

闪烁噪声条件下基于交互多模框架的雷达目标跟踪算法*

占荣辉¹, 李祖检¹, 滕书华²

(1. 国防科技大学 电子科学学院, 湖南 长沙 410073; 2. 湖南第一师范学院 电子信息学院, 湖南 长沙 410205)

摘要:针对传统雷达目标跟踪算法在处理闪烁噪声时面临的性能下降问题,提出一种将容积卡尔曼估计器与交互多模框架相结合的高性能滤波算法。该算法将目标状态建模为高斯分布,将闪烁噪声建模为混合高斯分布,同时将其发生概率建模为一阶马尔可夫过程;在此基础上,利用交互多模框架实现对不同高斯噪声分量的匹配滤波处理。为了减轻非线性观测条件对目标跟踪精度的影响,进一步采用容积卡尔曼估计器作为高斯近似滤波器,对目标状态进行递推预测和更新。仿真结果表明:所提算法较传统高斯混合滤波器和粒子滤波器具有更高的跟踪精度和更好的实时性能,同时还能对闪烁噪声出现时刻进行有效的估计。

关键词:闪烁噪声;交互多模;雷达目标跟踪;容积卡尔曼滤波器

中图分类号:TN95 文献标志码:A 文章编号:1001-2486(2023)05-140-10

Radar target tracking algorithm in the framework of interacting multiple model with glint noise

ZHAN Ronghui¹, LI Zujian¹, TENG Shuhua²

(1. College of Electronic Science and Technology, National University of Defense Technology, Changsha 410073, China;

2. Electronic Information School, Hunan First Normal University, Changsha 410205, China)

Abstract: Aiming at the problem of the performance degradation of the traditional tracking algorithm when confronted with glint noise, a high performance filtering method named as IMM-CKF was proposed by integrating the CKF (cubature Kalman filter) into the framework of IMM (interacting multiple model). In the proposed algorithm, the target state was modeled as Gaussian distribution, the glint noise was modeled as Gaussian mixture distribution, and the occurrence probability of the glint noise was modeled as the first-order Markov process. An IMM framework was then used to implement model-matched filtering for each Gaussian component. To further mitigate the impact of nonlinear observation condition on tracking accuracy, the CKF was utilized as Gaussian approximation filter to realize recursive prediction and update of the target state. Simulation results show that the proposed method not only has higher tracking accuracy than traditional algorithms such as Gaussian sum filter and particle filter, but also has better real time ability. Additionally, the IMM-CKF can effectively estimate the existence of glint noise.

Keywords: glint noise; interacting multiple model; radar target tracking; cubature Kalman filter

通常情况下,雷达传感器的测量误差主要受系统噪声和处理方法的影响,相应的噪声可称为“常规噪声”。但当雷达对复杂运动目标进行探测时,由于目标与传感器之间的几何关系将动态发生变化,目标上不同散射中心的相对相位也会随机发生变化,从而造成回波相位波前的畸变。波前在观测方向上的倾斜和随机摆动现象被称为闪烁效应,由此导致的测量噪声常被称为“闪烁噪声”。理论上,凡尺度能与雷达波长相比拟,具有两个或两个以上散射中心的任何体目标都可能产生闪烁效应,且目标与传感器越近,因闪烁效应产生的量测噪声也就越大。这对诸如导弹制导等

应用场合极为不利,将严重影响制导控制精度,甚至会引起导弹脱靶。

与常规噪声的高斯分布特性不同,闪烁噪声具有明显的拖尾特性,是典型的非高斯噪声。目前,关于闪烁噪声的分布主要有高斯-拉普拉斯混合模型^[1]、混合高斯模型^[2-4]、t分布模型^[5-7]、高斯-t混合分布模型^[8-9]等。文献[10]采用QQ-plot方法对闪烁噪声进行拟合,得到了模型参数的估计结果。值得一提的是,雷达的测量结果是在极坐标(或球坐标)下得到的,而观测方程具有非线性特点,因此闪烁噪声条件下的目标跟踪问题是典型的非线性、非高斯问题。

* 收稿日期:2022-07-12

基金项目:国家自然科学基金资助项目(62271491,61471370)

作者简介:占荣辉(1978—),男,江西玉山人,副研究员,博士,硕士生导师,E-mail:zhanrh@nudt.edu.cn;

滕书华(通信作者),男,河北唐山人,正高级工程师,博士,硕士生导师,E-mail:dxy-tengshuhua@hnfnu.edu.cn

众所周知,卡尔曼滤波器(Kalman filter, KF)是线性、高斯假设条件下的最优状态估计器,在目标跟踪领域得到广泛的应用。在非线性、非高斯条件下,最优滤波假设不再成立,国内外学者对此展开了广泛的研究。在现有针对闪烁噪声的跟踪算法中,根据原理框架的不同,可粗略分为以下三类:

第一类是以非线性评分函数(score function)为基础的跟踪算法。在文献[11]中,作者最早通过引入评分函数对KF进行修正。作为一种早期提出的方法,该算法涉及烦琐的混合分布函数卷积操作,在系统状态维数较高时异常复杂,因此在实际中应用较少。为此,文献[1]在评分函数的基础上,通过将量测噪声分布进行正交展开来避免卷积操作,然而该算法需要对量测向量进行解耦合,同时也存在复杂的矩生成函数(moment generating function, MGF)计算问题,因此实用性也较差。

第二类是以高斯混合(Gaussian mixture, GM)或高斯和(Gaussian sum, GS)滤波为基础的跟踪算法^[12-13]。这类算法将目标状态、过程噪声、测量噪声分别用多个高斯分量来近似,并采用多个并行的高斯近似滤波器完成状态估计。高斯近似滤波器的具体实现形式有扩展卡尔曼滤波器(extended KF, EKF)、不敏卡尔曼滤波器(unscented KF, UKF)和容积卡尔曼滤波器(cubature KF, CKF)等^[14-15]。该类算法的优点是实现结构比较简单,不足之处是:高斯分量数会随递推步数(时间周期数)呈指数增长,因此在每步递推完成后,需要进行高斯分量剪枝和合并处理;对高斯分量数的确定缺乏严格的理论指导,只能通过人工经验进行选择优化。

第三类是以粒子滤波(particle filter, PF)为基础的跟踪算法^[16-18]。这类算法主要利用了粒子滤波器对非线性、非高斯系统具有强大的近似能力这一特点,采用概率分布近似(而不是函数近似)手段进行目标状态估计。粒子的采样通常有两种方式:一是直接通过先验分布(状态转移函数)进行采样,这种采样方式比较简单,但粒子质量不高;二是从融合了量测信息的重要采样函数或建议分布^[19-21]中进行采样,该采样方式虽能在一定程度上改善粒子质量,但每一个粒子都需要一个类卡尔曼滤波器(卡尔曼滤波器及各种衍生形式)来实现完整的状态递推,导致算法复杂度急剧上升。除此之外,对于高维目标状态(相对一维标量状态),若要达到理想的密度近似效果和良好的跟踪性能,粒子规模必须足够大,相应

地,算法的复杂度也会非常高。

事实上,虽然闪烁噪声被建模为各种混合分布的形式,但闪烁噪声的发生本质上是概率事件。若将出现“常规噪声”和“闪烁噪声”视为两个不同的事件,对应两种不同的观测误差模型,同时将两种噪声出现的概率视为模型概率,并用一阶马尔可夫模型对相邻时刻噪声分布的变化特性进行建模,则可利用交互多模(interacting multiple model, IMM)框架^[22-23]进行处理,解决传统高斯混合滤波算法中对高斯分量选择缺乏理论指导、存在模型适配性差的问题。

受此思路启发,本文以闪烁噪声的高斯混合分布建模为基础,提出了一种交互多模框架下的高性能跟踪滤波算法。该算法用一个高斯分量对应一种噪声分布模型,同时通过一阶马尔可夫模型建立转移概率矩阵,并用免导数运算、对非线性系统具有强适应能力的容积卡尔曼滤波器与不同高斯分量相对应的模型匹配滤波,最后对各匹配滤波结果进行综合得到最终的跟踪结果。仿真结果证明了所提方法的有效性,并通过与传统高斯混合滤波算法、粒子滤波算法等进行比较说明了本文方法的性能优势和执行效率优势。

1 问题描述

考虑典型的目标跟踪系统,目标离散时间状态演化方程可描述为

$$\mathbf{x}_k = f(\mathbf{x}_{k-1}) + \mathbf{w}_{k-1} \quad (1)$$

其中: $\mathbf{x}_k \in \mathbb{R}^{n_x}$ 为 n_x 维状态矢量,通常包括目标位置、速度等分量; $f(\cdot)$ 为状态转移方程; \mathbf{w}_{k-1} 为高斯过程噪声,其均值为零,协方差矩阵为 \mathbf{Q}_{k-1} ,简记为 $\mathbf{w}_{k-1} \sim \mathcal{N}(\mathbf{w}_{k-1}; \mathbf{0}, \mathbf{Q}_{k-1})$ 。

相应地,目标观测方程可描述为

$$\mathbf{z}_k = h(\mathbf{x}_k) + \mathbf{v}_k \quad (2)$$

式中: $\mathbf{z}_k \in \mathbb{R}^{n_z}$ 为 n_z 维观测矢量,通常包含目标距离、角度等分量; $h(\cdot)$ 为观测方程; \mathbf{v}_k 为观测噪声。

通常情况下, \mathbf{v}_k 被视为均值为零、协方差矩阵为 \mathbf{R}_k 的高斯观测噪声。但在雷达受到干扰、复杂目标姿态发生急变等条件下,测量误差会出现跳变,称为“闪烁”,测量噪声的高斯分布特性不再满足。此时,常用的处理方法是将闪烁噪声建模为混合高斯分布,即

$$\begin{aligned} p(\mathbf{v}_k) &= (1 - \varepsilon)p_1(\mathbf{v}_k) + \varepsilon p_2(\mathbf{v}_k) \\ &= (1 - \varepsilon)\mathcal{N}(\mathbf{v}_k; \mathbf{0}, \mathbf{R}_k^{(1)}) + \varepsilon\mathcal{N}(\mathbf{v}_k; \mathbf{0}, \mathbf{R}_k^{(2)}) \end{aligned} \quad (3)$$

式中, $p(\cdot)$ 表示概率密度函数(probability

density function, PDF), ε 为闪烁概率, $\mathbf{R}_k^{(1)}$ 为常规噪声协方差矩阵, $\mathbf{R}_k^{(2)}$ 为闪烁噪声协方差矩阵, 且 $\mathbf{R}_k^{(2)}$ 远大于 $\mathbf{R}_k^{(1)}$ 。

2 混合高斯滤波算法

从贝叶斯估计的方法论来看, k 时刻状态量 \mathbf{x}_k 的后验密度能够提供该时刻状态完整的统计描述。滤波问题就是在给定观测 $\mathbf{z}_{1:k} = \{\mathbf{z}_i\}_{i=1}^k$ 的条件下得到时变状态量 \mathbf{x}_k 的估计, 这就要求建立 \mathbf{x}_k 后验概率密度函数 $p(\mathbf{x}_k | \mathbf{z}_{1:k})$ 。假定初始 PDF 为 $p(\mathbf{x}_0) \triangleq p(\mathbf{x}_0 | \mathbf{z}_0)$, 则理论上 $p(\mathbf{x}_k | \mathbf{z}_{1:k})$ 可通过两步递推计算求得:

1) 预测: 假定 $k-1$ 时刻状态的后验 PDF 为 $p(\mathbf{x}_{k-1} | \mathbf{z}_{1:k-1})$, 则其一步前向预测的 PDF 可根据 Chapman-Kolmogorov 方程计算为

$$p(\mathbf{x}_k | \mathbf{z}_{1:k-1}) = \int p(\mathbf{x}_k | \mathbf{x}_{k-1}) p(\mathbf{x}_{k-1} | \mathbf{z}_{1:k-1}) d\mathbf{x}_{k-1} \quad (4)$$

式中, $p(\mathbf{x}_k | \mathbf{x}_{k-1})$ 称为状态的转移密度函数, 用于描述一阶马尔可夫过程。

2) 更新: 在获得 $p(\mathbf{x}_k | \mathbf{z}_{1:k-1})$ 的基础上, 综合新近的观测 \mathbf{z}_k , 可通过式(5)所示的贝叶斯规则计算 k 时刻状态的后验 PDF:

$$p(\mathbf{x}_k | \mathbf{z}_{1:k}) = \frac{p(\mathbf{z}_k | \mathbf{x}_k) p(\mathbf{x}_k | \mathbf{z}_{1:k-1})}{p(\mathbf{z}_k | \mathbf{z}_{1:k-1})} \quad (5)$$

式中, $p(\mathbf{z}_k | \mathbf{z}_{1:k-1}) = \int p(\mathbf{z}_k | \mathbf{x}_k) p(\mathbf{x}_k | \mathbf{z}_{1:k-1}) d\mathbf{x}_k$ 为归一化系数。

理论上, 任何一个复杂的分布函数都可以用多个高斯分布求和的形式进行近似, 混合高斯滤波算法正是基于这一原理发展起来的。

假定 $k-1$ 时刻目标的状态分布可用 I 个混合高斯分量来近似, 各分量权值为 $\lambda_{k-1}^{(i)}$, 且 $\sum_{i=1}^I \lambda_{k-1}^{(i)} = 1$, 则目标的后验状态分布为

$$p(\mathbf{x}_{k-1} | \mathbf{z}_{1:k-1}) = \sum_{i=1}^I \lambda_{k-1}^{(i)} \mathcal{N}(\mathbf{x}_{k-1}; \hat{\mathbf{x}}_{k-1}^{(i)}, \mathbf{P}_{k-1}^{(i)}) \quad (6)$$

式中, 第 i 个高斯分量的均值和协方差矩阵分别为 $\hat{\mathbf{x}}_{k-1}^{(i)}$ 和 $\mathbf{P}_{k-1}^{(i)}$ 。

又设 k 时刻的过程噪声为包含 J 个分量的混合高斯分布, 即

$$p(\mathbf{w}_k) \approx \sum_{j=1}^J \alpha_k^{(j)} \mathcal{N}(\mathbf{w}_k; \mathbf{0}, \mathbf{Q}_k^{(j)}) \quad (7)$$

式中, $\sum_{j=1}^J \alpha_k^{(j)} = 1$ 。

根据式(1), k 时刻状态转移方程的先验分布为

$$p(\mathbf{x}_k | \mathbf{x}_{k-1}) = \sum_{j=1}^J \alpha_k^{(j)} \mathcal{N}(\mathbf{x}_k; f(\mathbf{x}_{k-1}), \mathbf{Q}_k^{(j)}) \quad (8)$$

由此可得 k 时刻更新(预测)状态分布为

$$\begin{aligned} p(\mathbf{x}_k | \mathbf{z}_{1:k-1}) &= \sum_{j=1}^J \sum_{i=1}^I \lambda_{k-1}^{(i)} \alpha_k^{(j)} \int_{\mathbf{R}^{n_x}} \mathcal{N}(\mathbf{x}_{k-1}; \hat{\mathbf{x}}_{k-1}^{(i)}, \mathbf{P}_{k-1}^{(i)}) \cdot \\ &\quad \mathcal{N}(\mathbf{x}_k; f(\mathbf{x}_{k-1}), \mathbf{Q}_k^{(j)}) d\mathbf{x}_{k-1} \\ &= \sum_{r=1}^{I \cdot J} \lambda_{klk-1}^{(r)} \mathcal{N}(\mathbf{x}_k; \hat{\mathbf{x}}_{klk-1}^{(r)}, \mathbf{P}_{klk-1}^{(r)}) \end{aligned} \quad (9)$$

式中, $r = (i-1)J + j$, $\lambda_{klk-1}^{(r)} = \lambda_{k-1}^{(i)} \cdot \alpha_k^{(j)}$, $\hat{\mathbf{x}}_{klk-1}^{(r)}$ 和 $\mathbf{P}_{klk-1}^{(r)}$ 则可通过高斯近似滤波器(或称类卡尔曼滤波器^[24])的时间更新步骤得到。

同理, 假定 k 时刻的观测噪声可建模为如下的混合高斯分布形式

$$p(\mathbf{v}_k) \approx \sum_{l=1}^L \beta_k^{(l)} \mathcal{N}(\mathbf{v}_k; \mathbf{0}, \mathbf{R}_k^{(l)}) \quad (10)$$

式中, $\sum_{l=1}^L \beta_k^{(l)} = 1$ 。

由式(2)可知, k 时刻观测方程的似然分布为

$$p(\mathbf{z}_k | \mathbf{x}_k) = \sum_{l=1}^L \beta_k^{(l)} \mathcal{N}(\mathbf{z}_k; h(\mathbf{x}_k), \mathbf{R}_k^{(l)}) \quad (11)$$

由此可得 k 时刻测量更新(修正)状态分布为

$$\begin{aligned} p(\mathbf{x}_k | \mathbf{z}_{1:k}) &= \sum_{l=1}^L \sum_{r=1}^{I \cdot J} \beta_k^{(l)} \lambda_{klk-1}^{(r)} \int_{\mathbf{R}^{n_x}} \mathcal{N}(\mathbf{x}_k; \hat{\mathbf{x}}_{klk-1}^{(r)}, \mathbf{P}_{klk-1}^{(r)}) \cdot \\ &\quad \mathcal{N}(\mathbf{z}_k; h(\mathbf{x}_k), \mathbf{R}_k^{(l)}) d\mathbf{x}_k \\ &= \sum_{n=1}^{I \cdot J \cdot L} \lambda_k^{(n)} \mathcal{N}(\mathbf{x}_k; \hat{\mathbf{x}}_k^{(n)}, \mathbf{P}_k^{(n)}) \end{aligned} \quad (12)$$

式中: $\lambda_k^{(n)} = \frac{\lambda_{klk-1}^{(r)} \cdot \gamma^{(n)} \cdot \beta_k^{(l)}}{\sum_{r=1}^{I \cdot J} \sum_{l=1}^L \lambda_{klk-1}^{(r)} \cdot \gamma^{(n)} \cdot \beta_k^{(l)}}$, $n = (r-1)L + l$, $\gamma^{(n)} = \mathcal{N}(\mathbf{z}_k; \hat{\mathbf{z}}_{klk-1}^{(n)}, \mathbf{P}_{zz}^{(n)})$, $\hat{\mathbf{z}}_{klk-1}^{(n)}$ 、 $\mathbf{P}_{zz}^{(n)}$ 分别表示第 n 个分量观测的预测及相应的预测误差协方差矩阵。

$\hat{\mathbf{x}}_k^{(n)}$ 和 $\mathbf{P}_k^{(n)}$ 则可通过高斯近似滤波器的量测更新步骤得到。

由此可见, 经过一步完整的递推过程, 高斯分量已由 $k-1$ 时刻的 I 个变为 k 时刻的 $I \cdot J \cdot L$ 个, 随着递推的进行, 高斯分量数将爆炸式增长。为此需要在每一步递推完成后, 进行剪枝处理, 主要是舍弃权值较小的高斯分量, 同时对相似性较强(空间距离小)的高斯分量进行合并, 具体方法可参考文献[25]。经剪枝后, 重新恢复成 I 个高斯分量, 即

$$\{\lambda_k^{(n)}, \hat{\mathbf{x}}_k^{(n)}, \mathbf{P}_k^{(n)}\}_{n=1}^{I \cdot J \cdot L} \rightarrow \{\tilde{\lambda}_k^{(i)}, \tilde{\mathbf{x}}_k^{(i)}, \tilde{\mathbf{P}}_k^{(i)}\}_{i=1}^I \quad (13)$$

3 交互多模框架下的 CKF 算法

3.1 闪烁噪声等效建模

由式(3)可知,尽管在不同的时刻,观测噪声可能不同(闪烁噪声出现是随机的),但在某一特定时刻,噪声的分布特性是确定的,即 $\mathbf{v}_k \sim \mathcal{N}(\mathbf{v}_k; \mathbf{0}, \mathbf{R}_k^{(1)})$ 或 $\mathbf{v}_k \sim \mathcal{N}(\mathbf{v}_k; \mathbf{0}, \mathbf{R}_k^{(2)})$ 。为此,假定用 r_k 表示模式变量,用 $\mathcal{M} = \{\mathcal{M}_j\}_{j=1}^2$ 代表模式集,其中 \mathcal{M}_1 为“出现常规噪声”事件, \mathcal{M}_2 为“出现闪烁噪声”事件,则式(2)可进一步表示为

$$\mathbf{z}_k = h(\mathbf{x}_k) + \mathbf{v}_k(r_k) \quad (14)$$

式中, $r_k \in \mathcal{M}$ 。

由模型条件可知, $\Pr(r_k = \mathcal{M}_1) = 1 - \varepsilon$, $\Pr(r_k = \mathcal{M}_2) = \varepsilon$, 其中 $\Pr(\cdot)$ 表示概率函数。

假定用 π_{ij} 表示 $k-1$ 时刻由模式 \mathcal{M}_i 向 k 时刻模式 \mathcal{M}_j 转移的条件概率,则有

$$\pi_{ij} \triangleq \Pr\{r_k = \mathcal{M}_j | r_{k-1} = \mathcal{M}_i\} \quad i, j = 1, 2 \quad (15)$$

相邻时刻的模式变化关系可用一阶马尔可夫链来描述,其转化关系如图1所示。

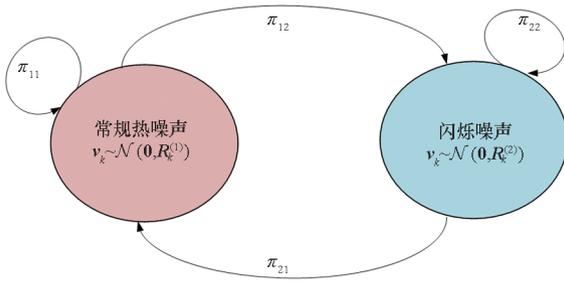


图1 常规噪声和闪烁噪声的模式转移关系

Fig.1 Mode transition relation of normal noise and glint noise

根据图1中所示的模式转化关系,可得转移概率矩阵为

$$\mathbf{\Pi} = [\pi_{ij}] = \begin{bmatrix} \pi_{11} & \pi_{12} \\ \pi_{21} & \pi_{22} \end{bmatrix} = \begin{bmatrix} 1 - \varepsilon & \varepsilon \\ 1 - \varepsilon & \varepsilon \end{bmatrix} \quad (16)$$

需要说明的是,式(16)与传统为机动目标跟踪所设计的转移概率矩阵是不同的,因为后者在主对角线上的元素体现的是下一个时刻保持与当前时刻一致运动模式的概率,因此其取值往往较大^[26]。

对于每一个 \mathcal{M}_i , $\mathcal{M}_j \in \mathcal{M}$, 初始模型概率表示为

$$\eta_0^{(i)} \triangleq \Pr\{r_0 = \mathcal{M}_i\} \quad i = 1, 2 \quad (17)$$

而对于 k 时刻的每一个 $\eta_k^{(i)}$, 满足条件

$$\begin{cases} \eta_k^{(i)} \geq 0 \\ \sum_{i=1}^2 \eta_k^{(i)} = 1 \end{cases} \quad (18)$$

3.2 交互多模递推框架

考虑由式(1)和式(14)组成的包含多种不同模型的混合系统估计问题,根据全概公式,系统状态的后验 PDF 为

$$\begin{aligned} p(\mathbf{x}_k | \mathbf{z}_{1:k}) &= \sum_{j=1}^2 p(\mathbf{x}_k | r_k(j), \mathbf{z}_{1:k}) \Pr(r_k(j) | \mathbf{z}_{1:k}) \\ &= \sum_{j=1}^2 p(\mathbf{x}_k | r_k(j), \mathbf{z}_{1:k-1}, \mathbf{z}_k) \eta_k^{(j)} \end{aligned} \quad (19)$$

为表示方便,式中用 $r_k(j)$ 代表 $r_k = \mathcal{M}_j$ 。

以各模型为条件的状态后验 PDF 为

$$\begin{aligned} p(\mathbf{x}_k | r_k(j), \mathbf{z}_{1:k-1}, \mathbf{z}_k) &= \frac{p(\mathbf{z}_k | r_k(j), \mathbf{x}_k)}{p(\mathbf{z}_k | r_k(j), \mathbf{z}_{1:k-1})} p(\mathbf{x}_k | r_k(j), \mathbf{z}_{1:k-1}) \end{aligned} \quad (20)$$

将全概公式再次用于式(20)中的第二项,则

$$\begin{aligned} p(\mathbf{x}_k | r_k(j), \mathbf{z}_{1:k-1}) &= \sum_{i=1}^2 p(\mathbf{x}_k | r_k(j), r_{k-1}(i), \mathbf{z}_{1:k-1}) \Pr(r_{k-1}(i) | r_k(j), \mathbf{z}_{1:k-1}) \end{aligned} \quad (21)$$

在高斯近似假设条件下,式(21)可进一步表示为

$$\begin{aligned} p(\mathbf{x}_k | r_k(j), \mathbf{z}_{1:k-1}) &= \sum_{i=1}^2 p(\mathbf{x}_k | r_k(j), r_{k-1}(i), \hat{\mathbf{x}}_{k-1}^{(i)}, \mathbf{P}_{k-1}^{(i)}) \mu_{k-1}[i|j] \end{aligned} \quad (22)$$

式中, $\mu_{k-1}[i|j] \triangleq \Pr(r_{k-1}(i) | r_k(j), \mathbf{z}_{1:k-1})$ 称为混合概率。

进一步假定式(22)所示的混合形式为高斯混合分布,而后采用单个高斯分量对其进行矩匹配近似,由此可得

$$\begin{aligned} p(\mathbf{x}_k | r_k(j), \mathbf{z}_{1:k-1}) &= \mathcal{N}(\mathbf{x}_k; E[\mathbf{x}_k | r_k(j), \sum_{i=1}^2 \hat{\mathbf{x}}_{k-1}^{(i)} \mu_{k-1}[i|j]], \mathbf{Cov}(\mathbf{x})) \end{aligned} \quad (23)$$

式中, $\mathbf{Cov}(\mathbf{x}) = E[(\mathbf{x} - \bar{\mathbf{x}})(\mathbf{x} - \bar{\mathbf{x}})^T]$ 代表协方差矩阵。

利用式(23)实现的多模滤波算法称为 IMM 算法,图2给出了该算法的实现结构,包含2个并行的交互滤波器,混合过程在每个滤波器的输入阶段完成,并以 $\mathbf{z}_{1:k-1}$ 为条件。混合概率的计算将在下文的具体算法流程中详细说明。

3.3 基于 CKF 匹配滤波的算法实现

由式(19)可知,为了实现对目标状态的估计,在每一时刻 k ,需要用2个与模式 $r_k(j)$ ($j=1, 2$) 匹配的高斯近似滤波器进行处理。为了提高对非线性观测条件的适应能力,本文采用 CKF 来

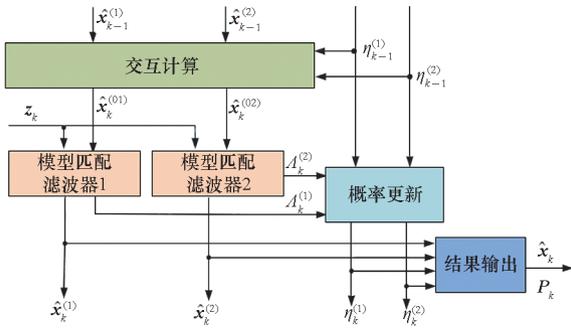


图 2 IMM 算法框架

Fig. 2 Framework of IMM algorithm

实现。为此,假定 $k-1$ 时刻与模式 $r_{k-1}(j)$ 相对应的目标状态估计、估计误差协方差矩阵及模型概率为 $\{\hat{\mathbf{x}}_{k-1}^{(j)}, \mathbf{P}_{k-1}^{(j)}, \eta_{k-1}^{(j)}\}$, 由此可得如算法 1 所示的交互多模滤波算法流程,其中 Step 3 中 CKF 算法的详细实现过程见算法 2。

算法 1 交互多模滤波

Alg. 1 Interacting multiple model filtering

Step 1: 混合概率计算

在 $\mathbf{z}_{1:k-1}$ 和 k 时刻模式 $r_k(j)$ 已知的条件下, $k-1$ 时刻模式 $r_{k-1}(i)$ 的概率为

$$\begin{aligned} \mu_{k-1}[i|j] &\triangleq \Pr(r_{k-1}(i) | r_k(j), \mathbf{z}_{1:k-1}) \\ &= \Pr(r_k(j) | r_{k-1}(i), \mathbf{z}_{1:k-1}) \cdot \\ &\quad \Pr(r_{k-1}(i) | \mathbf{z}_{1:k-1}) / \bar{c}_j \end{aligned} \quad (24)$$

式(24)中的混合概率可进一步表示为

$$\mu_{k-1}[i|j] = \pi_{ij} \eta_{k-1}^{(i)} / \bar{c}_j \quad i, j = 1, 2 \quad (25)$$

式中,归一化常数 $\bar{c}_j = \sum_{i=1}^r \pi_{ij} \eta_{k-1}^{(i)}$ 。

Step 2: 模型混合

由 $\hat{\mathbf{x}}_{k-1}^{(i)}$ 出发,根据式(23)计算与 $r_k(j)$ 相对应的混合初始条件

$$\hat{\mathbf{x}}_{k-1}^{(0j)} = \sum_{i=1}^2 \hat{\mathbf{x}}_{k-1}^{(i)} \mu_{k-1}[i|j] \quad j = 1, 2 \quad (26)$$

相应的协方差矩阵为

$$\begin{aligned} \mathbf{P}_{k-1}^{(0j)} &= \sum_{i=1}^2 \mu_{k-1}[i|j] \{ (\hat{\mathbf{x}}_{k-1}^{(i)} - \hat{\mathbf{x}}_{k-1}^{(0j)}) (\hat{\mathbf{x}}_{k-1}^{(i)} - \hat{\mathbf{x}}_{k-1}^{(0j)})^T + \mathbf{P}_{k-1}^{(i)} \} \\ &\quad j = 1, 2 \end{aligned} \quad (27)$$

Step 3: 基于 CKF 的模型匹配滤波

利用 Step 2 的结果作为模式 $r_k(j)$ 的匹配滤波输入,结合量测 \mathbf{z}_k 和 CKF 滤波过程得到 $\hat{\mathbf{x}}_k^{(j)}$ 和 $\mathbf{P}_k^{(j)}$, 即

$$\{\hat{\mathbf{x}}_{k-1}^{(0j)}, \mathbf{P}_{k-1}^{(0j)}, \mathbf{Q}_{k-1}, \mathbf{z}_k, \mathbf{R}_k^{(j)}\}_{j=1}^2 \rightarrow \{\hat{\mathbf{x}}_k^{(j)}, \mathbf{P}_k^{(j)}, \mathbf{P}_{zz,k}^{(j)}\}_{j=1}^2 \quad (28)$$

Step 4: 似然函数计算

$$\begin{aligned} \Lambda_k^{(j)} &= p(\mathbf{z}_k | r_k(j), \mathbf{z}_{1:k-1}) = p(\mathbf{z}_k | r_k(j), \hat{\mathbf{x}}_{k-1}^{(0j)}, \mathbf{P}_{k-1}^{(0j)}) \\ &\approx \mathcal{N}(\mathbf{z}_k; \hat{\mathbf{z}}_{k|k-1}^{(j)}, \mathbf{P}_{zz,k}^{(j)}) \end{aligned} \quad (29)$$

Step 5: 模型概率更新

$$\begin{aligned} \eta_k^{(j)} &\triangleq \Pr(r_k(j) | \mathbf{z}_{1:k}) = p(\mathbf{z}_k | r_k(j), \mathbf{z}_{1:k-1}) \Pr(r_k(j) | \mathbf{z}_{1:k-1}) / c \\ &= \Lambda_k^{(j)} \sum_{i=1}^2 p(r_k(j) | r_{k-1}(i), \mathbf{z}_{1:k-1}) \Pr(r_{k-1}(i) | \mathbf{z}_{1:k-1}) / c \\ &= \Lambda_k^{(j)} \sum_{i=1}^2 \pi_{ij} \eta_{k-1}^{(i)} / c \end{aligned} \quad (30)$$

式中, $c = \sum_{j=1}^2 \Lambda_k^{(j)} \bar{c}_j$ 。

Step 6: 状态和协方差估计

基于模型的条件估计及协方差为

$$\hat{\mathbf{x}}_k = \sum_{j=1}^2 \hat{\mathbf{x}}_k^{(j)} \eta_k^{(j)} \quad (31)$$

$$\mathbf{P}_k = \sum_{j=1}^2 \eta_k^{(j)} [(\hat{\mathbf{x}}_k^{(j)} - \hat{\mathbf{x}}_k) (\hat{\mathbf{x}}_k^{(j)} - \hat{\mathbf{x}}_k)^T + \mathbf{P}_k^{(j)}] \quad (32)$$

需要指出的是,算法 1 中 Step 6 只用于结果输出,并不参与递推计算。

算法 2 基于 CKF 的模型匹配滤波

Alg. 2 Model-matched filtering based on CKF

输入: $\{\hat{\mathbf{x}}_{k-1}^{(0j)}, \mathbf{P}_{k-1}^{(0j)}, \mathbf{Q}_{k-1}, \mathbf{z}_k, \mathbf{R}_k^{(j)}\}_{j=1}^2$

输出: $\{\hat{\mathbf{x}}_k^{(j)}, \mathbf{P}_k^{(j)}, \mathbf{P}_{zz,k}^{(j)}\}_{j=1}^2$

Step 1: 计算容积分点及相应的权值

$$\begin{cases} \chi_l = \sqrt{n_x} [\mathbf{1}]_l & l = 1, 2, \dots, 2n_x \\ w_l = 1 / (2n_x) \end{cases}$$

式中: $[\mathbf{1}] \in \mathbb{R}^{n_x}$, 且有

$$[\mathbf{1}]_l \in \left\{ \begin{bmatrix} 1 \\ 0 \\ \vdots \\ 0 \end{bmatrix}, \begin{bmatrix} 0 \\ 1 \\ \vdots \\ 0 \end{bmatrix}, \dots, \begin{bmatrix} 0 \\ 0 \\ \vdots \\ 1 \end{bmatrix}, \begin{bmatrix} -1 \\ 0 \\ \vdots \\ 0 \end{bmatrix}, \begin{bmatrix} 0 \\ -1 \\ \vdots \\ 0 \end{bmatrix}, \dots, \begin{bmatrix} 0 \\ 0 \\ \vdots \\ -1 \end{bmatrix} \right\}$$

Step 2: 时间更新

1) 滤波协方差矩阵分解

$$\mathbf{P}_{k-1}^{(0j)} = \mathbf{S}_{k-1}^{(j)} (\mathbf{S}_{k-1}^{(j)})^T$$

2) 计算求积点

$$\xi_{l,k-1}^{(j)} = \mathbf{S}_{k-1}^{(j)} \cdot \chi_l + \hat{\mathbf{x}}_{k-1}^{(0j)}$$

3) 传播求积分点

$$\xi_{l,k|k-1}^{(j)} = f(\xi_{l,k-1}^{(j)})$$

4) 状态预测

$$\hat{\mathbf{x}}_{k|k-1}^{(j)} = \sum_{l=1}^{2n_x} w_l \xi_{l,k|k-1}^{(j)}$$

5) 状态预测协方差矩阵

$$\mathbf{P}_{k|k-1}^{(j)} = \sum_{l=1}^{2n_x} w_l \xi_{l,k|k-1}^{(j)} (\xi_{l,k|k-1}^{(j)})^T - \hat{\mathbf{x}}_{k|k-1}^{(j)} (\hat{\mathbf{x}}_{k|k-1}^{(j)})^T + \mathbf{Q}_{k-1}$$

Step 3: 量测更新

1) 矩阵分解

$$\mathbf{P}_{k|k-1}^{(j)} = \mathbf{S}_{k|k-1}^{(j)} (\mathbf{S}_{k|k-1}^{(j)})^T$$

2) 计算求积分点

$$\xi_{l,k|k-1}^{(j)} = \mathbf{S}_{k|k-1}^{(j)} \cdot \chi_l + \hat{\mathbf{x}}_{k|k-1}^{(j)}$$

3) 传播求积点

$$\boldsymbol{\varepsilon}_{l,k|k-1}^{(j)} = h(\boldsymbol{\zeta}_{l,k|k-1}^{(j)})$$

4) 观测的预测

$$\hat{\boldsymbol{z}}_{kl|k-1}^{(j)} = \sum_{l=1}^{2n_x} w_l \boldsymbol{\varepsilon}_{l,k|k-1}^{(j)}$$

5) 预测误差协方差

$$\boldsymbol{P}_{zz,k}^{(j)} = \sum_{l=1}^{2n_x} w_l (\boldsymbol{\varepsilon}_{l,k|k-1}^{(j)} - \hat{\boldsymbol{z}}_{kl|k-1}^{(j)}) (\boldsymbol{\varepsilon}_{l,k|k-1}^{(j)} - \hat{\boldsymbol{z}}_{kl|k-1}^{(j)})^T + \boldsymbol{R}_k^{(j)}$$

6) 计算互协方差矩阵

$$\boldsymbol{P}_{xz,k}^{(j)} = \sum_{l=1}^{2n_x} w_l (\boldsymbol{\zeta}_{l,k|k-1}^{(j)} - \hat{\boldsymbol{x}}_{kl|k-1}^{(j)}) (\boldsymbol{\varepsilon}_{l,k|k-1}^{(j)} - \hat{\boldsymbol{z}}_{kl|k-1}^{(j)})^T$$

7) 计算卡尔曼增益

$$\boldsymbol{K}_k^{(j)} = \boldsymbol{P}_{xz,k}^{(j)} (\boldsymbol{P}_{zz,k}^{(j)})^{-1}$$

8) 状态更新

$$\hat{\boldsymbol{x}}_k^{(j)} = \hat{\boldsymbol{x}}_{kl|k-1}^{(j)} + \boldsymbol{K}_k^{(j)} (\boldsymbol{z}_k - \hat{\boldsymbol{z}}_{kl|k-1}^{(j)})$$

9) 状态协方差矩阵更新

$$\boldsymbol{P}_k^{(j)} = \boldsymbol{P}_{kl|k-1}^{(j)} - \boldsymbol{K}_k^{(j)} \boldsymbol{P}_{zz,k}^{(j)} (\boldsymbol{K}_k^{(j)})^T$$

4 仿真实验

4.1 仿真条件

以雷达制导跟踪应用为例,假定 k 时刻目标状态为 $\boldsymbol{x}_k^l = [x_k^l \ x_k^l \ y_k^l \ y_k^l]^T$, 目标的运动方程满足

$$\boldsymbol{x}_k^l = \boldsymbol{F} \boldsymbol{x}_{k-1}^l + \boldsymbol{w}_k^l \quad (33)$$

式中:

$$\boldsymbol{F} = \begin{bmatrix} \boldsymbol{F}_1 & \mathbf{0} \\ \mathbf{0} & \boldsymbol{F}_1 \end{bmatrix}$$

$$\boldsymbol{F}_1 = \begin{bmatrix} 1 & T_s \\ 0 & 1 \end{bmatrix}$$

T_s 为采样间隔; \boldsymbol{w}_k^l 为目标运动的过程噪声,这里假定服从均值为零、协方差矩阵为 \boldsymbol{Q}_k 的高斯分布,且有

$$\boldsymbol{Q}_k = q^2 \cdot \begin{bmatrix} \boldsymbol{Q}_1 & \mathbf{0} \\ \mathbf{0} & \boldsymbol{Q}_1 \end{bmatrix}$$

$$\boldsymbol{Q}_1 = \begin{bmatrix} T_s^3/3 & T_s^2/2 \\ T_s^2/2 & T_s \end{bmatrix}$$

导弹的运动方程为

$$\boldsymbol{x}_k^m = \boldsymbol{F} \boldsymbol{x}_{k-1}^m + \boldsymbol{G} \boldsymbol{u}_{k-1} + \boldsymbol{w}_k^m \quad (34)$$

式中: $\boldsymbol{x}_k^m = [x_k^m \ \dot{x}_k^m \ y_k^m \ \dot{y}_k^m]^T$, $\boldsymbol{u} = [a_x \ a_y]^T$, $\boldsymbol{G} =$

$$\begin{bmatrix} -T_s^2/2 \\ -T_s \end{bmatrix}, a_x = [c_1 \ c_2] \begin{bmatrix} x_k \\ \dot{x}_k \end{bmatrix}, a_y = [c_1 \ c_2] \begin{bmatrix} y_k \\ \dot{y}_k \end{bmatrix}, \boldsymbol{x}_k =$$

$\boldsymbol{x}_k^l - \boldsymbol{x}_k^m$ 为相对状态; $c_1 = N_g/t_{go}^2$, $c_2 = N_g/t_{go}$, N_g 为导引系数, t_{go} 为剩余时间,相关算式基于能量消耗最小化制导控制理论^[27]生成; \boldsymbol{w}_k^m 为导弹运动的

过程噪声,同样建模为服从零均值、协方差为 \boldsymbol{Q}_k 的高斯分布。

雷达传感器位于弹载观测坐标系的原点,其观测量包括目标的距离和方位角,观测方程的具体形式为

$$\boldsymbol{z}_k = h(\boldsymbol{x}_k) + \boldsymbol{v}_k = \begin{bmatrix} \sqrt{x_k^2 + y_k^2} \\ \arctan(y_k/x_k) \end{bmatrix} + \begin{bmatrix} v_{r,k} \\ v_{\phi,k} \end{bmatrix} \quad (35)$$

式中:距离和方位观测噪声 \boldsymbol{v}_k 的分布形式同式(3),闪烁噪声出现的概率为 ε 。

4.2 仿真结果

仿真中假定目标和导弹的初始状态分别为

$$\boldsymbol{x}_0^l = [20 \text{ km} \ -100 \text{ m/s} \ 1.5 \text{ km} \ 50 \text{ m/s}]^T$$

$$\boldsymbol{x}_0^m = [40 \text{ km} \ -1 \ 000 \text{ m/s} \ 15 \text{ km} \ -150 \text{ m/s}]^T$$

相应的过程噪声强度为 $q^l = q^m = 2 \text{ m/s}^2$, 采样间隔 $T_s = 0.5 \text{ s}$, 常规噪声和闪烁噪声协方差矩阵分别为 $\boldsymbol{R}_k^{(1)} = \text{diag}[(20 \text{ m})^2 \ (0.2^\circ)^2]$, $\boldsymbol{R}_k^{(2)} = 25\boldsymbol{R}_k^{(1)}$, $\varepsilon = 0.25$ 。进一步假定要求导弹在 $T_f = 60 \text{ s}$ 时间内完成对目标的拦截,导引系数 $N_g = 2.5$, $t_{k,go} = T_f - k \cdot T_s (k = 1, 2, \dots)$ 。图3给出了前30 s 弹目接近的过程。

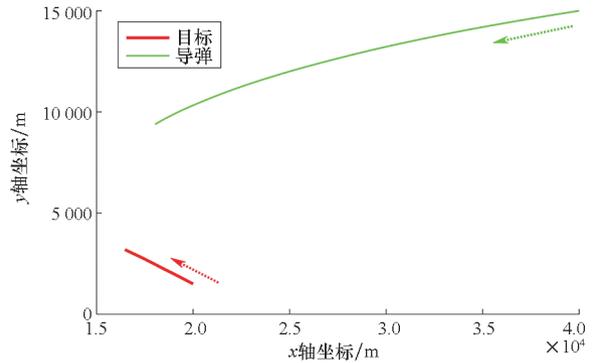


图3 导弹拦截目标实验场景

Fig. 3 Experimental scenario for target intercept

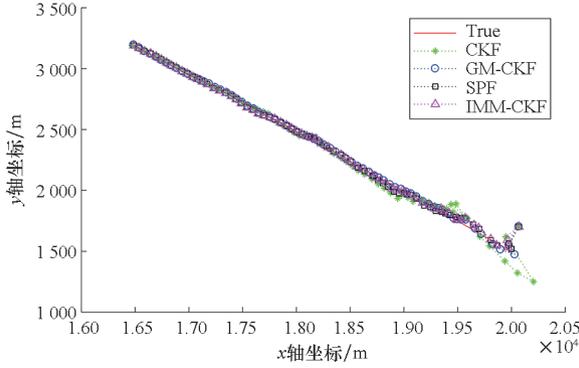
分别采用CKF、高斯混合容积卡尔曼滤波器GM-CKF、标准粒子滤波器(standard PF, SPF)以及本文提出的交互多模容积卡尔曼滤波器IMM-CKF对目标进行跟踪,目标初始状态估计误差的协方差矩阵为

$$\boldsymbol{P}_0 = \text{diag}[(200 \text{ m})^2 \ (100 \text{ m/s})^2 \ (200 \text{ m})^2 \ (100 \text{ m/s})^2]$$

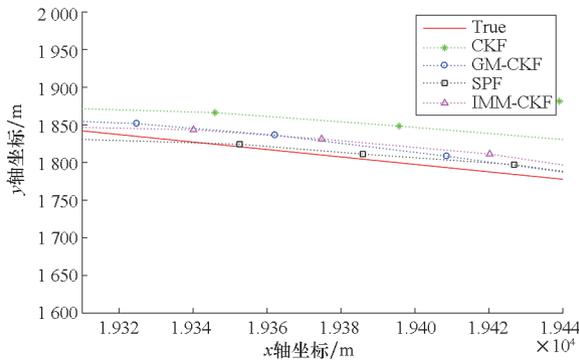
跟踪算法的初始状态由均值为 \boldsymbol{x}_0^l 、协方差为 \boldsymbol{P}_0 的高斯模型随机生成。在GM-CKF算法中,对目标状态、过程噪声和观测噪声进行模型近似的高斯分量数分别为 $I = 10$, $J = 1$, $L = 2$ 。对于SPF算法,粒子从先验分布(转移函数)中进行采样,粒子数为20 000。在CKF算法中,根据二阶矩近似方法,

将相应的协方差矩阵设为 $R_k = (1 - \varepsilon)R_k^{(1)} + \varepsilon R_k^{(2)}$ 。

单次实验跟踪结果如图 4 所示,由此可见,尽管四种算法都能实现对目标的跟踪,但其估计效果各有差异,这一点在图 4(b)可以更清楚地看出来;同时不难发现,相较其他几种算法,CKF 的误差最大,尤其是在闪烁噪声出现的时刻更加明显。



(a) 原图
(a) Original view

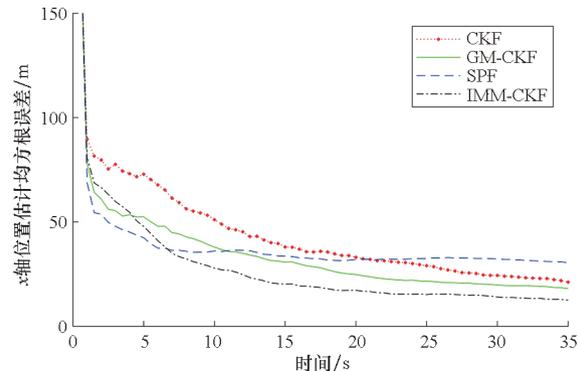


(b) 局部放大图
(b) Partially zoomed view

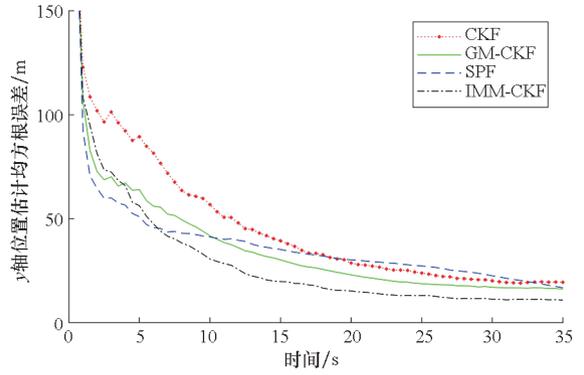
图 4 单次实验跟踪结果示例

Fig. 4 Target tracking for a single run

为了得到量化的分析评估结果,对 500 次 Monte Carlo 实验结果进行统计,得到目标位置估计的均方根误差 (root mean square error, RMSE) 如图 5 所示。



(a) x 轴
(a) Axis-x



(b) y 轴
(b) Axis-y

图 5 目标位置估计均方根误差

Fig. 5 RMSE for target position estimation

由图 5 中的结果可以清楚地看出,在存在较大初始误差的情况下,四种跟踪算法经数个周期的递推滤波以后,目标位置估计误差快速下降,并逐渐趋于收敛状态。相比之下,尽管 SPF 算法在前几个观测周期以最快的速度使估计误差降至一定的范围,但在 6 s 之后误差下降缓慢(见 y 轴)或已趋于不变(见 x 轴),最终的跟踪精度相对偏低;CKF、GM-CKF 和 IMM-CKF 三种算法的跟踪误差虽呈现一致的收敛趋势,但其误差大小各不相同,且渐次减小。整体上,IMM-CKF 算法表现出最佳的跟踪效果,这种优势在目标跟踪持续 6 s 之后更加明显。

为考察不同闪烁噪声对跟踪性能的影响,在前述仿真条件不变的情况下, ε 分别取 0.05, 0.10, 0.15, 0.20, 0.25, 0.40, 重新开展实验,并对 6 s 以后各观测周期的均方根误差取均值,得到平均均方根误差 (averaged RMSE, ARMSE), 如表 1 所示。表中各个不同 ε 取值条件下第一行数据对应 x 轴方向 ARMSE, 第二行数据对应 y 轴方向 ARMSE。

表 1 不同闪烁噪声条件下的位置估计 ARMSE

Tab. 1 ARMSE for target position estimation with different glint probability

单位: m				
ε	CKF	GM-CKF	SPF	IMM-CKF
0.05	20.90	16.25	21.13	15.46
	20.28	15.78	20.27	14.93
0.10	25.16	18.04	27.52	16.37
	24.73	17.33	26.51	15.66
0.15	28.75	20.99	32.61	17.05
	28.31	20.18	31.73	16.42
0.20	32.01	23.44	33.20	18.09
	31.54	23.39	32.63	17.47
0.25	35.16	27.15	35.68	19.22
	34.04	26.58	34.01	18.57
0.40	41.56	36.59	39.85	23.13
	41.45	35.01	38.30	22.57

由表1可知,随着闪烁概率的增大,四种算法的 ARMSE 也随之增大;相比之下,IMM-CKF 算法的性能最优,且闪烁噪声出现的概率越大,IMM-CKF 相对于其他算法的性能增量越明显。

由前文交互多模算法推导过程可知,IMM-CKF 除了能跟踪目标状态,还可根据模型概率对闪烁噪声出现与否进行在线估计。图6给出了 $\varepsilon = 0.10$ 时,单次仿真中在不同时刻出现真实闪烁噪声的示例,以及由 IMM-CKF 得到的闪烁噪声出现概率的估计结果。由图6中的结果不难看出,对于大部分真实出现过闪烁噪声的时间点,IMM-CKF 也相应呈现较高(或接近1)的闪烁噪声模型概率,说明估计结果与实际情况吻合良好。

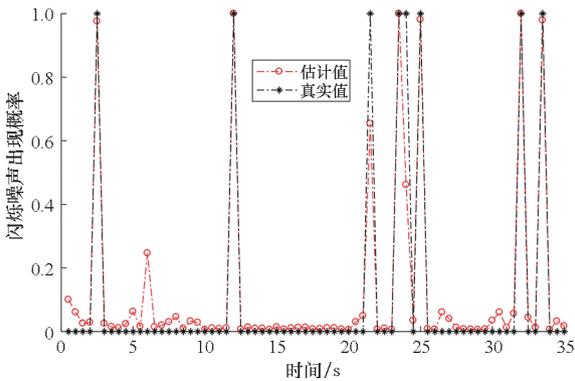


图6 闪烁噪声出现时刻估计结果

Fig.6 Estimation result for the existence of the glint noise

为定量评估 IMM-CKF 对闪烁噪声出现时刻的估计性能,表2中给出了不同 ε 条件下对闪烁噪声出现时刻估计的准确率,其结果通过500次 Monte Carlo 实验得到。

表2 不同仿真条件下对闪烁噪声出现时刻估计的准确率

Tab.2 Estimation accuracy for the existence of glint noise under different simulation conditions

ε	准确率
0.05	0.859
0.10	0.780
0.15	0.778
0.20	0.772
0.25	0.767
0.40	0.753

由表2可以看出,总体上,所提 IMM-CKF 能以较高的准确率对闪烁噪声出现的时刻进行估

计,这对目标姿态突变检测和干扰判断应用等具有重要的参考价值;同时可以看出,随着闪烁概率的增大,对闪烁噪声出现时刻估计的准确率也有所下降,这与实际情况也是相符的。究其原因,主要是因为滤波算法是通过递推更新的形式实现的,即当前时刻的模式概率估计结果同时受当前时刻量测和前一时刻估计结果的影响。当闪烁概率增大时,模型跳变得更为频繁,导致短时间内(或瞬时)得到稳定的模型概率估计结果难度增大,从而造成闪烁噪声出现时刻估计精度降低的现象。

根据文献[28-29]中的分析,对于维数为 n_x 的状态量,CKF 的算法复杂度为 $O(n_x^3)$;在粒子数为 M 的情况下,SPF 的算法复杂度为 $O(Mn_x^2)$ 。类似地,可得 IMM-CKF 和 GM-CKF 的算法复杂度分别为 $O(2n_x^3)$ 和 $O(20n_x^3)$ 。为更直观体现算法执行效率,本文对 CKF、GM-CKF、SPF 和 IMM-CKF 算法的运行时间进行了比较。仿真实验是在 MATLAB2020a 环境下完成的,计算机的基本配置是 4.00 GHz Intel(R) Core(TM) i7-6700 和 16 GB RAM。对500次仿真消耗的总时间进行统计,得到单次跟踪的平均时间如表3所示,表中第三列是以 CKF 算法运行时间为基准得到的归一化时间消耗。

表3 四种算法单次仿真运行时间

Tab.3 Running time of the four algorithms for a single run

跟踪算法	运行时间/ms	归一化运行时间
CKF	7.4	1.00
GM-CKF	152.5	20.61
SPF	220.7	29.82
IMM-CKF	16.2	2.19

由表3中的结果可以看出,由于 CKF 算法仅采用了单个高斯分量,因此耗时最少。GM-CKF 和 IMM-CKF 的运行时间分别是 CKF 的 20.61 倍和 2.19 倍,这与 GM-CKF 采用了 20 个高斯分量、IMM-CKF 采用了 2 个高斯分量进行近似的实际情况也是完全吻合的。相比之下,SPF 耗时最长,且是在粒子并行采样条件下得到的结果。由此可见,IMM-CKF 算法复杂度适中,实时性强,非常便于工程实现与应用。

5 结论

闪烁噪声是影响目标跟踪精度的一种重要

因素,在制导雷达应用中,如何克服闪烁噪声的影响直接关系到导弹对目标的命中精度和作战效能。本文以闪烁噪声的混合高斯建模为基础,将闪烁噪声发生概率建成一阶马尔可夫模型,并在混合系统理论框架下导出了高斯近似解的表达式,同时通过容积点求解非线性积分方程,从而得到交互多模容积卡尔曼滤波器 IMM-CKF。结合典型制导跟踪示例,对所提算法的有效性进行了验证。仿真结果表明,本文算法复杂度仅为传统 CKF 的 2 倍,明显低于 GM-CKF 和 SPF 算法,且在不同闪烁概率条件下均取得了一致最优的跟踪性能;除此之外,所提算法还能对闪烁噪声出现时刻进行有效的估计,具有很好的实用价值。

参考文献 (References)

- [1] WU W R. Target tracking with glint noise [J]. IEEE Transactions on Aerospace and Electronic Systems, 1993, 29(1): 174 - 185.
- [2] BILIK I, TABRIKIAN J. Maneuvering target tracking in the presence of glint using the nonlinear Gaussian mixture Kalman filter [J]. IEEE Transactions on Aerospace and Electronic Systems, 2010, 46(1): 246 - 262.
- [3] CUI H R, WANG X X, GAO S H, et al. A Gaussian mixture regression model based adaptive filter for non-Gaussian noise without a priori statistic [J]. Signal Processing, 2022, 190: 108314.
- [4] LI M Z, JING Z L, ZHU H Y, et al. Multi-sensor measurement fusion based on minimum mixture error entropy with non-Gaussian measurement noise [J]. Digital Signal Processing, 2022, 123: 103377.
- [5] SHAN C H, ZHOU W D. A zigzag maneuver target tracking algorithm with colored glint measurement noise [C]// Proceedings of IEEE International Conference on Mechatronics and Automation, 2021.
- [6] HUANG Y L, ZHANG Y G, LI N, et al. A novel robust student's t -based Kalman filter [J]. IEEE Transactions on Aerospace and Electronic Systems, 2017, 53(3): 1545 - 1554.
- [7] LIU Z X, HUANG B J, ZOU Y N, et al. Multi-object Bayesian filter for jump Markov system under glint noise [J]. Signal Processing, 2019, 157: 131 - 140.
- [8] HUANG Y L, ZHANG Y G, ZHAO Y X, et al. A novel robust Gaussian-student's t mixture distribution based Kalman filter [J]. IEEE Transactions on Signal Processing, 2019, 67(13): 3606 - 3620.
- [9] JIA G L, HUANG Y L, BAI M B, et al. A novel robust Kalman filter with non-stationary heavy-tailed measurement noise [J]. IFAC-PapersOnLine, 2020, 53(2): 368 - 373.
- [10] WU W R. Maximum likelihood identification of glint noise [J]. IEEE Transactions on Aerospace and Electronic Systems, 1996, 32(1): 41 - 51.
- [11] HEWER G A, MARTIN R D, ZEH J. Robust preprocessing for Kalman filtering of glint noise [J]. IEEE Transactions on Aerospace and Electronic Systems, 1987, AES - 23(1): 120 - 128.
- [12] KOTTAKKI K K, BHARTIYA S, BHUSHAN M. State estimation of nonlinear dynamical systems using nonlinear update based unscented gaussian sum filter [J]. Journal of Process Control, 2014, 24(9): 1425 - 1443.
- [13] 许红, 谢文冲, 王永良. 角闪烁噪声下的高斯和容积卡尔曼滤波算法 [J]. 系统工程与电子技术, 2019, 41(2): 229 - 235.
XU H, XIE W C, WANG Y L. Gaussian sum cubature Kalman tracking filter with angle glint noise [J]. Systems Engineering and Electronics, 2019, 41(2): 229 - 235. (in Chinese)
- [14] JULIER S, UHLMANN J, DURRANT-WHITE H F. A new method for the nonlinear transformation of means and covariances in filters and estimators [J]. IEEE Transactions on Automatic Control, 2000, 45(3): 477 - 482.
- [15] ARASARATNAM I, HAYKIN S. Cubature Kalman filters [J]. IEEE Transactions on Automatic Control, 2009, 54(6): 1254 - 1269.
- [16] 胡洪涛, 敬忠良, 李安平, 等. 非高斯条件下基于粒子滤波的目标跟踪 [J]. 上海交通大学学报, 2004, 38(12): 1996 - 1999.
HU H T, JING Z L, LI A P, et al. Particle filter based target tracking in non-Gaussian environment [J]. Journal of Shanghai Jiao Tong University, 2004, 38(12): 1996 - 1999. (in Chinese)
- [17] KIM J, TANDALE M, MENON P K, et al. Particle filter for ballistic target tracking with glint noise [J]. Journal of Guidance, Control, and Dynamics, 2010, 33(6): 1918 - 1921.
- [18] DU H Y, WANG W J, BAI L. Observation noise modeling based particle filter: an efficient algorithm for target tracking in glint noise environment [J]. Neurocomputing, 2015, 158: 155 - 166.
- [19] 刘望生, 李亚安. 闪烁噪声下目标跟踪的改进粒子滤波算法 [J]. 兵工学报, 2011, 32(1): 91 - 95.
LIU W S, LI Y A. Target tracking based on modified particle filter algorithm in glint noise environment [J]. Acta Armamentarii, 2011, 32(1): 91 - 95. (in Chinese)
- [20] MUKHERJEE A, SENGUPTA A. Likelihood function modeling of particle filter in presence of non-stationary non-Gaussian measurement noise [J]. Signal Processing, 2010, 90(6): 1873 - 1885.
- [21] 崔彦凯, 梁晓庚. 闪烁噪声环境下机动目标跟踪的改进的高斯 - 厄米特粒子滤波 [J]. 计算机工程与科学, 2013, 35(9): 187 - 190.
CUI Y K, LIANG X G. An IGHPF algorithm of maneuvering target tracking in glint environment [J]. Computer Engineering & Science, 2013, 35(9): 187 - 190. (in Chinese)
- [22] LI X R, 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.
- [23] HWANG I, SEAH C E, LEE S J. A study on stability of the interacting multiple model algorithm [J]. IEEE Transactions on Automatic Control, 2017, 62(2): 901 - 906.
- [24] KULIKOV G Y, KULIKOVA M V. A comparative study of Kalman-like filters for state estimation of turning aircraft in presence of glint noise [J]. IFAC-PapersOnLine, 2020, 53(2): 2335 - 2340.
- [25] NIE Y F, ZHANG T. Improved pruning algorithm for Gaussian mixture probability hypothesis density filter [J]. Journal of Systems Engineering and Electronics, 2018, 29(2): 229 - 235.
- [26] MA Y H, WANG Z L, ZHANG Y Z, et al. Maneuvering target tracking based on adaptive turning rate interactive multiple model [C] // Proceedings of IEEE 17th International Conference on Control & Automation, 2022.
- [27] 刘兴堂. 导弹制导控制系统分析、设计与仿真 [M]. 西安: 西北工业大学出版社, 2006: 79 - 83.
- LIU X T. Control system analysis, design and simulation for missile guidance [M]. Xi'an: Northwestern Polytechnical University Press, 2006: 79 - 83. (in Chinese)
- [28] 李春辉, 马健, 杨永建, 等. 低复杂度自适应容积卡尔曼滤波算法 [J]. 北京航空航天大学学报, 2022, 48(4): 716 - 724.
- LI C H, MA J, YANG Y J, et al. Low-complexity adaptive cubature Kalman filter algorithm [J]. Journal of Beijing University of Aeronautics and Astronautics, 2022, 48(4): 716 - 724. (in Chinese)
- [29] KARLSSON R, SCHON T, GUSTAFSSON F. Complexity analysis of the marginalized particle filter [J]. IEEE Transactions on Signal Processing, 2005, 53(11): 4408 - 4411.