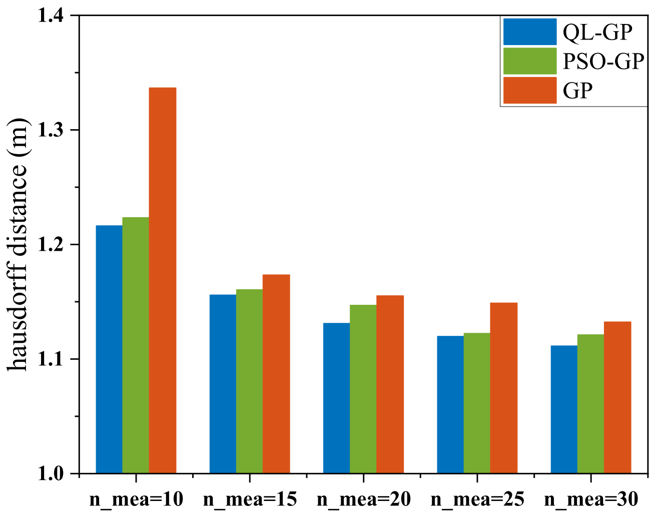


Figure 20. The mean hausdorff distance with varying numbers of measurements.

Table 1. Performance of different algorithms in different scenarios.

Scenario	Extended shape	algorithm	RMSE(m)	hausdorff distance (m)
1	pentagram	QL-GP	**0.4714**	**1.1311**
		PSO-GP	0.4868	1.1468
		GP	0.5072	1.1551
		RHM	0.5276	1.2354
	pentagram to cross	QL-GP	**0.5113**	**1.178**
		PSO-GP	0.5255	1.1965
		GP	0.5502	1.2064
		RHM	0.6112	1.3152
2	pentagram	QL-GP	**0.5038**	**1.1349**
		PSO-GP	0.5241	1.1586
		GP	0.5412	1.1972
		RHM	0.5675	1.2456
	cross	QL-GP	**0.4877**	**1.0909**
		PSO-GP	0.5012	1.1099
		GP	0.5167	1.1191
		RHM	0.5433	1.2069

proposed QL-GP algorithm, which enhances the capability of estimating the shape of extended objects by adaptively optimizing GP model hyperparameters through RL methods.

Table 1 presents the average RMSE and hausdorff distance for the four algorithms across different tracking scenarios. The table shows that QL-GP and PSO-GP, optimized for hyperparameters, show better estimation performance compared to GP and RHM. Moreover, QL-GP exhibits superior optimization performance compared to PSO-GP, further confirming the effectiveness of the proposed algorithm.

Table 2 summarises the single execution time for each algorithm in tracking a cross-shaped extended object in Scenario 2, conducted on a unified experimental platform (3.9 GHz Intel Core i3 7100, Python 3.11.5). The results reveal that, although RHM and traditional GP methods incur lower computational costs, their lack of adaptability to complex scenarios (e.g., dynamic object deformation or unknown parameters) significantly constrains their tracking and estimation accuracy in such environment. While QL-GP exhibits a higher runtime than RHM and traditional GP, it achieves hyperparameter optimisation with only a modest additional computational cost, striking an effective balance between tracking accuracy and computational efficiency. Furthermore, its average computational load is lower than that of PSO-GP, which relies on global search.

Table 2. Compute load comparison.

Algorithms	Single execution time (s)
QL-GP	0.2562
PSO-GP	0.2794
GP	0.1827
RHM	0.0948

Consequently, QL-GP ensures performance improvement while maintaining real-time capability and resource efficiency, offering a highly practical solution for EOT in engineering applications.

6. Conclusion

The proposed QL-GP algorithm is designed to bolster the GP model's proficiency in estimating the contour of extended object by dynamically optimizing its hyperparameters. This novel approach seamlessly incorporates QL into the GP framework, empowering it to learn the requisite hyperparameters autonomously, circumventing the conventional reliance on predefined empirical values. Embracing a RL paradigm, QL-GP adapts its hyperparameters through environmental interactions, surpassing the performance of the conventional GP model. Simulation results affirm the remarkable capabilities of the QL-GP algorithm in accurately estimating both the centroid and contour of extended object. It should be noted that the training-execution separation strategy relies on the statistical consistency between the online environment and the training environment. When real-world scenarios deviate from the training data distribution, such as encountering new object deformation patterns or significant changes in measurement noise, the fixed policy may struggle to adapt, leading to reduced estimation accuracy (e.g., increased hausdorff distance and RMSE). To enhance the method's adaptability in unknown environments, future work could incorporate lightweight online incremental adjustment mechanisms, enabling the policy to fine-tune Q-values during execution based on environmental feedback. Alternatively, a periodic policy retraining framework could be employed, using data collected online to update the offline policy, thereby maintaining long-term model stability. Furthermore, incorporating domain randomisation strategies during the training phase, by expanding the diversity of object and sensor conditions in simulated environments, could improve the policy's generalisation and robustness when facing novel scenarios.

Acknowledgments

This work is supported by the National Natural Science Foundation of China (62163023, 61873116, 62366031, 62363023), the Gansu Provincial Basic Research Innovation Group of China (25JRRA058), the Central Government's Funds for Guiding Local Science and Technology Development of China (25ZYJA040), the Gansu Provincial Key Talent Project of China (2024RCXM86) and the Gansu Provincial Special Fund for Military-Civilian Integration Development of China.

Data availability statement

The data cannot be made publicly available upon publication because no suitable repository exists for hosting data in this field of study. The data that support the findings of this study are available upon reasonable request from the authors.

Appendix

For the state-space models (1) and (12), prediction and update are performed using the CKF algorithm.

(1) predict step

$$\tilde{x}_{k|k-1} = F\tilde{x}_{k-1|k-1} \tag{57}$$

$$P_{k|k-1} = FP_{k-1|k-1}F^{\mathrm{T}} + Q_k \tag{58}$$

where $\tilde{x}_{k-1|k-1}$ and $P_{k-1|k-1}$ represent the initial state and covariance.

(2) update step

i) generate basic cubature points

$$\xi_i = \sqrt{\frac{n_t}{2}}\,[1]_i \tag{59}$$

where $w_i = \frac{1}{n_t}$, $i = 1, 2, \ldots, n_t$, and n_t is the total number of cubature points, $n_t = 2n_x$ (n_x represents the dimensionality of the state).

ii) compute cubature points

$$S_{k|k-1} = \sqrt{P_{k|k-1}} \tag{60}$$

$$\chi^i_{k|k-1} = \tilde{x}_{k|k-1} + S_{k|k-1}\xi_i \tag{61}$$

iii) propagate cubature points through the observation equation

$$Z^i_{k|k-1} = h(\chi^i_{k|k-1})|_{s=s_k,a=a_k} \tag{62}$$

iv) compute the predicted observation

$$z_{k|k-1} = \frac{1}{m}\sum_{i=1}^{m} Z^i_{k|k-1} \tag{63}$$

v) calculate the covariance matrix

$$R_{k,m} = p(\theta^P_{k,m}(x^p_k))R^f_{k,m}|_{s=s_k,a=a_k}\ p(\theta^P_{k,m}(x^p_k))^{\mathrm{T}} + R_k \tag{64}$$

$$P^{zz}_{k|k-1} = \frac{1}{m}\sum_{i=1}^{m} Z^i_{k|k-1}(Z^i_{k|k-1})^{\mathrm{T}} - z_{k|k-1}(z_{k|k-1})^{\mathrm{T}} + R_{k,m} \tag{65}$$

$$P^{xz}_{k|k-1} = \frac{1}{m}\sum_{i=1}^{m} \chi^i_{k|k-1}(Z^i_{k|k-1})^{\mathrm{T}} - \tilde{x}_{k|k-1}(z_{k|k-1})^{\mathrm{T}} \tag{66}$$

vi) compute the gain

$$W_k = P^{xz}_{k|k-1}(P^{zz}_{k|k-1})^{-1} \tag{67}$$

vii) calculate the estimated value

$$\tilde{x}_{k|k} = \tilde{x}_{k|k-1} + W_k(z_k - z_{k|k-1}) \tag{68}$$

$$P_{k|k} = P_{k|k-1} - W_k P^{zz}_{k|k-1}(W_k)^{\mathrm{T}} \tag{69}$$

Given that the CKF offers higher filtering accuracy and stability compared to the EKF, the CKF is chosen for nonlinear filtering.

ORCID iDs

Ziwen Zhao https://orcid.org/0000-0001-7298-5599
Hui Chen https://orcid.org/0000-0002-1102-9912

References

[1] Dahal P, Mentasti S, Arrigoni S, Braghin F, Matteucci M and Cheli F 2023 Extended object tracking in curvilinear road coordinates for autonomous driving *IEEE Trans. Intell. Veh.* **8** 1266–78

[2] Haag S, Duraisamy B, Govaers F, Fritzsche M, Dickmann J and Koch W 2021 Extended object tracking assisted adaptive multi-hypothesis clustering for radar in autonomous driving domain *Proc. Int. Radar Symp. (IRS)* pp 1–10

[3] Haag S, Duraisamy B, Govaers F, Koch W, Fritzsche M and Dickmann J 2019 Extended object tracking assisted adaptive clustering for radar in autonomous driving applications *Proc. Sens. Data Fusion: Trends, Solut., Appl. (SDF)* pp 1–7

[4] Zhou H, Jiang F and Lu H 2023 Body-part joint detec-tion and association via extended object representation *IEEE Int. Conf. on Multimedia and Expo (ICME)* pp 168–73

[5] Meyer F and Williams J L 2021 Scalable detection and tracking of geometric extended objects *IEEE Trans. Signal Process.* **69** 6283–98

[6] Wu Z-J, Wang C-X, Li Y-C and Zhou Z-Q 2022 Extended target estimation and recognition based on multimodel approach and waveform diversity for cognitive radar *IEEE Trans. Geosci. Remote Sens.* **60** 1–14

[7] Li Y, Wei P, Wei Y, Gao L and Zhang H 2022 Loopy sum-product algorithm based joint detection, tracking and classification of extended objects with analytic implementations *Signal Process* **196** 108520

[8] Tuncer B, Kumru M and Özkan E 2019 Extended target tracking and classification using neural net-works *Proc. Int. Conf. Inf. Fusion (FUSION)* pp 1–7

[9] Cao W, Lan J and Li X R 2018 Extended object tracking and classification using radar and ESM sensor data *IEEE Signal Process Lett.* **25** 90–4

[10] Koch J W 2008 Bayesian approach to extended object and cluster tracking using random matrices *IEEE Trans. Aerosp. Electron. Syst.* **44** 1042–59

[11] Baum M and Hanebeck U D 2014 Extended object tracking with random hypersurface models *IEEE Trans. Aerosp. Electron. Syst.* **50** 149–59

[12] Wahlstrom N and Ozkan E 2015 Extended target tracking using gaussian processes *IEEE Trans. Signal Process.* **63** 4165–78

[13] Feldmann M, Franken D and Koch W 2011 Tracking of extended objects and group targets using random matrices *IEEE Trans. Signal Process.* **59** 1409–20

[14] Yang S and Baum M 2019 Tracking the orientation and axes lengths of an elliptical extended object *IEEE Trans. Signal Process.* **67** 4720–9

[15] Tuncer B, Orguner U and Ozkan E 2022 Multi-ellipsoidal extended target tracking with variational bayes inference *IEEE Trans. Signal Process.* **70** 3921–34

[16] Tuncer B and Ozkan E 2021 Random matrix based extended target tracking with orientation: a new model and inference *IEEE Trans. Signal Process.* **69** 1910–23

[17] Sun L, Yu H, Fu Z, He Z and Zou J 2021 Modeling and tracking of maneuvering extended object with random hypersurface *IEEE Sens. J.* **21** 20552–62

[18] Zhang X, Yan Z, Chen Y and Yuan Y 2022 A novel particle filter for extended target tracking with random hypersurface model *Appl. Math. Comput.* **425** 127081

[19] Sun L, Zhang J, Yu H, Fu Z and He Z 2022 Maneuvering extended object tracking with modified star-convex random hypersurface model based on minimum cosine distance *Remote Sens* **14** 4376

[20] Seeger M 2004 Gaussian processes for machine learning *Int. J. Neur. Syst.* **14** 69–106

[21] Thormann K, Baum M and Honer J 2018 Extended target tracking using gaussian processes with high-resolution automotive radar *Proc. Int. Conf. Inf. Fusion (FUSION)* pp 1764–70

[22] Li Q and Song L 2020 Axisymmetric extended target tracking using Gaussian process *Proc. IEEE Int. Conf. Inf. Commun. Signal Process. (ICICSP)* pp 306–11

[23] Kumru M and Özkan E 2021 Three-dimensional ex-tended object tracking and shape learning using Gaussian processes *IEEE Trans. Aerosp. Electron. Syst.* **57** 2795–814

[24] Yang D, Guo Y, Yin T and Lin B 2023 Cost-effective gaussian processes based extended target tracking *IEEE Trans. Aerosp. Electron. Syst.* **59** 8282–96

[25] Aftab W, Hostettler R, De Freitas A, Arvaneh M and Mihaylova L 2019 Spatio-temporal gaussian process models for extended and group object tracking with irregular shapes *IEEE Trans. Veh. Technol.* **68** 2137–51

[26] Clifton J and Laber E 2020 Q-learning: theory and applications *Annual Review of Statistics and Its Application* **7** 279–301

[27] Shi Q, Lam H-K, Xiao B and Tsai S-H 2018 Adaptive PID controller based on Q -learning algorithm *CAAI Trans. Intell. Technol.* **3** 235–44

[28] Maoudj A and Hentout A 2020 Optimal path planning approach based on Q-learning algorithm for mobile robots *Appl. Soft Comput.* **97** 106796

[29] Zhu Q, Su S, Tang T, Liu W, Zhang Z and Tian Q 2022 An eco-driving algorithm for trains through distributing energy: a Q-learning approach *ISA Trans* **122** 24–37

[30] Wang R, Zhuang Z, Tao H, Paszke W and Stojanovic V 2023 Q-learning based fault estimation and fault tolerant iterative learning control for MIMO systems *ISA Trans* **142** 123–35

[31] Ni X, Hu W, Fan Q, Cui Y and Qi C 2024 A Q-learning based multi-strategy integrated artificial bee colony algorithm with application in unmanned vehicle path planning *Expert Syst. Appl.* **236** 121303

[32] Xiong K, Wei C and Zhang H 2021 Q-learning for noise covariance adaptation in extended KALMAN filter *Asian J. Control* **23** 1803–16

[33] Dai X, Fourati H and Prieur C 2022 A dynamic grid-based Q-learning for noise covariance adaptation in ekf and its application in navigation *Proc. IEEE Conf Decis Control (CDC)* pp 4984–9

[34] Xiong K and Wei C 2021 Q-learning-based target selection for bearings-only autonomous navigation *J .Syst. Sci. Complex* **34** 1401–25

[35] Xiong K, Zhou P and Wei C 2024 Spacecraft autonomous navigation using line-of-sight directions of non-cooperative targets by improved Q-learning based extended Kalman filter *Proc. Inst. Mech. Eng. Part G J. Aerosp. Eng.* **238** 182–97

[36] Bekhtaoui Z, Meche A, Dahmani M and Me-raim K A 2017 Maneuvering target tracking using q-learning based Kalman filter *Proc. Int. Conf. Electr. Eng. - Boumerdes (ICEE-B)* pp 1–5

[37] Alexe B, Deselaers T and Ferrari V 2012 Measuring the objectness of image windows *IEEE Trans. Pattern Anal. Mach. Intell.* **34** 2189–202

[38] Taha A A and Hanbury A 2015 An Efficient algorithm for calculating the exact hausdorff distance *IEEE Trans. Pattern Anal. Mach. Intell.* **37** 2153–63

[39] Guo Y, Li Y, Tharmarasa R, Kirubarajan T, Efe M and Sarikaya B 2019 GP-PDA filter for extended target tracking with measurement origin uncertainty *IEEE Trans. Aerosp. Electron. Syst.* **55** 1725–42

[40] Tang X, Li M, Tharmarasa R and Kirubarajan T 2019 Seamless tracking of apparent point and extended targets using gaussian process PMHT *IEEE Trans. Signal Process.* **67** 4825–38

[41] Kennedy J and Eberhart R 1995 Particle swarm optimization *Proceedings of ICNN'95-International Conference on Neural Networks* pp 1942–8

sensors

MDPI

Article

Multi-Maneuvering Target Tracking Based on a Gaussian Process

Ziwen Zhao and Hui Chen *

School of Electrical and Information Engineering, Lanzhou University of Technology, Lanzhou 730050, China; ziw@lut.edu.cn

* Correspondence: chenh@lut.edu.cn

Abstract: Aiming at the uncertainty of target motion and observation models in multi-maneuvering target tracking (MMTT), this study presents an innovative data-driven approach based on a Gaussian process (GP). Traditional multi-model (MM) methods rely on a predefined set of motion models to describe target maneuvering. However, these methods are limited by the finite number of available models, making them unsuitable for handling highly complex and dynamic real-world scenarios, which, in turn, restricts the adaptability and flexibility of the filter. In addition, traditional methods often assume that observation models follow ideal linear or simple nonlinear relationships. However, these assumptions may be biased in actual application and so lead to degradation in tracking performance. To overcome these limitations, this study presents a learning-based algorithm-leveraging GP. This non-parametric GP approach enables learning an unlimited range of target motion and observation models, effectively mitigating the problems of model overload and mismatch. This improves the algorithm's adaptability in complex environments. When the motion and observation models of multiple targets are unknown, the learned models are incorporated into the cubature Kalman probability hypothesis density (PHD) filter to achieve an accurate MMTT estimate. Our simulation results show that the presented approach delivers high-precision tracking of complex multi-maneuvering target scenarios, validating its effectiveness in addressing model uncertainty.

Keywords: data-driven; multi-maneuvering target tracking; Gaussian process; model-free tracking

check for updates

Citation: Zhao, Z.; Chen, H. Multi-Maneuvering Target Tracking Based on a Gaussian Process. *Sensors* **2024**, *24*, 7270. https://doi.org/10.3390/s24227270

Academic Editor: Wenling Li

Received: 27 September 2024
Revised: 9 November 2024
Accepted: 9 November 2024
Published: 14 November 2024

1. Introduction

Maneuvering target tracking (MTT) involves monitoring the velocity, acceleration, position, and other state information of a moving target by using sensors that predict and track the target's trajectory using algorithms. This technology has extensive applications in video surveillance, robotic vision, and military operations [1–6]. MTT remains challenging due to external environment and disturbance effects, where the target motion may exhibit irregular and highly dynamic characteristics [7].

Traditional MTT methods, which are model-driven (MD), describe the dynamic characteristics of the target through reasonable assumptions and modeling of target motion. These methods utilize recursive filtering techniques to process sensor measurements and system noise. The interactive multiple model (IMM) algorithm is a typical representative of this category. It employs multiple motion models simultaneously to describe different target motion models. It dynamically adjusts the weights of each model during filtering to achieve optimal state estimation of the target [8–13]. To further improve the flexibility and adaptability of MTT, ref. [14] proposed the variable-structure IMM algorithm, which handles changes in target motion models more effectively. However, these methods are primarily designed for single-target tracking problems. With an increasing number of targets in the surveillance area, the applicability of these methods decreases significantly.

The multi-model (MM) approach is an effective solution for multi-maneuvering target tracking (MMTT), which is widely applied to solving multi-target tracking problems with various motion patterns. Several MMTT filtering algorithms have been developed, based on this approach. For example, refs. [15,16] introduced the MM probability hypothesis density (PHD) filter; ref. [17] proposed an MM cardinalized PHD filter, to address the problem of inaccurate target cardinality estimation in MMTT; ref. [18] proposed a variable-structure MM-PHD (VSMM-PHD) filter to improve the efficiency and accuracy of MMTT. Unlike the traditional MM-PHD filter, VSMM-PHD uses a different set of models for each target at different times, better adapting to changes in target motion. In addition, refs. [19–21] developed various MM MeMBer filters to meet the needs of different tracking scenarios. However, as the uncertainty of target trajectories increases and the diversity of target motion patterns increases, model-based methods become increasingly inadequate for handling such complex variations. These methods are subject to certain limitations in practical application. Firstly, model-based methods rely heavily on initial conditions. Inaccurate initial settings can adversely affect estimation performance. Secondly, although increasing the number of models can improve tracking accuracy, an excessive number of models significantly increases the computational cost and complexity.

To overcome the limitations of the traditional methods, the data-driven (DD) approach, which is mainly based on a Gaussian process (GP) [22], provides promising alternatives. By contrast, GP-based techniques can learn the underlying models and model parameters from training data through non-parametric regression, thus eliminating the dependence on motion models in the classic MTT approach. The advantage of this strategy is its ability to adapt to different target motion models and produce more reliable state estimates. In recent years, GP-based target-tracking methods have increasingly become a popular research area, and they provide a substitute for traditional methods. A GP is a non-parametric machine learning regression method based on Bayesian inference. The distribution of output variables is modeled through a GP, which updates this distribution, using observational data. As a flexible model, a GP can adapt to various input and output data in multi-dimensional spaces and perform adaptive optimization based on the data. Moreover, a GP can seamlessly integrate state space models and Bayesian filtering. For instance, refs. [23,24] demonstrates using the GP to learn prediction and observation models from training data; ref. [25] combines Kalman filtering (KF) with a GP to create an efficient GP estimator for a spatiotemporal dynamic GP. Furthermore, ref. [26] modeled unknown perturbations as the GP and proposed an adaptive KF to improve the estimation performance.

Recent studies have applied the GP to MTT to address issues related to unknown target motion models or mismatches between motion models and actual target motion. For example, ref. [27] proposed a model-free MTT method that leverages the flexibility of the GP to enable switching between a large number of models and state estimates. Another study [28] introduced a DD method for MTT and smoothing, which showed significant performance improvements compared to traditional MD methods; ref. [29] presented a new GP-based approach to learning motion models and applied it within particle filtering to track targets in different surveillance regions. Furthermore, ref. [30] used a GP to approximate the transition density of the Bayesian optimal Bernoulli filter and proposed a particle implementation of the Bernoulli filter to handle unknown target motion model transitions, while [31] proposed a hybrid strategy that combines DD and MD approaches and effectively improves the tracking performance of strong maneuverable targets by integrating the advantages of both methods. Despite the success of these GP-based approaches in a variety of application scenarios, research for model-free MMTT in the context of random finite set (RFS) theory [32] has not yet been implemented. Further exploration of this area is essential to advance the development of MMTT technology.

To this end, this study proposes a novel MMTT algorithm to improve tracking accuracy in complex environments. The main contributions of this paper are as follows:

(1) A data-driven MMTT state estimation method is proposed by combining a GP and the PHD filter. The method models the MMTT motion and observation models as nonlinear functions over time. It uses a GP to learn the unknown characteristics of the target motion and observation models from training data.

(2) Based on the GP model learning, a cubature Kalman filter (CKF) [33] is utilized to propagate the uncertainty of the system to achieve accurate estimation. The GP

possess provide model-learning capability, while the CKF efficiently handles nonlinear system through the 'cubature sampling' technique. This innovative design allows the GP-PHD filter to achieve excellent tracking accuracy and stability in uncertain and complex environments.

(3) To verify the effectiveness of the proposed algorithm, two groups of simulation experiments with different scenarios are designed. The results demonstrate that, compared to the traditional MD method, the GP-based method offers significant advantages in an environment with unpredictable and highly dynamic target motion.

Furthermore, the existing GP-based MTT algorithms are limited to scenarios involving a single target. However, the proposed algorithm overcomes this limitation, enabling simultaneous tracking of multi-maneuvering targets. This capability is particularly important in scenarios with numerous targets and frequent dynamic changes. The proposed method imposes no restrictions on the number of targets. It can effectively handle target generation, disappearance and maneuvering behavior, demonstrating its applicability and flexibility in complex MMTT scenarios.

The remainder of the paper is organized as follows. Section 2 introduces the problem definition and background, and Section 3 introduces the Gaussian process. A detailed implementation of the proposed algorithm is given in Section 4. Simulation results are provided in Section 5, and Section 6 concludes the paper.

2. Problem Definition and Background

2.1. System Model

Consider a discrete-time dynamic model with a transfer dynamics equation and observation equation as

$$x_{t+1} = f(x_t) + \phi_t \tag{1}$$

$$z_t = g(x_t) + \varsigma_t \tag{2}$$

where $x_t = [\zeta_t, \dot{\zeta}_t, \varphi_t, \dot{\varphi}_t]^T$ represents the state for a target in two-dimensional space at time t, $(\zeta_t, \dot{\zeta}_t)$ represents the position along the x- and y-axis, $(\varphi_t, \dot{\varphi}_t)$ denotes the corresponding velocity; z_t denotes sensor measurement. The f and g are nonlinear process transfer functions and observation functions; ϕ_t and ς_t are the zero mean, white additive Gaussian process and measurement noise, respectively.

2.2. Multi-Target Bayesian Filtering

Based on the RFS theory [32], the state and measurement sets for multiple targets are represented as RFS $X_t = \{x_{t,1}, \ldots, x_{t,n_x}\}$ and $Z_t = \{z_{t,1}, \ldots, z_{t,n_z}\}$, respectively; n_x and n_z specify the number for targets and measurements, respectively. According to the Chapman–Kolmogorov Equation [34], the multi-target prediction equation at time t can be derived as

$$f_{t|t-1}(X_t|Z_{1:t-1}) = \int f_{t|t-1}\left(X_t \middle| X_{t|t-1}\right) f_{t-1}\left(X_{t|t-1}|Z_{1:t-1}\right)\delta X_{t|t-1} \tag{3}$$

where $f_{t|t-1}$ and $f_{t-1}\left(X_{t|t-1}|Z_{1:t-1}\right)$ denote a multi-target state transfer function and state at time $t-1$, respectively. According to Bayes' rule, after a new set Z_t of measurements is received at time t, the multi-target update equation is given by

$$f_{t|t}\left(X_{t|t}|Z_{1:t}\right) = \frac{L_{t|t}\left(Z_{t|t} \middle| X_{t|t}\right) f_{t|t-1}\left(X_{t|t}|Z_{1:t-1}\right)}{\int L_{t|t}\left(Z_{t|t} \middle| X_{t|t}\right) f_{t|t-1}\left(X_{t|t}|Z_{1:t-1}\right)\delta X_{t|t}} \tag{4}$$

2.3. PHD Filter

Suppose v_t and $v_{t|t-1}$ denote the intensity functions corresponding to multi-target posterior density p_t alongside predicted density $p_{t|t-1}$, respectively. The multi-target intensity function at time $t-1$ is given by the ν_t, and its prediction equation of the PHD filter can be expressed as

$$\nu_{t|t-1}(x) = \int p_{s,t}(x')f_{t|t-1}(x|x')\nu_{t-1}(x')dx' + \int \beta_{t|t-1}(x|x')\nu_{t-1}(x')dx' + \gamma_t(x) \quad (5)$$

where $p_{s,t}(x')$ denotes the probability of surviving, $f_{t|t-1}(x|x')$ represents the transition probability density of a single target. At time t, $\beta_{t|t-1}(x|x')$ and $\gamma_t(x)$ represent the intensity of the spawned and birth targets, respectively. Given the set of measurements Z_t at time t, the update Equation for the PHD filter is

$$\nu_{t|t}(x) = (1 - p_{d,t}(x))\nu_{t|t-1}(x) + \sum_{z\in Z_t} \frac{p_{d,t}(x)g_t(z|x)\nu_{t|t-1}(x)}{\kappa_t(z) + \int p_{d,t}(x')g_t(z|x')\nu_{t|t-1}(x')dx'} \quad (6)$$

where $p_{d,t}(x)$ denotes detection probability, $g_t(z|x)$ represents the measurement likelihood of a single target, and $\kappa_t(z)$ signifies the intensity for clutter.

3. Gaussian Process

Using a training dataset, the GP is a complex non-parametric learning algorithm primarily used to learn unknown functions. The dataset contains input–output pairs, and the GP provides a mapping between them. The critical aspect that comprises GP involves the flexibility of modeling as it facilitates simulating the behavior of a system in the face of uncertainty.

3.1. Basic Gaussian Process Model

The GP represents a distribution of the function based on the training data. Suppose there is a set of training data $T_d = \langle X, y\rangle$, where d-dimensional input vector x_i are arranged in the matrix $X = [x_1, x_2, \ldots, x_n]$, where n denotes the number of training points, and $y = [y_1, y_2, \ldots, y_n]$ is the vector containing the scalar training output. Assume that the measurement values are derived from the noise process

$$y_i = h(x_i) + \varepsilon \quad (7)$$

where ε is additive Gaussian white noise with zero mean and variance is σ_n^2. The Gaussian predictive distribution on the output y_* for training data $T_d = \langle X, y\rangle$ and test inputs x_*, the mean and variance are specified by the GP, i.e.,

$$GP_m(x_*, T_d) = k_*^T K^{-1} y \quad (8)$$

$$GP_v(x_*, T_d) = k(x_*, x_*) - k_*^T K^{-1} k_* \quad (9)$$

In this case, k_* is a vector formed by the kernel values between the test input x_* and the training input X, where k indicates the kernel function for the GP, and the training input values are represented by the $n \times n$ dimensional kernel matrix K, which means that, $k_*[i] = k(x_*, x_i)$ and $K[i,j] = k\big(x_i, x_j\big)$. It should be emphasized that process noise, both the correlation between the test input and the training data, influence the prediction uncertainty as reflected by variance GP_v.

The exact application scenario determines the kernel function to be utilized, with the squared exponential or the Gaussian kernel with additive noise being the most popular

$$k\big(x, x'\big) = \sigma_f^2 e^{-\frac{1}{2}(x-x')A(x-x')^T} + \sigma_n^2\delta \quad (10)$$

and the signal variance is given by σ_f^2, thus regulating the degree of prediction uncertainty in the area of low training data density. The length scale of the process is contained in the diagonal matrix A, for example, $A = diag\big([1/a_1^2, 1/a_1^2, \ldots, 1/a_d^2]\big)$. In different input dimensions, the length scale reacts to how smooth the operation is overall. σ_n^2 is the final

GP parameter, which controls the noise of the whole process. Figure 1 illustrates a one-dimensional GP example. In the figure, the red × denotes the training point, the blue curve represents the prediction result, and the blue shading represents uncertainty. It can be seen from the figure that the uncertainty is lower near the training points and increases in areas away from the training points.

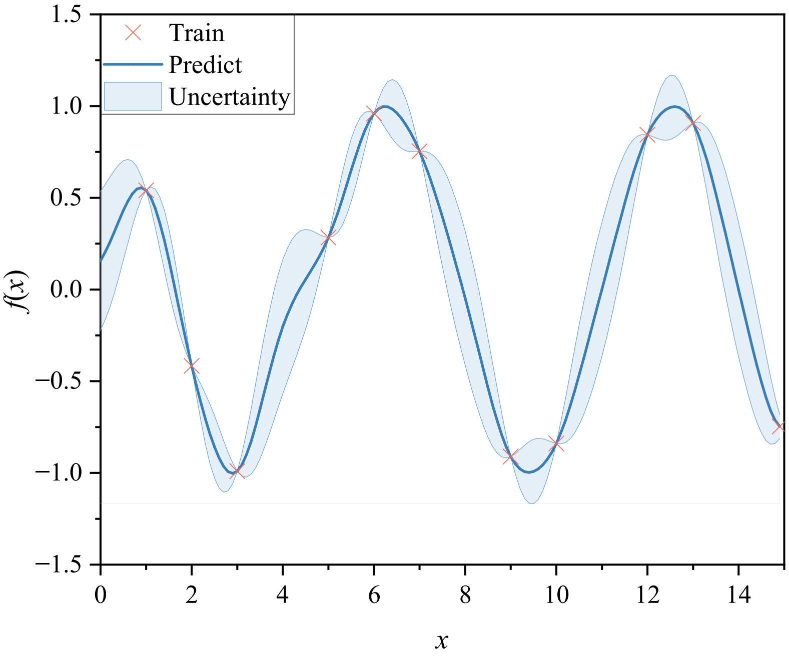


Figure 1. One-dimensional GP.

3.2. Hyperparameter Learning

The hyperparameters of the GP are represented by $\theta = \left[A, \sigma_f, \sigma_n\right]$. By maximizing the log marginal likelihood of the training output for a given input, they can be trained

$$\theta_{\max} = \arg\max_{\theta}\{\log(p(y|X,\theta))\} \tag{11}$$

It is possible to express the logarithmic component in (11) as

$$\log(p(y|X,\theta)) = -\frac{1}{2}y^T\left(K(X,X)+\sigma_n^2 I\right)^{-1}y - \frac{1}{2}\log\left|K(X,X)+\sigma_n^2 I\right| - \frac{n}{2}\log 2\pi \tag{12}$$

Numerical optimization methods such as conjugate gradient ascent can be employed to solve this optimization problem [21]. To perform this optimization, it is essential to use the partial derivatives of the log-likelihood, as given below

$$\frac{\partial}{\partial\theta_t}\log(p(y|X,\theta)) = \frac{1}{2}tr\left[\left(K^{-1}y\right)\left(K^{-1}y\right)^T\frac{\partial K}{\partial\theta_t}\right] \tag{13}$$

Each element of $\frac{\partial K[i,j]}{\partial\theta_t}$ in (13) represents a partial derivative of a kernel function in regard to its hyperparameters

$$\frac{\partial k(x_i,x_j)}{\partial\sigma_f} = 2\sigma_f e^{-\frac{1}{2}(x_i-x_j)A(x_i-x_j)^T} \tag{14}$$

$$\frac{\partial k(x_i,x_j)}{\partial\sigma_n} = 2\sigma_n\delta \tag{15}$$

$$\frac{\partial k(x_i, x_j)}{\partial A_{ii}} = -\frac{1}{2}\left(x_i[i] - x_j[i]\right)^2 \sigma_f^2 e^{-\frac{1}{2}\left(x_i - x_j\right)A\left(x_i - x_j\right)^T} \tag{16}$$

Due to the non-convex nature of this optimization problem, finding the global optimal solution cannot be guaranteed. However, in practical applications, such optimization problems often yield satisfactory results.

3.3. Learning Prediction and Observation Models Using Gaussian Process

The GP is possibly employed straight to the Bayesian filter in (3), and it has been shown to satisfy the conditions for learning predictive and observational models. In the context of the application in this work, the model needs to provide both expected mean and predicted uncertainty or noise. The GP inherently satisfies both objectives in its unique manner.

The training data are obtained by dynamically sampling and observing the system. They are expected to be representative of the system, i.e., they can span the state space encountered during normal operation. A set of input–output relations forms the training data for each GP. In the predictive model, state and control variables (x_t, u_t) are mapped to state transitions $\Delta x_t = x_{t+1} - x_t$. Then, the previous state is added to the state transition to determine the subsequent state. The state x_t is mapped into observation z_t using the observation model. Consequently, the training dataset for prediction and observation should have the following form

$$T_p = \left\langle (X, u), X' \right\rangle \tag{17}$$

$$T_o = \langle X, Z \rangle \tag{18}$$

where the matrix containing the real states is indicated by X, and the matrix created when these states experience a transfer of control in application u is $X' = [\Delta x_1, \Delta x_2, \ldots, \Delta x_t]$. The observation matrix for the corresponding state X is denoted by Z. The prediction and observation models for the GP are subsequently obtained as

$$p(x_t | x_{t-1}, u_{t-1}) \approx N\left(GP_m\left([x_{t-1}, u_{t-1}], T_p\right), GP_v\left([x_{t-1}, u_{t-1}], T_p\right)\right) \tag{19}$$

$$p(z_t | x_t) \approx N(GP_m(x_t, T_o), GP_v(x_t, T_o)) \tag{20}$$

It is important to note that the mean and variance of these models, for both input and training data, are nonlinear functions, even though they correspond to a Gaussian distribution. Moreover, due to their local Gaussian character, these models are seamlessly integrated into Bayesian filters.

The GP is typically defined in the case of scalar outputs. However, the GP Bayesian filter is represented for the vector output model by learning a distinct GP for each output dimension. Since the output dimensions are no longer interdependent, the resulting noise covariance matrix of the GP becomes diagonal.

4. Gaussian Process Bayesian Filter

In the following phase, a GP model will be introduced into the Bayesian filter to address the uncertainty in the motion and observation models of MMTT.

4.1. Gaussian Process for System Model

Some existing MD methods represent the motion and observation states of a target through one or more defined equations of motion and observations. However, GP-based approaches eliminate the need for precise equations of motion and observation. This reduces the reliance on the target motion and observation models by encoding the target state through the learned GP state and observation models.

The GP state model GP^f and observation model GP^h can be used to express the state and measurement equations as shown below

$$x_t = GP_m^f\left([x_{t-1}, u_{t-1}], T_p\right) + \phi_{t-1} \tag{21}$$

$$z_t = GP_m^h(x_t, T_o) + \varsigma_t \tag{22}$$

where

$$\phi_{t-1} \sim N\left(0, GP_v^f\left([x_{t-1}, u_{t-1}], T_p\right)\right) \tag{23}$$

$$\varsigma_t \sim N\left(0, GP_v^h(x_t, T_o)\right) \tag{24}$$

4.2. GP-CK-PHD Gaussian Mixture Implementation

Based on the Gaussian mixture (GM) recursive construction of the standard PHD filter, the posterior intensity of the multi-target state is expressed as a weighted sum of multiple non-Gaussian functions, derived through the recursive propagation in (5) and (6). Gaussian functions can approximate each non-Gaussian component, and similar to the CKF method, the 'cubature sampling' approach can be used to calculate the GM approximating components of the posterior intensity at subsequent time steps while approximating the weight of each component.

Therefore, this study proposes a nonlinear GM implementation based on the GP-PHD filter to address the challenges posed by uncertain motion and observation models in MMTT. This method leverages the GP learning approach and employs cubature sampling for propagation, making it an effective solution for tackling the problems of uncertain motion and observation models under nonlinear conditions in MMTT.

Considering the properties of nonlinear systems, it is impossible to represent the posterior intensity explicitly in GM form, so it is necessary to approximate the non-Gaussian component of the posterior intensity using an appropriate Gaussian distribution. The GM form for the birth RFS intensity is

$$\gamma_t(x) = \sum_{a=1}^{J_{\gamma,t}} w_{\gamma,t}^a N\left(x; m_{\gamma,t}^a, P_{\gamma,t}^a\right) \tag{25}$$

where $J_{\gamma,t}$, $w_{\gamma,t}^a$, $m_{\gamma,t}^a$, $P_{\gamma,t}^a$, $a = 1, \ldots, J_{\gamma,t}$ are the model parameter given to determine the birth intensity. The particular procedure is described below:

(1) Consider the following as an approximation of the posterior intensity at time $t-1$ can be approximated by

$$\nu_{t-1}(x) \approx \sum_{a=1}^{J_{t-1}} w_{t-1}^a N\left(x; m_{t-1}^a, P_{t-1}^a\right) \tag{26}$$

Then, at the time t, the predicted intensity is

$$\nu_{t|t-1}(x) = \nu_{s,t|t-1}(x) + \gamma_t(x) \tag{27}$$

where

$$\nu_{s,t|t-1}(x) \approx p_{s,t} \sum_{j=1}^{J_{t-1}} w_{t-1}^j N\left(x; m_{s,t|t-1}^j, P_{s,t|t-1}^j\right) \tag{28}$$

According to the Cubature rule, $2n$ weighted Cubature sampling points $\left[x_{t|t-1}^l, w_{t|t-1}^l\right]$ are selected, and the quantity of sampling points is $l = 0, 1, \ldots, 2n$. Then, the model of the unknown system is linearized, where

$$x_{l,t-1} = x_{t-1} \pm \sqrt{P_{t-1}}\alpha_l \tag{29}$$

$$x^{l}_{t|t-1} = GP_m\left([x_{l,t-1}, u_{t-1}], T_p\right) \tag{30}$$

$$Q_t = GP_v\left([x_{t-1}, u_{t-1}], T_p\right) \tag{31}$$

$$m^{j}_{s,t|t-1} = \frac{1}{2n}\sum_{l=0}^{2n} x^{l}_{t|t-1} \tag{32}$$

$$P^{j}_{s,t|t-1} = \frac{1}{2n}\sum_{l=0}^{2n}\left(x^{l}_{t|t-1} - m^{j}_{s,t|t-1}\right)\left(x^{l}_{t|t-1} - m^{j}_{s,t|t-1}\right)^{T} + Q_t \tag{33}$$

(2) Suppose that a Gaussian mixture can be used to roughly represent the predicted intensity at time t, i.e.,

$$\nu_{t|t-1}(x) \approx \sum_{j=1}^{J_{t|t-1}} w^{j}_{t|t-1} N\left(x; m^{j}_{t|t-1}, P^{j}_{t|t-1}\right) \tag{34}$$

Then the posterior intensity at time t is likewise in the structure of a GM, denoted as

$$\nu_t(x) = (1 - p_{d,t})\nu_{t|t-1}(x) + \sum_{z\in Z_t} \nu_{d,t}(x; z) \tag{35}$$

where

$$\nu_{d,t}(x; z) = \sum_{j=1}^{J_{t|t-1}} w^{j}_{t}(z) N\left(x; m^{j}_{t|t}(z); P^{j}_{t|t}\right) \tag{36}$$

$$w^{j}_{t}(z) = \frac{p_{d,t} w^{j}_{t|t-1} q^{j}_{t}(z)}{\kappa_t(z) + p_{d,t}\sum\limits_{j=1}^{J_{t|t-1}} w^{j}_{t|t-1} q^{j}_{t}(z)} \tag{37}$$

$$w^{j}_{t|t-1} = p_{s,t} w^{j}_{t-1} \tag{38}$$

$$q^{j}_{t}(z) = N\left(z; \eta^{j}_{t|t-1}, S^{j}_{t}\right) \tag{39}$$

$$m^{j}_{t|t}(z) = m^{j}_{s,t|t-1} + K^{j}_{t}\left(z - \eta^{j}_{t|t-1}\right) \tag{40}$$

$$x^{l}_{t|t} = m^{j}_{s,t|t-1} \pm \sqrt{P^{j}_{s,t|t-1}}\alpha_l \tag{41}$$

$$z^{l}_{t|t-1} = GP_m\left(x^{l}_{t|t}, T_o\right), \quad l = 0, \ldots, 2n \tag{42}$$

$$R_t = GP_v\left(m^{j}_{t|t-1}, T_o\right) \tag{43}$$

$$\eta^{j}_{t|t-1} = \frac{1}{2n}\sum_{l=0}^{2n} z^{l}_{t|t-1} \tag{44}$$

$$S^{j}_{t} = \frac{1}{2n}\sum_{l=0}^{2n}\left(z^{l}_{t|t-1} - \eta^{j}_{t|t-1}\right)\left(z^{l}_{t|t-1} - \eta^{j}_{t|t-1}\right)^{T} + R_t \tag{45}$$

$$P^{j}_{xz,t} = \frac{1}{2n}\sum_{l=0}^{2n}\left(x^{l}_{t|t-1} - m^{j}_{s,t|t-1}\right)^{T}\left(z^{l}_{t|t-1} - \eta^{j}_{t|t-1}\right)^{T} \tag{46}$$

$$K^{j}_{t} = P^{j}_{xz,t}\left(S^{j}_{t}\right)^{-1} \tag{47}$$

$$P^{j}_{t|t} = P^{j}_{t|t-1} - K^{j}_{t} S^{j}_{t}\left(K^{j}_{t}\right)^{-1} \tag{48}$$

Given the GM intensity $\nu_{t|t-1}$ and ν_t, the appropriate weights can be summed jointly to yield the associated expected number of targets $\hat{n}_{t|t-1}$ and $\hat{n}_t$.

According to the prediction step, the mean value of the predicted number of targets is

$$\hat{n}_{t|t-1} = \hat{n}_{t-1} p_{s,t} + \sum_{j=1}^{J_{\gamma,t}} w^j_{\gamma,t} \tag{49}$$

According to the update step, the mean value of the updated target number is

$$\hat{n}_t = \hat{n}_{t|t-1}(1 - p_{d,t}) + \sum_{z \in Z_t} \sum_{j=1}^{J_{t|t-1}} w^j_t(z) \tag{50}$$

(3) Pruning & Merging

The GP-PHD filter encounters the same computational challenges as the standard GM-PHD filter, especially the growth of the Gaussian components over time. To address this issue, an efficient pruning strategy is employed to reduce the number of Gaussian components passed to subsequent time steps [15]. The specific steps of the GP-PHD algorithm are described in Algorithm 1.

Algorithm 1 The GP-PHD algorithm

```
Input: $\{w^a_{t-1}, m^a_{t-1}, P^a_{t-1}\}_{a=1}^{J_{t-1}}, Z_t, T_p, T_o$
Predict
(1) predict newborn targets
$a = 0$
for $j = 1 : J_{\gamma,t}$ do
    $a = a + 1$
    $w^i_{t|t-1} = w^j_{\gamma,t}, m^i_{t|t-1} = m^j_{\gamma,t}, P^i_{t|t-1} = P^j_{\gamma,t}$
end for
(2) predict existing targets
for $j = 1 : J_{t-1}$ do
    $a = a + 1$
    use (29)–(33) calculate the predictive parameters $m^j_{s,t|t-1}$ and $P^j_{s,t|t-1}$ for the birth targets
end for
$J_{t|t-1} = i$
Update
for $j = 1 : J_{t|t-1}$ do
    $w^a_t = (1 - p_{d,t})w^a_{t|t-1}, m^a_t = m^a_{t|t-1}, P^a_t = P^a_{t|t-1}$
end for
$q = 0$
for $b = 1 : length(Z_t)$ do
    $q = q + 1$
    for $j = 1 : J_{t|t-1}$ do
        $w^j_t = p_{d,t} w^j_{t|t-1} q^j_t(z)$
        use (36), (38)–(48) calculate the update parameters $m^j_{t|t}$ and $P^j_{t|t}$
    end for
    use (37) calculate the update parameters $w^j_t$
end for
$J_t = qJ_{t|t-1} + J_{t|t-1}$
Output: $\{w^i_t, m^i_t, P^i_t\}_{i=1}^{J_t}$
```

5. Simulation Experiments

5.1. Performance Evaluation

To evaluate the effectiveness of the proposed GP-PHD filtering algorithm in this part, employ the Generalized Optimal Subpattern Assignment (GOSPA) distance [35], which is defined as

$$d_p^{(c,\alpha)}(X,Y) \triangleq [\min_{\gamma\in\Gamma}(\sum_{(i,j)\in\gamma} d\left(x_i,y_j\right)^p + \frac{c^p}{\alpha}(|X|+|Y|-\alpha|\gamma|))]^{\frac{1}{p}} \tag{51}$$

The parameters are assigned to $c = 50$, $p = 2$, $\alpha = 2$.

5.2. Simulation Results

(1) Scenario 1: For a two-dimensional surveillance region $[-800,800]\times[-800,800]$ m contains clutter and an unknown number of targets which evolve over time. Each target moves autonomously according to its motion model

$$x_t = F_{CV/CT}x_{t-1} + \phi_t \tag{52}$$

$$F_{CV} = \begin{bmatrix} 1 & \Delta & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & \Delta \\ 0 & 0 & 0 & 1 \end{bmatrix} \tag{53}$$

$$F_{CT} = \begin{bmatrix} 1 & \frac{(\sin\theta)}{\theta} & 0 & -\frac{(1-\cos\theta)}{\theta} \\ 0 & \cos\theta & 0 & -\sin\theta \\ 0 & \frac{(1-\cos\theta)}{\theta} & 1 & \frac{(\sin\theta)}{\theta} \\ 0 & \sin\theta & 0 & \cos\theta \end{bmatrix} \tag{54}$$

with $\phi_t \sim N(0,Q_t)$

$$Q_t = \sigma^2\begin{bmatrix} \Delta^4/4 & \Delta^3/2 & 0 & 0 \\ \Delta^3/2 & \Delta^2 & 0 & 0 \\ 0 & 0 & \Delta^4/4 & \Delta^3/2 \\ 0 & 0 & \Delta^3/2 & \Delta^3/2 \end{bmatrix} \tag{55}$$

where $\sigma = 0.1$, $\Delta = 1$ s represents the sampling interval. Model 1 is a CV model (M1); Model 2 has a turn rate of $\theta = -9$ °/s and represents a left-turning model (M2); Model 3 is a right-turn model and the turn rate is $\theta = 6$ °/s (M3). For each target, the survival probability and detection probability are $p_{s,t} = 0.97$ and $p_{d,t} = 0.95$, respectively. The observation consists of the orientation and distance

$$z_t = \begin{bmatrix} \arctan\left(\frac{\zeta_y}{\zeta_x}\right) \\ \sqrt{\zeta_x^2+\zeta_y^2} \end{bmatrix} + \varsigma_t \tag{56}$$

where $\varsigma_t \sim N(0,R_t)$, $R_t = diag\left(\left[\sigma_\theta^2,\sigma_r^2\right]^T\right)$, $\sigma_\theta = 2\times(\pi/180)$ rad/s, $\sigma_r = 10$ m. The clutter model is modeled using a uniform Poisson model with a clutter rate $\lambda_c = 10$. Additionally, a GM of the form is also utilized as the birth model of the target

$$\gamma_t(x) = \sum_{i=1}^{5} w_b^i N\left(x; m_b^i, P_b^i\right) \tag{57}$$

where $w_b^i = 0.1$ and
$m_b^1 = \begin{bmatrix} 50 & 0 & 250 & 0 \end{bmatrix}^T, m_b^2 = \begin{bmatrix} -250 & 0 & -250 & 0 \end{bmatrix}^T,$
$m_b^3 = \begin{bmatrix} -250 & 0 & 250 & 0 \end{bmatrix}^T, m_b^4 = \begin{bmatrix} 250 & 0 & -250 & 0 \end{bmatrix}^T,$
$m_b^5 = \begin{bmatrix} 0 & 0 & 150 & 0 \end{bmatrix}^T, P_b^i = diag\left([200,100,200,100]^T\right).$

The length of the training data $L_1 = 1000$, and the length of testing data $L_2 = 100$. The real trajectories used for training and testing are distinct, i.e., the training and testing data are from different datasets but follow the same motion model. For the targets' motion process, the testing target moves in M2 at $20 \sim 40$ s, M3 at $60 \sim 80$ s, and M1 at other moments. Figure 2 displays the trajectory of the test targets. Furthermore, the efficacy of the proposed approach is evaluated by averaging 500 independent Monte Carlo (MC) experiments.

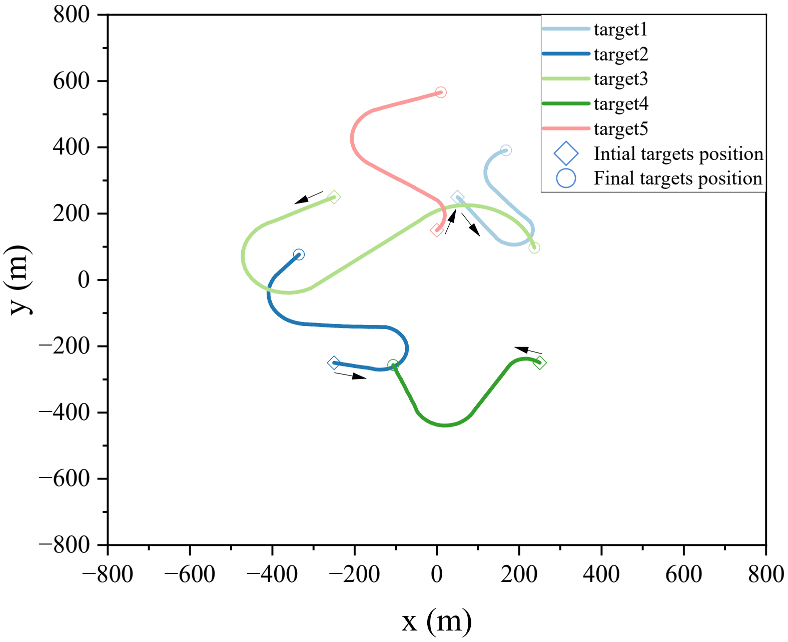


Figure 2. True trajectory of maneuvering targets .

Figures 3 and 4 illustrate the cardinality estimation and cardinality estimation error with detection probability $p_d = 0.95$, respectively. The results in Figure 3 indicate that both the GP-PHD, VSMM-PHD, and MM-PHD filters outperform the single-model PHD filter in terms of performance and stability of cardinality estimation. When there is a significant model mismatch, the cardinality estimate error of the single model PHD filter increases observably and, therefore, cannot accurately estimate the actual number of targets in the environment. In contrast, the GP-PHD, VSMM-PHD, and MM-PHD filters show similar performance in MMTT cardinality estimation. A closer analysis reveals that the GP-PHD filter outperforms the others in target cardinality estimation. The histogram with error bars for cardinality estimation errors of several algorithms is shown in Figure 4, which is intended to visually and accurately present the mean value of cardinality estimation errors and their fluctuations of each algorithm so as to provide strong support for the comparison of different algorithms in terms of cardinality estimation accuracy. In this figure, the height of the histogram represents the mean value of the cardinality estimation error, while the error bars serve as a quantitative indicator of the fluctuation or uncertainty of the data, and the longer the error bars are, the greater the fluctuation or uncertainty of the data. After careful analysis, it can be clearly observed that the proposed GP-PHD algorithm performs the best in terms of the mean value of cardinality estimation error with the smallest mean value, which fully proves the excellent performance of the algorithm in the task of multi-maneuvering target cardinality estimation. Meanwhile, the VSMM-PHD and MM-PHD filters exhibit similar performance in cardinality estimation, but the VSMM-PHD filter shows a slight advantage in the mean value of cardinality estimation error. In contrast, the other single-model algorithms perform poorly in terms of both the mean cardinality estimation error and the range of fluctuation, which are large and fluctuate significantly, demonstrating significant shortcomings in cardinality estimation performance.

This phenomenon further underscores the accuracy and stability of the GP-PHD algorithm for cardinality estimation of multi-maneuvering targets in complex environments.

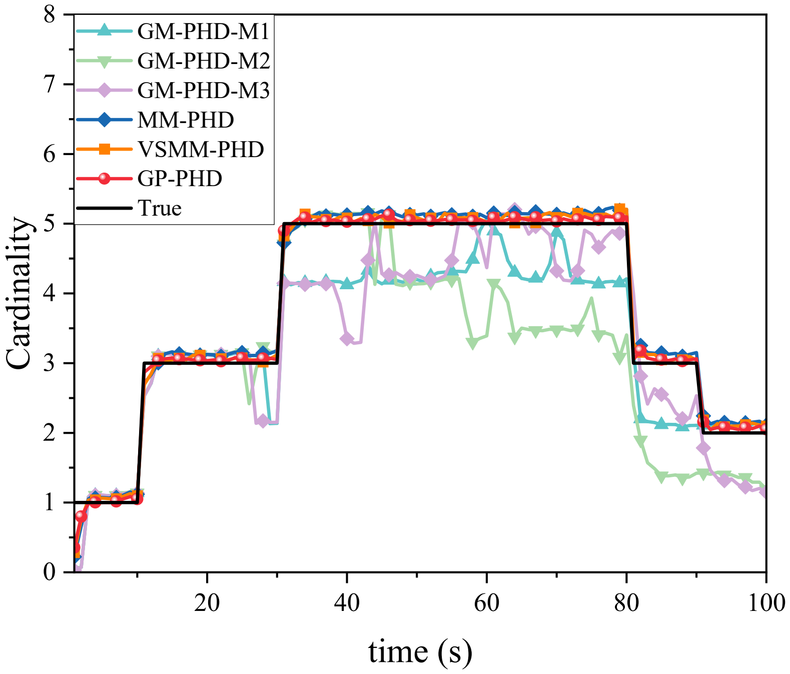


Figure 3. Cardinality estimation comparison under $p_d = 0.95$.

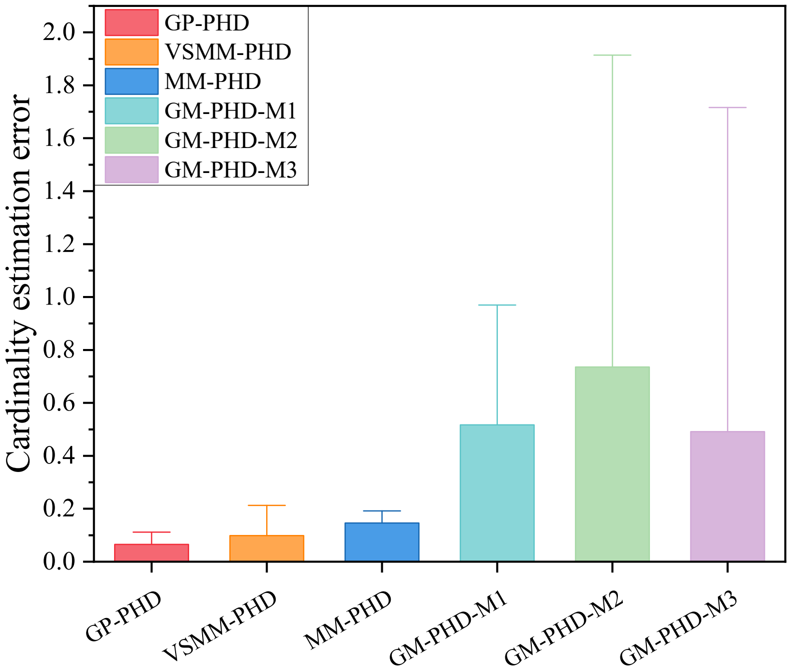


Figure 4. Cardinality estimation error comparison under $p_d = 0.95$.

Figures 5 and 6 show the GOSPA distance with detection probability $p_d = 0.95$, under various clutter conditions. Figure 5 demonstrates that the GP-PHD filter has an advantage over the VSMM-PHD, MM-PHD, and other single-model filters. By better adapting to changes in maneuvering target kinematics, the GP-PHD filter results in a smaller GOSPA distance. This is due to the GP's ability to model the target's dynamic properties flexibly, automatically learn the target's motion models, adapt to different motion trajectories, and thus reduce the position estimation error. In addition, the precise modeling of the target motion can also effectively cope with the uncertainty of the target potential, thus reducing the occurrence of missed targets and false detections. This property plays a crucial role in reducing the GOSPA distance. Therefore, the GP-PHD filter outperforms other algorithms in terms of GOSPA distance. For instance, during the 40–60 s and 60–80 s, when the motion model of the maneuvering target changes, the GP-PHD filter maintains stable estimation performance with minimal degradation in accuracy. In contrast, the VSMM-PHD and MM-PHD filters do not perform as well as the GP-PHD filter because the multi-model approach generally suffers from model assumption limitations and model switching lags. These issues lead to increased errors in target location and cardinality estimation, thereby adversely affecting the GOSPA distance. Furthermore, when a single-model PHD filter is used for estimation, significant estimation errors are often observed due to the mismatch between the model and the actual target motion. Figure 6 illustrates the average GOSPA distance under varying clutter conditions. The average GOSPA distance for all algorithms tends to increase as the clutter density increases. However, the average GOSPA distance of the GP-PHD filter is less sensitive to the clutter density, maintaining the best estimation performance across all conditions. This further highlights the advantages of the GP-PHD filter in MMTT and its strong adaptability to complex environments.

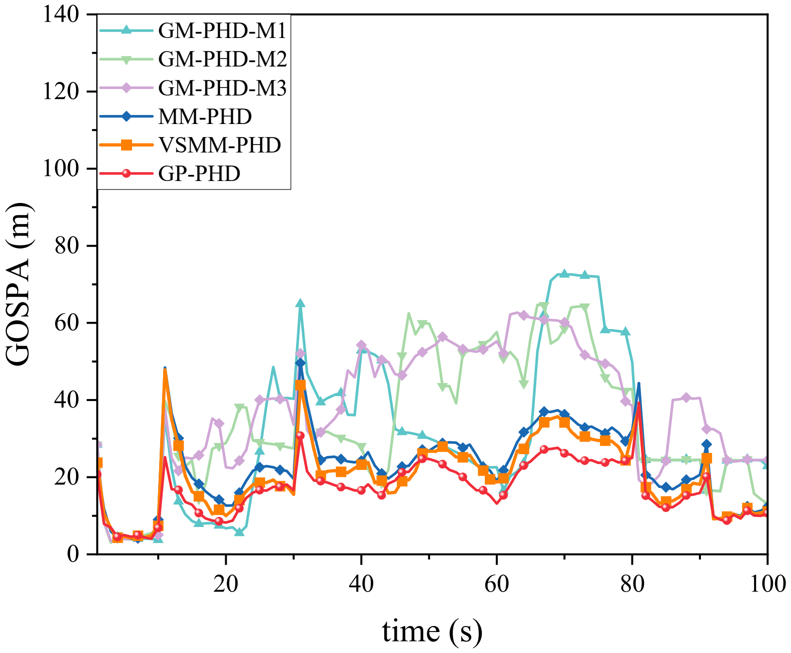


Figure 5. GOSPA distance under $p_d = 0.95$.

To thoroughly assess the performance of the proposed algorithm in a low-signal-to-noise ratio (SNR) environment, Figure 7 demonstrates the average GOSPA distance of the algorithm under different settings of the measurement noise covariance. An increase in the measurement noise covariance matrix R_t, a key parameter affecting the SNR, leads to a reduction in SNR. It can be observed through Figure 7 that the GP-PHD filter exhibits the smallest GOSPA distance in each noise level test, highlighting its significant advantage in target tracking accuracy and robustness to noise interference. This advantage stems from the GP filter's non-parametric modeling capability, which not only effectively learns the features of the target model but also adapts to the unknown characteristics of the noise covariance. Meanwhile, the VSMM-PHD and MM-PHD filters perform acceptably under initial low-noise conditions. Still, the GOSPA distance of these two filters increases rapidly with the growth in the measurement noise covariance, indicating a significant deficiency in their adaptability in high-noise environments. The performance degradation of the other single-model PHD filters is more significant in the presence of increased noise, underscoring the limitations of the single-model algorithm in terms of flexibility and estimation accuracy.

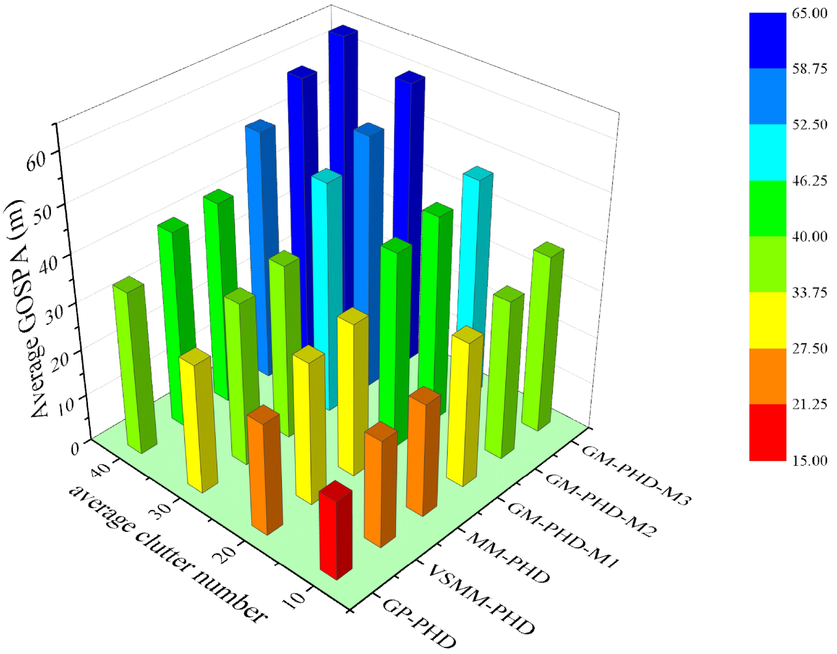


Figure 6. Average GOSPA distance under different clutter number under $p_d = 0.95$.

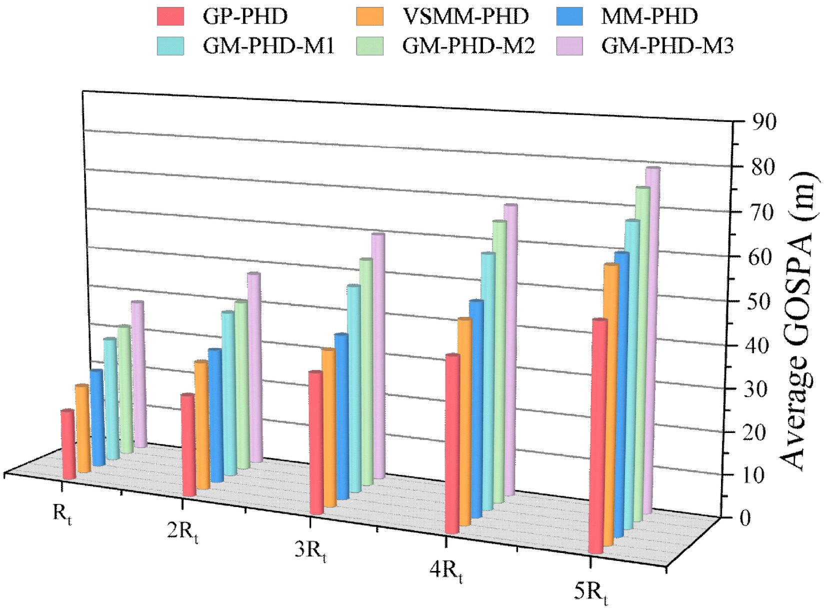


Figure 7. Average GOSPA distance under different R_t under $p_d = 0.95$.

Figures 8–10 evaluate the tracking performance of different algorithms with a detection probability of 0.7. Figures 8 and 9 show that a lower detection probability significantly affects the cardinality estimation of multi-maneuvering targets, with all algorithms exhibiting some bias. However, the cardinality estimation of the GP-PHD filter remains closer to the actual situation. In contrast, the VSMM-PHD and MM-PHD filters show more significant deviations, while the other single-model methods deviate even more. Figure 9 further illustrates this phenomenon using cardinality estimation error statistics. Despite the impact of low detection probability, the GP-PHD filter maintains better robustness in cardinality estimation and outperforms traditional MD algorithms. Figure 10 compares the

GOSPA distance and shows that the proposed GP-PHD filter outperforms both MM-PHD and single-model PHD filters. This also highlights that the GP-PHD filter is beneficial in MMTT estimate. The GP-PHD filter demonstrates superior performance by maintaining a lower GOSPA distance even under challenging conditions with low detection probability.

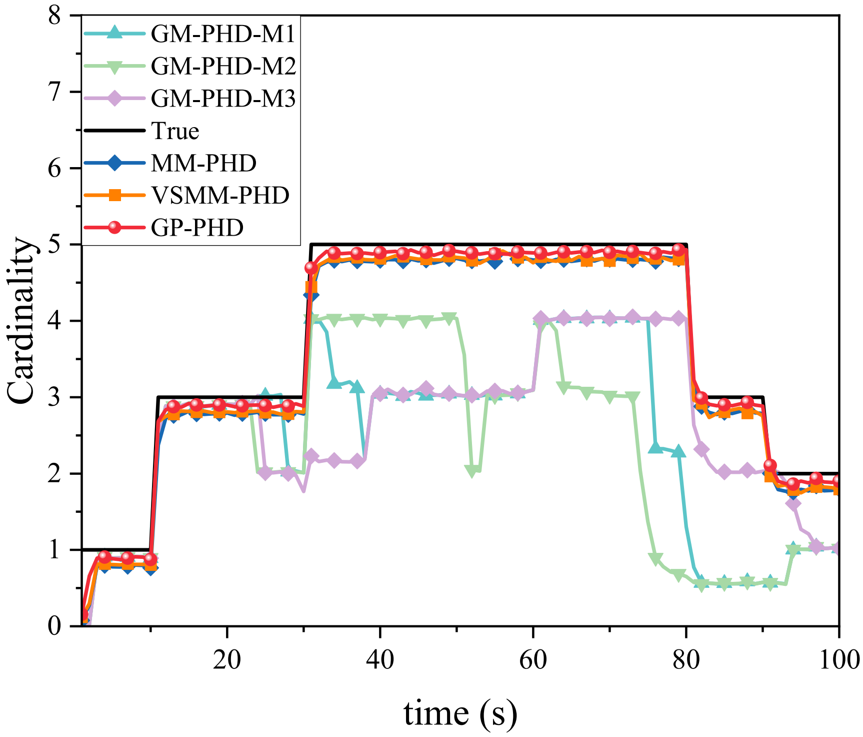


Figure 8. Cardinality estimation comparison under $p_d = 0.7$.

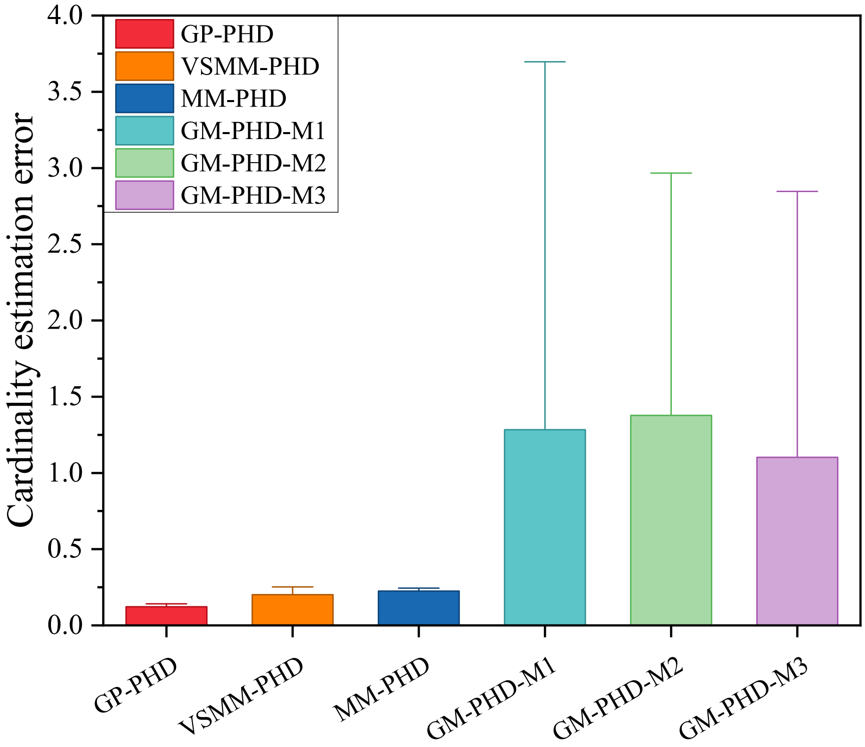


Figure 9. Cardinality estimation error comparison under $p_d = 0.7$.

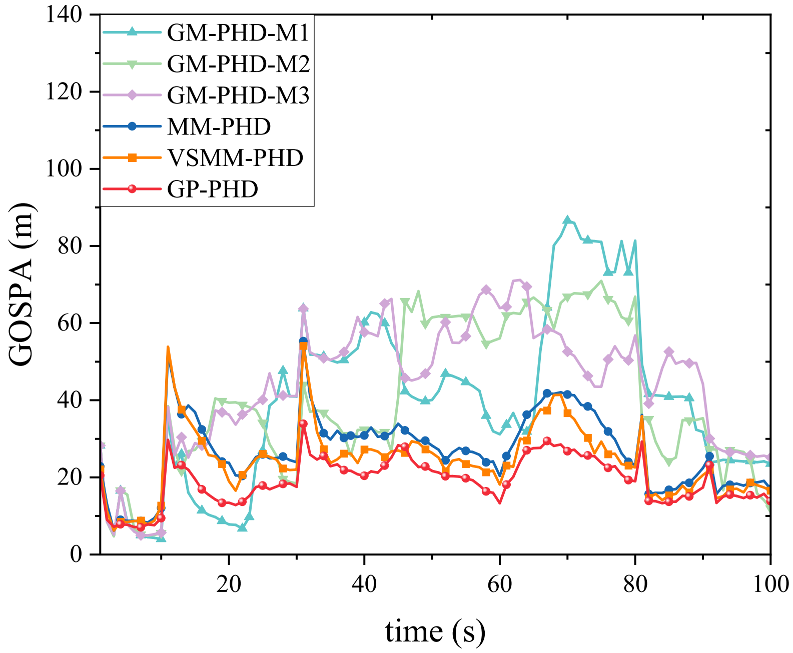


Figure 10. GOSPA distance under $p_d = 0.7$.

Table 1 presents the average GOSPA distance of various filtering algorithms for 500 MC experiments at a detection probability of 0.7 under different clutter conditions. As the amount of clutter increases, the average GOSPA distance for all filters increases accordingly. However, the proposed GP-PHD filter exhibits a low average statistical error in these scenarios, highlighting its superiority in estimating multi-maneuvering target motion states when facing uncertain motion and observation models. In contrast, the MD MM-PHD filter performs slightly worse than the GP-PHD filter algorithm, while the other three single-model PHD filters perform poorly in low detection probability scenarios due to mismatched motion models. This difference shows that the GP-PHD filter maintains robust performance even under challenging conditions with low detection probability and high clutter rates.

Table 1. Average GOSPA distance statistics in different λ_c.

	$\lambda_c = 10$	$\lambda_c = 20$	$\lambda_c = 30$	$\lambda_c = 40$
GP-PHD	**18.81**	**28.82**	**39.19**	**53.52**
VSMM-PHD	24.26	36.43	47.57	64.93
MM-PHD	27.03	38.68	50.13	67.65
GM-PHD-M1	39.46	49.49	62.61	74.69
GM-PHD-M2	39.85	52.84	64.05	77.04
GM-PHD-M3	45.07	54.24	68.35	79.02

(2) Scenario 2: A more complex MMTT environment is designed to further validate the effectiveness of the proposed approach. In this experimental setup, the maneuverability of the targets is significantly increased, imposing higher demands on the estimation performance of the MTT algorithms. The targets' motion models still include M1, M2, and M3, but the turning rates of M2 and M3 have significantly changed with $\theta = -12\ °/\text{s}$ and $\theta = 12\ °/\text{s}$. This complex environment makes the trajectories of targets more diverse and uncertain, which poses greater challenges to the adaptability and robustness of tracking algorithms. Through this setup, the performance of the GP-PHD filter in highly dynamic and complex environments can be comprehensively evaluated and compared with other traditional MD algorithms. In addition, a GM of the form is employed, as well as the target birth model

$$\gamma_t(x) = \sum_{i=1}^{5} w_b^i N\left(x; m_b^i, P_b^i\right) \tag{58}$$

with $w_b^i = 0.1$ and
$m_b^1 = \left[\begin{array}{cccc} 50 & 0 & 250 & 0 \end{array}\right]^T, m_b^2 = \left[\begin{array}{cccc} -250 & 0 & -250 & 0 \end{array}\right]^T,$
$m_b^3 = \left[\begin{array}{cccc} -250 & 0 & 250 & 0 \end{array}\right]^T, m_b^4 = \left[\begin{array}{cccc} 250 & 0 & -250 & 0 \end{array}\right]^T,$
$m_b^5 = \left[\begin{array}{cccc} -100 & 0 & -100 & 0 \end{array}\right]^T.$

The remaining of the multi-target motion and tracking environment parameters are set as in Scenario 1. The testing targets move in M2 during 10 $\sim$ 30 s and 41 $\sim$ 60 s, M3 during 31 $\sim$ 40 s and 61 $\sim$ 90 s, and M1 during the other intervals. The efficacy of the proposed approach is further validated through the aggregating 500 independent MC experiments. In Scenario 2, the actual trajectory used for testing is shown in Figure 11. As can be seen in the figure, the maneuverability of the targets has significantly increased due to changes in their turning rates. The intense maneuver introduces more significant uncertainty, which poses a more substantial challenge for tracking moving targets.

Figure 12 compares the cardinality estimation for MMTT in a highly dynamic scenario. It is observed that the high maneuverability of the target movements significantly influences the cardinality estimation of multiple targets. The GP-PHD, VSMM-PHD, and MM-PHD filters exhibit varying degrees of deviation in their cardinality estimation. However, the GP-PHD filter, with its ability to learn motion models, better adapts to different maneuvering variations and outperforms both the VSMM-PHD and MM-PHD in multi-target cardinality estimation. Other single-model approaches generally fail to account for such maneuvering variations and, in most cases, do not accurately estimate the cardinality of multiple targets.

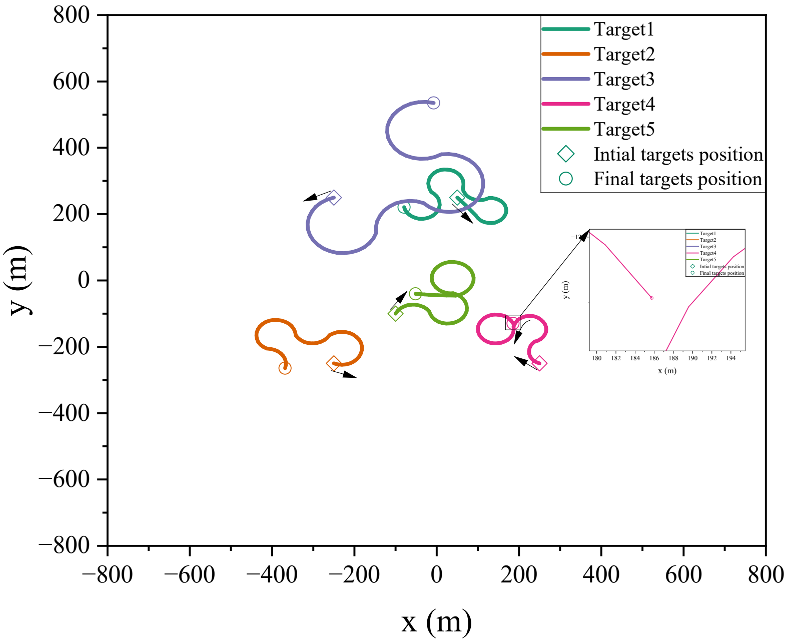


Figure 11. True trajectory of maneuvering targets.

Figure 13 further elucidates the differences between the algorithms using the cardinality estimation error statistics. The results indicate that although the cardinality estimation error statistics of the GP-PHD, VSMM-PHD, and MM-PHD filters exhibit similar performance, notable differences still exist. Compared to the VSMM-PHD and MM-PHD filters, the GP-PHD filter demonstrates smaller mean and median of the error statistics of cardinality estimation, highlighting its higher stability and accuracy in multi-target cardinality estimation. For the VSMM-PHD and MM-PHD filters, it is observed that there is no significant difference between the two in terms of cardinality estimation error, with the VSMM-PHD filter exhibiting a slight advantage. Other single-model filters exhibit issues such as scattered data, high variability, and numerous outliers, which render them inadequate for such a highly dynamic environment. The performance illustrated in Figures 12 and 13 underscores the robustness and adaptability of the GP-PHD filter in tracking highly maneuverable targets. The GP-PHD filter's ability to learn and adapt to different motion models ensures a more accurate and reliable cardinality estimate, even in challenging scenarios with significant target maneuverability.

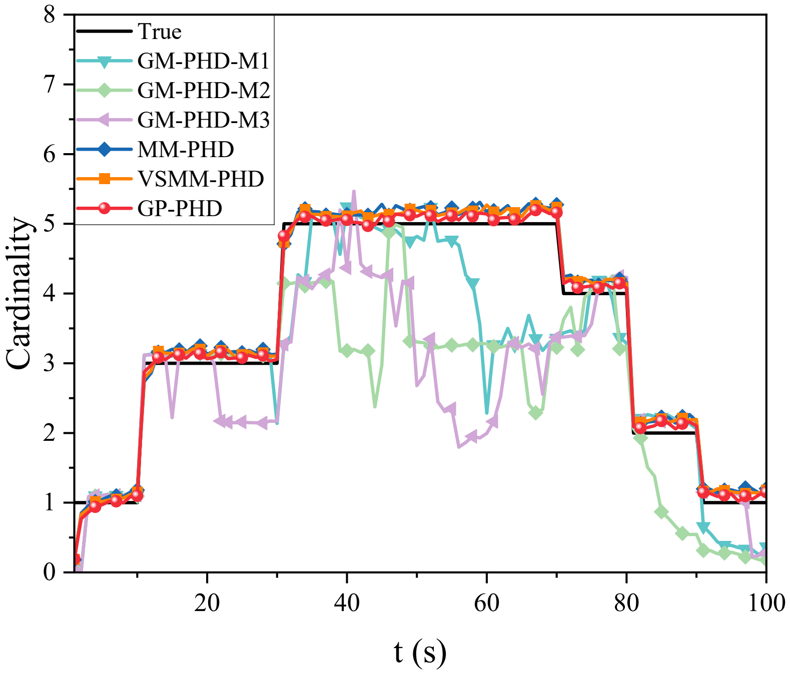


Figure 12. Cardinality estimation comparison under $p_d = 0.95$.

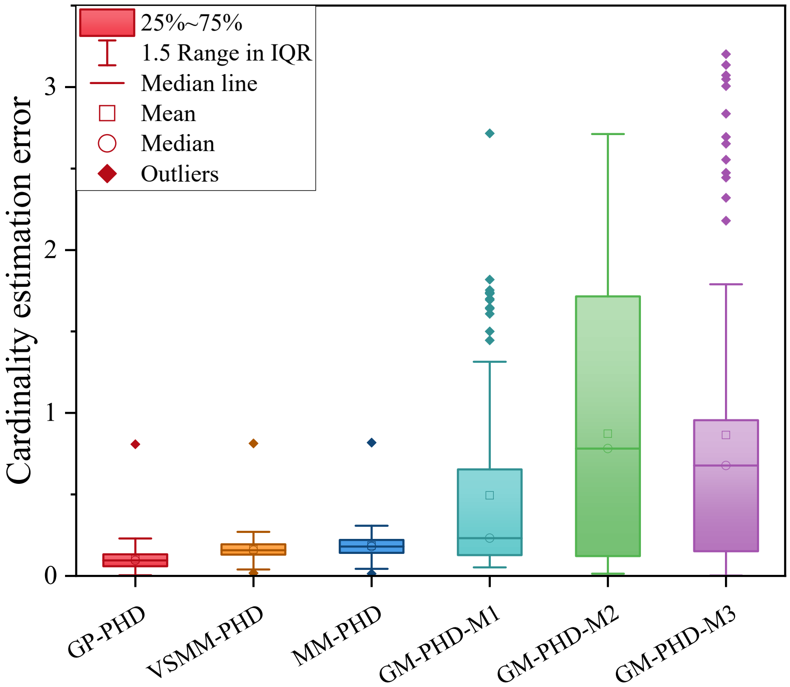


Figure 13. Cardinality estimateion error comparison under $p_d = 0.95$.

Figures 14 and 15 present the GOSPA distance and the average GOSPA distance for MMTT. Figure 14 shows that the GP-PHD, VSMM-PHD, and MM-PHD filters exhibit smaller GOSPA distance than other single-model filters, indicating higher accuracy in estimating target positions, missed detections, and false alarms. Variations in target states lead to fluctuations in GOSPA distance, as observed in periods such as 40 $\sim$ 50 s and 50 $\sim$ 70 s, where changes in target motion states and increased target counts result in significant increases in GOSPA distance. Notably, the GP-PHD filter shows a more stable GOSPA distance variation and is less sensitive to environmental changes than the other filters.

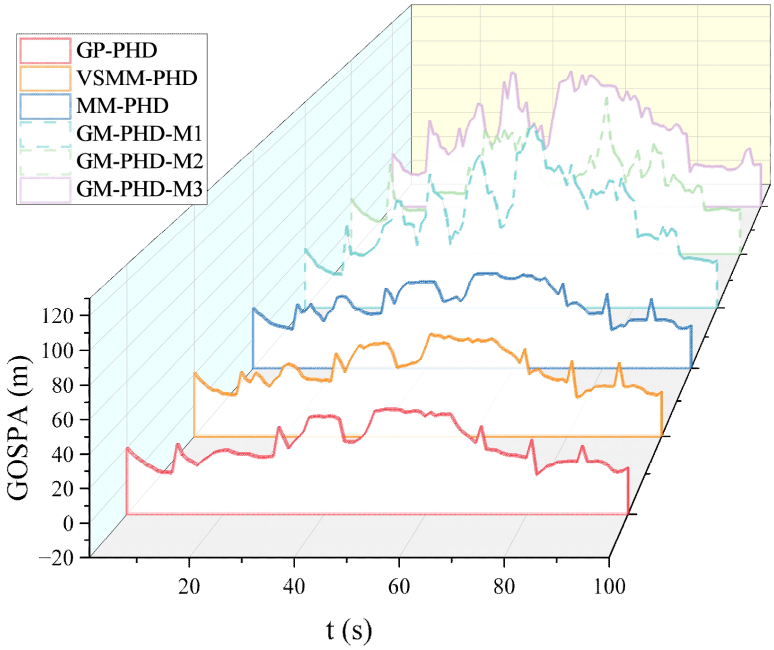


Figure 14. GOSPA distance under $p_d = 0.95$.

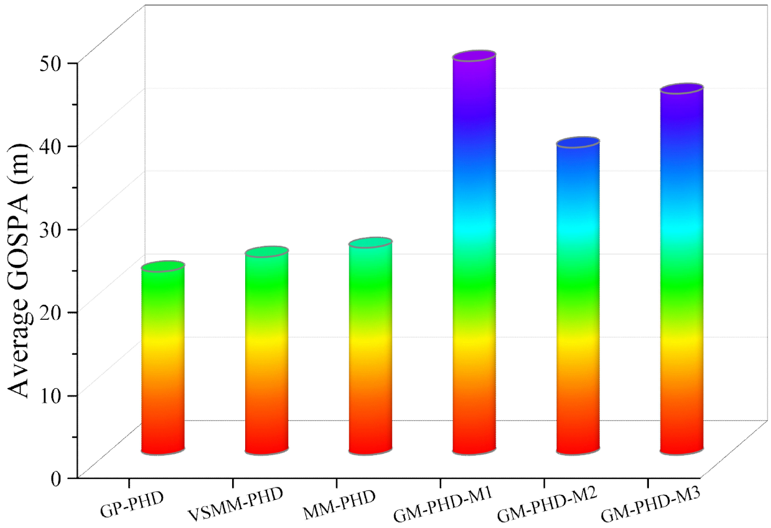


Figure 15. Average GOSPA distance under $p_d = 0.95$.

Figure 15 displays the average GOSPA distance, with the GP-PHD filter exhibiting the smallest average GOSPA distance, further confirming its superiority in MMTT. These results highlight the robustness and adaptability of the GP-PHD filter in complex scenarios. Compared to traditional methods, the GP-PHD filter can estimate MMTT states more accurately and achieve a smaller GOSPA distance, thereby underscoring its effectiveness.

Overall, the GP-PHD filter maintains a smaller GOSPA distance even under significant changes in target motion, demonstrating its superiority in handling dynamic and complex environments. It can adapt to various target motion models while ensuring precise tracking, greatly enhancing the potential application of the GP-PHD filter.

Table 2 presents the average GOSPA distance for different detection probability conditions. The table shows that as detection probability decreases, the estimated performance of both GP-PHD and other filters shows a declining trend. However, the performance of the GP-PHD filter consistently outperforms that of VSMM-PHD, MM-PHD, and single-model PHD filters. This advantage is particularly important in real-world applications, where environmental factors can cause fluctuations in detection probability, making it essential to reliably and accurately track targets under diverse and challenging conditions. The GP-PHD filter maintains higher tracking accuracy even at low detection probabilities, indicating its adaptability and robustness in highly uncertain environments. In contrast, VSMM-PHD, MM-PHD, and single-model PHD filters exhibit noticeable performance degradation under low detection probability conditions and struggle to track multiple maneuvering targets reliably. This further underscores the advantage of the GP-PHD filter in MMTT applications, especially in dynamic and uncertain target motion and observation models.

Table 2. Average GOSPA distance statistics in different p_d.

	$p_d = 0.95$	$p_d = 0.85$	$p_d = 0.8$	$p_d = 0.75$	$p_d = 0.7$
GP-PHD	**22.03**	**23.87**	**25.63**	**27.72**	**30.64**
VSMM-PHD	23.15	24.04	26.49	28.93	32.75
MM-PHD	23.62	25.31	27.92	30.22	34.93
GM-PHD-M1	42.51	44.86	47.14	49.81	52.39
GM-PHD-M2	33.81	35.63	36.74	39.23	41.98
GM-PHD-M3	37.88	40.62	44.17	46.31	49.46

Pruning plays a crucial role in the proposed algorithm, and it largely determines the computational efficiency of the algorithm. Figure 16 deeply analyzes the impact of pruning on the performance of the algorithm in practical applications by analyzing the execution time. Both cases are implemented in the MATLAB (2021b) environment on a computer equipped with a 3.9 GHz CPU (Inter Core i3-7100) (Santa Clara, CA, USA). From the comparison of the data in the figure, it is obvious that the algorithm with pruning algorithm maintains a stable and efficient performance at all time points. In contrast, the running time of the unpruned algorithm increases sharply with the increase in the number of Gaussian components. This trend significantly reduces the applicability of the algorithm in practical scenarios. Therefore, introducing the pruning step is of great significance in ensuring the real-time and practicality of the algorithm.

(3) Summary: Through a series of simulation experiments in different scenarios, the proposed GP-PHD filter demonstrates superior robustness when compared to the traditional tracking methods, and it effectively adapts to the complexity and uncertainty of the target motion in the tracking scenarios more effectively. This advantage is primarily reflected in the following aspects: (1) The GP-PHD filter can adaptively capture the dynamic behavior of the target without reliance on specific model assumptions, due to the modeling flexibility of GP. This characteristic makes the method particularly suitable for handling complex and variable target motion scenarios and can effectively address sudden maneuvers and nonlinear motion trajectories of the target. (2) The GP model has the ability to deal with similarities and differences in target motion, which makes the GP-PHD filter able to accurately distinguish and track the trajectories of different targets in complex scenarios when facing multi-target interactions. (3) The GP model can effectively deal with the uncertainty and noise in the observation, and the filter can still maintain excellent tracking performance even under conditions of low detection probability or serious clutter interference. Therefore, the GP-PHD filter shows its unique advantages and wide applica-

bility in dealing with the challenges in the field of MMTT and offers an effective solution to the MMTT problem in complex environments.

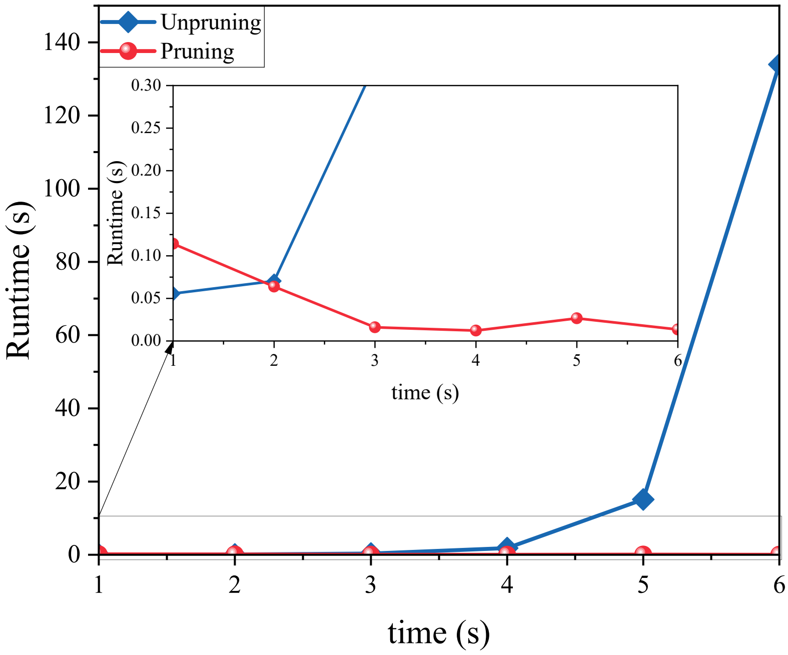


Figure 16. Runtime comparison.

6. Conclusions

This study proposes a model-free GP-PHD filter to effectively address the challenges of target motion and observation model uncertainty in MMTT. The filter leverages the GP to learn the unknown maneuvering targets' motion and observation models and employs the 'cubature sampling' method to create GM approximation of the posterior intensity for the next time step. Additionally, the study provides a concrete implementation of this filter utilizing the GM method. The experiments compare the performance of the GP-PHD filter with the VSMM-PHD, MM-PHD, and single-model GM-PHD filters. The results demonstrate that the GP-PHD filter exhibits robust adaptability in learning uncertain target motion and observation models, outperforming the VSMM, MM, and single-model methods. These advantages make the GP-PHD filter a preferred solution for MMTT. Its ability to learn and adapt to various target motion models ensures more accurate and reliable tracking in complex scenarios with highly maneuverable targets. In future research, applying the GP-PHD filter in multi-extended target tracking will be further explored for more challenging tracking tasks.

Author Contributions: Z.Z. and H.C. contributed to the study's conception and design. Formal analysis, Z.Z. and H.C.; experimental analysis, Z.Z.; writing-original draft preparation, Z.Z.; review and suggestion, Z.Z. and H.C.; funding acquisition, H.C. All authors have read and agreed to the published version of the manuscript.

Funding: This work is supported by the National Natural Science Foundation of China (62163023, 61873116), the Industrial Support Project of Education Department of Gansu Province (2021CYZC–02), the Special Funds Project for Civil-Military Integration Development in Gansu Province in 2023 and the Key Talent Project of Gansu Province in 2024.

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: The data are contained within this article.

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

References

1. He, Y. Mission-Driven Autonomous Perception and Fusion Based on UAV Swarm. *Chin. J. Aeronaut.* **2020**, *33*, 2831–2834. [CrossRef]
2. Fan, C.; Song, C.; Wang, M. Small Video Satellites Visual Tracking Control for Arbitrary Maneuvering Targets. In Proceedings of the 2022 IEEE International Conference on Robotics and Biomimetics (ROBIO); Xishuangbanna, China, 5–9 December 2022; pp. 951–957.
3. Yu, J.; Shi, Z.; Dong, X.; Li, Q.; Lv, J.; Ren, Z. Impact Time Consensus Cooperative Guidance Against the Maneuvering Target: Theory and Experiment. *IEEE Trans. Aerosp. Electron. Syst.* **2023**, *59*, 4590–4603. [CrossRef]
4. Zhang, H.; Liu, W.; Zong, B.; Shi, J.; Xie, J. An Efficient Power Allocation Strategy for Maneuvering Target Tracking in Cognitive MIMO Radar. *IEEE Trans. Signal Process.* **2021**, *69*, 1591–1602. [CrossRef]
5. Lucena de Souza, M.; Gaspar Guimarães, A.; Leite Pinto, E. A Novel Algorithm for Tracking a Maneuvering Target in Clutter. *Digital Signal Process.* **2022**, *126*, 103481. [CrossRef]
6. Wang, S.; Jiang, F.; Zhang, B.; Ma, R.; Hao, Q. Development of UAV-Based Target Tracking and Recognition Systems. *IEEE Trans. Intell. Transp. Syst.* **2020**, *21*, 3409–3422. [CrossRef]
7. Rong Li, X.; Jilkov, V.P. Survey of Maneuvering Target Tracking. Part I. Dynamic Models. *IEEE Trans. Aerosp. Electron. Syst.* **2003**, *39*, 1333–1364. [CrossRef]
8. Li, W.; Jia, Y. An Information Theoretic Approach to Interacting Multiple Model Estimation. *IEEE Trans. Aerosp. Electron. Syst.* **2015**, *51*, 1811–1825. [CrossRef]
9. Xu, H.; Pan, Q.; Xu, H.; Quan, Y. Adaptive IMM Smoothing Algorithms for Jumping Markov System with Mismatched Measurement Noise Covariance Matrix. *IEEE Trans. Aerosp. Electron. Syst.* **2024**, *60*, 5467–5480. [CrossRef]
10. Xu, W.; Xiao, J.; Xu, D.; Wang, H.; Cao, J. An Adaptive IMM Algorithm for a PD Radar with Improved Maneuvering Target Tracking Performance. *Remote Sens.* **2024**, *16*, 1051. [CrossRef]
11. Li, X.; Lu, B.; Li, Y.; Lu, X.; Jin, H. Adaptive Interacting Multiple Model for Underwater Maneuvering Target Tracking with One-Step Randomly Delayed Measurements. *Ocean Eng.* **2023**, *280*, 114933. [CrossRef]
12. Han, B.; Huang, H.; Lei, L.; Huang, C.; Zhang, Z. An Improved IMM Algorithm Based on STSRCKF for Maneuvering Target Tracking. *IEEE Access* **2019**, *7*, 57795–57804. [CrossRef]
13. Lu, C.; Feng, W.; Li, W.; Zhang, Y.; Guo, Y. An Adaptive IMM Filter for Jump Markov Systems with Inaccurate Noise Covariances in the Presence of Missing Measurements. *Digital Signal Process.* **2022**, *127*, 103529. [CrossRef]
14. Kirubarajan, T.; Bar-Shalom, Y.; Pattipati, K.R.; Kadar, I. Ground Target Tracking with Variable Structure IMM Estimator. *IEEE Trans. Aerosp. Electron. Syst.* **2000**, *36*, 26–46. [CrossRef]
15. Pasha, S.A.; Vo, B.-N.; Tuan, H.D.; Ma, W.-K. A Gaussian Mixture PHD Filter for Jump Markov System Models. *IEEE Trans. Aerosp. Electron. Syst.* **2009**, *45*, 919–936. [CrossRef]
16. Sithiravel, R.; McDonald, M.; Balaji, B.; Kirubarajan, T. Multiple Model Spline Probability Hypothesis Density Filter. *IEEE Trans. Aerosp. Electron. Syst.* **2016**, *52*, 1210–1226. [CrossRef]
17. Georgescu, R.; Willett, P. The multiple model CPHD tracker. *IEEE Trans. Signal Process.* **2012**, *60*, 1741–1751. [CrossRef]
18. Dong, P.; Jing, Z.; Li, M.; Pan, H. The variable structure multiple model GM-PHD filter based on likely model set algorithm. In Proceedings of the 2016 19th International Conference on Information Fusion (FUSION), Heidelberg, Germany, 5–8 July 2016; IEEE: New York, NY, USA, 2016; pp. 2289–2295.
19. Dunne, D.; Kirubarajan, T. Multiple Model Multi-Bernoulli Filters for Manoeuvering Targets. *IEEE Trans. Aerosp. Electron. Syst.* **2013**, *49*, 2679–2692. [CrossRef]
20. Reuter, S.; Scheel, A.; Dietmayer, K. The multiple model labeled multi-Bernoulli filter. In Proceedings of the 2015 18th International Conference on Information Fusion (FUSION), Washington, DC, USA, 6–9 July 2015; IEEE: New York, NY, USA, 2015; pp. 1574–1580.
21. Punchihewa, Y.; Vo, B.N.; Vo, B.T. A generalized labeled multi-Bernoulli filter for maneuvering targets. In Proceedings of the 2016 19th International Conference on Information Fusion (FUSION), Heidelberg, Germany, 5–8 July 2016; IEEE: New York, NY, USA, 2016; pp. 980–986.
22. Seeger, M. Gaussian Processes for Machine Learning. *Int. J. Neural Syst.* **2004**, *14*, 69–106. [CrossRef] [PubMed]
23. Ko, J.; Fox, D. GP-BayesFilters: Bayesian Filtering Using Gaussian Process Prediction and Observation Models. *Auton Robot.* **2009**, *27*, 75–90. [CrossRef]
24. Kowsari, E.; Safarinejadian, B. Applying GP-EKF and GP-SCKF for Non-Linear State Estimation and Fault Detection in a Continuous Stirred-Tank Reactor System. *Trans. Inst. Meas. Control* **2017**, *39*, 1486–1496. [CrossRef]
25. Todescato, M.; Carron, A.; Carli, R.; Pillonetto, G.; Schenato, L. Efficient Spatio-Temporal Gaussian Regression via Kalman Filtering. *Automatica* **2020**, *118*, 109032. [CrossRef]
26. Lee, T. Adaptive learning Kalman filter with Gaussian process. In Proceedings of the 2020 American Control Conference (ACC), Denver, CO, USA, 1–3 July 2020; IEEE: New York, NY, USA, 2020; pp. 4442–4447.
27. Aftab, W.; Mihaylova, L. A Gaussian Process Regression Approach for Point Target Tracking. In Proceedings of the 2019 22th International Conference on Information Fusion (FUSION), Ottawa, ON, Canada, 2–5 July 2019; IEEE: New York, NY, USA, 2019; pp. 1–8.

28. Aftab, W.; Mihaylova, L. A Learning Gaussian Process Approach for Maneuvering Target Tracking and Smoothing. *IEEE Trans. Aerosp. Electron. Syst.* **2021**, *57*, 278–292. [CrossRef]
29. Sun, M.; Davies, M.E.; Proudler, I.K.; Hopgood, J.R. A Gaussian Process Regression based Dynamical Models Learning Algorithm for Target Tracking. *arXiv* **2022**, arXiv:2211.14162.
30. Hu, Z.; Li, T. A Particle Bernoulli Filter Based on Gaussian Process Learning for Maneuvering Target Tracking. In Proceedings of the 2022 30th European Signal Processing Conference (EUSIPCO), Belgrade, Serbia, 29 August–2 September 2022; IEEE: New York, NY, USA, 2022; pp. 777–781.
31. Guo, Q.; Teng, L.; Yin, T.; Guo, Y.; Wu, X.; Song, W. Hybrid-Driven Gaussian Process Online Learning for Highly Maneuvering Multi-Target Tracking. *Front. Inform. Technol. Electron. Eng.* **2023**, *24*, 1647–1656. [CrossRef]
32. Mahler, R.P.S. Multitarget Bayes Filtering via First-Order Multitarget Moments. *IEEE Trans. Aerosp. Electron. Syst.* **2003**, *39*, 1152–1178. [CrossRef]
33. Arasaratnam, I.; Haykin, S. Cubature Kalman Filters. *IEEE Trans. Autom. Control* **2009**, *54*, 1254–1269. [CrossRef]
34. Vo, B.N.; VO, B.T.; Clark, D. Bayesian multiple target filtering using random finite sets. In *Integrated Tracking, Classification, and Sensor Management*; Wiley: Hoboken, NJ, USA, 2013; pp. 75–126. [CrossRef]
35. Rahmathullah, A.S.; García-Fernández, Á.F.; Svensson, L. Generalized Optimal Sub-Pattern Assignment Metric. In Proceedings of the 2017 20th International Conference on Information Fusion (FUSION), Xi'an, China, 10–13 July 2017; IEEE: New York, NY, USA, 2017; pp. 1–8.

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.

厚尾噪声条件下的学生t泊松多伯努利混合滤波器

赵子文[1], 陈 辉[1†], 连 峰[2], 张光华[2]

(1. 兰州理工大学 电气工程与信息工程学院, 甘肃 兰州 730050; 2. 西安交通大学 自动化科学与工程学院, 陕西 西安 710049)

摘要: 针对运动过程和观测过程均受到异常噪声干扰的复杂不确定性多目标跟踪问题, 本文创新性地提出了学生t混合泊松多伯努利混合滤波器. 首先, 直接将广域分布的异常噪声特性建模为学生t分布. 随后, 将泊松多伯努利混合滤波器的泊松点过程(PPP)和多伯努利混合(MBM)的概率密度参数合理的近似为学生t混合形式. 其次, 基于多目标概率密度的学生t混合模型, 详细推导了泊松多伯努利混合滤波器学生t混合共轭先验形式, 建立了学生t混合泊松多伯努利混合的闭式递推框架. 最后, 通过带显著拖尾分布特性的过程噪声和量测噪声共同干扰的复杂多目标跟踪仿真实验, 验证了所提滤波算法的有效性.

关键词: 随机有限集; 多目标跟踪; 学生t混合; 厚尾噪声; 泊松多伯努利混合

引用格式: 赵子文, 陈辉, 连峰, 等. 厚尾噪声条件下的学生t泊松多伯努利混合滤波器. 控制理论与应用, 2024, 41(9): 1598 – 1609

DOI: 10.7641/CTA.2023.20625

A Student's t Poisson multi-Bernoulli mixture filter in the presence of heavy-tailed noise

ZHAO Zi-wen[1], CHEN Hui[1†], LIAN Feng[2], ZHANG Guang-hua[2]

(1. School of Electrical and Information Engineering, Lanzhou University of Technology, Lanzhou Gansu 730050, China;
2. School of Automation Science and Engineering, Xi'an Jiaotong University, Xi'an Shaanxi 710049, China)

Abstract: Aiming at the complex uncertainty multi-target tracking where both the motion process and observation process are disturbed by anomalous noise, this paper innovatively proposes a Student's t mixture Poisson multi-Bernoulli mixture filter. First, the anomalous noise characteristics of the wide-area distribution are directly modeled as the Student's t distribution. Subsequently, the probability density parameters of the Poisson point process (PPP) and the multi-Bernoulli mixture (MBM) of the Poisson multi-Bernoulli mixture filter are reasonably approximated by the Student's t mixture form. Moreover, based on the Student's t mixture model which approximates the multi-target probability density, the Student's t mixture conjugate prior form of Poisson multi-Bernoulli mixture filter is derived in detail and a closed-form recursive framework of Student's t mixture Poisson multi-Bernoulli mixture is established. Finally, the effectiveness of the proposed filtering algorithm is verified by complex multi-target tracking simulation experiments under the joint interference of process noise and measurement noise with significant trailing distribution characteristics.

Key words: random finite set; multi-target tracking; Student's t mixture; heavy-tail noise; Poisson multi-Bernoulli mixture

Citation: ZHAO Ziwen, CHEN Hui, LIAN Feng, et al. A Student's t Poisson multi-Bernoulli mixture filter in the presence of heavy-tailed noise. *Control Theory & Applications*, 2024, 41(9): 1598 – 1609

1 引言

多目标跟踪(multi-target tracking, MTT)是雷达信号处理中的一个关键性问题, 例如, 在自动驾驶、环境监测、空中交通管制和导航制导等领域中发挥着核心作用[1–4]. 多目标跟踪的一个实质性困难是目标和量测之间的数据关联, 数据关联是一个时变最优决策问

收稿日期: 2022–07–13; 录用日期: 2023–08–05.

†通信作者. E-mail: huich78@hotmail.com; Tel.: +86 931-2973506.

本文责任编委: 胡德文.

国家自然科学基金项目(62163023, 61873116, 62173266, 62103318), 甘肃省教育厅产业支撑计划项目(2021CYZC–02), 甘肃省教育厅优秀研究生"创新之星"项目(2022CXZX–468), 2023年甘肃省军民融合发展专项资金项目, 2024年甘肃省重点人才项目资助.

Supported by the National Natural Science Foundation of China (62163023, 61873116, 62173266, 62103318), the Industrial Support Project of Education Department of Gansu Province (2021CYZC–02), the outstanding graduate Innovation Star project of Education Department of Gansu Province (2022CXZX–468), the Special Funds Project for Civil-Military Integration Development in Gansu Province in 2023 and the Key Talent Project of Gansu Province in 2024.

题, 对其决策的质量在很大程度上决定了多目标跟踪问题的求解质量. 在已有的研究算法中, 联合概率数据关联 (joint probabilistic data association, JPDA)[5]、多假设跟踪 (multiple hypothesis tracking, MHT)[6]和随机有限集[7−8]是多目标跟踪最为重要的解决方案. 特别的, 由于随机有限集–多目标跟踪基于完备、严密的随机有限集概率统计方法, 在解决大量非传统安全防御体系与信号处理新模式的多源多目标跟踪融合问题上具有天生的优势, 所以对该问题的研究成为近年来多目标跟踪研究的主流. 在有限集统计(finite set statistics, FISST)理论框架下, 一些基于随机有限集理论的多目标滤波器被开发出来, 可从其航迹的身份属性上将它们分为两类: 无标签多目标滤波器和标签多目标滤波器. 其中, 基于随机有限集的无标签多目标滤波器主要由概率假设密度滤波器(probability hypothesis density, PHD)[9]、势概率假设密度滤波器(cardinalised PHD, CPHD)[10]和多伯努利(multi-Bernouolli, MB)滤波器[11−12]组成. 而基于随机有限集的标签多目标滤波器主要包括标签多伯努利滤波器(labeled MB, LMB)[13]、广义标签多伯努利滤波器(generalized LMB, GLMB)[14−15]、标签多伯努利混合滤波器(LMB mixture, LMBM)[16]和边缘化δ-GLMB滤波器(marginalized δ-GLMB, Mδ-GLMB)[17]. 这些方法的一个关键特征是赋予了多目标跟踪问题更为统一严密且易于处理和管理的优越属性, 这也是避免直接应对数据关联的巧妙近似化以后的结果, 即根据量测和目标的对应关系进行了进一步的广义推理. 特别的, 对于标准的多目标似然函数, 广义标签多伯努利和标签多伯努利混合密度已经被证明是共轭的[16]. 此外, 以上所有提到的随机有限集滤波器都可以通过采用高斯混合 (Gaussian mixture, GM)[18−19] 或者序贯蒙特卡罗 (sequential Monte Carlo, SMC)[20−22]技术来有效实现, 对多目标跟踪问题提供了标准的求解模式.

最近, 有学者提出一种新的无标签多目标滤波器, 即泊松多伯努利混合 (Poisson multi-Bernoulli mixture, PMBM) 滤波器, 它是泊松随机有限集和多伯努利混合 (multi-Bernouolli mixture, MBM) 随机有限集的卷积[23−24]. 与其它基于随机有限集的无标签滤波器相比, 泊松多伯努利混合滤波器的一个独特且重要的特征是共轭先验, 这意味着在多目标跟踪模型中, 如果多目标密度在初始时刻是共轭先验的形式, 那么所有后续的预测和更新的多目标密度也将是共轭先验的形式[25−26]. 共轭先验的重要性在于, 在假设模型下, 可以知道理论上密度的精确形式, 这与直接计算多目标预测和更新相比, 共轭先验提供了一种更为方便的方法. 因此, 泊松多伯努利混合滤波器在许多应用中也被广泛的采用[27−33]. 多目标跟踪滤波方法发展至今, 泊松多伯努利混合滤波器在诸多有效的随机有限集–多目标跟踪方法中脱颖而出, 且更加有效, 原因在于: 1) 泊松多伯努利混合更能有效的表示未检测目标(潜在目标)的信息; 2) 泊松多伯努利混合密度伴随着更少的全局假设, 相较于广义标签多伯努利有更加高效的递推结构[24, 27−28].

为建立通用和易处理的系统噪声的随机建模方式, 基于随机有限集的多目标滤波器都将过程噪声和量测噪声假设为高斯分布, 但在许多实际跟踪应用中, 这些噪声可能会频繁的出现异常值, 直观上, 这些异常值可能是偏离了它们本应该存在位置的样本, 在分布特性上呈现出明显的拖尾现象. 在多目标跟踪中, 噪声的异常值可能由系统环境中突然的扰动或传感器的偶发故障等引起, 而目标运动的随机机动也会带来过程噪声的异常. 拖尾或厚尾噪声的存在使得多目标跟踪系统的随机建模过程出现很大的偏差, 严重影响了多目标滤波器的跟踪估计性能. 为了使滤波器更适应厚尾噪声环境, 有学者提出了高斯假设下通过重加权同时处理厚尾过程噪声和量测噪声的方法, 但在高斯假设下, 该方法处理厚尾非高斯噪声的能力受限[34−35]. 粒子滤波器可以将过程噪声和量测噪声近似为任意分布, 但是在高维系统中, 粒子滤波的计算非常复杂[36]. 高斯和滤波器(Gaussian sum filter, GSF)也是一种处理厚尾噪声的方法, 但除了实现过程中分量扩张带来的计算复杂度, 该方法也难以利用有限的高斯和灵活适应拖尾噪声的异常变化[37]. 最近几年, 学生t分布由于能够更为直观和合理的表征异常噪声的拖尾分布特性而被高度关注. 与高斯噪声相比, 厚尾噪声分布在离均值较远的地方的可能性增大, 而学生t分布能较好的包容这些噪声异常值, 进而推动滤波器对厚尾噪声干扰表现出更好的鲁棒性. 对此, 国内外一些学者已经提出了一些基于学生t分布的滤波器和平滑器[38−40]. 在这些滤波器和平滑器中, 噪声被建模为学生t分布, 而为保持递推滤波的封闭性近似, 后验分布近似为高斯分布. 此外, 有学者也提出了另一种完全基于学生t分布的滤波器, 该滤波器将噪声建模为学生t分布, 并将后验概率密度也近似为学生t分布, 以此来获得在厚尾过程噪声和量测噪声条件下的跟踪滤波问题的闭式解[41−43]. 所以, 结合最新的泊松多伯努利混合滤波器, 利用其研究异常噪声条件下基于学生t分布建模的复杂多目标跟踪问题, 形成更高效的多目标密度(包括未检测到的目标)的联合递推, 具有非常重大的现实意义.

本文的主要贡献是考虑到异常噪声的影响下含未检测目标的复杂多目标跟踪问题, 提出学生 t 混合 (Student's t mixture, STM)泊松多伯努利混合滤波器. 首先, 为了能更好的表征具有厚尾过程噪声和量测噪声的多目标跟踪系统, 将目标的状态和量测模型建模为学生t分布, 其能对异常噪声体现更好的包容性. 其

次, 本文所提算法的重点是如何构造STM-PMBM共轭先验, 来为多目标概率密度函数的封闭求解提供便利. 根据泊松多伯努利混合滤波器的特性, 多目标预测密度中未检测到的目标由泊松分布近似, 检测到的目标由多伯努利分布近似, 将它们的概率参数均近似表示为学生t混合分布, 多目标先验和后验密度视为具有学生t混合空间分布的泊松多伯努利混合, 并进一步研究如何应对运动过程和观测过程都受异常噪声干扰的复杂多目标跟踪问题. 根据泊松多伯努利混合滤波器的递推框架, 在学生t混合近似多目标密度分量的条件下, 详细推导并提出了高度复杂不确定条件下泊松多伯努利混合滤波器的学生t混合实现. 最后通过仿真实验, 验证了所提算法的有效性.

2 背景

2.1 多目标贝叶斯滤波

假设$k-1$时刻的多目标转移函数和多目标状态分别为$T_{k|k-1}(\cdot|\cdot)$和$f_{k-1}(\boldsymbol{X}|\boldsymbol{Z}_{1:k-1})$, 其中$\boldsymbol{Z}_{1:k-1}$是从1到$k-1$时刻接收到的有限组量测值的集合, 定义为$\boldsymbol{Z}_{1:k-1}=(\boldsymbol{Z}_1,\cdots,\boldsymbol{Z}_{k-1})$, 根据Chapman-Kolmogorov方程[7]可以得到k时刻的多目标预测方程为

$$\begin{aligned}&f_{k|k-1}(\boldsymbol{X}_k \mid \boldsymbol{Z}_{1:k-1})=\\&\int T_{k|k-1}(\boldsymbol{X}_k \mid \boldsymbol{X}_{k|k-1})\times\\&f_{k-1}(\boldsymbol{X}_{k|k-1} \mid \boldsymbol{Z}_{1:k-1})\delta\boldsymbol{X}_{k|k-1},\end{aligned}\tag{1}$$

其中: 多目标的状态$\boldsymbol{X}_k=\{\boldsymbol{x}_{k,1},\cdots,\boldsymbol{x}_{k,N(k)}\}\subset\mathcal{X}$, $\boldsymbol{x}_{k,N(k)}$表示k时刻目标的状态, $N(k)$表示k时刻目标的数目, $\mathcal{X}$表示状态空间. 目标状态$\boldsymbol{x}_{k,N(k)}=[\boldsymbol{x}^i_{k,p}\ \boldsymbol{x}^i_{k,\mathrm{v}}]^{\mathrm{T}}$包含位置$\boldsymbol{x}^i_{k,p}$和速度$\boldsymbol{x}^i_{k,\mathrm{v}}$. $T_{k|k-1}(\boldsymbol{X}_k|\boldsymbol{X}_{k|k-1})$表示多目标转移密度[7].

当k时刻接收到一个新的量测集合$\boldsymbol{Z}_k$时, 根据贝叶斯规则, k时刻的多目标更新方程为

$$\begin{aligned}&f_k(\boldsymbol{X}_k \mid \boldsymbol{Z}_{1:k})=\\&\frac{L_k(\boldsymbol{Z}_k \mid \boldsymbol{X}_k)f_{k|k-1}(\boldsymbol{X}_k \mid \boldsymbol{Z}_{1:k-1})}{\int L_k(\boldsymbol{Z}_k \mid \boldsymbol{X}_k)f_{k|k-1}(\boldsymbol{X}_k \mid \boldsymbol{Z}_{1:k-1})\delta\boldsymbol{X}_k},\end{aligned}\tag{2}$$

其中: 多目标量测$\boldsymbol{Z}_k=\{\boldsymbol{z}^1_k,\cdots,\boldsymbol{z}^{M(k)}_k\}\subset\mathcal{Z}$, $\boldsymbol{z}^{M(k)}_k$表示k时刻目标量测, $M(k)$表示k时刻量测的数目, $\mathcal{Z}$表示量测空间. $L_k(\boldsymbol{Z}_k|\boldsymbol{X}_k)$是多目标量测似然, 其标准形式为$L_k(\boldsymbol{Z}_k|\boldsymbol{X}_k)\triangleq\frac{\delta\beta_{k+1}}{\delta Z}(\varnothing|\boldsymbol{X})=[\frac{\delta\beta_{k+1}}{\delta Z}(\mathcal{Z}|X)]_{\mathcal{Z}=\varnothing}$, 另一种不同形式的等价表达式为$l(\boldsymbol{Z}|\boldsymbol{X})=\mathrm{e}^{-\lambda_c}\sum\limits_{\boldsymbol{Z}^c\uplus\boldsymbol{Z}_1\cdots\uplus\boldsymbol{Z}_n=\boldsymbol{Z}}[c(\cdot)]^{Z_c}\prod\limits_{i=1}^{n}\hat{l}(\boldsymbol{Z}_i|\boldsymbol{x}_i)$.

上式中包含的积分是集合积分[7], 定义为

$$\begin{aligned}&\int f(\boldsymbol{X})\delta\boldsymbol{X}=\\&\sum_{n=0}^{\infty}\frac{1}{n!}\int_{\boldsymbol{X}^n}f(\{\boldsymbol{x}_1,\cdots,\boldsymbol{x}_n\})\mathrm{d}\boldsymbol{x}_1\cdots\mathrm{d}\boldsymbol{x}_n,\end{aligned}$$

为了便于表示, 将$f_k(\boldsymbol{X}_k|\boldsymbol{Z}_{1:k})$缩写为$f_k(\boldsymbol{X})$.

2.2 泊松多伯努利混合随机有限集

定义1和定义2分别给出了未检测到的目标和潜在检测到的目标的定义, 可以帮助更好的理解泊松多伯努利混合随机有限集.

定义 1 在k时刻存在但从未被检测到的目标定义为未检测到的目标, 记为$\boldsymbol{X}^{\mathrm{u}}_k$.

定义 2 对于新的量测, 其可以产生于第1次检测到的新目标, 也可以是对应于另一个先前检测到的目标或者杂波. 考虑到它可能存在或者不存在, 可称它为潜在检测到的目标, 记为$\boldsymbol{X}^{\mathrm{d}}_k$.

多目标状态随机有限集$\boldsymbol{X}_k$定义为在量测集合为$\boldsymbol{Z}_{1:k}$的条件下, 随机有限集$\boldsymbol{X}^{\mathrm{u}}_k$和$\boldsymbol{X}^{\mathrm{d}}_k$的不相交并集, 即, $\boldsymbol{X}^{\mathrm{u}}_k\cup\boldsymbol{X}^{\mathrm{d}}_k=\boldsymbol{X}_k$且$\boldsymbol{X}^{\mathrm{u}}_k\cap\boldsymbol{X}^{\mathrm{d}}_k=\varnothing$. 因此, 泊松多伯努利混合随机有限集的后验密度可以通过有限集统计理论卷积定义为

$$f_k(\boldsymbol{X})=\sum_{\boldsymbol{Y}\subseteq\boldsymbol{X}}f^{\mathrm{p}}_k(\boldsymbol{Y})f^{\mathrm{mbm}}_k(\boldsymbol{X}-\boldsymbol{Y}),\tag{3}$$

其中$f^{\mathrm{p}}_k(\cdot)$是一个泊松密度, 定义为

$$f^{\mathrm{p}}_k(\boldsymbol{X})=\mathrm{e}^{-\int\mu_k(\boldsymbol{x})\mathrm{d}\boldsymbol{x}}[\mu_k(\cdot)]^{\boldsymbol{X}},\tag{4}$$

其中$\mu_k(\boldsymbol{x})$是强度函数. 对于一个非空集合$\boldsymbol{X}$和函数$f(\boldsymbol{x})$, $[f(\boldsymbol{x})]^{\boldsymbol{X}}=\prod\limits_{\boldsymbol{x}\in\boldsymbol{X}}f(\boldsymbol{x})$, 且$[f(\cdot)]^{\varnothing}=1$. 此外, $f^{\mathrm{mbm}}_k(\cdot)$是一个多伯努利混合密度, 定义为

$$f^{\mathrm{mbm}}_k(\boldsymbol{X})\propto\sum_{j\in\boldsymbol{I}}\sum_{\boldsymbol{X}_1\uplus\cdots\uplus X_n=\boldsymbol{X}}\prod_{i=1}^{n}w^{j,i}f^{j,i}(\boldsymbol{X}_i),\tag{5}$$

其中: $\propto$表示比例符号, $w^{j,i}$和$f^{j,i}(X_i)$定义为假设权值和第j个全局假设中第i个伯努利密度, $f^{j,i}(\boldsymbol{X}_i)$定义为

$$f^{j,i}(\boldsymbol{X}_i)=\begin{cases}1-r^{j,i}, & \boldsymbol{X}_i=\varnothing,\\ r^{j,i}f^{j,i}(x), & \boldsymbol{X}_i=\{x\},\\ 0, & \text{其他},\end{cases}\tag{6}$$

其中: $r^{j,i}$表示存在概率; 如果状态密度存在, $f^{j,i}(\boldsymbol{x})$表示状态密度. 可以看出, 泊松多伯努利混合随机有限集是多个多伯努利的多目标密度的归一化加权和, 参数化表示为$\{w^{j,i},r^{j,i},f^{j,i}(\boldsymbol{x})\}_{j\in I,i\in I^j}$, 其中$I$是多伯努利的索引集合(也叫做全局假设集合). 特别的, 当只有一个全局假设, 即$|I|=1$时, 多伯努利混合随机有限集退化为多伯努利随机有限集

$$f^{\mathrm{mb}}_k(\boldsymbol{X})\propto\sum_{\boldsymbol{X}_1\uplus\cdots\uplus\boldsymbol{X}_n=\boldsymbol{X}}\prod_{i=1}^{n}w^{1,i}f^{1,i}(\boldsymbol{X}_i).\tag{7}$$

2.3 泊松多伯努利混合递推

2.3.1 预测步

定理1推导出了泊松多伯努利混合滤波器的预测步骤[23]. 该定理表明: 泊松点过程(Poisson point process, PPP)分量(4)的预测遵循标准概率假设密度的预测步骤, 并且以与多目标多伯努利(multiple target multi-Bernoulli, MeMBer)滤波器等效的方式独立预测多伯努利轨迹(不包括新生目标的出现, 因为使用的是泊松点过程新生模型而不是多伯努利新生模型)[9, 19].

定理 1 假设$k-1$时刻的后验分布形式如式(3)–(6)所示. 那么k时刻的预测分布有相同的形式

$$\lambda^{\mathrm{u}}_{k|k-1}(\boldsymbol{x}) = \lambda^{\mathrm{b}}(\boldsymbol{x}) + \int f_{k|k-1}(\boldsymbol{x}\,|\boldsymbol{x}')P^{\mathrm{s}}(\boldsymbol{x}')\lambda^{\mathrm{u}}_{k-1|k-1}(\boldsymbol{x}')\mathrm{d}\boldsymbol{x}', \tag{8}$$

$$n_{k|k-1} = n_{k-1|k-1},\; h^i_{k|k-1} = h^i_{k-1|k-1}\,\forall i, \tag{9}$$

$$w^{i,a^i}_{k|k-1} = w^{i,a^i}_{k-1|k-1}\,\forall i, a^i, \tag{10}$$

$$r^{i,a^i}_{k|k-1} = r^{i,a^i}_{k-1|k-1}\langle f^{i,a^i}_{k-1|k-1}, P^{\mathrm{s}}\rangle\,\forall i, a^i, \tag{11}$$

$$f^{i,a^i}_{k|k-1}(\boldsymbol{x}) = \frac{\int f_{k|k-1}(\boldsymbol{x}|\boldsymbol{x}')P^{\mathrm{s}}(\boldsymbol{x}')f^{i,a^i}_{k-1|k-1}(\boldsymbol{x}')}{\langle f^{i,a^i}_{k-1|k-1}, P^{\mathrm{s}}\rangle}\,\forall i, a^i, \tag{12}$$

其中: $\lambda^{\mathrm{b}}(\boldsymbol{x})$表示泊松点过程新生强度, $f_{k|k-1}(x|x')$表示目标转移概率密度函数, $P^{\mathrm{s}}(\boldsymbol{x})$表示目标的存活概率, $\lambda^{\mathrm{u}}_{k-1|k-1}(x')$表示上一时刻泊松点过程密度, $n_{k|k-1}$表示预测轨迹的数目, $h^i_{k|k-1}$表示轨迹i中的目标假设的数目, $w^{i,a^i}_{k|k-1}$表示假设权值, $r^{i,a^i}_{k|k-1}$表示存在概率, $f^{i,a^i}_{k|k-1}(\boldsymbol{x})$表示概率密度函数.

2.3.2 量测更新步

定理 2 推导出了泊松多伯努利混合滤波器的量测更新步骤[23]. 泊松点过程强度的更新方程如式(13)所示, 这与不带量测的概率假设密度更新相同[9, 19].

定理 2 假设预测分布的形式如式(8)–(12)所示. 那么更新分布 (通过量测集合$\boldsymbol{Z}_k = \{\boldsymbol{z}^1_k, \cdots, \boldsymbol{z}^{m_k}_k\}$更新)有相同的形式, 其中$P^{\mathrm{d}}(\boldsymbol{x})$表示目标的检测概率, 更新轨迹的数目$n_{k|k} = n_{k|k-1} + m_k$,

$$\lambda^{\mathrm{u}}_{k|k}(\boldsymbol{x}) = \left\{1 - P^{\mathrm{d}}(\boldsymbol{x})\right\}\lambda^{\mathrm{u}}_{k|k-1}(\boldsymbol{x},). \tag{13}$$

对于从先前持续存在轨迹 ($i \in \{1, \cdots, n_{k|k-1}\}$) 的一个假设, 其可能包含在先前每个假设组合中, 也可能是漏检, 或者是使用m_k个新的量测值的其中一个进行更新, 因此假设的数目变为$h^i_{k|k} = h^i_{k|k-1}(1 + m_k)$. 对于漏检假设($i \in \{1, \cdots, n_{k|k-1}\}, a^i \in \{1, \cdots, h_{k|k-1}\}$)

$$w^{i,a^i}_{k|k} = w^{i,a^i}_{k|k-1}(1 - r^{i,a^i}_{k|k-1} + r^{i,a^i}_{k|k-1}\langle f^{i,a^i}_{k|k-1}, 1 - P^{\mathrm{d}}\rangle), \tag{14}$$

$$r^{i,a^i}_{k|k} = \frac{r^{i,a^i}_{k|k-1}\langle f^{i,a^i}_{k|k-1}, 1 - P^{\mathrm{d}}\rangle}{1 - r^{i,a^i}_{k|k-1} + r^{i,a^i}_{k|k-1}\langle f^{i,a^i}_{k|k-1}, 1 - P^{\mathrm{d}}\rangle}, \tag{15}$$

$$f^{i,a^i}_{k|k-1}(\boldsymbol{x}) = \frac{\left\{1 - P^{\mathrm{d}}(\boldsymbol{x})\right\}f^{i,a^i}_{k|k-1}(\boldsymbol{x})}{\langle f^{i,a^i}_{k|k-1}, 1 - P^{\mathrm{d}}\rangle}. \tag{16}$$

对于当前轨迹假设的更新 ($i \in \{1, \cdots, n_{k|k-1}\}$, $a^i = \tilde{a}^i + h^i_{k|k-1}j$, $\tilde{a}^i \in \{1, \cdots, h^i_{k|k-1}\}$, $j \in \{1, \cdots, m_k\}$, 即用量测$\boldsymbol{z}^j_k$更新先前假设$\tilde{a}^i$)

$$w^{i,a^i}_{k|k} = w^{i,\tilde{a}^i}_{k|k-1}r^{i,\tilde{a}^i}_{k|k-1}\langle f^{i,\tilde{a}^i}_{k|k-1}, f(\boldsymbol{z}^j_k\,|\cdot)P^{\mathrm{d}}\rangle, \tag{17}$$

$$r^{i,a^i}_{k|k} = 1, \tag{18}$$

$$f^{i,a^i}_{k|k}(\boldsymbol{x}) = \frac{f(\boldsymbol{z}^j_k\,|\boldsymbol{x})P^{\mathrm{d}}(\boldsymbol{x})f^{i,\tilde{a}^i}_{k|k-1}(\boldsymbol{x})}{\langle f^{i,\tilde{a}^i}_{k|k-1}, f(\boldsymbol{z}^j_k\,|\cdot)P^{\mathrm{d}}\rangle}. \tag{19}$$

最后, 对于新的轨迹, $i \in \left\{n_{k|k-1} + j\right\}$(即新轨迹从量测$\boldsymbol{z}^j_t$开始)

$$h^i_{k|k} = 2, \tag{20}$$

$$w^{i,1}_{k|k} = 1,\; r^{i,1}_{k|k} = 0, \tag{21}$$

$$w^{i,2}_{k|k} = \lambda^{\mathrm{fa}}(\boldsymbol{z}^j_k) + \langle\lambda^{\mathrm{u}}_{k|k-1}, f(\boldsymbol{z}^j_k|\cdot)P^{\mathrm{d}}\rangle, \tag{22}$$

$$r^{i,2}_{k|k} = \frac{\langle\lambda^{\mathrm{u}}_{k|k-1}, f(\boldsymbol{z}^j_k|\cdot)P^{\mathrm{d}}\rangle}{\lambda^{\mathrm{fa}}(\boldsymbol{z}^j_k) + \langle\lambda^{\mathrm{u}}_{k|k-1}, f(\boldsymbol{z}^j_k|\cdot)P^{\mathrm{d}}\rangle}, \tag{23}$$

$$f^{i,2}_{k|k}(\boldsymbol{x}) = \frac{f(\boldsymbol{z}^j_k|\boldsymbol{x})P^{\mathrm{d}}(\boldsymbol{x})\lambda^{\mathrm{u}}_{k|k-1}(\boldsymbol{x})}{\langle\lambda^{\mathrm{u}}_{k|k-1}, f(\boldsymbol{z}^j_k|\cdot)P^{\mathrm{d}}\rangle}, \tag{24}$$

其中: 式(20)表示有两个目标假设, 其中第一个假设包含了量测与另一轨迹相关联的情况, 因此, 新轨迹的存在概率为0, 如式(21)所示; $\lambda^{\mathrm{fa}}(\boldsymbol{x})$表示杂波量测泊松点过程的强度; a^i表示索引用于第i个目标的假设; $f(\boldsymbol{z}\,|\boldsymbol{x})$表示目标量测似然.

3 学生t混合实现

在本节中, 本文详细推导过程噪声和量测噪声均为厚尾噪声条件下基于学生t分布的泊松多伯努利混合滤波器. 首先, 将过程噪声和量测噪声均建模为学生t分布. 其次, 在每个时刻, 多目标滤波器的概率密度函数$f(\boldsymbol{x}_k\,|\boldsymbol{z}_{1:k})$同样服从学生t分布. 之后, 详细推导了泊松多伯努利混合滤波器的学生t混合实现算法.

3.1 系统建模

对于如下状态空间模型所示的线性随机系统:

$$\boldsymbol{x}_{k+1} = \boldsymbol{F}_k\boldsymbol{x}_k + \boldsymbol{w}_k, \tag{25}$$

$$\boldsymbol{z}_k = \boldsymbol{H}_k\boldsymbol{x}_k + \boldsymbol{v}_k, \tag{26}$$

其中: $\boldsymbol{x}_k \in \mathbb{R}^{d_x}$表示$k$时刻的状态, $\boldsymbol{z}_k \in \mathbb{R}^{d_z}$表示量测, $\boldsymbol{w}_k$和$\boldsymbol{v}_k$分别表示适当维度的过程噪声和量测噪声. 系统矩阵$\boldsymbol{F}_k$和$\boldsymbol{H}_k$是已知的, 且初始状态$\boldsymbol{x}_0$和噪声是边缘t分布. 由于过程噪声和量测噪声都存在异常值, 假设过程噪声和量测噪声向量具有厚尾分布, 因此将过程噪声和量测噪声模拟为学生t分布

$$p(\boldsymbol{w}_k) = \mathrm{St}(\boldsymbol{w}_k; 0, \boldsymbol{Q}_k, v_1), \tag{27}$$

$$p(\boldsymbol{v}_k) = \mathrm{St}(\boldsymbol{v}_k; 0, \boldsymbol{R}_k, v_2), \tag{28}$$

其中: $\mathrm{St}(\boldsymbol{x}; \boldsymbol{\mu}, \boldsymbol{\Sigma}, v)$表示均值为$\boldsymbol{\mu}$、尺度矩阵为$\boldsymbol{\Sigma}$、自由度参数为$v$的学生t概率密度函数; $\boldsymbol{Q}_k$和$\boldsymbol{R}_k$分别表示过程噪声和量测噪声的尺度矩阵; v_1和v_2分别表示过程噪声和量测噪声的自由度参数. 假设系统的初始状态$\boldsymbol{x}_0$服从均值向量为$\hat{\boldsymbol{x}}_{0|0}$, 尺度矩阵为$\boldsymbol{P}_{0|0}$, 自由度参数为$v_3$的学生t分布, 即

$$p(\boldsymbol{x}_0) = \mathrm{St}(\boldsymbol{x}_0; \hat{\boldsymbol{x}}_{0|0}, \boldsymbol{P}_{0|0}, v_3), \tag{29}$$

此外, 假设$\boldsymbol{x}_0$, $\boldsymbol{w}_k$, $\boldsymbol{v}_k$互不相关.

3.2 闭式递推

在所提出的STM-PMBM滤波器中, 假设杂波参数和目标的生成模型为已知先验, 类似于泊松多伯努利混合滤波器[23]. 然后通过将状态模型的学生t分布形式代入泊松多伯努利混合滤波器递推式, 可以直接推导出厚尾噪声条件下STM-PMBM滤波器的闭式解. 图1为该算法的流程图.

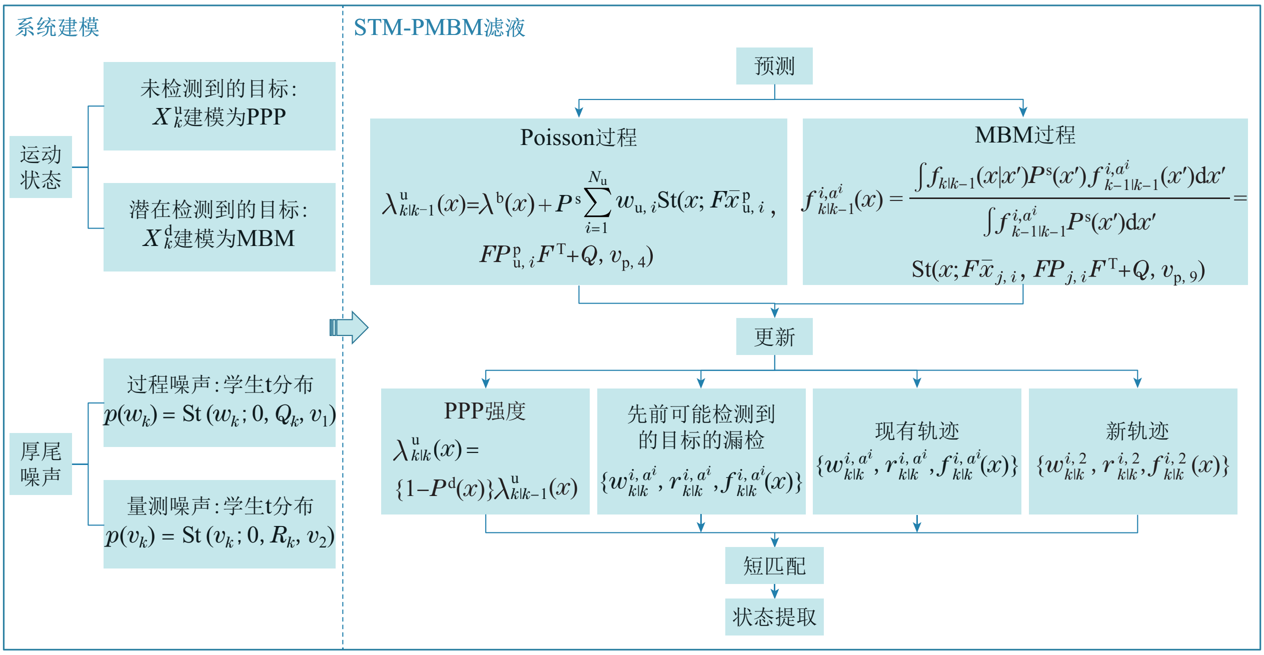

系统建模
运动状态
未检测到的目标: X_k^{u}建模为PPP
潜在检测到的目标: X_k^{d}建模为MBM
厚尾噪声
过程噪声: 学生t分布
$p(w_k) = \mathrm{St}\,(w_k; 0, Q_k, v_1)$
量测噪声: 学生t分布
$p(v_k) = \mathrm{St}\,(v_k; 0, R_k, v_2)$
STM-PMBM滤液
预测
Poisson过程
$\lambda^{\mathrm{u}}_{k|k-1}(x)=\lambda^{\mathrm{b}}(x)+P^{\mathrm{s}}\sum_{i=1}^{N_{\mathrm{u}}} w_{\mathrm{u},i}\mathrm{St}(x; F\bar{x}^{\mathrm{p}}_{\mathrm{u},i}, FP^{\mathrm{p}}_{\mathrm{u},i}F^{\mathrm{T}}+Q, v_{\mathrm{p},4})$
MBM过程
$f^{i,a^i}_{k|k-1}(x) = \frac{\int f_{k|k-1}(x|x')P^{\mathrm{s}}(x')f^{i,a^i}_{k-1|k-1}(x')\mathrm{d}x'}{\int f^{i,a^i}_{k-1|k-1}P^{\mathrm{s}}(x')\mathrm{d}x'} = \mathrm{St}(x; F\bar{x}_{j,i}, FP_{j,i}F^{\mathrm{T}}+Q, v_{\mathrm{p},9})$
更新
PPP强度
$\lambda^{\mathrm{u}}_{k|k}(x) = \{1-P^{\mathrm{d}}(x)\}\lambda^{\mathrm{u}}_{k|k-1}(x)$
先前可能检测到的目标的漏检
$\{w^{i,a^i}_{k|k}, r^{i,a^i}_{k|k}, f^{i,a^i}_{k|k}(x)\}$
现有轨迹
$\{w^{i,a^i}_{k|k}, r^{i,a^i}_{k|k}, f^{i,a^i}_{k|k}(x)\}$
新轨迹
$\{w^{i,2}_{k|k}, r^{i,2}_{k|k}, f^{i,2}_{k|k}(x)\}$
短匹配
状态提取

图 1 STM-PMBM滤波算法

Fig. 1 STM-PMBM filtering algorithm

3.2.1 预测步

假设多目标存活概率$P^{\mathrm{s}}(\boldsymbol{x})$是一个常值, $k-1$时刻泊松点过程强度, 泊松点过程新生强度, 目标状态转移方程分别为

$$\lambda^{\mathrm{u}}_{k-1|k-1}(\boldsymbol{x}') = \sum_{i=1}^{N_{\mathrm{u}}} w_{\mathrm{u},i}\mathrm{St}(\boldsymbol{x}'; \bar{\boldsymbol{x}}^{\mathrm{p}}_{\mathrm{u},i}, \boldsymbol{P}^{\mathrm{p}}_{\mathrm{u},i}, v_4), \tag{30}$$

$$\lambda^{\mathrm{b}}(\boldsymbol{x}) = \sum_{i=1}^{N_{\mathrm{b}}} w^{\mathrm{p}}_{\mathrm{b},i}\mathrm{St}(\boldsymbol{x}; \bar{\boldsymbol{x}}^{\mathrm{p}}_{\mathrm{b},i}, \boldsymbol{P}^{\mathrm{p}}_{\mathrm{b},i}, v_5), \tag{31}$$

$$f_{k|k-1}(\boldsymbol{x}\,|\boldsymbol{x}') = \mathrm{St}(\boldsymbol{x}; \boldsymbol{F}\boldsymbol{x}', Q, v_6), \tag{32}$$

其中N_{u}, N_{b}, $\bar{\boldsymbol{x}}^{\mathrm{p}}_{\mathrm{u},i}$, $\boldsymbol{P}^{\mathrm{p}}_{\mathrm{u},i}$, $\bar{\boldsymbol{x}}^{\mathrm{p}}_{\mathrm{b},i}$, $\boldsymbol{P}^{\mathrm{p}}_{\mathrm{b},i}$, $w_{\mathrm{u},i}$, $w^{\mathrm{p}}_{\mathrm{b},i}$是给定的模型参数. 将式(30)–(32)代入式(8)可得

$$\begin{gathered}\lambda^{\mathrm{u}}_{k|k-1}(\boldsymbol{x}) = \\ \lambda^{\mathrm{b}}(\boldsymbol{x}) + \int \mathrm{St}(\boldsymbol{x}; \boldsymbol{F}\boldsymbol{x}', \boldsymbol{Q}, v_6)P^{\mathrm{s}}(\boldsymbol{x}') \times \\ \sum_{i=1}^{N_{\mathrm{u}}} w_{\mathrm{u},i}\mathrm{St}(\boldsymbol{x}'; \bar{\boldsymbol{x}}^{\mathrm{p}}_{\mathrm{u},i}, \boldsymbol{P}^{\mathrm{p}}_{\mathrm{u},i}, v_4)\mathrm{d}\boldsymbol{x}'.\end{gathered} \tag{33}$$

为求解上式中的积分形式, 根据Chapman-Kolmogorov方程和Roth等人的学生t随机变量的仿射变换得到引理1[42–43].

引理 1 若$\boldsymbol{P}$和$\boldsymbol{Q}$均为正定矩阵, 则有

$$\begin{gathered}\int \mathrm{St}(\boldsymbol{x}; \boldsymbol{F}\boldsymbol{\xi}, \boldsymbol{Q}, v_7)\mathrm{St}(\boldsymbol{\xi}; \boldsymbol{m}, \boldsymbol{P}, v_8)\mathrm{d}\boldsymbol{\xi} = \\ \mathrm{St}(\boldsymbol{x}; \boldsymbol{F}\boldsymbol{m}, \boldsymbol{F}\boldsymbol{P}\boldsymbol{F}^{\mathrm{T}} + \boldsymbol{Q}, v_8).\end{gathered} \tag{34}$$

根据引理1, 式(33)的泊松点过程预测强度可以表示为

$$\begin{gathered}\lambda^{\mathrm{u}}_{k|k-1}(\boldsymbol{x}) = \\ \lambda^{\mathrm{b}}(\boldsymbol{x}) + P^{\mathrm{s}} \times \\ \sum_{i=1}^{N_{\mathrm{u}}} w_{\mathrm{u},i}\mathrm{St}(\boldsymbol{x}; \boldsymbol{F}\bar{\boldsymbol{x}}^{\mathrm{p}}_{\mathrm{u},i}, \boldsymbol{F}\boldsymbol{P}^{\mathrm{p}}_{\mathrm{u},i}\boldsymbol{F}^{\mathrm{T}} + \boldsymbol{Q}, v_{\mathrm{p},4}),\end{gathered} \tag{35}$$

其中$v_{\mathrm{p},4} = v_4$. 假设$k-1$时刻多伯努利混合的概率密度函数为

$$f^{i,a^i}_{k-1|k-1}(\boldsymbol{x}') = \mathrm{St}(\boldsymbol{x}'; \bar{\boldsymbol{x}}_{j,i}, \boldsymbol{P}_{j,i}, v_9). \tag{36}$$

根据式(10)–(11)得到伯努利分量预测的权值和存在概率为

$$w^{i,a^i}_{k|k-1} = w^{i,a^i}_{k-1|k-1}, \tag{37}$$

$$r^{i,a^i}_{k|k-1} = P^{\mathrm{s}} r^{i,a^i}_{k-1|k-1}, \tag{38}$$

将式(32)(36)代入式(12), 根据引理1得到预测多伯努利混合过程伯努利分量的密度为

$$\begin{aligned} & f^{i,a^i}_{k|k-1}(\boldsymbol{x}) = \\ & \frac{\displaystyle\int f_{k|k-1}(\boldsymbol{x}\,|\boldsymbol{x}')P^{\mathrm{s}}(\boldsymbol{x}')f^{i,a^i}_{k-1|k-1}(\boldsymbol{x}')\mathrm{d}\boldsymbol{x}'}{\displaystyle\int f^{i,a^i}_{k-1|k-1}(\boldsymbol{x}')P^{\mathrm{s}}(\boldsymbol{x}')\mathrm{d}\boldsymbol{x}'} = \\ & \frac{P^{\mathrm{s}}\displaystyle\int \mathrm{St}(\boldsymbol{x};\boldsymbol{F}\boldsymbol{x}',\boldsymbol{Q},v_6)\mathrm{St}(\boldsymbol{x}';\bar{\boldsymbol{x}}_{j,i},\boldsymbol{P}_{j,i},v_9)\mathrm{d}\boldsymbol{x}'}{P^{\mathrm{s}}\displaystyle\int f^{i,a^i}_{k-1|k-1}(\boldsymbol{x}')\mathrm{d}\boldsymbol{x}'} = \\ & \mathrm{St}(\boldsymbol{x};\boldsymbol{F}\bar{\boldsymbol{x}}_{j,i},\boldsymbol{F}\boldsymbol{P}_{j,i}\boldsymbol{F}^{\mathrm{T}}+\boldsymbol{Q},v_{\mathrm{p},9}), \end{aligned} \tag{39}$$

其中$v_{\mathrm{p},9}=v_9$. 接下来, 算法1(见表1)给出了本文所提STM-PMBM滤波算法预测部分的伪代码.

表 1 预测算法

Table 1 Prediction algorithm

```
输入: $n_{k-1|k-1}, (r^{i,a}_{k-1|k-1}, \bar{\boldsymbol{x}}^{i}_{k-1|k-1}, \boldsymbol{P}^{i}_{k-1|k-1})$,
    $n^{\mathrm{u}}_{k-1|k-1}, (\lambda^{\mathrm{u}}_{k-1|k-1}, \bar{\boldsymbol{x}}^{\mathrm{u},i}_{k-1|k-1}, \boldsymbol{P}^{\mathrm{u},i}_{k-1|k-1})$,
    $v_4, v_5, v_9$.
预测当前轨迹:
$n_{k|k-1} = n_{k-1|k-1}$
for $i = 1 : n_{k|k-1}$ do
    根据式(38)–(39)计算得到$k$时刻当前轨迹的预测参数($r^{i,a^i}_{k|k-1}, \bar{\boldsymbol{x}}^{i}_{k|k-1}, \boldsymbol{P}^{i}_{k|k-1}$);
end for
预测当前PPP强度:
$n^{\mathrm{u}}_{k|k-1} = n^{\mathrm{u}}_{k-1|k-1} + n^{\mathrm{b}}$
for $i = 1 : n^{\mathrm{u}}_{k|k-1}$ do
    根据式(35)计算得到$k$时刻PPP分量的预测强度($\lambda^{\mathrm{u}}_{k|k-1}, \bar{\boldsymbol{x}}^{\mathrm{u},i}_{k|k-1}, \boldsymbol{P}^{\mathrm{u},i}_{k|k-1}$);
end for
将新生强度合并到PPP:
for $i = 1 : n^{\mathrm{b}}$ do
    根据式 (31) 计算得到$k$时刻合并后的PPP强度($\lambda^{\mathrm{u},i+n^{\mathrm{u}}_{k-1|k-1}}_{k|k-1}, \bar{\boldsymbol{x}}^{\mathrm{u},i+n^{\mathrm{u}}_{k-1|k-1}}_{k|k-1}, \boldsymbol{P}^{\mathrm{u},i+n^{\mathrm{u}}_{k-1|k-1}}_{k|k-1}$);
end for
输出: $n_{k|k-1}, (r^{i,a^i}_{k|k-1}, \bar{\boldsymbol{x}}^{i}_{k|k-1}, \boldsymbol{P}^{i}_{k|k-1}), n^{\mathrm{u}}_{k|k-1}$,
    $(\lambda^{\mathrm{u},i}_{k|k-1}, \bar{\boldsymbol{x}}^{\mathrm{u},i}_{k|k-1}, \boldsymbol{P}^{\mathrm{u},i}_{k|k-1}), v_{\mathrm{p},4}, v_5, v_{\mathrm{p},9}$.
```

3.2.2 量测更新步

假设多目标检测概率$P^{\mathrm{d}}(\boldsymbol{x})$是恒定的, $f(\boldsymbol{z}^j_k\,|\cdot) = \mathrm{St}(\boldsymbol{z};\boldsymbol{H}\boldsymbol{x},\boldsymbol{R},v_{10})$, 则泊松部分的预测密度(35)可以写为

$$\lambda^{\mathrm{u}}_{k|k-1}(\boldsymbol{x}) = \sum_{i=1}^{N_\mu} w_{\mu,i}\mathrm{St}(\boldsymbol{x};\bar{\boldsymbol{x}}_{\mu,i},\boldsymbol{P}_{\mu,i},v_{\mathrm{p},4}), \tag{40}$$

其中: $N_\mu, \bar{\boldsymbol{x}}_{\mu,i}, w_{\mu,i}, \boldsymbol{P}_{\mu,i}$是泊松部分给定的模型参数, 且多伯努利混合参数为$w^{i,a^i}_{k|k-1}, f^{i,a^i}_{k|k-1}(\boldsymbol{x})=\mathrm{St}(\boldsymbol{x};\bar{\boldsymbol{x}}_{j,i},\boldsymbol{P}_{j,i},v_{\mathrm{p},9}), r^{i,a^i}_{k|k-1}$. 给定量测集合$\boldsymbol{Z}_k$, STM-PMBM滤波器的更新分为以下4部分.

1) 更新泊松点过程强度.

$$\lambda^{\mathrm{u}}_{k|k}(\boldsymbol{x}) = \{1-P^{\mathrm{d}}(\boldsymbol{x})\}\lambda^{\mathrm{u}}_{k|k-1}(\boldsymbol{x}). \tag{41}$$

2) 更新先前可能检测到的目标的漏检.

$$\begin{aligned} & w^{i,a^i}_{k|k} = \\ & w^{i,a^i}_{k|k-1}(1-r^{i,a^i}_{k|k-1}+r^{i,a^i}_{k|k-1}\langle f^{i,a^i}_{k|k-1}, 1-P^{\mathrm{d}}\rangle) = \\ & w^{i,a^i}_{k|k-1}(1-r^{i,a^i}_{k|k-1}+r^{i,a^i}_{k|k-1}(1-P^{\mathrm{d}})), \end{aligned} \tag{42}$$

$$\begin{aligned} & r^{i,a^i}_{k|k} = \frac{r^{i,a^i}_{k|k-1}\langle f^{i,a^i}_{k|k-1}, 1-P^{\mathrm{d}}\rangle}{1-r^{i,a^i}_{k|k-1}+r^{i,a^i}_{k|k-1}\langle f^{i,a^i}_{k|k-1}, 1-P^{\mathrm{d}}\rangle} = \\ & \frac{r^{i,a^i}_{k|k-1}(1-P^{\mathrm{d}})\displaystyle\int f^{i,a^i}_{k|k-1}(\boldsymbol{x})\mathrm{d}\boldsymbol{x}}{1-r^{i,a^i}_{k|k-1}+r^{i,a^i}_{k|k-1}(1-P^{\mathrm{d}})\displaystyle\int f^{i,a^i}_{k|k-1}(\boldsymbol{x})\mathrm{d}\boldsymbol{x}} = \\ & \frac{r^{i,a^i}_{k|k-1}(1-P^{\mathrm{d}})}{1-r^{i,a^i}_{k|k-1}+r^{i,a^i}_{k|k-1}(1-P^{\mathrm{d}})}, \end{aligned} \tag{43}$$

$$\begin{aligned} & f^{i,a^i}_{k|k}(\boldsymbol{x}) = \frac{\{1-P^{\mathrm{d}}(\boldsymbol{x})\}f^{i,a^i}_{k|k-1}(\boldsymbol{x})}{\langle f^{i,a^i}_{k|k-1}, 1-P^{\mathrm{d}}\rangle} = \\ & \frac{(1-P^{\mathrm{d}})f^{i,a^i}_{k|k-1}(\boldsymbol{x})}{(1-P^{\mathrm{d}})\displaystyle\int f^{i,a^i}_{k|k-1}(\boldsymbol{x})\mathrm{d}\boldsymbol{x}} = f^{i,a^i}_{k|k-1}(\boldsymbol{x}), \end{aligned} \tag{44}$$

其中$w^{i,a^i}_{k|k}, r^{i,a^i}_{k|k}, f^{i,a^i}_{k|k}(\boldsymbol{x})$分别表示更新后的假设权值, 存在概率和概率密度函数.

3) 更新当前轨迹.

$$\begin{aligned} & w^{i,a^i}_{k|k} = w^{i,\tilde{a}^i}_{k|k-1}r^{i,\tilde{a}^i}_{k|k-1}\langle f^{i,\tilde{a}^i}_{k|k-1}, f(\boldsymbol{z}^j_k\,|\cdot)P^{\mathrm{d}}\rangle = \\ & w^{i,\tilde{a}^i}_{k|k-1}r^{i,\tilde{a}^i}_{k|k-1}P^{\mathrm{d}}\int f^{i,\tilde{a}^i}_{k|k-1}(\boldsymbol{x})f(\boldsymbol{z}^j_k\,|\cdot)\mathrm{d}\boldsymbol{x} = \\ & w^{i,\tilde{a}^i}_{k|k-1}r^{i,\tilde{a}^i}_{k|k-1}P^{\mathrm{d}}\mathrm{St}(\boldsymbol{z};\boldsymbol{H}\bar{\boldsymbol{x}}_{j,i},\boldsymbol{H}\boldsymbol{P}_{j,i}\boldsymbol{H}^{\mathrm{T}}+\boldsymbol{R},v_{\mathrm{p},9}), \end{aligned} \tag{45}$$

$$r^{i,a^i}_{k|k} = 1, \tag{46}$$

$$\begin{aligned} & f^{i,a^i}_{k|k}(\boldsymbol{x}) = \frac{f(\boldsymbol{z}^j_k\,|\boldsymbol{x})P^{\mathrm{d}}(\boldsymbol{x})f^{i,\tilde{a}^i}_{k|k-1}(\boldsymbol{x})}{\langle f^{i,\tilde{a}^i}_{k|k-1}, f(\boldsymbol{z}^j_k\,|\cdot)P^{\mathrm{d}}\rangle} = \\ & \frac{P^{\mathrm{d}}f(\boldsymbol{z}^j_k\,|\boldsymbol{x})f^{i,\tilde{a}^i}_{k|k-1}(\boldsymbol{x})}{P^{\mathrm{d}}\displaystyle\int f^{i,\tilde{a}^i}_{k|k-1}(\boldsymbol{x})f(\boldsymbol{z}^j_k\,|\cdot)\mathrm{d}\boldsymbol{x}} = \\ & \frac{\mathrm{St}(\boldsymbol{z};\boldsymbol{H}\boldsymbol{x},\boldsymbol{R},v_{10})\mathrm{St}(\boldsymbol{x};\bar{\boldsymbol{x}}_{j,i},\boldsymbol{P}_{j,i},v_{\mathrm{p},9})}{\displaystyle\int \mathrm{St}(\boldsymbol{z};\boldsymbol{H}\boldsymbol{x},\boldsymbol{R},v_{10})\mathrm{St}(\boldsymbol{x};\bar{\boldsymbol{x}}_{j,i},\boldsymbol{P}_{j,i},v_{\mathrm{p},9})\mathrm{d}x}, \end{aligned} \tag{47}$$

为求解上式分子中两个学生t分布的乘积形式, 引入引理2[42–43].

引理 2　若$\boldsymbol{P}$和$\boldsymbol{R}$均为正定矩阵, 则有

$$\begin{aligned} & \mathrm{St}(\boldsymbol{z};\boldsymbol{H}\boldsymbol{x},\boldsymbol{R},v_{11})\mathrm{St}(\boldsymbol{x};\boldsymbol{m},\boldsymbol{P},v_{12}) = \\ & q(\boldsymbol{z})\mathrm{St}(\boldsymbol{x};\tilde{\boldsymbol{m}},\tilde{\boldsymbol{P}},\tilde{v}_{12}), \end{aligned} \tag{48}$$

其中:

$$q(\boldsymbol{z})=\mathrm{St}(\boldsymbol{z};\boldsymbol{Hm},\boldsymbol{S},v_{12}),\tag{49}$$

$$\tilde{\boldsymbol{m}}=\boldsymbol{m}+\boldsymbol{PH}^{\mathrm{T}}\boldsymbol{S}^{-1}(\boldsymbol{z}-\boldsymbol{Hm}),\tag{50}$$

$$\tilde{\boldsymbol{P}}=\frac{v_{12}+\varDelta_{\boldsymbol{z}}^2}{\tilde{v}_{12}}\boldsymbol{P}(\boldsymbol{I}-\boldsymbol{H}^{\mathrm{T}}\boldsymbol{S}^{-1}\boldsymbol{H}),\tag{51}$$

$$\boldsymbol{S}=\boldsymbol{HPH}^{\mathrm{T}}+\boldsymbol{R},\ \tilde{v}_{12}=v_{12}+\mathrm{d}\boldsymbol{z},\tag{52}$$

$$\varDelta_{\boldsymbol{z}}^2=(\boldsymbol{z}-\boldsymbol{Hm})^{\mathrm{T}}\boldsymbol{S}^{-1}(\boldsymbol{z}-\boldsymbol{Hm}),\tag{53}$$

其中d$\boldsymbol{z}$表示系统的量测维度. 根据引理1和引理2, 式(47)可以简化为

$$f_{k|k}^{i,a^i}(\boldsymbol{x})=\frac{q_1(\boldsymbol{z})\mathrm{St}(\boldsymbol{x};\boldsymbol{m}_1,\boldsymbol{P}_1,\tilde{v}_9)}{\mathrm{St}(\boldsymbol{z};\boldsymbol{H}\bar{\boldsymbol{x}}_{j,i},\boldsymbol{HP}_{j,i}\boldsymbol{H}^{\mathrm{T}}+\boldsymbol{R},v_{\mathrm{p},9})},\tag{54}$$

其中:

$$q_1(\boldsymbol{z})=\mathrm{St}(\boldsymbol{z};\boldsymbol{H}\bar{\boldsymbol{x}}_{j,i},\boldsymbol{S}_1,v_{\mathrm{p},9}),\tag{55}$$

$$\boldsymbol{m}_1=\bar{\boldsymbol{x}}_{j,i}+\boldsymbol{P}_{j,i}\boldsymbol{H}^{\mathrm{T}}\boldsymbol{S}_1^{-1}(\boldsymbol{z}-\boldsymbol{H}\bar{\boldsymbol{x}}_{j,i}),\tag{56}$$

$$\boldsymbol{P}_1=\frac{v_{\mathrm{p},9}+\varDelta_{\boldsymbol{z}_1}^2}{\tilde{v}_9}\boldsymbol{P}_{j,i}(\boldsymbol{I}-\boldsymbol{H}^{\mathrm{T}}\boldsymbol{S}_1^{-1}\boldsymbol{H}),\tag{57}$$

$$\boldsymbol{S}_1=\boldsymbol{HP}_{j,i}\boldsymbol{H}^{\mathrm{T}}+\boldsymbol{R},\tag{58}$$

$$\tilde{v}_9=v_{\mathrm{p},9}+\mathrm{d}\boldsymbol{z},\tag{59}$$

$$\varDelta_{\boldsymbol{z}_1}^2=(\boldsymbol{z}-\boldsymbol{H}\bar{\boldsymbol{x}}_{j,i})^{\mathrm{T}}\boldsymbol{S}_1^{-1}(\boldsymbol{z}-\boldsymbol{H}\bar{\boldsymbol{x}}_{j,i}).\tag{60}$$

4) 更新新生成的轨迹.

$$w_{k|k}^{i,2}=\lambda^{\mathrm{fa}}(\boldsymbol{z}_k^j)+\langle\lambda_{k|k-1}^{\mathrm{u}},f(\boldsymbol{z}_k^j|\cdot)P^{\mathrm{d}}\rangle,\tag{61}$$

其中:

$$\begin{aligned}&\langle\lambda_{k|k-1}^{\mathrm{u}},f(\boldsymbol{z}_k^j|\cdot)P^{\mathrm{d}}\rangle=\\&\int\lambda_{k|k-1}^{\mathrm{u}}(\boldsymbol{x})f(\boldsymbol{z}_k^j|\cdot)P^{\mathrm{d}}(\boldsymbol{x})\mathrm{d}\boldsymbol{x}=\\&P^{\mathrm{d}}\int\sum_{i=1}^{N_\mu}w_{\mu,i}\mathrm{St}(\boldsymbol{x};\bar{\boldsymbol{x}}_{\mu,i},\boldsymbol{P}_{\mu,i},v_{\mathrm{p},4})\times\\&\mathrm{St}(\boldsymbol{z};\boldsymbol{Hx},\boldsymbol{R},v_{10})\mathrm{d}x=\\&P^{\mathrm{d}}\sum_{i=1}^{N_\mu}w_{\mu,i}\mathrm{St}(\boldsymbol{z};\boldsymbol{H}\bar{\boldsymbol{x}}_{\mu,i},\boldsymbol{HP}_{\mu,i}\boldsymbol{H}^{\mathrm{T}}+\boldsymbol{R},v_{\mathrm{p},4}),\end{aligned}\tag{62}$$

将式(62)代入式(61)得

$$\begin{aligned}&w_{k|k}^{i,2}=\lambda^{\mathrm{fa}}(\boldsymbol{z}_k^j)+\langle\lambda_{k|k-1}^{\mathrm{u}},f(\boldsymbol{z}_k^j|\cdot)P^{\mathrm{d}}\rangle=\\&\lambda^{\mathrm{fa}}(\boldsymbol{z}_k^j)+P^{\mathrm{d}}\sum_{i=1}^{N_\mu}w_{\mu,i}\times\\&\mathrm{St}(\boldsymbol{z};\boldsymbol{H}\bar{\boldsymbol{x}}_{\mu,i},\boldsymbol{HP}_{\mu,i}\boldsymbol{H}^{\mathrm{T}}+\boldsymbol{R},v_{\mathrm{p},4}),\end{aligned}\tag{63}$$

$$r_{k|k}^{i,2}=\frac{\langle\lambda_{k|k-1}^{\mathrm{u}},f(\boldsymbol{z}_k^j|\cdot)P^{\mathrm{d}}\rangle}{\lambda^{\mathrm{fa}}(\boldsymbol{z}_k^j)+\langle\lambda_{k|k-1}^{\mathrm{u}},f(\boldsymbol{z}_k^j|\cdot)P^{\mathrm{d}}\rangle},\tag{64}$$

将式(62)代入式(64), 得

$$\begin{aligned}&r_{k|k}^{i,2}=\frac{\langle\lambda_{k|k-1}^{\mathrm{u}},f(\boldsymbol{z}_k^j|\cdot)P^{\mathrm{d}}\rangle}{\lambda^{\mathrm{fa}}(\boldsymbol{z}_k^j)+\langle\lambda_{k|k-1}^{\mathrm{u}},f(\boldsymbol{z}_k^j|\cdot)P^{\mathrm{d}}\rangle}=\\&\frac{P^{\mathrm{d}}\sum\limits_{i=1}^{N_\mu}w_{\mu,i}\mathrm{St}(\boldsymbol{z};\boldsymbol{H}\bar{\boldsymbol{x}}_{\mu,i},\boldsymbol{HP}_{\mu,i}\boldsymbol{H}^{\mathrm{T}}+\boldsymbol{R},v_{\mathrm{p},4})}{w_{k|k}^{i,2}},\end{aligned}\tag{65}$$

$$f_{k|k}^{i,2}(\boldsymbol{x})=\frac{f(\boldsymbol{z}_k^j|\boldsymbol{x})P^{\mathrm{d}}(\boldsymbol{x})\lambda_{k|k-1}^{\mathrm{u}}(\boldsymbol{x})}{\langle\lambda_{k|k-1}^{\mathrm{u}},f(\boldsymbol{z}_k^j|\cdot)P^{\mathrm{d}}\rangle},\tag{66}$$

其中, 根据引理2

$$\begin{aligned}&f(\boldsymbol{z}_k^j|\boldsymbol{x})\lambda_{k|k-1}^{\mathrm{u}}(x)=\\&\sum_{i=1}^{N_\mu}w_{\mu,i}\mathrm{St}(\boldsymbol{x};\bar{\boldsymbol{x}}_{\mu,i},\boldsymbol{P}_{\mu,i},v_{\mathrm{p},4})\mathrm{St}(\boldsymbol{z};\boldsymbol{Hx},\boldsymbol{R},v_{10})=\\&\sum_{i=1}^{N_\mu}w_{\mu,i}q_2(\boldsymbol{z})\mathrm{St}(\boldsymbol{x};\boldsymbol{m}_2,\boldsymbol{P}_2,\tilde{v}_4),\end{aligned}\tag{67}$$

$$q_2(\boldsymbol{z})=\mathrm{St}(\boldsymbol{z};\boldsymbol{H}\bar{\boldsymbol{x}}_{\mu,i},\boldsymbol{S}_2,v_{\mathrm{p},4}),\tag{68}$$

$$\boldsymbol{m}_2=\bar{\boldsymbol{x}}_{\mu,i}+\boldsymbol{P}_{\mu,i}\boldsymbol{H}^{\mathrm{T}}\boldsymbol{S}_2^{-1}(\boldsymbol{z}-\boldsymbol{H}\bar{\boldsymbol{x}}_{\mu,i}),\tag{69}$$

$$\boldsymbol{P}_2=\frac{v_{\mathrm{p},4}+\varDelta_{\boldsymbol{z}_2}^2}{\tilde{v}_4}\boldsymbol{P}_{\mu,i}(\boldsymbol{I}-\boldsymbol{H}^{\mathrm{T}}\boldsymbol{S}_2^{-1}\boldsymbol{H}),\tag{70}$$

$$\boldsymbol{S}_2=\boldsymbol{HP}_{\mu,i}\boldsymbol{H}^{\mathrm{T}}+\boldsymbol{R},\tag{71}$$

$$\tilde{v}_4=v_{\mathrm{p},4}+\mathrm{d}\boldsymbol{z},\tag{72}$$

$$\varDelta_{\boldsymbol{z}_2}^2=(\boldsymbol{z}-\boldsymbol{H}\bar{\boldsymbol{x}}_{\mu,i})^{\mathrm{T}}\boldsymbol{S}_2^{-1}(\boldsymbol{z}-\boldsymbol{H}\bar{\boldsymbol{x}}_{\mu,i}),\tag{73}$$

将式(67)代入式(66), 得

$$\begin{aligned}&f_{k|k}^{i,2}(\boldsymbol{x})=\frac{f(\boldsymbol{z}_k^j|\boldsymbol{x})P^{\mathrm{d}}(\boldsymbol{x})\lambda_{k|k-1}^{\mathrm{u}}(\boldsymbol{x})}{\langle\lambda_{k|k-1}^{\mathrm{u}},f(\boldsymbol{z}_k^j|\cdot)P^{\mathrm{d}}\rangle}=\\&\frac{P^{\mathrm{d}}\sum\limits_{i=1}^{N_\mu}w_{\mu,i}q_2(\boldsymbol{z})\mathrm{St}(\boldsymbol{x};\boldsymbol{m}_2,\boldsymbol{P}_2,\tilde{v}_4)}{P^{\mathrm{d}}\int\lambda_{k|k-1}^{\mathrm{u}}(\boldsymbol{x})f(\boldsymbol{z}_k^j|\cdot)\mathrm{d}\boldsymbol{x}}=\\&\frac{\sum\limits_{i=1}^{N_\mu}w_{\mu,i}q_2(\boldsymbol{z})\mathrm{St}(\boldsymbol{x};\boldsymbol{m}_2,\boldsymbol{P}_2,\tilde{v}_4)}{\sum\limits_{i=1}^{N_\mu}w_{\mu,i}\mathrm{St}(\boldsymbol{z};\boldsymbol{H}\bar{\boldsymbol{x}}_{\mu,i},\boldsymbol{HP}_{\mu,i}\boldsymbol{H}^{\mathrm{T}}+\boldsymbol{R},v_{\mathrm{p},4})}.\end{aligned}\tag{74}$$

以上即STM-PMBM滤波器的递推过程, 但是从式(59)和式(72)中可以看出, 后验概率密度的自由度参数随着时间的推移在持续的增加, 并且随着自由度参数的增加, t分布会退化近似为高斯分布, 厚尾特性将丢失. 因此, 为了保留后验概率密度函数的厚尾特性和封闭的学生t分布形式, 本文使用二阶矩匹配的方法[40–41]修正后验概率密度函数的均值$\boldsymbol{m}$和协方差矩阵$\boldsymbol{P}$, 得到近似后验概率密度函数$\mathrm{St}(\boldsymbol{x}_k;\boldsymbol{m}',\boldsymbol{P}',v)$, 即

$$\boldsymbol{m}'_{k|k}=\boldsymbol{m}_{k|k},\ \frac{v}{v-2}\boldsymbol{P}'_{k|k}=\frac{v'}{v'-2}\boldsymbol{P}_{k|k},\tag{75}$$

$$v'=v+\mathrm{d}\boldsymbol{z}.\tag{76}$$

为了完整起见, 算法2(见表2)给出本文所提STM-PMBM滤波算法更新部分的伪代码.

表 2 更新算法

Table 2 Update algorithm

```
输入: $n_{k|k-1}$, $(r^i_{k|k-1}, \bar{\boldsymbol{x}}^i_{k|k-1}, \boldsymbol{P}^i_{k|k-1})$, $n^{\rm u}_{k|k-1}$,
  $(\lambda^{\rm u}_{k|k-1}, \bar{\boldsymbol{x}}^{{\rm u},i}_{k|k-1}, \boldsymbol{P}^{{\rm u},i}_{k|k-1})$, $\upsilon_{{\rm p},4}, \upsilon_5, \upsilon_{{\rm p},9}$, $\boldsymbol{Z}_k = (\boldsymbol{z}^1, \cdots, \boldsymbol{z}^{m_k})$.
更新当前轨迹
for $i = 1 : n_{k|k-1}$ do
    生成漏检假设
    根据式(42)–(44)计算得到$k$时刻: $w^{i,1}_{k|k}, r^{i,1}_{k|k}, \bar{\boldsymbol{x}}^{i,1}_{k|k}, \boldsymbol{P}^{i,1}_{k|k}$;
    使用量测更新生成假设
    for $j = 1 : m_k$ do
        根据式(45)–(47)(54)–(60)计算得到$k$时刻: $w^{i,j}_{k|k}, r^{i,j}_{k|k}, \bar{\boldsymbol{x}}^{i,j}_{k|k}, \boldsymbol{P}^{i,j}_{k|k}$;
    end for
end for
用量测更新泊松点过程, 为每个量测生成新的轨迹:
for $i = 1 : n^{\rm u}_{k|k-1}$ do
    $\boldsymbol{S}_k = \boldsymbol{H}\boldsymbol{P}^{{\rm u},i}_{k|k-1}\boldsymbol{H}^{\rm T} + \boldsymbol{R}$
    $\boldsymbol{K}_k = \boldsymbol{P}^{{\rm u},i}_{k|k-1}\boldsymbol{H}^{\rm T}/\boldsymbol{S}_k$
    $\boldsymbol{P}_k = \boldsymbol{P}^{{\rm u},i}_{k|k-1} - \boldsymbol{K}_k\boldsymbol{H}\boldsymbol{P}^{{\rm u},i}_{k|k-1}$
end for
for $j = 1 : m_k$ do
    $i = n_{k|k-1} + j$
    for $k = 1 : n^{\rm u}_{k|k-1}$ do
        根据式(61)–(74)计算得到$k$时刻: $w^{i,2}_{k|k}, r^{i,2}_{k|k}, \bar{\boldsymbol{x}}^{i,2}_{k|k}, \boldsymbol{P}^{i,2}_{k|k}$;
    end for
end for
更新未知目标的强度
$n^{\rm u}_{k|k} = n^{\rm u}_{k|k-1}$
for $i = 1 : n^{\rm u}_{k|k-1}$ do
    根据式(41)计算得到$k$时刻: $\lambda^{{\rm u},i}_{k|k}, \bar{\boldsymbol{x}}^{{\rm u},i}_{k|k}, \boldsymbol{P}^{{\rm u},i}_{k|k}$;
end for
输出: $n^{\rm u}_{k|k}$, $\lambda^{{\rm u},i}_{k|k}$, $\bar{\boldsymbol{x}}^{{\rm u},i}_{k|k}$, $\boldsymbol{P}^{{\rm u},i}_{k|k}$, $n_{k|k}$, $w^{i,a}_{k|k}$, $r^{i,a}_{k|k}$, $\bar{\boldsymbol{x}}^{i,a}_{k|k}$, $\boldsymbol{P}^{i,a}_{k|k}$.
```

4 仿真实验

4.1 多目标跟踪性能评价指标

对于多目标跟踪滤波器的性能评价, 本文选用最新的广义最优子模式分配(generalized optimal subpattern assignment metric, GOSPA)距离来评价本文所提滤波算法的性能[44], GOSPA定义为

$$d_p^{(c,2)}(\boldsymbol{X},\boldsymbol{Y}) \triangleq \Big[\min_{\gamma\in\Gamma}\Big(\sum_{(i,j)\in\gamma} d(\boldsymbol{x}_i,\boldsymbol{y}_j)^p + \frac{c^p}{2}(|\boldsymbol{X}|+|\boldsymbol{Y}|-2|\gamma|)\Big)\Big]^{\frac{1}{p}}, \tag{77}$$

其中: Γ表示所有可能的二维赋值集的集合, c表示截断距离, p衡量了对定位分量中的异常值惩罚的敏感性. 从上式可以看出, $(d_p^{(c,2)}(\cdot,\cdot))^2$可以被分解为3种形式: $c^2_{\rm l}(\cdot,\cdot) = \sum\limits_{(i,j)\in\gamma} d(\boldsymbol{x}_i,\boldsymbol{y}_j)^p$表示定位误差(localization error, LE), $c^2_{\rm m}(\cdot,\cdot) = \dfrac{c^p(|\boldsymbol{X}|-|\gamma|)}{2}$表示漏检误差(missed targets error, ME), $c^2_{\rm f}(\cdot,\cdot) = \dfrac{c^p(|\boldsymbol{Y}|-|\gamma|)}{2}$表示虚警误差(false targets error, FE). 在本文仿真中, 设置$c = 8, p = 1, \alpha = 2$.

4.2 仿真场景

为了验证本文所提算法的有效性, 通过设计复杂的多目标跟踪实验, 将STM-PMBM滤波算法与GM-PMBM, STM-PHD, GM-PHD滤波算法进行跟踪性能的充分比较. 设置$[-400, 400]\,{\rm m} \times [-400, 400]\,{\rm m}$的多目标跟踪监控区域内有5个受异常噪声干扰的目标, 每个目标在总检测时长的中间时刻进行初始化, 并向前向后传递. 目标的状态由位置和速度组成, 即, $\boldsymbol{x}_k = [p_{\boldsymbol{x},k}\ \ p_{\boldsymbol{y},k}\ \ v_{\boldsymbol{x},k}\ \ v_{\boldsymbol{y},k}]^{\rm T}$, 观测向量为$\boldsymbol{z}_k = [z_{\boldsymbol{x},k}\ \ z_{\boldsymbol{y},k}]^{\rm T}$. 此外, 目标的运动模型如式(25)–(26)所示. 状态转移矩阵和量测矩阵设置为

$$\boldsymbol{F}_k = \begin{bmatrix} 1 & T \\ 0 & 1 \end{bmatrix} \otimes \boldsymbol{I}_2,\ \boldsymbol{H} = [1\ \ 0] \otimes \boldsymbol{I}_2. \tag{78}$$

初始未知目标的强度被假设为$\lambda^{\rm u}_{0|0}(\boldsymbol{x}) = 10{\rm St}\{\boldsymbol{x}; 0, \boldsymbol{P}, \upsilon_{\rm u}\}$, 其中$\boldsymbol{P} = {\rm diag}\{[150^2\ \ 60^2\ \ 150^2\ \ 60^2]^{\rm T}\}$; 新生强度$\lambda^{\rm b}(\boldsymbol{x}) = 0.05{\rm St}\,\{\boldsymbol{x}; 0, \boldsymbol{P}, \upsilon_{\rm b}\}$; 每次扫描的平均杂波数$\lambda^{\rm fa} = 10$. 目标的存活概率$P^{\,\rm s} = 0.99$, 检测概率$P^{\,\rm d} = 0.99$. 总检测时长$T_{\rm d} = 50\,{\rm s}$, 检测时间间隔$T = 1\,{\rm s}$, 自由度参数均设为3. 每个目标在总检测时长中间时刻的初始状态分别设置为: $\boldsymbol{m}_1 = [135\ \ 10\ \ 105\ \ 8]^{\rm T}$, $\boldsymbol{m}_2 = [125\ \ -8\ \ 85\ \ 5]^{\rm T}$, $\boldsymbol{m}_3 = [-140\ \ -5\ \ -95\ \ -10]^{\rm T}$, $\boldsymbol{m}_4 = [-160\ \ 9\ \ 100\ \ 6]^{\rm T}$, $\boldsymbol{m}_5 = [210\ \ -11\ \ -200\ \ 9]^{\rm T}$.

4.2.1 当仅有量测噪声为厚尾噪声时

为了验证厚尾噪声条件下的多目标跟踪性能, 考虑量测噪声受异常噪声干扰, 即多目标的观测过程呈现明显的拖尾现象. 在仿真实验中, 过程噪声建模为$\boldsymbol{w}_k \sim \mathcal{N}(\boldsymbol{w}_k; 0, \boldsymbol{Q}_k)$, 其中

$$\boldsymbol{Q}_k = q\begin{bmatrix} \dfrac{T^3}{3} & \dfrac{T^2}{2} \\ \dfrac{T^2}{2} & T \end{bmatrix} \otimes \boldsymbol{I}_2, \tag{79}$$

其中$q = 0.01$. 厚尾量测噪声建模为$\boldsymbol{v}_k \sim {\rm St}(\boldsymbol{v}_k; 0, \boldsymbol{R}_k, \upsilon_2)$, 即

$$v_k \sim \begin{cases} \mathcal{N}(0, \boldsymbol{R}_k), & {\rm w.p.}\ \ 1-p_{\rm m}, \\ \mathcal{N}(0, 100\boldsymbol{R}_k), & {\rm w.p.}\ \ p_{\rm m}, \end{cases} \tag{80}$$

其中: w.p.表示以一定的概率出现, $p_{\rm m} = 0.1$表示量测噪声异常值出现的概率, $\boldsymbol{R}_k = \sigma^2\boldsymbol{I}_2$表示量测噪声协方差矩阵, 其中$\sigma = 2\,{\rm m}$. 为了验证本文所提算法

的性能, 进行了100次独立的蒙特卡洛(Monte Carlo, MC)实验, 针对k时刻的特定指标求其统计平均值.

图2是多目标运动的真实轨迹. 为了清楚的体现4种算法的差异, 分别对MC实验中多目标势估计, GOSPA距离和定位误差进行统计分析, 统计结果如图3–5所示.

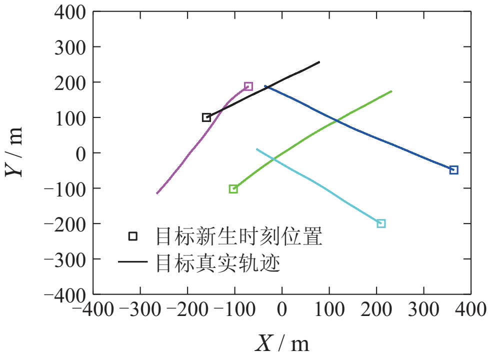


图 2 多目标运动轨迹

Fig. 2 Multi-target motion trajectory

从图3中可以看出, 在异常的量测噪声的影响下, 对于多目标势的估计, 随着时间的推移和目标数目的渐变, STM-PMBM滤波器还是比GM-PMBM, STM-PHD, GM-PHD滤波器更能准确的估计空间目标势及其变化, 显示了前者在复杂的厚尾噪声条件下拥有更好的多目标检测性能.

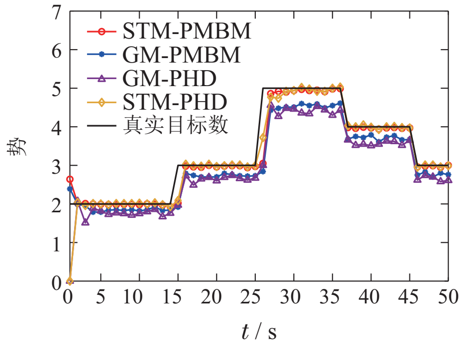


图 3 多目标势估计统计

Fig. 3 Multi-target cardinality estimation statistics

从图4中可以看出, 本文所提算法的GOSPA距离较小, 由于GOSPA考虑了多目标状态估计、势估计、漏检率、错误航迹生成率等综合的多目标跟踪性能, 所以更能客观的反映出: 本文所提算法受异常观测噪声影响下, 对比传统算法依然可表现出更为优越的跟踪性能, 所提算法在复杂跟踪环境下的鲁棒性和自适应性得以充分验证. 而传统的GM-PMBM, STM-PHD, GM-PHD滤波算法的GOSPA距离统计在此仿真环境下表现出种种问题, 包括目标势发生瞬变时GOSPA的迅速提升(对势变化的适应性很差), 整体GOSPA距离统计的不稳定, 这都体现了传统算法很难应对受异常观测噪声干扰下的多目标跟踪问题. 图5多目标定位误差统计下, 在仅量测噪声为厚尾噪声在情况下, 4种滤波器都表现出较稳定的定位性能.

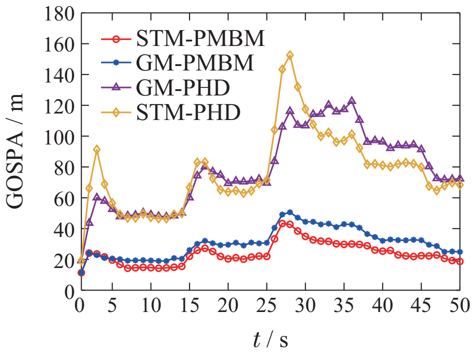


图 4 GOSPA距离统计

Fig. 4 GOSPA distance statistics

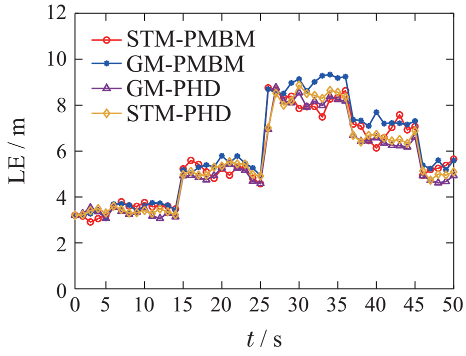


图 5 多目标位置估计误差统计

Fig. 5 Multi-target location estimation error statistics

图6是在厚尾噪声水平p_{m}变化的条件下, 4种滤波器的GOSPA距离平均值统计, 以此来说明异常噪声水平的增大对4种滤波器的影响. 从图6中可以看出, 4个滤波器的GOSPA距离的平均值都随着p_{m}的增加而增加, 但本文所提算法的多目标整体跟踪性能评价GOSPA在任何的异常噪声水平下都明显优于传统算法. 注意到, 所提算法的GOSPA距离随着噪声异常水平的增加, 要显著低于传统算法GOSPA的增长速率, 这即说明了本文算法有效抑制异常噪声的能力, 也充分验证了所提算法在广域异常量测噪声条件下多目标跟踪性能的鲁棒性.

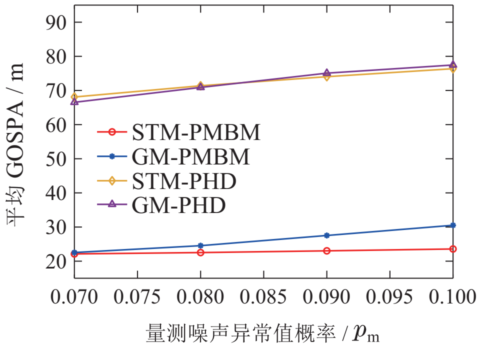


图 6 不同p_{m}时GOSPA距离均值统计

Fig. 6 GOSPA distance mean statistics for different p_{m}

4.2.2 过程噪声和量测噪声都为厚尾噪声的情况

为了进一步验证本文算法的有效性, 设计更为复杂的异常噪声条件, 即过程噪声和量测噪声都为厚尾噪声的仿真实验, 既包括了复杂过程(多目标机动), 也考虑了量测的厚尾特性. 其中, 未知动态模型或目标位置和速度的突变, 会产生厚尾过程噪声; 而传感器自身的不可靠性会产生厚尾量测噪声. 厚尾过程噪声可被建模为$\boldsymbol{w}_k \sim \mathrm{St}(\boldsymbol{w}_k; 0, \boldsymbol{Q}_k, v_1)$, 即

$$\boldsymbol{w}_k \sim \begin{cases} \mathcal{N}(0, \boldsymbol{Q}_k), & \text{w.p. } 1-p_{\mathrm{p}}, \\ \mathcal{N}(0, 100\boldsymbol{Q}_k), & \text{w.p. } p_{\mathrm{p}}, \end{cases} \tag{81}$$

其中$p_{\mathrm{p}} = 0.1$表示过程噪声异常值出现的概率,

$$\boldsymbol{Q}_k = \begin{bmatrix} \dfrac{T^3}{3} & \dfrac{T^2}{2} \\ \dfrac{T^2}{2} & T \end{bmatrix} \otimes \begin{bmatrix} \sigma_q^2 & 0 \\ 0 & \sigma_q^2 \end{bmatrix}, \tag{82}$$

其中$\sigma_q = 1$ m/s. 目标量测方程的建模及其他参数的设置与仿真1相同. 在厚尾过程噪声下, 多目标运动将呈现明显的随机机动趋势, 如图7所示, 跟踪复杂度大大增加. 因此, 本文同样对MC实验中多目标势估计、GOSPA距离和多目标估计位置误差具体分析这4种算法的差异, 统计结果如图8–10所示.

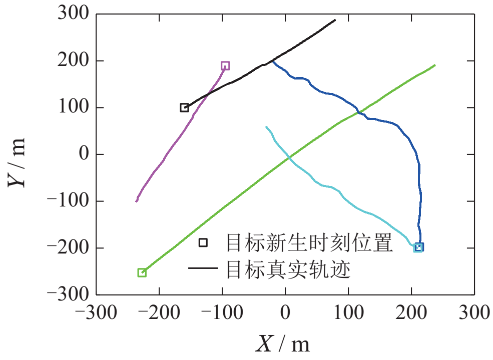


图 7 多目标运动轨迹

Fig. 7 Multi-target motion trajectory

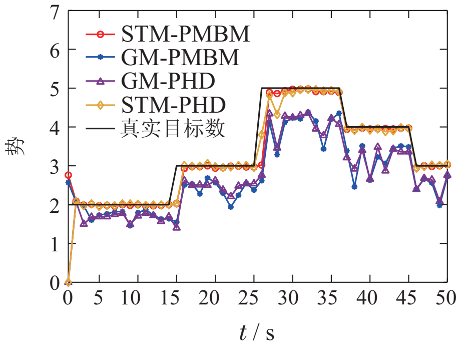


图 8 多目标势估计统计

Fig. 8 Multi-target cardinality estimation statistics

从图8中可以看出, 即便在更为复杂的异常噪声条件下, 多目标不仅伴随着拖尾严重的观测噪声分布, 还伴随着过程噪声的异常噪声而出现的目标的随机机动运动. 在这种复杂的跟踪条件下, GM-PMBM, GM-PHD 滤波器对多目标势估计的误差更大, 对目标势变化的适应性变得更差, 对监控区域的目标检测能力大大下降, 传统滤波器很难在如此恶劣的随机不确定性条件下获得较好的估计性能, STM-PHD相较于传统的高斯滤波器有更好的估计效果. 相比较之下, 本文所提算法对于异常的过程噪声和量测噪声的共存体, 依然能够在目标检测上体现出极强的鲁棒性, 所以图8中的势估计平均统计是平稳和更加精确的.

图9是多目标跟踪的GOSPA距离, 可以看出, 传统的GM-PMBM滤波器由于轻尾分布描述特性无法准确表征异常噪声分布, 这表示为难以识别目标的随机机动和量测过程的离群异动, 跟踪性能相较于本文所提算法大幅度降低. STM-PHD和GM-PHD滤波器由于当目标出现漏检或虚假检测时跟踪估计性能较差, 导致GOSPA距离较大. 而本文算法的GOSPA即便在如此复杂不确定性条件下, 表现出相较于传统算法明显的跟踪估计优势, 体现了滤波器在异常噪声干扰下极强的稳健性和适应性. 图10多目标定位误差统计下, 本文所提算法的位置估计误差相对较小, 展现出更好的多目标跟踪估计性能.

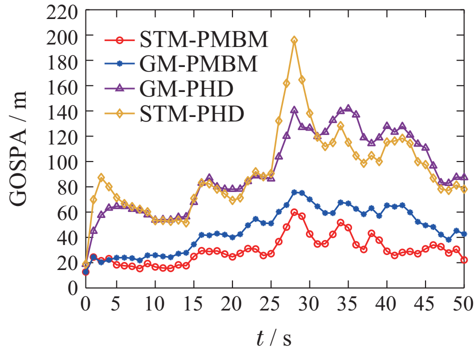


图 9 GOSPA距离统计

Fig. 9 GOSPA distance statistics

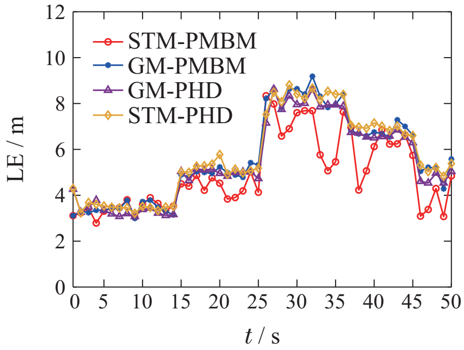


图 10 多目标位置估计误差统计

Fig. 10 Multi-target location estimation error statistics

图11是在厚尾量测噪声水平p_{m}为0.1时, 不同厚尾过程噪声水平p_{p}条件下4种滤波器GOSPA距离的

平均值统计. 从图中可以看出, 随着过程噪声异常噪声水平的增大, 由于多目标机动概率显著增大, 传统算法的GOSPA呈显著增长趋势, 而本文算法抑制异常过程噪声的能力尤为突出, 体现了明显的多目标异常运动(机动)的跟踪优势. 图12是在p_p为0.1时, 4种滤波器在不同p_m条件下GOSPA距离的平均值统计. 从图12中依旧可以看出, 随着异常量测噪声水平增大时, 仍然体现出更好的跟踪性能.

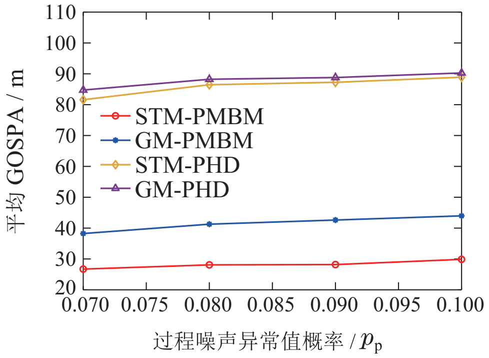


图 11 不同p_p时GOSPA距离均值统计

Fig. 11 GOSPA distance mean statistics for different p_p

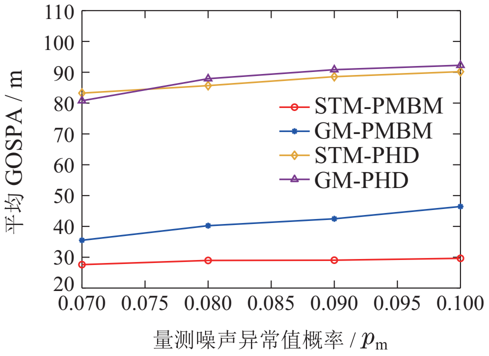


图 12 不同p_m时GOSPA距离均值统计

Fig. 12 GOSPA distance mean statistics for different p_m

图11和图12相辅相成, 联合构成了广域异常噪声(异常过程噪声与异常量测噪声)的复杂跟踪环境, 无论在任何异常的噪声水平下, 本文所提算法都能体现出比传统算法更好的多目标跟踪性能, 滤波器抑制异常噪声、滤波器在复杂环境下的鲁棒性和稳健性得以充分证明.

5 结论

本文主要研究在过程噪声和量测噪声都为厚尾噪声的条件下, 提出学生t泊松多伯努利混合滤波器进行异常噪声条件下的多目标跟踪. 仿真实验表明, 与传统算法相比, 本文方法在处理非高斯的厚尾噪声时有其独特的优势, 这为广域复杂不确定性多目标跟踪问题找到了一个更为行之有效的解决方案. 在未来的工作中, 可以基于现代高分辨率传感器, 进一步开展本文算法在扩展目标跟踪应用中的拓展性研究.

参考文献:

[1] SOLDI G, MEYER F, BRACA P, et al. Self-tuning algorithms for multisensor-multitarget tracking using belief propagation. *IEEE Transactions on Signal Processing*, 2019, 67(15): 3922 – 3937.

[2] GRANSTROM K, SVENSSON L, REUTER S, et al. Likelihood-based data association for extended object tracking using sampling methods. *IEEE Transactions on Intelligent Vehicles*, 2017, 3(1): 30 – 45.

[3] CLARK D E. Multi-sensor network information for linear-Gaussian multi-target tracking systems. *IEEE Transactions on Signal Processing*, 2021, 69(1): 4312 – 4325.

[4] MEMON S A, ULLAH I, KHAN U, et al. Smoothing linear multi-target tracking using integrated track splitting filter. *Remote Sensing*, 2022, 14(5): 1289 – 1307.

[5] AINSLEIGH P L, LUGINBUHL T E, WILLETT P K. A sequential target existence statistic for joint probabilistic data association. *IEEE Transactions on Aerospace and Electronic Systems*, 2020, 57(1): 371 – 381.

[6] BLACKMAN S S. Multiple hypothesis tracking for multiple target tracking. *IEEE Aerospace and Electronic Systems Magazine*, 2004, 19(1): 5 – 18.

[7] VO B N, VO B T. A multi-scan labeled random finite set model for multi-object state estimation. *IEEE Transactions on Signal Processing*, 2019, 67(19): 4948 – 4963.

[8] SAVIC V, WYMEERSCH H, LARSSON E G. Target tracking in confined environments with uncertain sensor positions. *IEEE Transactions on Vehicular Technology*, 2015, 65(2): 870 – 882.

[9] MAHLER R P S. Multitarget bayes filtering via first-order multitarget moments. *IEEE Transactions on Aerospace and Electronic Systems*, 2003, 39(4): 1152 – 1178.

[10] MAHLER R. PHD filters of higher order in target number. *IEEE Transactions on Aerospace and Electronic Systems*, 2007, 43(4): 1523 – 1543.

[11] VO B T, VO B N, CANTONI A. The cardinality balanced multi-target multi-Bernoulli filter and its implementations. *IEEE Transactions on Signal Processing*, 2008, 57(2): 409 – 423.

[12] YANG C, SHI Z, HAN K, et al. Optimization of particle CBMeMBer filters for hardware implementation. *IEEE Transactions on Vehicular Technology*, 2018, 67(9): 9027 – 9031.

[13] REUTER S, VO B T, VO B N, et al. The labeled multi-Bernoulli filter. *IEEE Transactions on Signal Processing*, 2014, 62(12): 3246 – 3260.

[14] VO B T, VO B N. Labeled random finite sets and multi-object conjugate priors. *IEEE Transactions on Signal Processing*, 2013, 61(13): 3460 – 3475.

[15] VO B N, VO B T, BEARD M. Multi-sensor multi-object tracking with the generalized labeled multi-Bernoulli filter. *IEEE Transactions on Signal Processing*, 2019, 67(23): 5952 – 5967.

[16] MAHLER R. Integral-transform derivations of exact closed-form multitarget trackers. *The 19th International Conference on Information Fusion (FUSION)*. Heidelberg, Germany: IEEE, 2016: 950 – 957.

[17] FANTACCI C, PAPI F. Scalable multisensor multitarget tracking using the marginalized δ-GLMB density. *IEEE Signal Processing Letters*, 2016, 23(6): 863 – 867.

[18] HUANG Q, XIE L, SU H. Estimations of time-varying birth cardinality distribution and birth intensity in Gaussian mixture CPHD filter for multi-target tracking. *Signal Processing*, 2022, 190: 108321.

[19] VO B N, MA W K. The Gaussian mixture probability hypothesis density filter. *IEEE Transactions on Signal Processing*, 2006, 54(11): 4091 – 4104.

[20] VO B N, SINGH S, DOUCET A. Sequential Monte Carlo methods for multitarget filtering with random finite sets. *IEEE Transactions on Aerospace and Electronic Systems*, 2005, 41(4): 1224 – 1245.

[21] YANG C, SHI Z, HAN K, et al. Optimization of particle CBMeMBer filters for hardware implementation. *IEEE Transactions on Vehicular Technology*, 2018, 67(9): 9027 – 9031.

[22] KROPFREITER T, MEYER F, HLAWATSCH F. Sequential Monte Carlo implementation of the track-oriented marginal multi-Bernoulli/Poisson filter. *The 19th International Conference on Information Fusion (FUSION)*. Heidelberg, Germany: IEEE, 2016: 972 – 979.

[23] WILLIAMS J L. Marginal multi-Bernoulli filters: RFS derivation of MHT, JIPDA, and association-based MeMBer. *IEEE Transactions on Aerospace and Electronic Systems*, 2015, 51(3): 1664 – 1687.

[24] GARCIA-FERNÁNDEZ A F, WILLIAMS J L, GRANSTROM K, et al. Poisson multi-Bernoulli mixture filter: Direct derivation and implementation. *IEEE Transactions on Aerospace and Electronic Systems*, 2018, 54(4): 1883 – 1901.

[25] FANTACCI C, VO B N, VO B T, et al. Robust fusion for multisensor multiobject tracking. *IEEE Signal Processing Letters*, 2018, 25(5): 640 – 644.

[26] WILLIAMS J L. An efficient, variational approximation of the best fitting multi-Bernoulli filter. *IEEE Transactions on Signal Processing*, 2014, 63(1): 258 – 273.

[27] TOLLICH J, FUNKEN S, OPITZ F. Gaussian Poisson multi Bernoulli mixture filter and its applications. *The 21st International Radar Symposium (IRS)*. Warsaw, Poland: Politechnika Warszawska, 2020: 167 – 172.

[28] XIA Y, GRANSTRCOM K, SVENSSON L, et al. Performance evaluation of multi-Bernoulli conjugate priors for multi-target filtering. *The 20th International Conference on Information Fusion (FUSION)*. Xi'an, China: IEEE, 2017: 1 – 8.

[29] GRANSTROM K, FATEMI M, SVENSSON L. Poisson multi-Bernoulli mixture conjugate prior for multiple extended target filtering. *IEEE Transactions on Aerospace and Electronic Systems*, 2019, 56(1): 208 – 225.

[30] SCHEIDEGGER S, BENJAMINSSON J, ROSENBERG E, et al. Mono-camera 3d multi-object tracking using deep learning detections and pmbm filtering. *IEEE Intelligent Vehicles Symposium (IV)*. Changshu, China: IEEE, 2018: 433 – 440.

[31] FATEMI M, GRANSTROM K, SVENSSON L, et al. Poisson multi-Bernoulli mapping using Gibbs sampling. *IEEE Transactions on Signal Processing*, 2017, 65(11): 2814 – 2827.

[32] FROHLE M, LINDBERG C, GRANSTROM K, et al. Multisensor Poisson multi-Bernoulli filter for joint target-sensor state tracking. *IEEE Transactions on Intelligent Vehicles*, 2019, 4(4): 609 – 621.

[33] LI G, KONG L, YI W, et al. Multiple model Poisson multi-Bernoulli mixture filter for maneuvering targets. *IEEE Sensors Journal*, 2020, 21(3): 3143 – 3154.

[34] ZHOU H, HUANG H, ZHAO H, et al. Adaptive unscented Kalman filter for target tracking in the presence of nonlinear systems involving model mismatches. *Remote Sensing*, 2017, 9(7): 657 – 676.

[35] WU H, CHEN S, YANG B, et al. Robust derivative-free cubature Kalman filter for bearings-only tracking. *Journal of Guidance, Control, and Dynamics*, 2016, 39(8): 1865 – 1870.

[36] GORDON N J, SMITH A F M. Approximate non-Gaussian bayesian estimation and modal consistency. *Journal of the Royal Statistical Society: Series B (Methodological)*, 1993, 55(4): 913 – 918.

[37] BILIK I, TABRIKIAN J. Maneuvering target tracking in the presence of glint using the nonlinear Gaussian mixture Kalman filter. *IEEE Transactions on Aerospace and Electronic Systems*, 2010, 46(1): 246 – 262.

[38] HUANG Y, ZHANG Y, LI N, et al. A novel robust Student's t-based Kalman filter. *IEEE Transactions on Aerospace and Electronic Systems*, 2017, 53(3): 1545 – 1554.

[39] HUANG Y, ZHANG Y, LI N, et al. A robust Gaussian approximate fixed-interval smoother for nonlinear systems with heavy-tailed process and measurement noises. *IEEE Signal Processing Letters*, 2016, 23(4): 468 – 472.

[40] HUANG Y, ZHANG Y, XU B, et al. A new outlier-robust Student's t based Gaussian approximate filter for cooperative localization. *IEEE/ASME Transactions on Mechatronics*, 2017, 22(5): 2380 – 2386.

[41] CHEN Hui, ZHANG Xingxing. Multiple extended target tracking in the presence of heavy tailed noise using multi-Bernouli filtering method. *Acta Automatica Sinca*, 2023, 49(7): 1573 – 1586.
(陈辉, 张星星. 基于多伯努利滤波的厚尾噪声条件下多扩展目标跟踪. 自动化学报, 2023, 49(7): 1573 – 1586.)

[42] ROTH M, OZKAN E, GUSTAFSSON F. A Student's t filter for heavy tailed process and measurement noise. *IEEE International Conference on Acoustics, Speech and Signal Processing*. Vancouver, Canada: IEEE, 2013: 5770 – 5774.

[43] HUANG Y, ZHANG Y, LI N, et al. Robust Student's t based nonlinear filter and smoother. *IEEE Transactions on Aerospace and Electronic Systems*, 2016, 52(5): 2586 – 2596.

[44] RAHMATHULLAH A S, GARCIA-FERNANDEZ A F, SVENSSON L. Generalized optimal sub-pattern assignment metric. *The 20th International Conference on Information Fusion (Fusion)*. Xi'an, China: IEEE, 2017: 1 – 8.

作者简介:

赵子文　博士研究生, 目前研究方向为多目标跟踪技术, E-mail: zwzhao0930@163.com;

陈　辉　教授, 博士生导师, 主要研究兴趣为多目标跟踪、数据融合、最优控制, E-mail: huich78@hotmail.com;

连　峰　教授, 博士生导师, 主要研究方向为目标跟踪、信息融合与传感器管理, E-mail: lianfeng1981@mail.xjtu.edu.cn;

张光华　讲师, 主要研究方向为信息融合与目标跟踪, E-mail: MichaelZgh@stu.xjtu.edu.cn.

高斯过程认知学习的多机动目标泊松多伯努利混合滤波器

赵子文[①] 陈 辉*[①] 连 峰[②] 张光华[②] 张文旭[①]

[①](兰州理工大学自动化与电气工程学院 兰州 730050)

[②](西安交通大学自动化科学与工程学院 西安 710049)

摘 要： 针对复杂不确定环境下的多机动目标跟踪(MMTT)问题，该文提出一种基于高斯过程(GP)数据驱动的多目标跟踪方法。GP作为一种非参数贝叶斯方法，可通过有限样本灵活推断无限维函数，更能够灵活地自适应复杂多变的目标机动模式。通过GP算法学习多机动目标不确定的运动与观测模型，能有效克服传统多模型(MM)方法中因预设模型过多或模型失配所导致的性能退化问题。然后，利用泊松多伯努利混合(PMBM)建立多目标跟踪滤波的共轭先验递推结构，并使用GP学习未知的多目标动力学和观测模型，从而最终提出高斯过程多机动目标PMBM滤波器。仿真结果表明，该方法在复杂多变的MMTT任务中展现出较高的跟踪精度，验证了其在处理MMTT问题上的有效性和鲁棒性。

关键词： 多机动目标跟踪；数据驱动；高斯过程；泊松多伯努利混合

中图分类号： TN911.7; TP274 **文献标识码：** A **文章编号：** 1009-5896(2025)08-2724-12

DOI: 10.11999/JEIT241139 **CSTR:** 32379.14.JEIT241139

1 引言

多目标跟踪(Multi-target Tracking, MTT)是雷达信号处理领域中的一项核心技术和挑战，其主要功能是同时实现对多个目标的检测、跟踪与识别。当前，多目标跟踪技术已广泛应用于军事[1,2]、移动机器人[3,4]以及导航[5,6]等各专业领域。尽管多目标跟踪技术取得了显著进展，但其在实际应用中仍面临两大主要挑战：一是量测的不确定性[7,8]，即目标和量测之间的关联问题；二是目标运动的不确定性[9–12]，即目标可能表现出未预期的运动行为或机动。这两方面的不确定性共同作用，对稳定、高精度多目标跟踪提出了很大的挑战。

近年来，随机有限集(Random Finite Set, RFS)[13]理论因能够有效处理目标数目不确定及多目标跟踪中的数据关联问题而受到了广泛关注，逐渐成为解决多目标跟踪难题的主要方案。在有限集理论框架下，已发展出形式多样的多目标滤波算法，包括概率假设密度(Probability Hypothesis Density, PHD)滤波器[14]、势概率假设密度(Cardinality PHD, CPHD)滤波器[15]、多伯努利(Multi-Bernoulli, MB)滤波器[16]、标签多伯努利(Labeled MB, LMB)滤波器[17]、广义标签多伯努利(Generalized LMB, GLMB)滤波器[18]以及泊松多伯努利混合(Poisson Multi-Bernoulli Mixture, PMBM)滤波器[19,20]。这些滤波器均可通过高斯混合[21]和序贯蒙特卡罗(Sequential Monte Carlo, SMC)[22]方法实现。在这些滤波器中，PMBM滤波器因其独特的共轭先验和高效的递推结构脱颖而出。

现有的多目标滤波器普遍假设所有目标遵循共同的动态模型[9]，然而在实际应用中，目标的运动模式未知且机动变化。单一确定模型难以准确描述机动目标的运动学行为，很容易导致滤波器性能下降或发散。针对这一问题，多模型(Multiple Model, MM)方法作为一种有效的手段，已在多机动目标跟踪(Multiple Maneuvering Target Tracking, MMTT)中得到了广泛应用。相继提出了多种适用于MMTT的滤波算法[23–29]。此外，基于PMBM共轭先验结构，文献[30]提出多模型泊松多伯努利混合(MM-PMBM)滤波器。尽管MM方法在提高目标跟踪适应性方面取得了显著进展，但随着现代目标轨迹不确定性增加，传统基于模型的方法在应对复杂应用场景时仍存在相当大的局限性。

为了克服传统模型驱动方法在多机动目标跟踪中的局限性，高斯过程(Gaussian Process, GP)[31]作为一种数据驱动策略，为该领域带来了创新性的

收稿日期：2024-12-27；改回日期：2025-07-11；网络出版：2025-07-25
*通信作者： 陈辉 chenh@lut.edu.cn
基金项目：国家自然科学基金 (62163023, 61873116, 62366031, 62363023)，甘肃省基础研究创新群体(25JRRA058)，中央引导地方科技发展资金项目(25ZYJA040)，甘肃省重点人才项目(2024RCXM86)，甘肃省军民融合发展专项资金(本基金无项目编号)
Foundation Items: The National Natural Science Foundation of China (62163023, 61873116, 62366031, 62363023), Gansu Provincial Basic Research Innovation Group of China (25JRRA058), The Central Government's Funds for Guiding Local Science and Technology Development of China (25ZYJA040), Gansu Provincial Key Talent Project of China (2024RCXM86), Gansu Provincial Special Fund for Military-Civilian Integration Development of China

解决方案。与传统的基于模型的方法相比，GP通过非参数回归直接从训练数据中学习底层模型及其参数，从而减少了对预定义模型的依赖性。此外，GP易于与状态空间模型和贝叶斯滤波框架集成，为传统的贝叶斯滤波方法提供了重要的补充。

近期的研究将GP应用于机动目标跟踪，旨在解决未知目标运动模型或运动模型失配的问题。文献[32]介绍了一种无模型的机动目标跟踪方法，实现了在大量运动模型之间的动态切换，并进行状态估计。另一项研究提出了一种数据驱动的机动目标跟踪和平滑方法，在跟踪性能上有了显著提升[33]。在文献[34]中，研究者提出了一种基于GP的新型运动模型学习方法，以实现对不同监视区域内目标的有效跟踪。文献[35]则利用GP来近似贝叶斯最优伯努利滤波器中的转移密度，以应对单个未知目标运动模型的转换问题。文献[36]提出了一种结合数据驱动与模型驱动方法的混合策略，有效提升了对强机动目标的跟踪性能。尽管基于GP的方法在多种应用场景中取得了显著成效，但在RFS框架下，针对无模型多机动目标跟踪的研究尚未开展。因此，进一步探索这一领域对于推动多机动目标跟踪技术的发展具有重要意义。

鉴于此，本文提出一种新颖的多机动目标跟踪算法，旨在提高复杂环境下高机动目标的跟踪精度。该方法通过GP从训练数据中学习未知的多目标运动模型和观测模型，并结合容积Kalman滤波器(Cubature Kalman Filter, CKF)[37]进行非线性状态传播。利用GP学习的动态模型递归PMBM分量，本文提出一种GP-PMBM滤波器，并最终利用高斯混合技术推导其解析实现，以有效应对多机动目标跟踪中的复杂性和不确定性问题。仿真实验表明，相较于传统的模型驱动方法，基于高斯过程的方法在复杂多机动场景下展现出优异的跟踪性能。

2　问题背景

2.1　多目标贝叶斯滤波

假设在k时刻，有N_k个目标状态$\boldsymbol{x}_k^1, \boldsymbol{x}_k^2, \cdots, \boldsymbol{x}_k^{N_k}$和$M_k$个量测$\boldsymbol{z}_k^1, \boldsymbol{z}_k^2, \cdots, \boldsymbol{z}_k^{M_k}$，则相应的多目标状态和观测由以下有限集给出

$$\boldsymbol{X}_k = \left\{\boldsymbol{x}_k^1, \boldsymbol{x}_k^2, \cdots, \boldsymbol{x}_k^i, \boldsymbol{x}_k^{i+1}, \cdots, \boldsymbol{x}_k^{N_k}\right\} \in \mathcal{X}, i = 1, 2, \cdots, N_k \tag{1}$$

$$\boldsymbol{Z}_k = \left\{\boldsymbol{z}_k^1, \boldsymbol{z}_k^2, \cdots, \boldsymbol{z}_k^j, \boldsymbol{z}_k^{j+1}, \cdots, \boldsymbol{z}_k^{M_k}\right\} \in \mathcal{Z}, j = 1, 2, \cdots, M_k \tag{2}$$

其中，$\mathcal{X}$和$\mathcal{Z}$分别表示状态空间和观测空间。此外，$\boldsymbol{X}_k$为多目标状态，$\boldsymbol{Z}^k = \{\boldsymbol{Z}_1, \boldsymbol{Z}_2, \cdots, \boldsymbol{Z}_k\}$为从1到$k$时刻采集到的量测集。

给定$k-1$时刻的多目标后验密度$f_{k-1}\left(\boldsymbol{X}\left|\boldsymbol{Z}^{k-1}\right.\right)$，预测多目标密度可以通过Chapman-Kolmogorov方程[13]得到

$$\begin{aligned} f_{k|k-1}\left(\boldsymbol{X}\left|\boldsymbol{Z}^{k-1}\right.\right) = \int & \varphi_{k|k-1}\left(\boldsymbol{X}\left|\boldsymbol{X}'\right.\right) \\ & \cdot f_{k-1}\left(\boldsymbol{X}'\left|\boldsymbol{Z}^{k-1}\right.\right)\boldsymbol{\delta X}' \end{aligned} \tag{3}$$

其中，$\varphi_{k|k-1}\left(\boldsymbol{X}\left|\boldsymbol{X}'\right.\right)$是从$\boldsymbol{X}'$到$\boldsymbol{X}$的多目标转移密度，并且$k$时刻的更新多目标密度可以根据量测集$\boldsymbol{Z}_k$和贝叶斯准则得到

$$f_k\left(\boldsymbol{X}\left|\boldsymbol{Z}^k\right.\right) = \frac{L_k\left(\boldsymbol{Z}_k\left|\boldsymbol{X}\right.\right) f_{k|k-1}\left(\boldsymbol{X}\left|\boldsymbol{Z}^{k-1}\right.\right)}{\int L_k\left(\boldsymbol{Z}_k\left|\boldsymbol{X}\right.\right) f_{k|k-1}\left(\boldsymbol{X}\left|\boldsymbol{Z}^{k-1}\right.\right)\boldsymbol{\delta X}} \tag{4}$$

其中，$L_k\left(\boldsymbol{Z}_k\left|\boldsymbol{X}\right.\right)$表示多目标似然函数。此外，式中所涉及的积分为集合积分[13]

$$\begin{aligned} \int f(\boldsymbol{X})\,\delta\boldsymbol{X} = \sum_{n=0}^{\infty}\frac{1}{n!}\int & f\left(\left\{\boldsymbol{x}^1, \boldsymbol{x}^2, \cdots, \boldsymbol{x}^n\right\}\right) \\ & \cdot \mathrm{d}\boldsymbol{x}^1\mathrm{d}\boldsymbol{x}^2\cdots\mathrm{d}\boldsymbol{x}^n \end{aligned} \tag{5}$$

2.2　PMBM滤波器

PMBM滤波器是一种基于RFS的滤波方法，它的递归需要分别预测和更新泊松点过程(Poisson Point Process, PPP)和MBM。PPP通过其强度函数或者一阶矩进行参数化$\mu(\boldsymbol{x}) = \lambda f(\boldsymbol{x})$，其中$\lambda$表示泊松率，$f(\boldsymbol{x})$是单目标概率密度函数，同时PPP的势服从泊松分布。泊松RFS对应的多目标密度为

$$f(\boldsymbol{X}) = \mathrm{e}^{-\int\mu(\boldsymbol{x})\mathrm{d}\boldsymbol{x}}[\mu(\cdot)]^{\boldsymbol{X}} \tag{6}$$

伯努利RFS密度$f(\boldsymbol{X})$可以表示为

$$f(\boldsymbol{X}) = \begin{cases} 1-r, & \boldsymbol{X} = \varnothing \\ rp(\boldsymbol{x}), & \boldsymbol{X} = \{\boldsymbol{x}\} \\ 0, & |\boldsymbol{X}| \geq 2 \end{cases} \tag{7}$$

其中，$r \in [0,1]$表示存在概率，$p(\boldsymbol{x})$表示目标$\boldsymbol{x}$存在时的概率密度函数。MB RFS是独立伯努利RFS的不相交并集

$$\boldsymbol{X} = \bigcup_{i=1}^{n}\boldsymbol{X}^i \tag{8}$$

其中，n表示伯努利分量的数量，则MB分布可以表示为

$$f^{\mathrm{mb}}(\boldsymbol{X}) = \sum_{\boldsymbol{X}^1\cup\boldsymbol{X}^2\cup\cdots\cup\boldsymbol{X}^n = \boldsymbol{X}}\ \prod_{i=1}^{n} f^i\left(\boldsymbol{X}^i\right) \tag{9}$$

MBM RFS密度是MBs的多目标密度的归一化加权和，参数化为$\left\{w^{j,i}, \left\{r^{j,i}, p^{j,i}(x)\right\}_{i\in I^j}\right\}_{j\in I}$，其中$I$是MBM中MB的索引集合，$j$是关于所有全局假设的索引。MBM的多目标分布为

$$f^{\mathrm{mbm}}\left(\boldsymbol{X}\right)\propto\sum_{j\in I}\sum_{\boldsymbol{X}^1\cup\boldsymbol{X}^2\cup\cdots\cup\boldsymbol{X}^n=\boldsymbol{X}}\prod_{i=1}^{n}w^{j,i}f^{j,i}\left(\boldsymbol{X}^i\right)\tag{10}$$

其中，‘$\propto$’表示比例。根据泊松RFS和MBM RFS密度，PMBM密度表示为

$$f\left(\boldsymbol{X}\right)=\sum_{\boldsymbol{X}^{\mathrm{u}}\cup\boldsymbol{X}^{\mathrm{d}}=\boldsymbol{X}}f^{\mathrm{p}}\left(\boldsymbol{X}^{\mathrm{u}}\right)f^{\mathrm{mbm}}\left(\boldsymbol{X}^{\mathrm{d}}\right)\tag{11}$$

其中，$f^{\mathrm{p}}\left(\cdot\right)$表示泊松RFS密度，$\boldsymbol{X}^{\mathrm{u}}$和$\boldsymbol{X}^{\mathrm{d}}$分别表示未检测到的目标和潜在检测到的目标。根据量测集合$\boldsymbol{Z}^k$，PMBM滤波器的递推过程包括预测和更新步，具体可参考文献[20]。

2.3 高斯过程

GP是一种用于机器学习和统计学的非参数化模型。GP的主要优点包括建模的灵活性，提供不确定性估计的能力以及从训练数据中学习噪声和平滑参数的能力。假设有一组训练数组$\boldsymbol{T}=\{(\boldsymbol{u}_1,\boldsymbol{v}_1),(\boldsymbol{u}_2,\boldsymbol{v}_2),\cdots,(\boldsymbol{u}_n,\boldsymbol{v}_n)\}$，该数据来自受噪声影响的过程

$$\boldsymbol{v}_i=f\left(\boldsymbol{u}_i\right)+\boldsymbol{\varepsilon}\tag{12}$$

其中，$\boldsymbol{u}_i$表示d维输入向量，v_i为对应的标量输出，$\boldsymbol{\varepsilon}\sim\mathcal{N}\left(0,\boldsymbol{I}\sigma^2\right)$表示高斯噪声项。令$\boldsymbol{U}=[\boldsymbol{u}_1,\boldsymbol{u}_2,\cdots,\boldsymbol{u}_n]$和$\boldsymbol{V}=[\boldsymbol{v}_1,\boldsymbol{v}_2,\cdots,\boldsymbol{v}_n]$分别表示输入和输出集合，并且输出$\boldsymbol{v}$的联合分布是输入$\boldsymbol{U}$的函数，则其可以表示为一个零均值的多元高斯分布

$$p\left(\boldsymbol{v}\right)=\mathcal{N}\left(0,\boldsymbol{K}\left(\boldsymbol{U},\boldsymbol{U}\right)+\sigma_n^2\boldsymbol{I}\right)\tag{13}$$

其中，$\boldsymbol{K}\left(\boldsymbol{U},\boldsymbol{U}\right)$是核矩阵，其元素表示为$\boldsymbol{K}_{ij}=k\left(\boldsymbol{u}_i,\boldsymbol{u}_j\right)$；$k\left(\boldsymbol{u},\boldsymbol{u}'\right)$表示核函数，它是对输入和输出之间紧密程度的度量；$\sigma_n^2\boldsymbol{I}$表示引入的高斯噪声，核心是为了在已知训练数据的基础上对高斯函数进行线性化。

核函数的选择取决于具体的应用，在所提方法中，选择常用的平方指数作为核函数。平方指数核在平滑性和参数解释性等方面的综合优势，使其成为机动目标跟踪建模的理想选择[31,33,34]。平方指数核函数表示为

$$k\left(\boldsymbol{u},\boldsymbol{u}'\right)=\sigma_f^2\exp\left(\left(-\frac{1}{2}\left(\boldsymbol{u}-\boldsymbol{u}'\right)\boldsymbol{W}\left(\boldsymbol{u}-\boldsymbol{u}'\right)^{\mathrm{T}}\right)\right)\tag{14}$$

其中，σ_f^2是信号的方差，$\boldsymbol{W}$是一个对角矩阵，包含了每个输入维度的长度尺度。

给定一组训练数据$\langle\boldsymbol{U},\boldsymbol{v}\rangle$和一个测试输入$\boldsymbol{u}^*$，高斯过程定义了一个关于输出$\boldsymbol{v}^*$的高斯预测分布，其均值为

$$\mathrm{GP}_{\mathrm{m}}\left(\boldsymbol{u}^*,\langle\boldsymbol{U},\boldsymbol{v}\rangle\right)=\boldsymbol{k}_*^{\mathrm{T}}\left[\boldsymbol{K}\left(\boldsymbol{U},\boldsymbol{U}\right)+\sigma_n^2\boldsymbol{I}\right]^{-1}\boldsymbol{v}\tag{15}$$

协方差为

$$\begin{aligned}\mathrm{GP}_{\Sigma}\left(\boldsymbol{u}^*,\langle\boldsymbol{U},\boldsymbol{v}\rangle\right)=&k\left(\boldsymbol{u}^*,\boldsymbol{u}^*\right)\\&-\boldsymbol{k}_*^{\mathrm{T}}\left[\boldsymbol{K}\left(\boldsymbol{U},\boldsymbol{U}\right)+\sigma_n^2\boldsymbol{I}\right]^{-1}\boldsymbol{k}_*\end{aligned}\tag{16}$$

其中，$\boldsymbol{k}_*$是测试输入$\boldsymbol{u}^*$和训练输入$\boldsymbol{U}$之间的核向量。式(14)和式(16)中的参数σ_f，σ_n和$\boldsymbol{W}$表示高斯过程超参数，记为$\theta=[\sigma_f,\sigma_n,\boldsymbol{W}]$。这些超参数可以在给定输入的情况下通过最大化训练输出的对数似然来学习，即

$$\theta_{\max}=\arg\max_{\theta}\left\{\lg\left(p\left(\boldsymbol{v}\left|\boldsymbol{U},\theta\right.\right)\right)\right\}\tag{17}$$

其可以使用共轭梯度等数值优化方法[31]来完成。

3 高斯过程多机动目标跟踪

3.1 学习运动和观测模型

假设多目标运动状态和观测状态分别按照状态转移函数f和观测函数h进行演化，即

$$\boldsymbol{x}_k=f\left(\boldsymbol{x}_{k-1}\right)+\boldsymbol{w}_k\tag{18}$$

$$\boldsymbol{z}_k=h\left(\boldsymbol{x}_k\right)+\boldsymbol{v}_k\tag{19}$$

其中，$\boldsymbol{w}_k\sim\mathcal{N}\left(0,\boldsymbol{Q}_k\right)$表示加性高斯过程噪声，$\boldsymbol{v}_k\sim\mathcal{N}\left(0,\boldsymbol{R}_k\right)$表示加性高斯观测噪声。高斯过程可以直接用于学习动态系统的运动和观测模型。这里学习的目标是不仅可以学习相关噪声协方差$\boldsymbol{Q}_k$和$\boldsymbol{R}_k$，还可以学习状态转移函数f和观测函数h。

每个GP的训练数据是一组输入输出关系。运动模型将状态映射为状态转移$\Delta\boldsymbol{x}_k=\boldsymbol{x}_k-\boldsymbol{x}_{k-1}$。通过将上一时刻的状态与状态转移进行整合，可以找到下一时刻的状态。观测模型从状态$\boldsymbol{x}_k$映射到观测$\boldsymbol{z}_k$。运动和观测训练数据集的形式可以分别表示为

$$\boldsymbol{D}_f=\langle\boldsymbol{X},\boldsymbol{X}'\rangle\tag{20}$$

$$\boldsymbol{D}_h=\langle\boldsymbol{X},\boldsymbol{Z}\rangle\tag{21}$$

其中，$\boldsymbol{X}$是真实状态矩阵，$\boldsymbol{X}'=[\Delta\boldsymbol{x}_1,\Delta\boldsymbol{x}_2,\cdots,\Delta\boldsymbol{x}_k]$表示状态转移矩阵，$\boldsymbol{Z}$是观测输出矩阵。

GP近似f和h的过程分别用GP^f和GP^h表示，即

$$\boldsymbol{x}_k=\mathrm{GP}_{\mathrm{m}}^f\left(\boldsymbol{x}_{k-1},\boldsymbol{D}_f\right)+\boldsymbol{w}_k\tag{22}$$

$$\boldsymbol{z}_k=\mathrm{GP}_{\mathrm{m}}^h\left(\boldsymbol{x}_k,\boldsymbol{D}_h\right)+\boldsymbol{v}_k\tag{23}$$

其中，$\boldsymbol{w}_k\sim\mathcal{N}\left(0,\ \mathrm{GP}_{\Sigma}^f\left(\boldsymbol{x}_{k-1},\ \boldsymbol{D}_f\right)\right)$，$\boldsymbol{v}_k\sim\mathcal{N}\left(0,\mathrm{GP}_{\Sigma}^h\left(\boldsymbol{x}_k,\boldsymbol{D}_h\right)\right)$。理想的学习效果是$\mathrm{GP}_{\mathrm{m}}^f$趋近于$f$，$\mathrm{GP}_{\mathrm{m}}^h$趋近于$h$，$\mathrm{GP}_{\Sigma}^f$和$\mathrm{GP}_{\Sigma}^h$分别趋近于$\boldsymbol{Q}$和$\boldsymbol{R}$。

3.2 高斯过程PMBM滤波器

在这一部分中，重点研究如何实现提出的GP-PMBM算法。与其他PMBM滤波器的实现方式类

似，基于高斯混合技术来实现GP-PMBM滤波器。首先需要做一些必要的假设。

(1) 每个目标的存在概率和检测概率与目标的状态无关，即$p_{\mathrm{S},k}(\boldsymbol{x}) = p_{\mathrm{S}}(\boldsymbol{x})$，$p_{\mathrm{D},k}(\boldsymbol{x}) = p_{\mathrm{D}}(\boldsymbol{x})$。

(2) 新生目标的强度是一个高斯混合形式，即

$$\lambda_k^b(\boldsymbol{x}) = \sum_{b=1}^{J_k^\lambda} \boldsymbol{w}_k^{\lambda,b} \mathcal{N}\left(\boldsymbol{x}; \boldsymbol{m}_k^{\lambda,b}, \boldsymbol{P}_k^{\lambda,b}\right) \tag{24}$$

其中，J_k^λ是高斯分量的个数，$w_k^{\lambda,b}$是第b个高斯分量的权重，$\boldsymbol{m}_k^{\lambda,b}$和$\boldsymbol{P}_k^{\lambda,b}$分别表示第$b$个高斯分量的均值和协方差。

(3) 假设$k-1$时刻的先验强度$\mu_{k-1}(\boldsymbol{x})$是一个高斯混合形式，即

$$\mu_{k-1}(\boldsymbol{x}) = \sum_{b=1}^{J_{k-1}^\mu} \boldsymbol{w}_{k-1}^{\mu,b} \mathcal{N}\left(\boldsymbol{x}; \boldsymbol{m}_{k-1}^{\mu,b}, \boldsymbol{P}_{k-1}^{\mu,b}\right) \tag{25}$$

并且假设单目标强度$p_{k-1}^{j,i}(\boldsymbol{x})$是一个高斯分布的形式，即

$$p_{k-1}^{j,i}(\boldsymbol{x}) = \mathcal{N}\left(\boldsymbol{x}, \boldsymbol{m}_{k-1}^{j,i}, \boldsymbol{P}_{k-1}^{j,i}\right) \tag{26}$$

然后给出GP-PMBM滤波器预测过程和更新过程的解析解。

(1) 预测步：对于泊松点过程，预测泊松强度可以表示为

$$\mu_{k|k-1}(\boldsymbol{x}) = \lambda_k^b(\boldsymbol{x}) + \sum_{b=1}^{J_{k-1}^\mu} w_{k|k-1}^{\mu,b} \cdot \mathcal{N}\left(\boldsymbol{x}; \boldsymbol{m}_{k|k-1}^{\mu,b}, \boldsymbol{P}_{k|k-1}^{\mu,b}\right) \tag{27}$$

其中，

$$w_{k|k-1}^{\mu,b} = p_{\mathrm{S},k} w_{k-1}^{\mu,b} \tag{28}$$

根据容积规则，选择$2n$个容积点，$l = 1, 2, \cdots, 2n$，未知模型可以计算为

$$\boldsymbol{x}_{l,k-1} = \boldsymbol{m}_{k-1}^{\mu,b} \pm \sqrt{\boldsymbol{P}_{k-1}^{\mu,b}} \alpha_l \tag{29}$$

$$\boldsymbol{x}_{k|k-1}^l = \mathrm{GP}_{\mathrm{m}}^f(\boldsymbol{x}_{l,k-1}, \boldsymbol{D}_f) \tag{30}$$

$$\boldsymbol{m}_{k|k-1}^{\mu,b} = \frac{1}{2n} \sum_{l=0}^{2n} \boldsymbol{x}_{k|k-1}^l \tag{31}$$

$$\boldsymbol{Q}_k = \mathrm{GP}_\Sigma^f(\boldsymbol{x}_{l,k-1}, \boldsymbol{D}_f) \tag{32}$$

$$\boldsymbol{P}_{k|k-1}^{\mu,b} = \frac{1}{2n} \sum_{l=0}^{2n} \left(\boldsymbol{x}_{k|k-1}^l - \boldsymbol{m}_{k|k-1}^{\mu,b}\right) \cdot \left(\boldsymbol{x}_{k|k-1}^l - \boldsymbol{m}_{k|k-1}^{\mu,b}\right)^{\mathrm{T}} + \boldsymbol{Q}_k \tag{33}$$

对于多伯努利混合，每个伯努利分量预测过程为

$$w_{k|k-1}^{j,i} = w_{k-1}^{j,i} \tag{34}$$

$$r_{k|k-1}^{j,i} = p_{\mathrm{S},k} r_{k-1}^{j,i} \tag{35}$$

$$p_{k|k-1}^{j,i}(\boldsymbol{x}) = \mathcal{N}\left(\boldsymbol{x}; \boldsymbol{m}_{k|k-1}^{j,i}, \boldsymbol{P}_{k|k-1}^{j,i}\right) \tag{36}$$

其中，

$$\boldsymbol{m}_{k|k-1}^{j,i} = \frac{1}{2n} \sum_{l=0}^{2n} \boldsymbol{x}_{k|k-1}^{j,i,l} \tag{37}$$

$$\boldsymbol{P}_{k|k-1}^{j,i} = \frac{1}{2n} \sum_{l=0}^{2n} \left(\boldsymbol{x}_{k|k-1}^{j,i,l} - \boldsymbol{m}_{k|k-1}^{j,i}\right) \cdot \left(\boldsymbol{x}_{k|k-1}^{j,i,l} - \boldsymbol{m}_{k|k-1}^{j,i}\right)^{\mathrm{T}} + \boldsymbol{Q}_k^{j,i} \tag{38}$$

$$\boldsymbol{x}_{l,k-1}^{j,i} = \boldsymbol{m}_{k-1}^{j,i} \pm \sqrt{\boldsymbol{P}_{k-1}^{j,i}} \alpha_l \tag{39}$$

$$\boldsymbol{x}_{k|k-1}^{j,i,l} = \mathrm{GP}_{\mathrm{m}}^f\left(\boldsymbol{x}_{l,k-1}^{j,i}, \boldsymbol{D}_f\right) \tag{40}$$

$$\boldsymbol{Q}_k^{j,i} = \mathrm{GP}_\Sigma^f\left(\boldsymbol{x}_{l,k-1}^{j,i}, \boldsymbol{D}_f\right) \tag{41}$$

(2) 更新步：

(a) 更新泊松分量：首先将预测泊松强度重新写为

$$\mu_{k|k-1}(\boldsymbol{x}) = \sum_{b=1}^{J_{k|k-1}^\mu} w_{k|k-1}^{\mu,b} \mathcal{N}\left(\boldsymbol{x}; \boldsymbol{m}_{k|k-1}^{\mu,b}, \boldsymbol{P}_{k|k-1}^{\mu,b}\right) \tag{42}$$

则更新后的泊松强度为

$$\mu_k(\boldsymbol{x}) = \sum_{b=1}^{J_k^\mu} w_k^{\mu,b} \mathcal{N}\left(\boldsymbol{x}; \boldsymbol{m}_k^{\mu,b}, \boldsymbol{P}_k^{\mu,b}\right) \tag{43}$$

其中，$w_k^{\mu,b} = (1 - p_{\mathrm{D}}(\boldsymbol{x})) w_{k|k-1}^{\mu,b}$，$\boldsymbol{m}_k^{\mu,b} = \boldsymbol{m}_{k|k-1}^{\mu,b}$，$\boldsymbol{P}_k^{\mu,b} = \boldsymbol{P}_{k|k-1}^{\mu,b}$。

(b) 更新伯努利分量：考虑以下3种情况

① 更新首次检测到的目标

$$r_k^{\mathrm{p}}(\boldsymbol{z}) = e_k(\boldsymbol{z}) / \rho_k^{\mathrm{p}}(\boldsymbol{z}) \tag{44}$$

$$p_k^{\mathrm{p}}(\boldsymbol{x}|\boldsymbol{z}) = \frac{1}{e_k(\boldsymbol{z})} \sum_{b=1}^{J_k^\mu} w_{k|k-1}^{\mu,b} \mathcal{N}\left(\boldsymbol{x}; \boldsymbol{m}_k^{\mathrm{p},b}, \boldsymbol{P}_k^{\mathrm{p},b}\right) \tag{45}$$

其中，

$$e_k(\boldsymbol{z}) = \sum_{b=1}^{J_{k|k-1}^\mu} w_k^b \mathcal{N}\left(\boldsymbol{z}; \boldsymbol{m}_k^b, \boldsymbol{P}_k^b\right) \tag{46}$$

$$\rho_k^{\mathrm{p}}(\boldsymbol{z}) = e_k(\boldsymbol{z}) + c(\boldsymbol{z}) \tag{47}$$

$$\boldsymbol{m}_k^{\mathrm{p},b} = \boldsymbol{m}_{k|k-1}^{\mu,b} + \boldsymbol{K}_k^{\mathrm{p}}\left(\boldsymbol{z} - \boldsymbol{m}_k^b\right) \tag{48}$$

$$\boldsymbol{P}_k^{\mathrm{p},b} = \boldsymbol{P}_{k|k-1}^{\mu,b} - \boldsymbol{K}_k^{\mathrm{p}} \boldsymbol{S}_k^{\mathrm{p}} (\boldsymbol{K}_k^{\mathrm{p}})^{\mathrm{T}} \tag{49}$$

$$\boldsymbol{x}_k^{\mathrm{p},l} = \boldsymbol{m}_{k|k-1}^{\mu,b} + \sqrt{\boldsymbol{P}_{k|k-1}^{\mu,b}}\alpha_l \tag{50}$$

$$\boldsymbol{z}_{k|k-1}^{\mathrm{p},l} = \mathrm{GP}_{\mathrm{m}}^{h}\left(\boldsymbol{x}_k^{\mathrm{p},l}, \boldsymbol{D}_h\right) \tag{51}$$

$$\boldsymbol{R}_k^{\mathrm{p}} = \mathrm{GP}_{\Sigma}^{h}\left(\boldsymbol{m}_{k|k-1}^{\mu,b}, \boldsymbol{D}_h\right) \tag{52}$$

$$\boldsymbol{m}_k^{b} = \frac{1}{2n}\sum_{l=0}^{2n}\boldsymbol{z}_{k|k-1}^{\mathrm{p},l} \tag{53}$$

$$\boldsymbol{S}_k^{\mathrm{p}} = \frac{1}{2n}\sum_{l=0}^{2n}\left(\boldsymbol{x}_k^{\mathrm{p},l} - \boldsymbol{m}_k^{b}\right)\left(\boldsymbol{x}_k^{\mathrm{p},l} - \boldsymbol{m}_k^{b}\right)^{\mathrm{T}} + \boldsymbol{R}_k^{\mathrm{p}} \tag{54}$$

$$\boldsymbol{P}_k^{b} = \boldsymbol{S}_k^{\mathrm{p}} \tag{55}$$

$$\boldsymbol{P}_{xz,k}^{\mathrm{p}} = \frac{1}{2n}\sum_{l=0}^{2n}\left(\boldsymbol{m}_{k|k-1}^{\mu,b} - \boldsymbol{x}_k^{\mathrm{p},l}\right) - \left(\boldsymbol{m}_k^{b} - \boldsymbol{z}_{k|k-1}^{\mathrm{p},l}\right)^{\mathrm{T}} \tag{56}$$

$$\boldsymbol{K}_k^{\mathrm{p}} = \boldsymbol{P}_{xz,k}^{\mathrm{p}}(\boldsymbol{S}_k^{\mathrm{p}})^{-1} \tag{57}$$

② 更新漏检

$$\boldsymbol{w}_k^{j,i,0} = \boldsymbol{w}_{k|k-1}^{j,i}\left(1 - r_{k|k-1}^{j,i} + r_{k|k-1}^{j,i}\left(1 - p_{\mathrm{D}}\left(\boldsymbol{x}\right)\right)\right) \tag{58}$$

$$r_k^{j,i,0} = \frac{r_{k|k-1}^{j,i}\left(1 - p_{\mathrm{D}}\left(\boldsymbol{x}\right)\right)}{1 - r_{k|k-1}^{j,i} + r_{k|k-1}^{j,i}\left(1 - p_{\mathrm{D}}\left(\boldsymbol{x}\right)\right)} \tag{59}$$

$$p_k^{j,i,0}\left(\boldsymbol{x}\right) = \mathcal{N}\left(\boldsymbol{x}; \boldsymbol{m}_k^{j,i,0}, \boldsymbol{P}_k^{j,i,0}\right) \tag{60}$$

其中，$\boldsymbol{m}_k^{j,i,0} = \boldsymbol{m}_{k|k-1}^{j,i}$，$\boldsymbol{P}_k^{j,i,0} = \boldsymbol{P}_{k|k-1}^{j,i}$。

③ 更新潜在检测到的目标

$$w_k^{j,i}\left(\boldsymbol{z}\right) = w_{k|k-1}^{j,i} r_{k|k-1}^{j,i}\mathcal{N}\left(\boldsymbol{z}; \boldsymbol{m}_k^{j,i,u}, \boldsymbol{P}_k^{j,i,u}\right) \tag{61}$$

$$r_k^{j,i}\left(\boldsymbol{z}\right) = 1 \tag{62}$$

$$p_k^{j,i}\left(\boldsymbol{x}\left|\boldsymbol{z}\right.\right) = \mathcal{N}\left(\boldsymbol{x}; \boldsymbol{m}_k^{j,i}, \boldsymbol{P}_k^{j,i}\right) \tag{63}$$

其中，

$$\boldsymbol{m}_k^{j,i} = \boldsymbol{m}_{k|k-1}^{j,i} + \boldsymbol{K}_k^{\mathrm{u}}\left(\boldsymbol{z} - \boldsymbol{m}_k^{j,i}\right) \tag{64}$$

$$\boldsymbol{P}_k^{j,i} = \boldsymbol{P}_{k|k-1}^{j,i} - \boldsymbol{K}_k^{\mathrm{u}}\boldsymbol{S}_k^{\mathrm{u}}(\boldsymbol{K}_k^{\mathrm{u}})^{\mathrm{T}} \tag{65}$$

$$\boldsymbol{K}_k^{\mathrm{u}} = \boldsymbol{P}_{xz,k}^{\mathrm{u}}(\boldsymbol{S}_k^{\mathrm{u}})^{-1} \tag{66}$$

$$\boldsymbol{P}_{xz,k}^{\mathrm{u}} = \frac{1}{2n}\sum_{l=0}^{2n}\left(\boldsymbol{m}_{k|k-1}^{j,i} - \boldsymbol{x}_k^{\mathrm{u},l}\right) - \left(\boldsymbol{m}_k^{j,i} - \boldsymbol{z}_{k|k-1}^{\mathrm{u},l}\right)^{\mathrm{T}} \tag{67}$$

$$\boldsymbol{S}_k^{\mathrm{u}} = \frac{1}{2n}\sum_{l=0}^{2n}\left(\boldsymbol{x}_k^{\mathrm{u},l} - \boldsymbol{m}_k^{j,i}\right)\left(\boldsymbol{x}_k^{\boldsymbol{u},l} - \boldsymbol{m}_k^{j,i}\right)^{\mathrm{T}} + \boldsymbol{R}_k^{\mathrm{u}} \tag{68}$$

$$\boldsymbol{m}_k^{j,i,u} = \frac{1}{2n}\sum_{l=0}^{2n}\boldsymbol{z}_{k|k-1}^{\mathrm{u},l} \tag{69}$$

$$\boldsymbol{P}_k^{j,i,\mathrm{u}} = \boldsymbol{S}_k^{\mathrm{u}} \tag{70}$$

$$\boldsymbol{R}_k^{\mathrm{u}} = \mathrm{GP}_{\Sigma}^{h}\left(\boldsymbol{m}_{k|k-1}^{j,i}, \boldsymbol{D}_h\right) \tag{71}$$

$$\boldsymbol{x}_k^{\mathrm{u},l} = \boldsymbol{m}_{k|k-1}^{j,i} + \sqrt{\boldsymbol{P}_{k|k-1}^{j,i}}\alpha_l \tag{72}$$

$$\boldsymbol{z}_{k|k-1}^{\mathrm{u},l} = \mathrm{GP}_{\mathrm{m}}^{h}\left(\boldsymbol{x}_k^{\mathrm{u},l}, \boldsymbol{D}_h\right) \tag{73}$$

由于当前的更新过程只得到了单目标假设，因此，接下来通过对所有可能的单目标假设进行组合，得到新的全局假设。为了降低计算的复杂度，利用Murty's算法实现这一过程，具体的实现步骤见文献[20]。

4 仿真分析

4.1 评价指标

为全面评估算法的性能，将所提数据驱动的自学习算法和传统多模型算法以及基于深度学习LSTM的数据驱动算法构造对照组，并使用广义最优子模式分配(Generalized Optimal SubPattern Assignment, GOSPA)误差[38]作为评价指标，其定义为

$$d_p^{(c,\alpha)}\left(\boldsymbol{X},\boldsymbol{Y}\right) \triangleq \left[\min_{\gamma\in\Gamma}\left(\sum_{(i,j)\in\gamma} d^p(\boldsymbol{x}_i, \boldsymbol{y}_j) + \frac{c^p}{\alpha}\left(|\boldsymbol{X}| + |\boldsymbol{Y}| - 2\left|\gamma\right|\right)\right)\right]^{\frac{1}{p}} \tag{74}$$

其中，$c = 50$, $p = 2$, $\alpha = 2$。

4.2 仿真结果

（1）场景1。2维监控区域$[-2\,000, 2\,000]$ m×$[-2\,000, 2\,000]$ m中包含杂波和不确定数量的目标。每个目标的状态由2维位置和速度组成，即$x_k = [p_{x,k}, \dot{p}_{x,k}, p_{y,k}, \dot{p}_{y,k}]^{\mathrm{T}}$，其中$[p_{x,k}, p_{y,k}]$表示目标的位置，$[\dot{p}_{x,k}, \dot{p}_{y,k}]$表示目标的速度。每个目标根据匀速直线(Constant Velocity, CV)或协同转弯(Coordinated Turn, CT)模型切换运动

$$\boldsymbol{x}_k = \boldsymbol{F}_{\mathrm{CV/CT}}\boldsymbol{x}_{k-1} + \boldsymbol{w}_k$$

$$\boldsymbol{F}_{\mathrm{CV}} = \begin{bmatrix} 1 & \varDelta & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & \varDelta \\ 0 & 0 & 0 & 1 \end{bmatrix},$$

$$\boldsymbol{F}_{\mathrm{CT}} = \begin{bmatrix} 1 & \dfrac{(\sin\theta)}{\theta} & 0 & -\dfrac{(1-\cos\theta)}{\theta} \\ 0 & \cos\theta & 0 & -\sin\theta \\ 0 & \dfrac{(1-\cos\theta)}{\theta} & 1 & \dfrac{(\sin\theta)}{\theta} \\ 0 & \sin\theta & 0 & \cos\theta \end{bmatrix}$$

其中，$\boldsymbol{w}_k \sim \mathcal{N}(0, \boldsymbol{Q}_k)$

$$\boldsymbol{Q}_k = \sigma^2 \begin{bmatrix} \Delta^4/4 & \Delta^3/2 & 0 & 0 \\ \Delta^3/2 & \Delta^2 & 0 & 0 \\ 0 & 0 & \Delta^4/4 & \Delta^3/2 \\ 0 & 0 & \Delta^3/2 & \Delta^3/2 \end{bmatrix}$$

其中，$\sigma = 0.1$，$\Delta = 1\ \mathrm{s}$表示采样间隔。模型1是常速模型，模型2的转弯率为$\theta = -9(^\circ)/\mathrm{s}$，模型3的转弯率为$\theta = 6(^\circ)/\mathrm{s}$。对于每个目标，存活概率和检测概率分别为$p_{\mathrm{S},k} = 0.97$，$p_{\mathrm{D},k} = 0.95$。观测模型为

$$\boldsymbol{z}_k = \begin{bmatrix} \sqrt{(p_{x,k} - p_{x,0})^2 + (p_{y,k} - p_{y,0})^2} \\ \arctan\left(\dfrac{p_{x,k} - p_{x,0}}{p_{y,k} - p_{y,0}}\right) \end{bmatrix} + \boldsymbol{v}_k$$

其中，$p_{x,0}$，$p_{y,0}$表示传感器的位置，设置为$(0,0)\ \mathrm{m}$; $\boldsymbol{v}_k \sim \mathcal{N}(0, \boldsymbol{R}_k)$, $\boldsymbol{R}_k = \mathrm{diag}\left(\left[\sigma_r^2, \sigma_\theta^2\right]^{\mathrm{T}}\right)$, $\sigma_r = 10\ \mathrm{m}$, $\sigma_\theta = (\pi/180)\ \mathrm{rad/s}$。杂波的分布建模为泊松模型，泊松率为$\lambda_{\mathrm{c}} = 10$。此外，采用高斯混合形式作为目标的新生模型

$$\gamma_k(\boldsymbol{x}) = \sum_{i=1}^{6} w_{\mathrm{b}}^i \mathcal{N}\left(\boldsymbol{x}; \boldsymbol{m}_{\mathrm{b}}^i, \boldsymbol{P}_{\mathrm{b}}^i\right)$$

其中，$w_{\mathrm{b}}^i = 0.1$，均值分别为$\boldsymbol{m}_{\mathrm{b}}^1 = [-300\ \ 0\ \ 100\ \ 0]^{\mathrm{T}}$，$\boldsymbol{m}_{\mathrm{b}}^2 = [800\ \ 0\ \ 400\ \ 0]^{\mathrm{T}}$，$\boldsymbol{m}_{\mathrm{b}}^3 = [50\ \ 0\ \ -600\ \ 0]^{\mathrm{T}}$，$\boldsymbol{m}_{\mathrm{b}}^4 = [-800\ \ 0\ \ -400\ \ 0]^{\mathrm{T}}$，$\boldsymbol{m}_{\mathrm{b}}^5 = [400\ \ 0\ \ 500\ \ 0]^{\mathrm{T}}$，$\boldsymbol{m}_{\mathrm{b}}^6 = [-500\ \ 0\ \ 600\ \ 0]^{\mathrm{T}}$，协方差矩阵为$\boldsymbol{P}_{\mathrm{b}}^i = \mathrm{diag}\left([200, 100, 200, 100]^{\mathrm{T}}\right)$。

此外，训练数据的长度$L_1 = 1\,000$，测试数据的长度$L_2 = 80$。用于训练和测试的真实轨迹是不同的，即训练数据和测试数据来自不同的数据集，但遵循相同的运动模型。在目标运动过程中，测试目标在10～30 s以模型2运动，在40～50 s以模型3运动，在其他时间步以模型1运动。测试目标的运动轨迹如图2所示。

图1展示了多机动目标的真实运动轨迹，包括目标的出生位置和消亡位置。图2则展示了单次实验中，所提出的GP-PMBM滤波器的跟踪估计结果。结果表明，所提算法能够在复杂动态环境中较准确地估计目标状态。接下来，通过500次蒙特卡罗(Monte Carlo, MC)实验，进一步验证了所提方法的有效性。

图3对比分析了多种算法在GOSPA误差指标上的堆叠面积随时间的动态演变趋势。经细致观察，不难发现所提出的GP-PMBM滤波器在GOSPA误差指标上的堆叠面积显著低于其他算法。相比之下，LSTM-PMBM算法虽然利用时序依赖建模在匀速运动阶段实现了较高的跟踪一致性，但在目标机动过程中，其误差面积增长明显快于GP-PMBM滤波器。此外，MM-PMBM和MM-GLMB滤波器在目标状态突变场景下表现出较大的误差波动，显示出一定的性能瓶颈；而MM-PHD滤波器由于仅传播概率密度函数，导致估计精度和稳定性均较差。这一结果表明，GP-PMBM滤波器在处理复杂机动的多目标跟踪问题上展现了更优的适用性和鲁棒性，适合在目标运动状态频繁变化的复杂环境中应用。

图4比较了不同算法的目标势估计结果及其标准差，直观反映了各算法在目标势估计中的动态变化及性能差异。从图中可以看出，GP-PMBM滤波

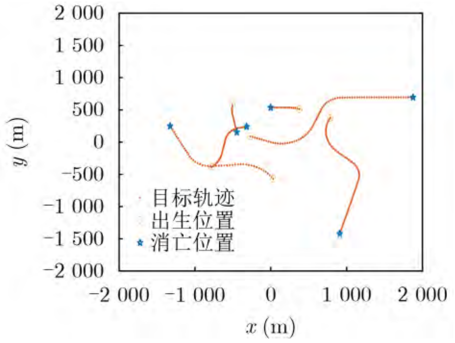


图 1 多机动目标真实轨迹

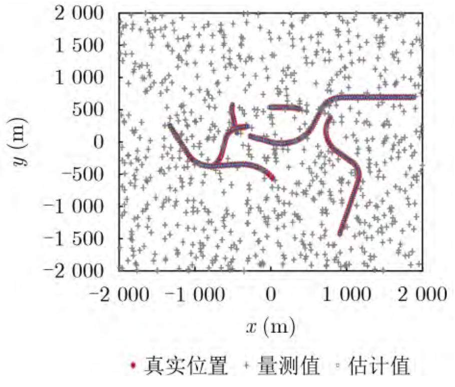


图 2 跟踪结果

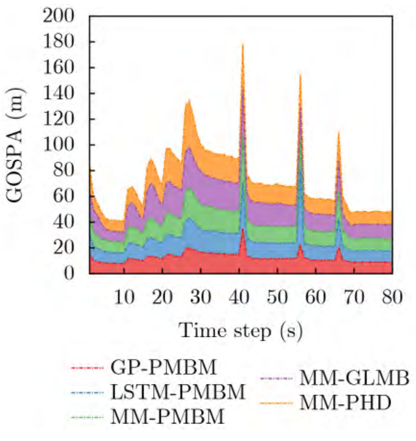
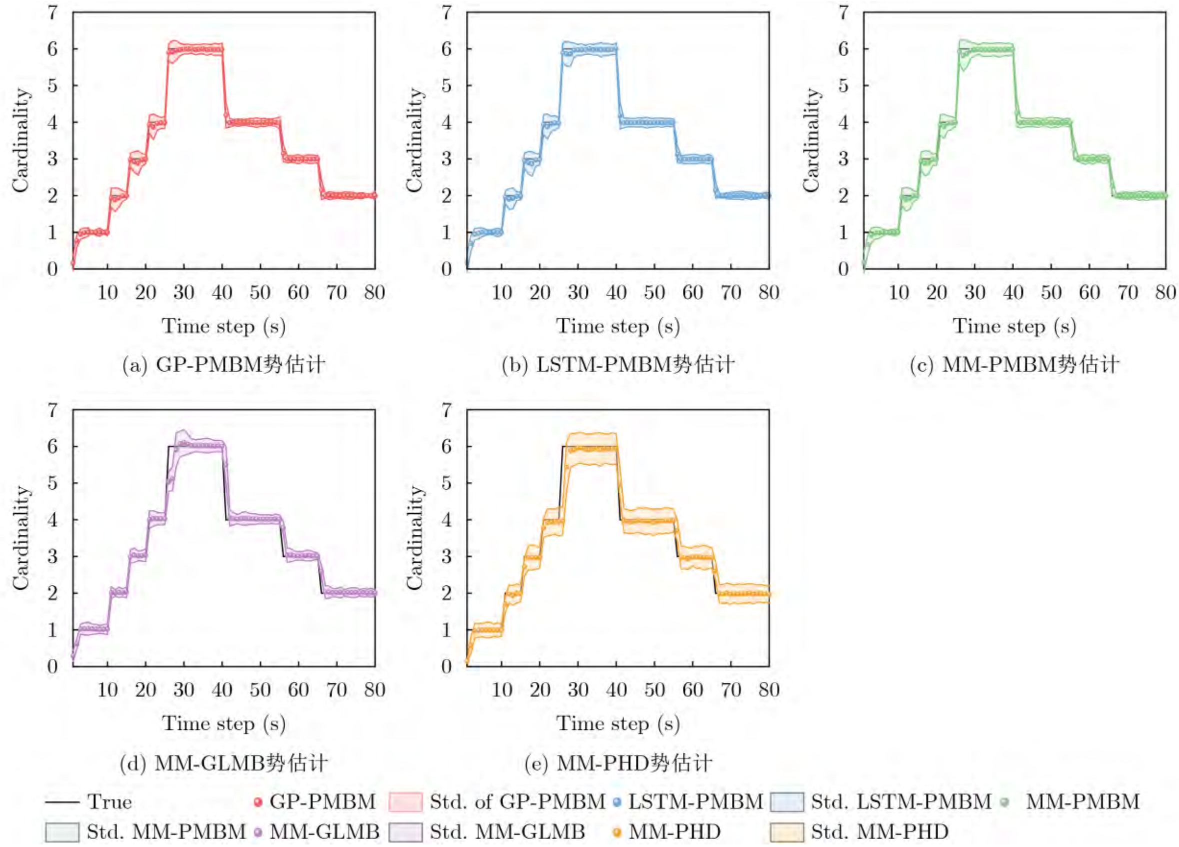


图 3 GOSPA堆叠面积



(a) GP-PMBM势估计

(b) LSTM-PMBM势估计

(c) MM-PMBM势估计

(d) MM-GLMB势估计

(e) MM-PHD势估计

图 4 多机动目标势估计

器表现出较高的估计精度，其标准差显著低于其他算法，体现了对目标场景变化很好的适应性和稳定性。LSTM-PMBM算法的势估计性能略逊于GP-PMBM滤波器，但整体性能优于传统的MM-PMBM方法。MM-PMBM滤波器虽然在大部分时段内具有较小的整体误差，但在目标势发生剧烈变化时，其误差波动较为明显。MM-GLMB与MM-PHD滤波器在剧烈变化场景中的估计性能较弱，说明其估计稳定性不足。

为了进一步验证所提算法在复杂跟踪环境中的性能，图5展示了5种滤波算法在不同杂波率下的平均GOSPA误差。仿真结果表明，随着杂波率的增加，各算法的跟踪性能均呈下降趋势，但降幅存在显著差异。具体而言，GP-PMBM滤波器在整个杂波率范围内均表现出最佳性能，其平均GOSPA误差始终保持在最低水平；而LSTM-PMBM算法则表现出次优性能，其误差增长趋势介于GP-PMBM与传统多模型方法之间，体现出较强的虚警抑制能力和一定的稳健性。相对而言，MM-PMBM和MM-GLMB滤波器的性能位居其后，而MM-PHD滤波器不仅平均GOSPA误差最高，误差随杂波率增加而显著上升。

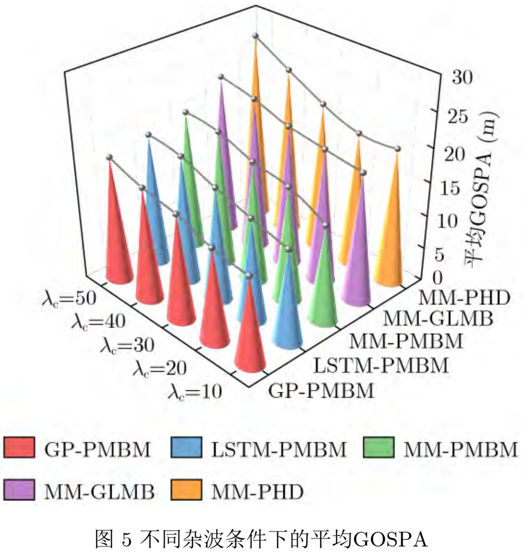


图 5 不同杂波条件下的平均GOSPA

图6展示了不同算法在不同检测概率条件下的平均GOSPA距离。图中数据表明，随着检测概率的降低，所有算法的平均GOSPA误差都出现了显著的增加。原因在于，较低的检测概率会增加目标漏检的情况，从而影响算法的准确性。然而，GP-PMBM滤波器在不同检测概率下的多机动目标跟踪性能最优，特别是在检测概率较低的情况下，仍能保持相对较好的跟踪效果，表现出较强的鲁棒性。

(2) 场景2。为了进一步验证所提方法的有效性，在场景2中构建了更为复杂的多机动目标跟踪环境。

在该场景中，监视区域中目标的数量更多，目标的机动性进一步增强，这对多机动目标跟踪算法的估计性能提出了更高的要求。目标的运动模式仍然包括模型1～模型3，但是模型2和模型3的转弯率发生了显著的变化，分别为$\theta = 9(^\circ)/\mathrm{s}$和$\theta = -15(^\circ)/\mathrm{s}$。这种复杂的环境设置使得目标的运动轨迹更加多样化，且更难以预测，从而对跟踪算法的适应性和鲁棒性提出了更大的挑战。通过与其他模型驱动的算法进行比较，可以全面评估所提算法在高动态复杂环境中的性能。此外，目标的初始位置和存活时间如表1所示。测试目标在10～30 s以模型2运动，在31～50 s和61～80 s以模型3运动，在其他时间步以模型1运动。多机动目标运动和跟踪环境的其余参数设置与场景1相同。

图7和图8分别展示了在单次蒙特卡罗实验中多机动目标的真实轨迹及GP-PMBM滤波器的跟踪结果。图中清晰展现了目标的高机动性及其在时空域中的交错，对滤波器的跟踪性能提出了更高的要求。尽管面临这些挑战，所提出的GP-PMBM滤波器依然能够较为准确地估计多机动目标的状态。

图9展示了MC实验中多机动目标势估计的热度图。通过观察热力图可以发现，GP-PMBM和MM-PMBM滤波器的势估计分布与真实值接近，整体趋势较为一致，体现了较高的估计精度及对目标动态变化的良好适应性。相较而言，LSTM-PMBM算法的势估计性能介于GP-PMBM和MM-PMBM滤波器之间；而MM-GLMB滤波器的势估计在部分时间步出现明显偏离，反映出其对目标势变化的适应性略逊一筹。此外，MM-PHD滤波器的势估计分布则存在较大偏差，表现为多个时间段内颜色的显著偏差，这表明其势估计性能较弱且稳定性不足。

图10展示了各滤波器的GOSPA误差变化趋势。观察结果表明，当目标势发生剧烈变化时，各滤波

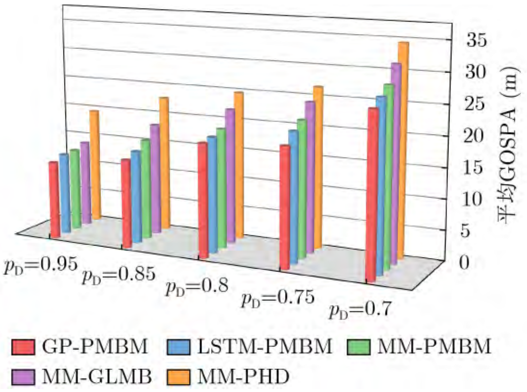


图 6 不同检测概率条件下的平均GOSPA

表 1　目标初始状态和存活时间

目标	初始状态 (m, v/s, m, v/s)	存活时间 (s)
1	$[\,-300\quad 0\quad 100\quad 0\,]^{\rm T}$	1～20
2	$[\,500\quad 0\quad 400\quad 0\,]^{\rm T}$	3～30
3	$[\,50\quad 0\quad -600\quad 0\,]^{\rm T}$	5～30
4	$[\,-800\quad 0\quad -400\quad 0\,]^{\rm T}$	10～35
5	$[\,400\quad 0\quad 500\quad 0\,]^{\rm T}$	15～65
6	$[\,-500\quad 0\quad 600\quad 0\,]^{\rm T}$	18～65
7	$[\,100\quad 0\quad 200\quad 0\,]^{\rm T}$	50～70
8	$[\,300\quad 0\quad -350\quad 0\,]^{\rm T}$	55～75
9	$[\,700\quad 0\quad -800\quad 0\,]^{\rm T}$	60～80

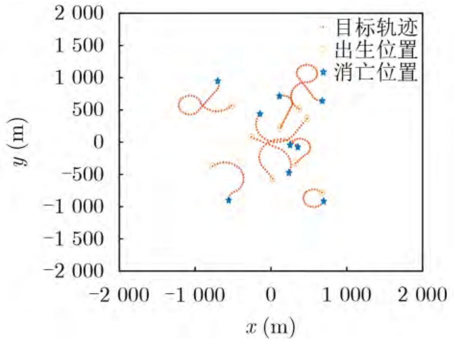


图 7 场景2中多机动目标真实轨迹

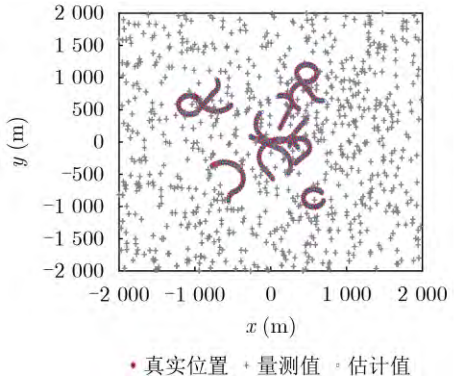


图 8 场景2中跟踪结果

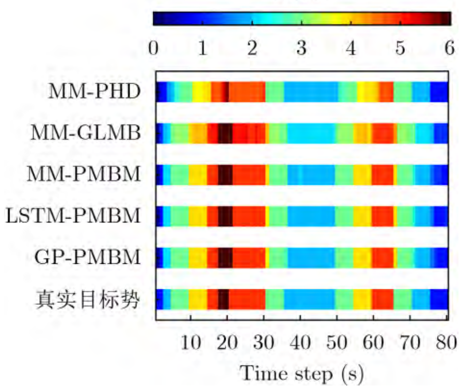


图 9 场景2中势估计分条热度

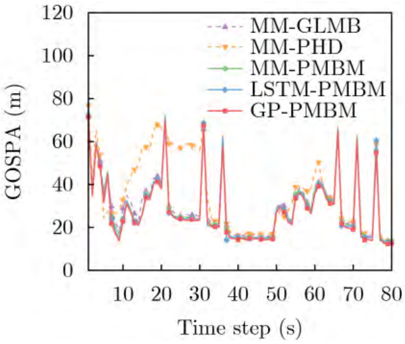


图 10 场景2中GOSPA误差

器均出现了不同程度的误差突变，这反映出它们对目标势快速变化的适应能力及在处理目标新增或消失等复杂情境时的性能差异。实验结果表明，GP-PMBM, LSTM-PMBM, MM-PMBM和MM-GLMB滤波器在目标势变化较为频繁的前30个时间步内能够迅速调整其多目标状态估计，有效降低了因目标新增或消失所带来的误差积累。

5 结束语

本文提出了一种无模型的GP-PMBM滤波器，旨在减少多机动目标跟踪中对特定运动模型选择的依赖。该滤波器通过GP卓越的学习能力，可以实现在大量模型之间进行切换，能够有效应对目标运动模式的复杂变化。实验结果表明，所提出的GP-PMBM滤波器在跟踪性能上优于基于深度学习的LSTM数据驱动方法和传统的MM方法。未来的研究将进一步探索GP-PMBM滤波器在多机动扩展目标跟踪中的应用潜力，以应对更具挑战性的跟踪任务。

参考文献

[1] DAI Jinhui, PU Wenqiang, YAN Junkun, *et al*. Multi-UAV collaborative trajectory optimization for asynchronous 3-D passive multitarget tracking[J]. *IEEE Transactions on Geoscience and Remote Sensing*, 2023, 61: 5101116. doi: 10.1109/TGRS.2023.3239952.

[2] DING Lintao, SHI Chenguang, and ZHOU Jianjiang. Collaborative route optimization and resource management strategy for multi-target tracking in airborne radar system[J]. *Digital Signal Processing*, 2023, 138: 104051. doi: 10.1016/j.dsp.2023.104051.

[3] RAMACHANDRAN R K, FRONDA N, PREISS J A, *et al*. Resilient multi-robot multi-target tracking[J]. *IEEE Transactions on Automation Science and Engineering*, 2024, 21(3): 4311–4327. doi: 10.1109/TASE.2023.3295373.

[4] KUMAR S and PARHI D R. Multi-target trajectory planning and control technique for autonomous navigation of multiple robots[J]. *ISA Transactions*, 2023, 138: 650–669. doi: 10.1016/j.isatra.2023.02.029.

[5] WANG Qiang, HUANG Xiao, TAO Hao, *et al*. Marine navigation radar multi-target tracking using adaptive innovation sequence-based joint probability data association[C]. 2024 IEEE 13th Data Driven Control and Learning Systems Conference (DDCLS), Kaifeng, China, 2024: 511–516. doi: 10.1109/DDCLS61622.2024.10606901.

[6] BLACK B, SELLERS T, LEI Tingjun, *et al*. Optimal multi-target navigation via graph-based algorithms in complex environments[C]. Proceedings of 2024 IEEE 33rd International Symposium on Industrial Electronics (ISIE), Ulsan, Korea, Republic of, 2024: 1–6. doi: 10.1109/ISIE54533.2024.10595682.

[7] BLACKMAN S S. Multiple hypothesis tracking for multiple target tracking[J]. *IEEE Aerospace and Electronic Systems Magazine*, 2004, 19(1): 5–18. doi: 10.1109/MAES.2004.1263228.

[8] MAHLER R P S. Statistical Multisource-Multitarget Information Fusion[M]. Boston: Artech, 2007.

[9] RONG LI X and JILKOV V P. Survey of maneuvering target tracking. Part I. Dynamic models[J]. *IEEE Transactions on Aerospace and Electronic Systems*, 2003, 39(4): 1333–1364. doi: 10.1109/TAES.2003.1261132.

[10] LI X R and JILKOV V P. Survey of maneuvering target tracking. Part V. Multiple-model methods[J]. *IEEE Transactions on Aerospace and Electronic Systems*, 2005, 41(4): 1255–1321. doi: 10.1109/TAES.2005.1561886.

[11] MAHLER R. On multitarget jump-Markov filters[C]. 2012 15th International Conference on Information Fusion, Singapore, 2012: 149–156.

[12] LI Xiaolong, SUN Zhi, YEO T S, *et al*. STGRFT for detection of maneuvering weak target with multiple motion models[J]. *IEEE Transactions on Signal Processing*, 2019, 67(7): 1902–1917. doi: 10.1109/TSP.2019.2899318.

[13] MAHLER R P S. Advances In Statistical Multisource-Multitarget Information Fusion[M]. Boston: Artech House, 2014.

[14] WEI Jingxin, LUO Feng, CHEN Shichao, *et al*. Robust fusion of GM-PHD filters based on geometric average[J]. *Signal Processing*, 2023, 206: 108912. doi: 10.1016/j.sigpro.2022.108912.

[15] HUANG Qiao, XIE Lei, and SU Hongye. Estimations of time-varying birth cardinality distribution and birth intensity in Gaussian mixture CPHD filter for multi-target tracking[J]. *Signal Processing*, 2022, 190: 108321. doi: 10.1016/j.sigpro.2021.108321.

[16] DAVIES E S and GARCÍA-FERNÁNDEZ á F. Information exchange track-before-detect Multi-Bernoulli filter for

superpositional sensors[J]. *IEEE Transactions on Signal Processing*, 2024, 72: 607–621. doi: 10.1109/TSP.2024.3349769.

[17] WEI Jingxin, LUO Feng, QI Jiawei, *et al*. Distributed fusion of Labeled Multi-Bernoulli filters based on arithmetic average[J]. *IEEE Signal Processing Letters*, 2024, 31: 656–660. doi: 10.1109/LSP.2024.3364506.

[18] CAO Chenghu and ZHAO Yongbo. Multi-sensor multi-target tracking with generalized labeled multi-Bernoulli filter based on track-before-detect observation model using Gaussian belief propagation[J]. *Digital Signal Processing*, 2024, 153: 104618. doi: 10.1016/j.dsp.2024.104618.

[19] ZHOU Yusong, ZHAO Jin, WU Sunyong, *et al*. A Poisson multi-Bernoulli mixture filter for tracking multiple resolvable group targets[J]. *Digital Signal Processing*, 2024, 144: 104279. doi: 10.1016/j.dsp.2023.104279.

[20] GARCÍA-FERNÁNDEZ á F, XIA Yuxuan, and SVENSSON L. Poisson multi-Bernoulli mixture filter with general target-generated measurements and arbitrary clutter[J]. *IEEE Transactions on Signal Processing*, 2023, 71: 1895–1906. doi: 10.1109/TSP.2023.3278944.

[21] VO B N and MA W K. The Gaussian mixture probability hypothesis density filter[J]. *IEEE Transactions on Signal Processing*, 2006, 54(11): 4091–4104. doi: 10.1109/TSP.2006.881190.

[22] VO B N, SINGH S, and DOUCET A. Sequential Monte Carlo methods for multitarget filtering with random finite sets[J]. *IEEE Transactions on Aerospace and Electronic Systems*, 2005, 41(4): 1224–1245. doi: 10.1109/TAES.2005.1561884.

[23] PUNITHAKUMAR K, KIRUBARAJAN T, and SINHA A. Multiple-model probability hypothesis density filter for tracking maneuvering targets[J]. *IEEE Transactions on Aerospace and Electronic Systems*, 2008, 44(1): 87–98. doi: 10.1109/TAES.2008.4516991.

[24] PASHA S A, VO B N, TUAN H D, *et al*. A Gaussian mixture PHD filter for jump Markov system models[J]. *IEEE Transactions on Aerospace and Electronic Systems*, 2009, 45(3): 919–936. doi: 10.1109/TAES.2009.5259174.

[25] GEORGESCU R and WILLETT P. The multiple model CPHD tracker[J]. *IEEE Transactions on Signal Processing*, 2012, 60(4): 1741–1751. doi: 10.1109/TSP.2012.2183128.

[26] DONG Peng, JING Zhongliang, GONG Deren, *et al*. Maneuvering multi-target tracking based on variable structure multiple model GMCPHD filter[J]. *Signal Processing*, 2017, 141: 158–167. doi: 10.1016/j.sigpro.2017.06.004.

[27] XUE Yu and FENG Xi'an. Joint multi-Gaussian mixture model and its application to multi-model multi-Bernoulli filter[J]. *Digital Signal Processing*, 2024, 153: 104616. doi: 10.1016/j.dsp.2024.104616.

[28] REUTER S, SCHEEL A, and DIETMAYER K. The multiple model labeled multi-Bernoulli filter[C]. 2015 18th International Conference on Information Fusion (Fusion), Washington, USA, 2015: 1574–1580.

[29] CAO Chenghu and ZHAO Yongbo. Multiple-model generalised labelled multi-Bernoulli filter with distributed sensors for tracking manoeuvring targets using belief propagation[J]. *IET Radar, Sonar & Navigation*, 2023, 17(5): 845–858. doi: 10.1049/rsn2.12382.

[30] LI Guchong, KONG Lingjiang, YI Wei, *et al*. Multiple model Poisson multi-Bernoulli mixture filter for maneuvering targets[J]. *IEEE Sensors Journal*, 2021, 21(3): 3143–3154. doi: 10.1109/JSEN.2020.3022669.

[31] SEEGER M. Gaussian processes for machine learning[J]. *International Journal of Neural Systems*, 2004, 14(2): 69–106. doi: 10.1142/S0129065704001899.

[32] AFTAB W and MIHAYLOVA L. A Gaussian process regression approach for point target tracking[C]. 2019 22th International Conference on Information Fusion (FUSION), Ottawa, Canada, 2019: 1–8. doi: 10.23919/FUSION43075.2019.9011310.

[33] AFTAB W and MIHAYLOVA L. A learning Gaussian process approach for maneuvering target tracking and smoothing[J]. *IEEE Transactions on Aerospace and Electronic Systems*, 2021, 57(1): 278–292. doi: 10.1109/TAES.2020.3021220.

[34] SUN Mengwei, DAVIES M E, PROUDLER I K, *et al*. A Gaussian process regression based dynamical models learning algorithm for target tracking[EB/OL]. https://doi.org/10.48550/arXiv.2211.14162, 2022.

[35] HU Zheng and LI Tiancheng. A particle Bernoulli filter based on Gaussian process learning for maneuvering target tracking[C]. Proceedings of 2022 30th European Signal Processing Conference (EUSIPCO), Belgrade, Serbia, 2022: 777–781. doi: 10.23919/EUSIPCO55093.2022.9909660.

[36] GUO Qiang, TENG Long, YIN Tianxiang, *et al*. Hybrid-driven Gaussian process online learning for highly maneuvering multi-target tracking[J]. *Frontiers of Information Technology & Electronic Engineering*, 2023, 24(11): 1647–1656. doi: 10.1631/FITEE.2300348.

[37] ARASARATNAM I and HAYKIN S. Cubature Kalman filters[J]. *IEEE Transactions on Automatic Control*, 2009, 54(6): 1254–1269. doi: 10.1109/TAC.2009.2019800.

[38] RAHMATHULLAH A S, GARCíA-FERNáNDEZ á F, and SVENSSON L. Generalized optimal sub-pattern assignment metric[C]. 2017 20th International Conference on

Information Fusion (Fusion), Xi’an, China, 2017: 1–8. doi: 10.23919/ICIF.2017.8009645.

赵子文：男，博士生，研究方向为多目标跟踪技术.

陈　辉：男，教授，博士生导师，博士，研究方向为多目标跟踪、数据融合、最优控制等.

连　峰：男，教授，博士生导师，博士，研究方向为目标跟踪、信息融合与传感器管理.

张光华：男，副教授，博士生导师，博士，研究方向为信息融合与目标跟踪.

张文旭：男，副教授，硕士生导师，博士，研究方向为深度强化学习、智能决策与数据挖掘和机器人技术.

责任编辑：马秀强

Multiple Maneuvering Target Poisson Multi-Bernoulli Mixture Filter for Gaussian Process Cognitive Learning

ZHAO Ziwen[1] CHEN Hui[1] LIAN Feng[2]
ZHANG Guanghua[2] ZHANG Wenxu[1]

[1](*School of Automation and Electrical Engineering, Lanzhou University of Technology, Lanzhou* 730050, *China*)

[2](*School of Automation Science and Engineering, Xi’an Jiaotong University, Xi’an* 710049, *China*)

Abstract:

Objective Multiple Maneuvering Target Tracking (MMTT) remains a critical yet challenging problem in radar signal processing and sensor fusion, particularly under complex and uncertain conditions. The primary difficulty arises from the unpredictable or highly dynamic nature of target motion. Conventional model-based methods, especially Multiple Model (MM) approaches, rely on predefined motion models to accommodate varying target behaviors. However, these methods face limitations, including sensitivity to initial parameter settings, high computational cost due to model switching, and degraded performance when actual target behavior deviates from the assumed model set. To address these limitations, this study proposes a data-driven MMTT method that combines Gaussian Process (GP) learning with the Poisson Multi-Bernoulli Mixture (PMBM) filter to improve robustness and tracking accuracy in dynamic environments without requiring extensive model assumptions.

Methods The proposed method exploits the data-driven modeling capability of GP, a non-parametric Bayesian inference approach that learns high-dimensional, nonlinear function mappings from limited historical data without specifying explicit functional forms. In this study, GP models both the state transition and observation processes of multi-target systems, reducing the dependence on predefined motion models. During the offline phase, historical target trajectories and sensor measurements are collected to build a training dataset. The squared exponential kernel is selected for its smoothness and infinite differentiability, which effectively captures the continuity and dynamic characteristics of target state evolution. GP hyperparameters, including length scale, signal variance, and observation noise variance, are jointly optimized by maximizing the log-marginal likelihood, ensuring generalization and expressiveness in complex environments. In the online filtering phase, the trained GP models are incorporated into the PMBM filter, forming a recursive GP-PMBM filtering structure. Within this framework, the PMBM filter employs a Poisson point process to represent undetected targets and a multi-Bernoulli mixture to characterize the posterior state distribution of detected targets. During the prediction step, the GP-derived nonlinear state transition model is propagated using the Cubature Kalman Filter (CKF). In the update step, the GP-learned observation model refines state estimates, enhancing both tracking accuracy and robustness.

Results and Discussions Extensive simulation experiments under two different MMTT scenarios validate the effectiveness and performance advantages of the proposed method. In Scenario 1, a moderate 2D surveillance environment with clutter and a varying number of targets is constructed. The GP-PMBM filter significantly outperforms existing methods, including LSTM-PMBM, MM-PMBM, MM-GLMB, and MM-PHD filters, based on the Generalized Optimal Sub-Pattern Assignment (GOSPA) metric (Fig. 3). In addition, the GP-PMBM filter achieves the lowest standard deviation in cardinality estimation, demonstrating high accuracy and stability (Fig. 4). Further experiments under different monitoring conditions confirm the robustness of GP-PMBM. When clutter rates vary, the GP-PMBM filter consistently achieves the lowest average GOSPA error, reflecting strong stability under interference (Fig. 5). As detection probability decreases, most algorithms show significant degradation in accuracy. However, GP-PMBM maintains superior tracking performance, achieving the lowest GOSPA distance across all detection conditions (Fig. 6). In Scenario 2, target motion becomes more complex, with increased maneuverability and higher–frequency birth–death dynamics. Despite these challenges, the GP-PMBM filter maintains superior tracking performance, even under highly maneuverable conditions and frequent target appearance and disappearance (Fig. 9, Fig. 10).

Conclusions This study proposes a novel GP-PMBM filtering framework for MMTT in complex environments. By integrating the data-driven learning capability of the GP with the PMBM filter, the proposed method addresses the limitations of conventional model-based tracking approaches. The GP-PMBM filter automatically learns unknown motion and observation models from historical data, eliminating the dependence on predefined model sets and significantly improving adaptability. Simulation results confirm that the GP-PMBM filter achieves superior tracking accuracy, improved cardinality estimation, and enhanced robustness under varying clutter levels and detection conditions. These results indicate that the proposed method is well-suited for environments characterized by frequent maneuvering changes and uncertain target behavior. Future work will focus on extending the GP-PMBM framework to multi-maneuvering extended target tracking tasks to address more challenging scenarios.

Key words: Multi-maneuvering target tracking; Data-driven; Gaussian Process (GP); Poisson Multi-Bernoulli Mixture (PMBM)

证书号第7214817号

专利公告信息

发明专利证书

发 明 名 称：基于高斯过程的概率假设密度滤波多机动目标跟踪方法

专 利 权 人：兰州理工大学

地　　　址：730050 甘肃省兰州市七里河区兰工坪路287号

发 明 人：赵子文;陈辉;张文旭;王旭昕;王莉;刘孟波

专 利 号：ZL 2023 1 1238475.X　　授权公告号：CN 117197491 B

专利申请日：2023年09月25日　　授权公告日：2024年07月19日

申请日时申请人：兰州理工大学

申请日时发明人：赵子文;陈辉;张文旭;王旭昕;王莉;刘孟波

国家知识产权局依照中华人民共和国专利法进行审查，决定授予专利权，并予以公告。

专利权自授权公告之日起生效。专利权有效性及专利权人变更等法律信息以专利登记簿记载为准。

局长
申长雨

申长雨

国家知识产权局
2024年07月19日