李义婷, 王帅, 苏小红,等. 基于交互式多模型粒子流滤波的机动目标跟踪算法[J]. 智能计算机与应用,2026,16(6):232-238. DOI:10.20169/j.issn.2095-2163.24101501

基于交互式多模型粒子流滤波的机动目标跟踪算法

李义婷[1,2], 王 帅[1,2], 苏小红[3], 韩 伟[4]

(1 福建师范大学 计算机与网络空间安全学院, 福州 350117;
2 福建师范大学 数字福建环境监测物联网实验室, 福州 350117;
3 哈尔滨工业大学 计算学部, 哈尔滨 150001; 4 福建师范大学 数学与统计学院, 福州 350117)

摘 要: 为了提高机动目标跟踪中状态估计的性能,提出一种交互式多模型粒子流滤波(Interactive Muti-Model Particle Flow Filtering, IMM-PFF)算法。该算法采用多模型描述目标的机动过程,利用粒子流滤波算法进行目标状态估计,交互输出滤波结果。避免单一模型造成的估计误差,同时克服了卡尔曼滤波和粒子滤波的局限性,有效提升状态估计精度。仿真实验结果证明本文所提新滤波算法的有效性和鲁棒性。

关键词: 目标跟踪; 状态估计; 交互式多模型; 粒子流滤波
中图分类号: TN953 **文献标志码**: A **文章编号**: 2095-2163(2026)06-0232-08

Mobile target tracking algorithm based on Interactive Multi-Model Particle Flow Filtering

LI Yiting[1,2], WANG Shuai[1,2], SU Xiaohong[3], HAN Wei[4]

(1 College of Computer and Cyber Security, Fujian Normal University, Fuzhou 350117, China;
2 Digital Fujian Internet-of-Things Laboratory of Environmental Monitoring, Fujian Normal University, Fuzhou 350117, China;
3 Faculty of Computing, Harbin Institute of Technology, Harbin 150001, China;
4 School of Mathematics and Statistics, Fujian Normal University, Fuzhou 350117, China)

Abstract: To enhance the performance of state estimation in target tracking, this paper proposes an Interactive Multi-Model Particle Flow Filtering algorithm. This algorithm employs multiple models to describe the maneuvering process of the target and utilizes the particle flow filtering algorithm for state estimation, with interactive outputs for the filtering results. This approach avoids the estimation errors caused by relying on a single model and overcomes the limitations of Kalman filtering and traditional particle filtering, thereby effectively improving the accuracy of state estimation. Simulation results demonstrate the effectiveness and robustness of the proposed filtering algorithm.

Key words: target tracking; state estimation; Interactive Multi-Model; Particle Flow Filtering

0 引 言

机动目标跟踪是目标跟踪领域中的重要课题之一,由于目标运动情况具有无先验性和不确定性等特点,导致难以建立准确的机动目标模型,获取精确的目标运动状态估计参数[1]。常见的机动模型有匀速运动模型(Constant Velocity, CV)、匀速转弯运动模型(Constant Turn, CT)、匀加速运动模型(Constant Acceleration, CA)。但是单一的运动模型在处理复杂的机动目标跟踪问题上容易出现较大的误差,甚至丢失目标,当滤波模型与目标机动模型不一致时,单一模型的滤波方法的精度会明显降低[2]。针对上述挑战,Blom 等学者[3]提出了交互式多模型(Interactive Multiple Model, IMM)算法,该算法的核心在于引入马尔科夫转移概率矩阵,通过对多个模型的状态估计进行概率加权,实现预测结果

基金项目: 福建省自然科学基金面上项目(2023J01297)。

作者简介: 李义婷(1998—),女,硕士研究生。主要研究方向:寿命预测,机器学习; 苏小红(1966—),女,博士,教授。主要研究方向:智能软件工程,软件质量保障和软件维护; 韩 伟(1983—),男,硕士。主要研究方向:故障诊断,寿命预测。
通信作者: 王 帅(1985—),女,博士,讲师。主要研究方向:故障诊断,寿命预测。Email: wangshuaiss@fjnu.edu.cn。

收稿日期: 2024-10-15

的交互融合。这一算法的提出，标志着目标跟踪从传统的单一模型匹配问题向更为灵活的机动目标跟踪方法的转变，可嵌套各类滤波器使用[4-6]。常用的滤波器算法主要有 2 类：卡尔曼滤波及其改进滤波器、粒子滤波及其改进滤波器算法。卡尔曼滤波在线性高斯系统环境下能够提供最优状态估计，但对于非线性系统或非高斯噪声情况，其性能表现则不尽如人意。改进的卡尔曼滤波算法对于非线性系统的滤波性能有一定提升，计算量远小于粒子滤波器，但在高度非线性系统中的滤波性能不佳[7]。粒子滤波使用蒙特卡罗采样和粒子权重来近似后验分布，从而对非线性非高斯过程可以逼近最优估计[8]。但由于重要性重采样，产生粒子贫化现象，从而影响滤波精度[9]。

近年来，许多学者对上述 2 种交互式多模型框架下的滤波算法进行了大量研究，用于提高目标跟踪滤波的性能。文献[10]对采用固定模型集的 IMM 算法进行了改进，创新性地提出了一种交互式多模型集自适应协同滤波算法，旨在通过动态调整模型集，提升跟踪精度和鲁棒性。文献[11]针对标准 IMM 算法使用固定转移概率矩阵而导致跟踪精度受限的问题，提出了一种自适应转移概率矩阵更新的高机动目标跟踪算法（ATPM－IMM）。文献[12]为了解决传统 IMM 算法因静态模型集而导致的精度不足问题，提出了一种多模型集自适应协同滤波算法，通过动态选择和管理多个模型集，更好适应目标运动模式的变化。文献[13]针对纯方位区间量测下机动目标跟踪问题，提出了一种交互式多模型箱粒子滤波（IMMBPF）算法。该算法结合了 IMM 算法的多模型切换能力和箱粒子滤波的高效处理能力，能够在纯方位量测条件下实现对机动目标的精确跟踪。

粒子权值退化与多样性匮乏问题是传统粒子滤波中相生相伴但难以完全去除的一对矛盾，前者表现为只有少部分粒子的权重较大，其余粒子的权重可以忽略不计，仍会占用计算资源；后者表现在粒子过于分布在局部状态空间，有可能会丢失正确的粒子[14]。对粒子滤波存在粒子退化问题的改进主要围绕优化重采样策略来展开研究，复制权重大的粒子而淘汰权重较小的粒子[15]。Daum 等学者[16-18]提出粒子流滤波算法（Particle Flow Filter，PFF；Daum-Huang Filter，DHF）以解决粒子贫化问题，通过构造同伦函数，形象地用“流动”来描述将粒子从先验概率分布移动向后验概率分布的过程。粒子流滤波赋予每个粒子一样的权重，无需重采样操作，不依赖建议密度函数。实验证明，在处理非线性问题中，PFF 比 EKF 或 UKF 的精度高出大约 2 个数量级[19]。为了达到与粒子滤波相同的精度，PFF 需要的粒子数量少 3 个数量级，而且在高维状态空间中 PFF 具有更好的性能[20]。粒子流滤波已经应用于纯方位目标跟踪[21]、雷达目标检测[22]、视觉定位[23]。在粒子流滤波方法中，流动策略起着关键作用，不同的流动策略会导致不同形式的粒子流动，通过设计合适的流动策略，充分利用观测的信息，引导粒子流动到合适的后验区域，提高预测的准确性[24]。

本文研究提出了一种交互式多模型粒子流滤波（Interactive Multi－Model Particle Flow Filtering，IMM-PFF）算法，IMM-PFF 算法由 3 个拥有独立机动模型的滤波器组成，分别为 CV、CA 和 CSCT 模型，3 个滤波器独立并行工作，根据当前的模型概率和协方差进行模型集的调整，以获得更加准确的目标跟踪效果。

1　粒子流滤波

1.1　贝叶斯滤波

贝叶斯滤波通过假定系统的状态和观测都遵循特定的概率分布，为非线性动态系统的状态估计问题提供了一种基于概率分布的解决方案。设 $\boldsymbol{x}_k$ 为目标在 k 时刻的状态向量，z_k 为观测值，状态方程和观测方程为：

$$\begin{cases}\boldsymbol{x}_k = f_k(\boldsymbol{x}_{k-1}, \boldsymbol{\mu}_k) \\ z_k = h_k(\boldsymbol{x}_k, \boldsymbol{\nu}_k)\end{cases} \tag{1}$$

其中，f_k 表示状态转移函数；$\boldsymbol{\mu}_k$ 表示过程噪声；h_k 表示测量函数；$\boldsymbol{\nu}_k$ 表示测量噪声。

贝叶斯滤波通常包含 2 个步骤：

（1）预测：根据系统上一时刻的状态估计和动态模型，利用贝叶斯定理推断系统当前时刻的状态。推得的数学公式如下：

$$p(\boldsymbol{x}_k \mid \boldsymbol{z}_{1:k-1}) = \int p(\boldsymbol{x}_{k-1} \mid \boldsymbol{z}_{1:k-1}) p(\boldsymbol{x}_k \mid \boldsymbol{x}_{k-1}) \mathrm{d}\boldsymbol{x}_{k-1} \tag{2}$$

（2）更新：将观测数据与预测估计进行比较，更新对系统状态的估计。更新步骤基于贝叶斯定理，利用观测数据来修正预测估计的不确定性。推得的数学公式如下：

$$p(\boldsymbol{x}_k \mid \boldsymbol{z}_{1:k}) = \frac{p(\boldsymbol{x}_k \mid \boldsymbol{z}_{1:k-1}) p(\boldsymbol{z}_k \mid \boldsymbol{x}_k)}{\int p(\boldsymbol{x}_k \mid \boldsymbol{z}_{1:k-1}) p(\boldsymbol{z}_k \mid \boldsymbol{x}_k) \mathrm{d}\boldsymbol{x}_k} \tag{3}$$

其中，$p(\boldsymbol{x}_k \mid \boldsymbol{x}_{k-1})$ 表示状态转移密度函数，$p(\boldsymbol{z}_k \mid \boldsymbol{x}_k)$ 表示似然函数。通过交替进行预测和更新步骤，可以不断迭代地更新对系统状态的估计，从而实现对系统状态的有效推断和预测。贝叶斯滤波方法在估计系统状态的同时，也提供了对估计不确定性的量化，使得对系统状态的推断更加可靠和准确。

1.2 粒子流滤波

粒子流滤波融合了拓扑学的同伦原理，通过一种创新的粒子迁移机制，将粒子从先验分布移动向后验分布上，以粒子流动的方式代替传统的贝叶斯公式，实现了贝叶斯滤波的功能[25]。为简化表达，设 $h(\boldsymbol{x}_k)$ 和 $g(\boldsymbol{x}_k)$ 分别代表贝叶斯滤波中的似然函数 $p(\boldsymbol{z}_k \mid \boldsymbol{x}_k)$ 和先验概率密度 $p(\boldsymbol{x}_k \mid \boldsymbol{z}_{k-1})$。定义对数同伦函数 $\log p(\boldsymbol{x}_k, \lambda)$ 为：

$$\log p(\boldsymbol{x}_k, \lambda) = \log g(\boldsymbol{x}_k) + \lambda \log h(\boldsymbol{x}_k) - \log K(\lambda) \tag{4}$$

其中，λ 表示伪离散时间，取值范围从 0 到 1，$K(\lambda)$ 表示归一化系数，定义公式为：

$$K(\lambda) = \int g(\boldsymbol{x}_k) h(\boldsymbol{x}_k)^{\lambda} \mathrm{d}\boldsymbol{x} \tag{5}$$

当 $\lambda = 0$ 时，粒子流处于先验分布状态，在 λ 从 0~1 的变化过程中，粒子逐渐移动，直到 $\lambda = 1$ 时，粒子移动到合适的后验分布区域。贝叶斯准则的粒子流动服从 Ito 随机微分方程[26]：

$$\mathrm{d}\boldsymbol{x} = f(\boldsymbol{x}, \lambda)\mathrm{d}\lambda + \sigma(\boldsymbol{x}, \lambda)\mathrm{d}\omega_{\lambda} \tag{6}$$

其中，$f(\boldsymbol{x}, \lambda)$ 可以认为是粒子从先验分布移动到后验分布的"速度场"，$\sigma(\boldsymbol{x}, \lambda)$ 为扩散系数。不考虑过程噪声，假设粒子流是零扩散的，$p(\boldsymbol{x}, \lambda)$ 满足零扩散项的 Fokker-Planck 方程归一化系数，对其 λ 求偏导，可得：

$$\frac{\partial p(\boldsymbol{x}, \lambda)}{\partial \lambda} = -\operatorname{div}(f(\boldsymbol{x}, \lambda) p(\boldsymbol{x}, \lambda)) \tag{7}$$

求解 $f(\boldsymbol{x}, \lambda)$ 是粒子流滤波算法的核心问题。根据不同的流动限制，式(7)有不同的解法[27]，精确粒子流滤波是其中较出名的一种。EDH 假设先验分布和似然分布都服从高斯分布，精确粒子流可以表示为：

$$f(\boldsymbol{x}, \lambda) = \boldsymbol{A}(\lambda)\boldsymbol{x} + \boldsymbol{b} \tag{8}$$

进一步解得：

$$\boldsymbol{A}(\lambda) = -\frac{1}{2}\boldsymbol{P}\boldsymbol{H}^{\mathrm{T}}(\lambda \boldsymbol{H}\boldsymbol{P}\boldsymbol{H}^{\mathrm{T}} + \boldsymbol{R})^{-1}\boldsymbol{H} \tag{9}$$

$$\boldsymbol{b} = (\boldsymbol{I} + 2\lambda \boldsymbol{A}(\lambda))[(\boldsymbol{I} + \lambda \boldsymbol{A}(\lambda))\boldsymbol{P}\boldsymbol{H}^{\mathrm{T}}\boldsymbol{R}^{-1}\boldsymbol{z} + \boldsymbol{A}(\lambda)\bar{\boldsymbol{x}}] \tag{10}$$

其中，$\boldsymbol{R}$ 表示测量噪声方差矩阵；$\boldsymbol{P}$ 表示为先验协方差矩阵；$\boldsymbol{H}$ 可以通过测量函数 h_k 线性化得到，求解粒子流的系数可以通过 EKF/UKF 计算得到。具体公式如下：

$$\boldsymbol{H} = \left.\frac{\partial h(\boldsymbol{x}, 0)}{\partial \boldsymbol{x}}\right|_{x = x_{\lambda}} \tag{11}$$

2 交互式多模型粒子流滤波算法

一般而言，单一运动模型难以精确地刻画目标的多种运动状态。交互式多模型算法采用一个包含多模型的集合来描述和预测目标的动态变化，每个模型都配备了相应的滤波器，并且算法能够自适应地调整各模型间的转移概率，通过概率加权的交互方式，IMM 算法综合了多个并行滤波器的预测结果[28]。

IMM-PFF 算法融合了 IMM 和粒子流滤波算法。首先计算混合概率和各模型的交互输入，然后根据先验概率密度函数初始化粒子，经过粒子流滤波后，再输出交互预测结果以及更新模型概率。IMM-PFF 算法原理如图 1 所示。

对于并行滤波器，考虑以下包含 M 个模型的线性离散时间马尔可夫系统[29]：

$$\boldsymbol{x}(k) = \boldsymbol{F}(k-1)\boldsymbol{x}(k-1) + \boldsymbol{G}(k-1) \cdot \boldsymbol{\omega}(k-1) \tag{12}$$

$$\boldsymbol{z}(k \mid k-1) = \boldsymbol{H}(k) \cdot \boldsymbol{x}(k \mid k-1) + \boldsymbol{v}(k) \tag{13}$$

其中，$\boldsymbol{F}(k-1)$ 和 $\boldsymbol{H}(k)$ 分别表示状态矩阵和观测矩阵；$\boldsymbol{\omega}(k-1) \sim \mathrm{N}(0, \boldsymbol{Q}(k-1))$ 和 $\boldsymbol{v}(k) \sim \mathrm{N}(0, \boldsymbol{R}(k))$ 分别表示状态噪声和观测噪声，两者是零均值标准的高斯白噪声序列；$\boldsymbol{x}(k)$ 和 $\boldsymbol{z}(k \mid k-1)$ 分别是 k 时刻的状态矢量和观测矢量。式(12)和(13)分别表示系统的状态方程和观测方程。模型 j 的先验概率和模型 i 到模型 j 的马尔可夫转移概率分别为：

$$\mu_j(0) = p(M_j(0)) \tag{14}$$

$$\pi_{ij} = p(M_j(k) \mid M_i(k-1)) \tag{15}$$

本文提出的 IMMPFF 算法的完整流程如下[30-31]：

(1)计算各模型的初始混合概率。定义公式为：

$$\mu_{i|j}(k-1 \mid k-1) = \frac{1}{\sum_{i=1}^{M} \pi_{ij} u_j^n(k-1)} \pi_{ij} u_i(k-1), \quad i, j \in M \tag{16}$$

其中，$\mu_{i|j}(k-1 \mid k-1)$ 表示 $k-1$ 时刻第 i 个模型的混合概率，M 表示模型个数。

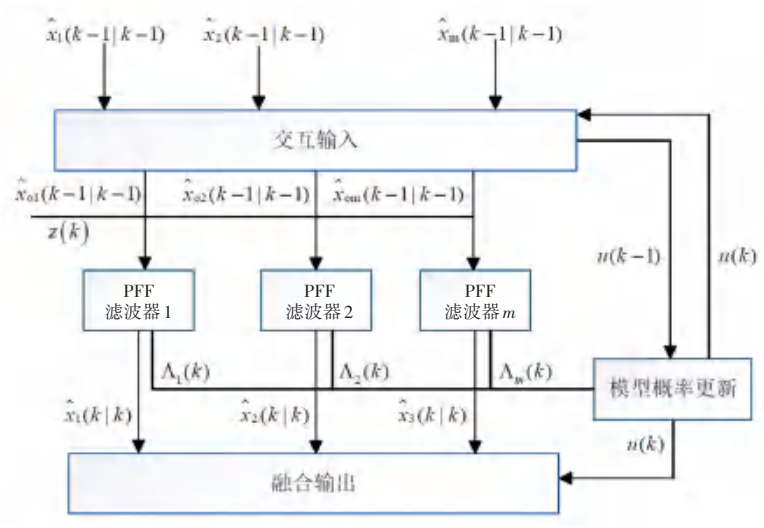


图 1　IMM-PFF 算法流程图

Fig. 1　Flowchart of IMM-PFF algorithm

(2) 估计混合状态输入。可由下式计算求得：

$$\bar{\boldsymbol{x}}_j(k-1\mid k-1)=\sum_{i=1}^{M}\hat{\boldsymbol{x}}_i(k-1\mid k-1)\mu_{i\mid j}(k-1\mid k-1) \tag{17}$$

$$\begin{aligned}\bar{\boldsymbol{p}}_j(k-1\mid k-1)=&\sum_{i=1}^{m}\mu_{i\mid j}(k-1\mid k-1)\cdot\\&\{\hat{\boldsymbol{p}}_i(k-1\mid k-1)+[\hat{\boldsymbol{x}}_i(k-1\mid k-1)-\\&\hat{\boldsymbol{x}}_j(k-1\mid k-1)]\cdot[\hat{\boldsymbol{x}}_i(k-1\mid k-1)-\\&\hat{\boldsymbol{x}}_j(k-1\mid k-1)]^{\mathrm{T}}\}\end{aligned} \tag{18}$$

其中，$\hat{\boldsymbol{x}}_i(k-1\mid k-1)$ 和 $\hat{\boldsymbol{p}}_i(k-1\mid k-1)$ 表示 $k-1$ 时刻第 i 个模型滤波器的状态估计值和状态协方差矩阵，$\bar{\boldsymbol{x}}_j(k-1\mid k-1)$ 和 $\bar{\boldsymbol{p}}_j(k-1\mid k-1)$ 是交互结果。

(3) 粒子流滤波。M 个独立的相同滤波器对各自输入数据同时进行滤波处理，得到 k 时刻的状态估计向量 $\hat{\boldsymbol{x}}_i(k\mid k)$ 及其对应的估计误差协方差矩阵 $\boldsymbol{p}_i(k\mid k)$。基于粒子流滤波方法的过程如下。

① 状态预测。公式如下：

$$\hat{\boldsymbol{x}}_j^n(k\mid k-1)\sim \mathrm{N}(\boldsymbol{F}_j(k-1)\bar{\boldsymbol{x}}_j(k-1\mid k-1),\boldsymbol{G}(k-1)\boldsymbol{Q}(k-1)\boldsymbol{G}(k-1)^{\mathrm{T}}) \tag{19}$$

$$\boldsymbol{p}_j(k\mid k-1)=\boldsymbol{F}_j(k-1)\bar{\boldsymbol{p}}_j(k\mid k-1)\boldsymbol{F}_j(k-1)^{\mathrm{T}}+\boldsymbol{G}(k-1)\boldsymbol{Q}(k-1)\boldsymbol{G}(k-1)^{\mathrm{T}} \tag{20}$$

$$\bar{\boldsymbol{x}}_j(k\mid k-1)=\sum_{i=1}^{N}\hat{\boldsymbol{x}}_i^n(k\mid k-1) \tag{21}$$

其中，$\hat{\boldsymbol{x}}_j^n(k\mid k-1)$ 表示第 j 个粒子流滤波在第 k 时刻第 n 个粒子的状态估计；$\boldsymbol{p}_j(k\mid k-1)$ 表示预测方差矩阵；$\boldsymbol{Q}(k-1)$ 表示 $k-1$ 时刻的过程噪声协方差矩阵；$\bar{\boldsymbol{x}}_j(k\mid k-1)$ 表示粒子集合的状态估计均值。

② 粒子流滤波更新。利用计算得出的 $\bar{\boldsymbol{x}}_j(k\mid k-1)$ 和 $\boldsymbol{p}_j(k\mid k-1)$ 根据式(9) 和式(10) 计算 $\boldsymbol{A}$ 和 $\boldsymbol{b}$。过程如下：

FOR $t=0:\lambda:1$

$$\hat{\boldsymbol{x}}_j^n(k\mid k-1)=\hat{\boldsymbol{x}}_j^n(k\mid k-1)+\lambda(\boldsymbol{A}\hat{\boldsymbol{x}}_j^n(k\mid k-1)+\boldsymbol{b}) \tag{22}$$

END

根据式(21)重新计算 $\bar{\boldsymbol{x}}_j(k\mid k-1)$。

③ 状态估计及其协方差矩阵。公式如下：

$$\boldsymbol{S}_j(k)=\boldsymbol{H}(k)\boldsymbol{p}_j(k\mid k-1)\boldsymbol{H}(k)^{\mathrm{T}}+\boldsymbol{R}(k) \tag{23}$$

$$\boldsymbol{K}_j(k)=\boldsymbol{p}_j(k\mid k-1)\boldsymbol{H}(k)^{\mathrm{T}}\boldsymbol{S}_j(k)^{-1} \tag{24}$$

$$\hat{\boldsymbol{x}}_j(k\mid k)=\bar{\boldsymbol{x}}_j(k\mid k-1)+\boldsymbol{K}_j(k)[\boldsymbol{z}(k)-\boldsymbol{H}(k)\bar{\boldsymbol{x}}_j(k\mid k-1)] \tag{25}$$

$$\boldsymbol{p}_j(k\mid k)=[\boldsymbol{I}-\boldsymbol{K}_j(k)\boldsymbol{H}(k)]\boldsymbol{p}_j(k\mid k-1) \tag{26}$$

(4) 更新模型后验概率。公式如下：

$$u_j(k)=\frac{1}{\sum_{j=1}^{M}\Lambda_j(k)\bar{C}_j}\Lambda_j(k)\bar{C}_j \tag{27}$$

其中，$\Lambda_j(k)$ 表示第 j 个模型的似然函数。

(5) 交互输出。计算公式如下：

$$\hat{\boldsymbol{x}}(k\mid k)=\sum_{j=1}^{M}\hat{\boldsymbol{x}}_j(k\mid k)u_j(k) \tag{28}$$

$$\boldsymbol{p}(k\mid k)=\sum_{j=1}^{M}\mu_j(k\mid k)\{\boldsymbol{p}(k\mid k)+$$

$$[\hat{\boldsymbol{x}}_j(k \mid k) - \hat{\boldsymbol{x}}(k \mid k)][\cdot]^{\mathrm{T}}\} \tag{29}$$

重复上述过程可以获得当前时刻输出的状态估计 $\hat{\boldsymbol{x}}(k \mid k)$ 和状态协方差 $\boldsymbol{p}(k \mid k)$。

3 仿真实验

为了验证 IMM-PFF 算法在目标跟踪任务中的性能表现,本文设计并实施了一系列三维空间中的机动目标跟踪实验,并将本文算法的跟踪性能与经典的交互式多模型卡尔曼滤波(IMM-KF)算法以及交互式粒子滤波(IMM-PF)算法进行比较分析。所有算法均在配备 12th Gen Intel(R) Core(TM) i7-12700H 处理器(主频为 2.3 GHz)的计算机上,通过 Matlab R2021b 进行运行和测试。

本文采用位置滤波值的均方根误差(Root mean square error, RMSE)来评价算法性能,即:

$$\mathrm{RMSE}_x(k) = \sqrt{\frac{1}{N}\sum_{k=1}^{N}[\hat{\boldsymbol{x}}_m(k) - \boldsymbol{x}(k)]^2} \tag{30}$$

$$\mathrm{RMSE}(k) = \sqrt{\mathrm{RMSE}_x^2(k) + \mathrm{RMSE}_y^2(k) + \mathrm{RMSE}_z^2(k)} \tag{31}$$

$$\mathrm{ARMSE}_x = \frac{1}{M}\sum_{m=1}^{M}\mathrm{RMSE}_x(m) \tag{32}$$

$$\mathrm{ARMSE} = \frac{1}{M}\sum_{m=1}^{M}\mathrm{RMSE}(m) \tag{33}$$

其中,M 表示蒙特卡洛实验次数;N 表示跟踪次数;$\boldsymbol{x}(k)$ 和 $\hat{\boldsymbol{x}}_m(k)$ 分别表示第 m 次蒙特卡洛实验在 k 时刻目标在 x 方向上的真实位置和估计位置;$\mathrm{RMSE}_x(k)$ 表示 k 时刻在 x 方向上的估计值的均方根误差;$\mathrm{RMSE}(k)$ 表示 k 时刻在三维空间中的估计值的均方根误差;ARMSE_x 表示在 x 方向上估计值的均方根误差均值;ARMSE 表示在三维空间中估计值的均方根误差均值。

3.1 实验设置

状态向量 $\boldsymbol{x} = [x,\dot{x},\ddot{x},y,\dot{y},\ddot{y},z,\dot{z},\ddot{z}]^{\mathrm{T}}$ 包含目标的位置、速度和加速度,x、$\dot{x}$、$\ddot{x}$ 分别表示目标在 x 方向上的位置、速度、加速度,y、z 同理。初始值为 $\boldsymbol{x}(0) = [5,5,0,5,5,0,5,5,0]^{\mathrm{T}}$。设置采样周期 $T = 0.05$ s,跟踪次数为 $N = 100$ 次,蒙特卡洛仿真次数 $M = 50$。目标运动轨迹(见图 2)如下:

(1)目标在 0~20 s、40~60 s、80~100 s 内做匀速运动 $v = 5$ m/s;

(2)目标在 20~40 s 内做匀加速运动,$\alpha = 5$ m/s^2;

(3)目标在 60~80 s 内做常速率协同转弯运动,转弯角速度 $\omega_x = -\frac{4}{3}$rad/s, $\omega_y = \frac{5}{6}$rad/s,$\omega_z = 1$ rad/s。

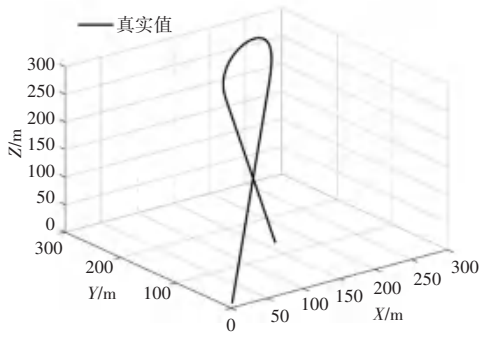


图 2 目标运动轨迹

Fig. 2 Target motion trajectory

模型集包含 3 个模型:

(1)匀速直线运动模型(Constant Velocity, CV)。具体公式如下:

$$\boldsymbol{F}1 = \begin{bmatrix} 1 & T & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 0 \end{bmatrix},\ \mathbf{F_CV} = \begin{bmatrix} \boldsymbol{F}1 & 0 & 0 \\ 0 & \boldsymbol{F}1 & 0 \\ 0 & 0 & \boldsymbol{F}1 \end{bmatrix} \tag{34}$$

(2)匀加速直线运动模型(Constant Acceleration, CA)。具体公式如下:

$$\boldsymbol{F}2 = \begin{bmatrix} 1 & T & T^2/2 \\ 0 & 1 & T \\ 0 & 1 & 1 \end{bmatrix},\ \mathbf{F_CA} = \begin{bmatrix} \boldsymbol{F}2 & 0 & 0 \\ 0 & \boldsymbol{F}2 & 0 \\ 0 & 0 & \boldsymbol{F}2 \end{bmatrix} \tag{35}$$

(3)常速率协同转弯运动模型(Constant Speed Coordination Turn, CSCT)。具体公式如下:

$$\boldsymbol{F}3_1 = \begin{bmatrix} 1 & \sin(\omega_1 T)/\omega_1 & (1-\cos(\omega_1 T))/\omega_1^2 \\ 0 & \cos(\omega_1 T) & \sin(\omega_1 T)/\omega_1 \\ 0 & -\omega_1\sin(\omega_1 T) & \cos(\omega_1 T) \end{bmatrix},$$

$$\boldsymbol{F}3_2 = \begin{bmatrix} 1 & \sin(\omega_2 T)/\omega_2 & (1-\cos(\omega_2 T))/\omega_2^2 \\ 0 & \cos(\omega_2 T) & \sin(\omega_2 T)/\omega_2 \\ 0 & -\omega_2\sin(\omega_2 T) & \cos(\omega_2 T) \end{bmatrix},$$

$$\boldsymbol{F}3_3 = \begin{bmatrix} 1 & \sin(\omega_3 T)/\omega_3 & (1-\cos(\omega_3 T))/\omega_3^2 \\ 0 & \cos(\omega_3 T) & \sin(\omega_3 T)/\omega_3 \\ 0 & -\omega_3\sin(\omega_3 T) & \cos(\omega_3 T) \end{bmatrix},$$

$$\mathbf{F_CSCT} = \begin{bmatrix} \boldsymbol{F}3_1 & 0 & 0 \\ 0 & \boldsymbol{F}3_2 & 0 \\ 0 & 0 & \boldsymbol{F}3_3 \end{bmatrix} \tag{36}$$

3 个模型的过程噪声协方差矩阵 $\boldsymbol{Q} = q \times \boldsymbol{I}_{9\times 9}$,$q = 5$;观测噪声协方差矩阵 $\boldsymbol{R} = r \times \boldsymbol{I}_{3\times 3}$,$r = 20$。观

测矩阵为：

$$\boldsymbol{H}=\begin{bmatrix}1&0&0&0&0&0&0&0&0\\0&0&0&1&0&0&0&0&0\\0&0&0&0&0&0&1&0&0\end{bmatrix} \tag{37}$$

各模型的概率初始值为[0.3,0.3,0.4]，模型的初始概率状态转移矩阵为：

$$\boldsymbol{\pi}=\begin{bmatrix}0.8&0.1&0.1\\0.1&0.8&0.1\\0.1&0.1&0.8\end{bmatrix} \tag{38}$$

3.2　算法跟踪性能对比

根据 3.1 节的实验参数设置，图 3 展示了 3 种算法(IMM-PFF、IMM-PF 和 IMM-KF)在单次实验的目标跟踪情况。3 种滤波算法通过交互式多模型方法来融合多种模型的信息，以提高目标状态的估计精度和跟踪能力。为了评估这些算法的性能，研究中进行了 50 次蒙特卡洛仿真实验，并通过式(30)~式(33)计算得到了目标空间位置估计误差，结果如图 4 所示。从图 4 中可以看出，IMM-PFF 在位置估计均方根误差方面表现出明显的优势，其误差值低于 IMM-KF，表明 IMM-PFF 在处理复杂环境和非线性动态系统时具有更高的精度。然而，IMM-PF 的 RMSE 比 IMM-PFF 更低，显示出在某些特定条件下，传统粒子滤波方法可能更具优势。3 种算法的执行时间如图 5 所示，实验结果表明，IMM-PFF 算法的运行时间介于 IMM-KF 和 IMM-PF 之间，这可能是由于 IMM-PFF 在每个时刻需要进行粒子流的计算，增加了计算复杂度，当所需时间比粒子滤波的重采样步骤更少。在第 1 次蒙特卡洛实验中，所需时间是第 2 次运行的 2 倍，随着实验次数的增加，3 种算法的运行时间逐渐趋于稳定。

进一步分析图 4 和图 5 中的数据，通过空间位置估计均方根误差(ARMSE)均值来衡量 3 种算法的状态估计性能，并通过算法的平均运行时间来验证其计算复杂度。表 1 列出了 3 种算法在各方向上位置估计的 RMSE 均值、空间位置估计的 RMSE 均值以及算法的平均运行时间。从表 1 可以看出，3 种算法中 IMM-PF 估计结果的均方根误差均值最小，表明其在状态估计精度上表现最佳。相比于 IMM-PF 算法，IMM-PFF 在 3 个方向的滤波精度分别降低了 34.7%、36.5%、22.9%，空间位置滤波精度降低了 31.4%。然而，与 IMM-KF 相比，IMM-PFF 在 x 和 z 方向上的滤波精度分别提升了 17.5% 和 13.5%，尽管在 y 方向上的滤波精度降低了 11.6%，但整体空间位置滤波精度提升了 7.0%。在计算复杂度方面，IMM-KF 算法的平均运行时间最少，IMM-PFF 次之，IMM-PF 算法的平均运行时间最长。PF 和 PFF 算法都属于粒子滤波框架，PF 通过大量粒子的采样和重采样过程，能够较好地逼近后验分布，从而提供高精度的状态估计。然而，这一过程也带来了较大的计算负担。PFF 算法在每个伪时间步长内动态调整粒子的位置，以更高效地逼近后验分布，减少了重采样步骤的计算开销。因此，IMM-PFF 在保持较高估计精度的同时，显著降低了计算复杂度。

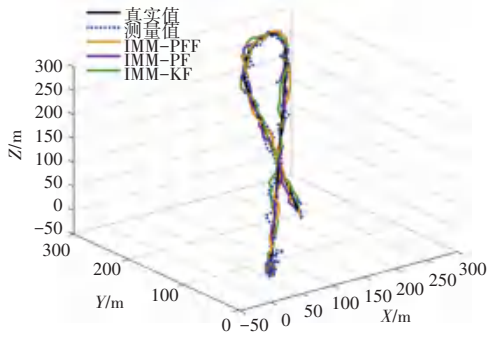


图 3　单次仿真的目标跟踪情况

Fig. 3　Target tracking situation in a single simulation

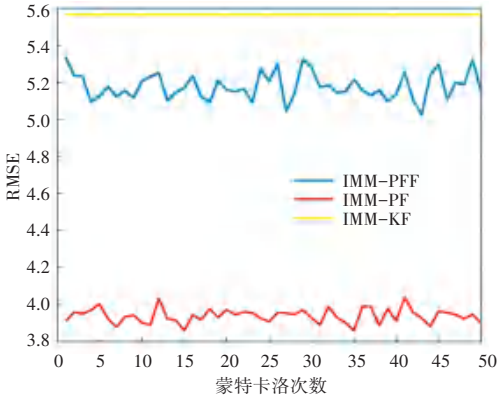


图 4　3 种算法的空间位置估计误差

Fig. 4　Spatial position estimation error of three algorithms

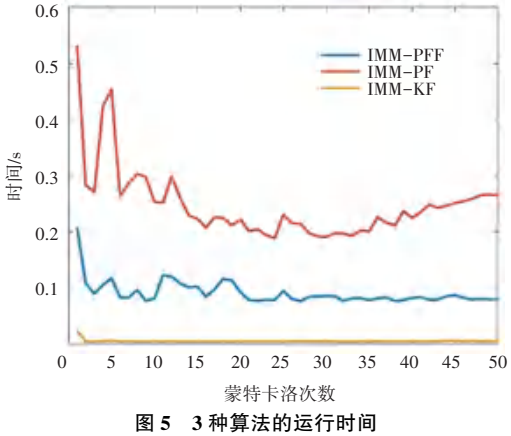


图 5　3 种算法的运行时间

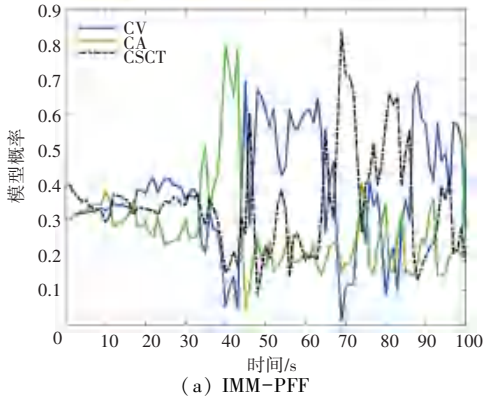
Fig. 5　Running time of three algorithms

表 1 3 种算法的目标位置估计误差和平均运行时间

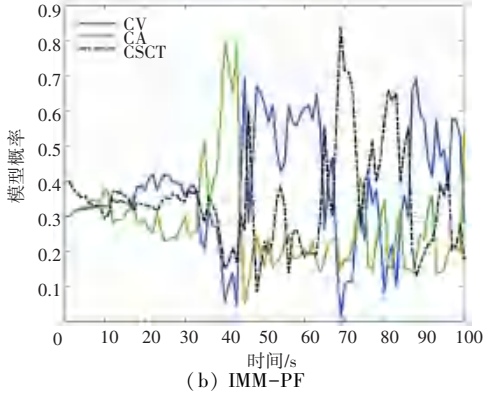
Table 1 Target position estimation error and average running time of three algorithms

模型	$ARMSE_x$	$ARMSE_y$	$ARMSE_z$	ARMSE	Time/s
IMM-PFF	2.777 31	3.324 0	2.837 3	5.176 9	0.090 5
IMM-PF	2.062 60	2.434 6	2.308 4	3.938 7	0.247 5
IMM-KF	3.366 10	2.978 4	3.280 0	5.564 2	0.004 6

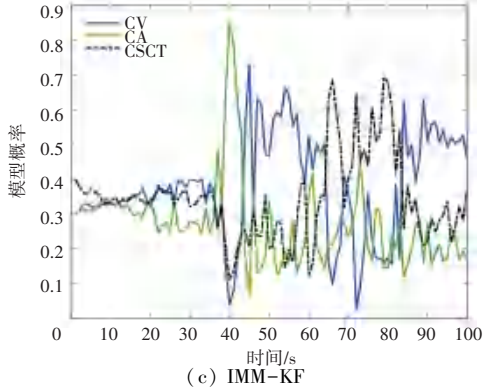
图 6 展示了 IMM 模型集检测到的各模型的概率,从图 6 中可以看出在[0,20)s、(40,60)s 和(80,100]s 内蓝色实线处于较高位置,表明目标匀速直线运动的概率更大,[20,40]s 绿色直线呈上升趋势,表明目标做匀加速运动的概率较大,[60,80]s 黑色曲线处于主要位置,表明目标做常速率协同转弯运动的概率更大,与目标的实际运动情况基本相符。



(a) IMM-PFF



(b) IMM-PF



(c) IMM-KF

图 6 单次仿真的模型概率

Fig. 6 Model probability of single simulation

3.3 观测噪声对滤波性能的影响

假设系统观测噪声的协方差矩阵 $\boldsymbol{R} = r \times \boldsymbol{I}_{3\times 3}$, $r = 20$,参数 r 在 20 至 200 的范围内均匀变化,步长为 10,同时保持其他所有参数的设置与第 3.1 节中的描述一致。为评估 3 种不同算法在目标空间位置估计上的性能,计算 3 种算法在不同观测噪声水平下的均方根误差。实验结果如图 7 所示。从图 7 中可以看到随着观测噪声的增大,RMSE 逐渐增大,但是 IMM-PFF 的峰值比其他算法更低,体现了其在噪声干扰下的优越性和稳定性,能够更有效地处理高噪声和非线性系统,从而提供更精确的状态估计。

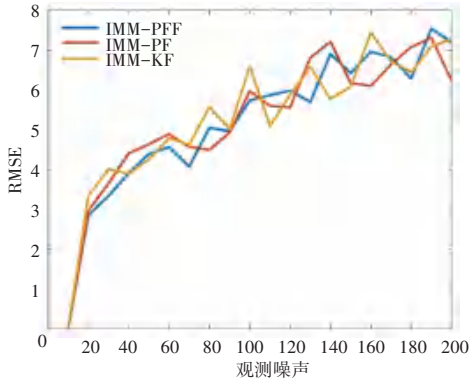


图 7 不同观测噪声对滤波性能的影响

Fig. 7 Influence of different observation noises on filtering performance

4 结束语

本文提出了一种交互式多模型粒子流滤波算法,针对包含匀速运动(CV)、匀加速运动(CA)和常速率协同转弯(CSCT)三种运动状态的目标位置估计进行了仿真和对比实验分析。实验结果表明,相比于传统的 IMM-PF 和 IMM-KF 滤波算法,IMM-PFF 滤波算法在估计精度和计算复杂度之间提供了一个较好的平衡,而且能根据目标运动状态的变化,及时匹配相应的运动模型,有效保证了估计的准确性和稳定性,从而给出了一个更合适的普适算法。

基于上述优点,IMM-PFF 滤波算法在实际应用中具有更重要的潜力和价值。为进一步提升算法性能,下一步的研究工作将包括 2 个方面:一是研究不同时间步长对 IMM-PFF 算法滤波性能的影响,以优化算法效率并提高实时处理能力;二是考虑引入其他粒子流动策略,探索并验证不同类型粒子流滤波算法在目标估计性能上的改进效果。

参考文献

[1] 曾浩, 母王强, 蒋阳, 等. 模型参数自适应的低复杂度 ATPM-VSIMM 算法[J]. 通信学报, 2023, 44(9): 25-35.

[2] HU K, WANG J, CHENG Y, et al. Adaptive filtering and smoothing algorithm based on variable structure interactive multiple model[J]. Scientific Reports, 2023, 13(1): 12993.
[3] BLOM H, BAR-SHALOM Y. The interacting multiple model algorithm for systems with Markovian switching coefficients[J]. IEEE Transactions on Automatic Control, 1988, 33(8): 780-783.
[4] 叶瑾, 许枫, 杨娟, 等. 一种基于多传感器的复合量测 IMM-EKF 数据融合算法[J]. 电子学报, 2020, 48(12): 2326-2330.
[5]王平波, 陈强, 卫红凯, 等. 一种基于模型概率单调性变化的自适应 IMM-UKF 改进算法[J]. 电子与信息学报, 2024, 46(1): 41-48.
[6] 刘望生, 潘海鹏, 王明环. 混响噪声下声源定位与跟踪的多特征自适应 IMM 粒子滤波算法[J]. 控制理论与应用, 2023, 40(3): 477-484.
[7] 马艳, 刘小东. 状态自适应无迹卡尔曼滤波算法及其在水下机动目标跟踪中的应用[J]. 兵工学报, 2019, 40(2): 361-368.
[8] 谢旭, 邱晓刚, 包亦正, 等. 动态数据驱动仿真综述[J]. 系统仿真学报, 2024, 36(8):1832-1842.
[9]XUE Li, HAN Yulan, NA Chunning. Robust interacting multiple model unscented particle filter for navigation[J]. Mathematical Problems in Engineering, 2020, 2020: 8871358.
[10]许红, 谢文冲, 袁华东, 等. 基于自适应的增广状态-交互式多模型的机动目标跟踪算法[J]. 电子与信息学报, 2020, 42(11): 2749-2755.
[11]曾浩, 母王强, 杨顺平. 高机动目标跟踪 ATPM-IMM 算法[J]. 通信学报, 2022, 43(7): 93-101.
[12]孔德明, 杨丹, 王书涛. 交互式多模型集自适应协同滤波目标跟踪算法研究[J]. 计量学报, 2021, 42(5): 638-644.
[13]张俊根. 基于 IMMBPF 的纯方位区间量测机动目标跟踪[J]. 控制工程, 2022, 29(12): 2277-2283.
[14]孙伟, 刘经洲. 基于 Huber 鲁棒容积裂变粒子滤波的协同导航方法[J]. 仪器仪表学报, 2022, 43(2): 166-175.
[15]孙玲莉, 杨超, 郭辉, 等. 基于粒子滤波的波叠加算法[J]. 智能计算机与应用, 2021, 11(3): 80-84.
[16] DAUM F, HUANG J. Nonlinear filters with log-homotopy[C]//Proceedings of Signal and Data Processing of Small Targets 2007. Bellingham, WA: SPIE, 2007, 6699: 423-437.
[17]DAUM F, HUANG J. Particle flow for nonlinear filters with log-homotopy[C]//Proceedings of Signal and Data Processing of Small Targets 2008. Bellingham, WA: SPIE, 2008, 6969: 414-425.
[18] CROUSE D. Particle Flowfilters: Biases and bias avoidance[C]//Proceedings of 2019 22th International Conference on Information Fusion (FUSION). Piscataway, NJ: IEEE, 2020: 1-8.
[19]DAUM F, HUANG J. Particle degeneracy: Root cause and solution[C]//Proceedings of Signal Processing, Sensor Fusion, and Target Recognition XX. Bellingham, WA: SPIE, 2011: 80500W.
[20]WANG Junjie, ZHAO Lingling, SU Xiaohong. Marginalized particle flow filter[J]. Circuits, Systems, and Signal Processing, 2019, 38(7): 3152-3169.
[21]COSTA R, WETTERGREN T. Computationally efficient angles-only tracking with particle flow filters[C]// Proceedings of Signal Processing, Sensor/Information Fusion, and Target Recognition XXIV. Bellingham, WA: SPIE, 2015: 947404.
[22]柳超, 王子微, 孙进平. 粒子流粒子滤波检测前跟踪方法[J]. 信号处理, 2019, 35(3): 342-350.
[23]杨琼楠, 吴力涛, 仇晨光. 基于粒子流滤波的视觉定位算法[J]. 信息技术, 2024 (8): 134-138.
[24]DAUM F, HUANG J, KRICHMAN M, et al. Seventeen dubious methods to approximate the gradient for nonlinear filters with particle flow[C]//Proceedings of Signal and Data Processing of Small Targets 2009. Bellingham, WA: SPIE, 2009: 7445.
[25]周德运. 基于新息误差的粒子滤波算法研究及其应用[D]. 武汉: 武汉科技大学, 2021.
[26]LI Chen, ZHEN Wu. A type of general forward-backward stochastic differential equations and applications[J]. Chinese Annals of Mathematics, Series B, 2011, 32(2): 279-292.
[27]DAUM F, HUANG J. Particle flow with non-zero diffusion for nonlinear filters, Bayesian decisions and transport[C]// Proceedings of Signal and Data Processing of Small Targets. Bellingham, WA: SPIE, 2013: 8857.
[28]陈世杰, 刘锡祥, 黄永江, 等. 基于 IMM 算法的多 AUV 协同定位水声传播延迟处理方法[J]. 水下无人系统学报, 2023, 31(2): 259-268.
[29]沈忱, MIHAYLOVA L. 跳变马尔可夫系统的最大混合相关熵状态估计[J]. 电子学报, 2021, 49(11): 2225-2233.
[30]徐博, 陈崇, 王连钊. 基于交互式多模型的车载 INS/OD/GPS 的定位方法[J]. 中国惯性技术学报, 2022, 30(1): 58-64.
[31]GUO Xie, SUN Lanlan, WEN Tao, et al. Adaptive transition probability matrix-based parallel IMM algorithm[J]. IEEE Transactions on Systems, Man, and Cybernetics: Systems, 2021, 51(5): 2980-2989.