

针对交叉目标场景的带标签 GM-PHD 改进算法

陈金广, 赵甜甜, 王明明, 王 伟

(西安工程大学 计算机科学学院, 西安 710048)

摘 要: 在多目标跟踪系统中, 当目标航迹较为接近或交叉时, 使用带标签高斯混合概率假设密度 (GM-PHD) 算法会出现目标漏检的现象。为此, 提出一种改进算法来解决存在交叉目标情况下的多目标跟踪问题。在更新结束后对估计得到的高斯项标签进行管理, 若估计目标数目减少, 则需要判断目标航迹是否较为接近或者交叉。若目标接近或交叉, 则对高斯项进行标签管理和权值重置, 并重新估计目标状态和航迹, 否则将目标减少视为正常的目标消亡现象, 直接进行航迹管理。实验结果表明, 与无标签算法及常规带标签算法相比, 该算法可以更好地解决由目标交叉导致的漏检问题, 并具有更高的稳定性。

关键词: 交叉目标; 目标跟踪; 概率假设密度滤波; 信息融合; 状态估计

中文引用格式: 陈金广, 赵甜甜, 王明明, 等. 针对交叉目标场景的带标签 GM-PHD 改进算法[J]. 计算机工程, 2018, 44(6): 316-320.

英文引用格式: CHEN Jinguang, ZHAO Tiantian, WANG Mingming, et al. Improved labeled GM-PHD algorithm for scenario with crossing target[J]. Computer Engineering, 2018, 44(6): 316-320.

Improved Labeled GM-PHD Algorithm for Scenario with Crossing Target

CHEN Jinguang, ZHAO Tiantian, WANG Mingming, WANG Wei

(School of Computer Science, Xi'an Polytechnic University, Xi'an 710048, China)

【Abstract】 In multi-target tracking system, some targets will be lost in the tracking when the targets are crossing or closed to each other used by Gaussian Mixture-Probability Hypothesis Density (GM-PHD). In order to solve this problem, an improved algorithm for scenario with crossing targets is put forward. After update, the estimated Gauss label is managed. If the estimated number of targets is reduced, it is needed to determine whether the targets are crossing or closed to each other. If these trajectories are closed to each other, the followed step is label management and weight reset of Gaussian items, and target states and tracks are re-estimated. Otherwise, the reduction of target will be regarded as a normal target extinction phenomenon, and track management is directly carried out. Experimental results show that the improved algorithm can solve the missed detection caused by the crossing targets successfully, and has good stability.

【Key words】 crossing target; target tracking; Probability Hypothesis Density (PHD) filtering; information fusion; state estimation

DOI: 10.19678/j.issn.1000-3428.0046745

0 概述

随机有限集 (Random Finite Set, RFS) 可以很好地处理目标数目有限且随机变化的多目标跟踪问题, 因此, 基于 RFS 的多目标跟踪已经多个领域研究的热点^[1]。Bayes 框架下基于随机有限集的多目标跟踪方法的可执行方案主要包括概率假设密度 (Probability Hypothesis Density, PHD)^[2] 滤波算法、势概率假设密度 (Cardinality Probability Hypothesis

Density, CPHD)^[3] 滤波算法以及多伯努利 (Multi-Bernoulli, MB)^[4] 滤波算法 3 种。对于 PHD 滤波算法现在主要有 2 种实现方案: 针对线性系统的高斯混合概率假设密度 (Gaussian Mixture-Probability Hypothesis Density, GM-PHD) 滤波算法^[5], 针对非线性系统的序贯蒙特卡罗-概率假设密度 (Sequential Monte Carlo-Probability Hypothesis Density, SMC-PHD) 滤波算法^[6]。上述算法大都将点目标和扩展目标分别进行跟踪, 但在实际领域存在点目标和扩

基金项目: 国家自然科学基金 (61601358); 陕西省自然科学基金基础研究计划项目 (2016JM6030); 中国纺织工业联合会科技指导性项目 (2016074); 陕西省教育厅科研计划项目 (15JK1291); 西安工程大学研究生创新基金 (CX201631)。

作者简介: 陈金广 (1977—), 男, 教授、博士, 主研方向为信息融合、目标跟踪; 赵甜甜, 硕士研究生; 王明明, 讲师、博士; 王 伟, 讲师、博士后。

收稿日期: 2017-04-11 **修回日期:** 2017-06-17 **E-mail:** xacjg@163.com

展目标同时存在的场景。针对该特殊场景, 文献[7]提出了基于随机集的点目标和扩展目标联合跟踪算法。针对基于 RFS 的多目标跟踪算法无法精确辨别单目标航迹的问题, 有诸多学者提出了解决目标航迹管理的方法。这些方法根据其本质可分为 2 类, 一类是对 PHD 滤波结果进行数据关联, 这种方法与传统的基于数据关联的多目标跟踪算法类似^[8-9]; 另一类则是通过为高斯项或者粒子加注标签来达到辨别单目标、形成航迹的目的^[10-12]。文献[13]提出了带标签 MB 滤波算法, 用于实现对目标轨迹的估计跟踪。

在目标运动过程中, 目标较为接近或者出现航迹交叉现象, 可能会导致目标出现漏估计。为解决该问题, 学者们给出了不少改进算法。文献[14]认为在目标发生交叉时, GM-PHD 滤波器性能降低的原因是违反了一对一假设, 故提出了一种约束权重的 GM-PHD 滤波方法, 通过约束方案来矫正违反一对一假设的目标权重, 提高滤波器的估计性能。文献[15]通过对 GM-PHD 算法进行改进, 实现了目标交叉情况下的航迹准确估计, 其思路是为可能漏检的那个高斯项重新分配权值, 并调整其他高斯项权值。该算法虽然可以解决漏检问题, 但是会导致衍生目标的漏检, 因此, 只适用于没有衍生目标出现的多目标跟踪。文献[16]针对粒子-PHD 提出了 2 种方案来为相邻时刻的目标状态估计进行关联, 并使用速度作为附加标准辨别邻近目标, 但是, 当相邻目标速度相同或者相近时该算法不能达到分辨目标的目的。对于目标航迹交叉的目标跟踪, 由于每个航迹假设概率是由似然概率的对数求和得到的, 因此如果目标被持续估计, 则航迹假设概率就会增大, 导致产生一些错误的估计关联。针对上述算法的不足, 本文提出一种改进算法, 通过对标签的管理以及权值的重新分配来解决交叉目标出现带来的目标漏检问题, 并经由实验证明该算法的有效性。

1 背景知识

在多目标跟踪系统中, 目标的数目、状态以及目标的量测信息都具有随机性, 可以使用随机有限集来表示 k 时刻的目标状态信息和量测信息。状态随机有限集表示为 $X_k = \{x_{k,1}, x_{k,2}, \dots, x_{k,M_k}\}$, 量测随机有限集表示为 $Z_k = \{z_{k,1}, z_{k,2}, \dots, z_{k,N_k}\}$ 。其中, $x_k = [\varepsilon_1, \varepsilon_2, \varepsilon_3, \varepsilon_4]^T$, $\varepsilon_1, \varepsilon_2$ 表示目标的位置分量, $\varepsilon_3, \varepsilon_4$ 表示速度分量信息, N_k, M_k 分别表示量测数目与目标数目。

高斯混合概率假设密度滤波算法是基于线性高斯假设的, 因此, PHD 滤波可以由混合高斯项来近似表示。使用 $N(\cdot; m, P)$ 表示变量服从高斯分布, 其中, m 表示均值, P 表示协方差。假设每一个目标都服从高斯线性模型, 则运动模型和观测模型可表示

如下:

$$f_{k|k-1}(x|\zeta) = N(x; F_{k-1}\zeta, Q_{k-1}) \quad (1)$$

$$g_{k|k-1}(z|x) = N(z; H_k x, R_k) \quad (2)$$

其中, F_{k-1} 是状态转移矩阵, Q_{k-1}, R_k 分别表示过程噪声协方差和观测噪声协方差, H_k 表示量测矩阵。带标签 GM-PHD 滤波算法则是在 GM-PHD 滤波算法的基础上实现的, 主要是通过为每一个高斯项附加一个标签信息来实现对单目标的辨别, 并通过对标签的管理实现对目标航迹的管理。

假设 $k-1$ 时刻的目标状态后验概率假设密度可表示为 J_{k-1} 个高斯项和的形式, 则目标的标签集可表示为:

$$T_{k-1} = \{t^1, t^2, \dots, t^{J_{k-1}}\} \quad (3)$$

而 k 时刻的目标集合中, 不仅包括从 $k-1$ 时刻存活到 k 时刻的存活目标, 还包括 k 时刻的衍生目标和新生目标。因此, 预测后的标签集为存活目标高斯项标签集 $T_{S,k|k-1}$ 、新生目标高斯项标签集 $T_{\gamma,k|k-1}$ 以及衍生目标高斯项标签集 $T_{\beta,k|k-1}$ 的并集:

$$T_{k|k-1} = T_{S,k|k-1} \cup T_{\gamma,k|k-1} \cup T_{\beta,k|k-1} \quad (4)$$

其中:

$$T_{\gamma,k|k-1} = \{t_{\gamma,k}^1, t_{\gamma,k}^2, \dots, t_{\gamma,k}^{J_{\gamma,k}}\} \quad (5)$$

$$T_{\beta,k|k-1} = \{t_{\beta,k}^1, t_{\beta,k}^2, \dots, t_{\beta,k}^{J_{\beta,k}}\} \quad (6)$$

$$T_{S,k|k-1} = T_{k-1} \quad (7)$$

在更新步骤中, 高斯项的标签不发生变化, 标签集中与预测标签集中元素相同, 但是由于一个目标状态与 N_k 个量测更新会产生 N_k 个高斯项, 因此在更新后的标签集中可能会有 $1 + N_k$ 个相同的标签。更新后的标签集合为:

$$T_{k|k} = T_{k|k-1} \quad (8)$$

对更新后的高斯项进行裁剪合并后, 对目标状态进行估计, 提取出真实目标状态标签集如下:

$$\hat{T}_k = \{t^i; w_k^i > w_{th}\} \quad (9)$$

在带标签 GM-PHD 滤波算法中根据高斯分量权值和的大小实时估计目标数目, 并且通过设定的提取阈值 w_{th} 来提取目标状态。更新结束后, 若目标的高斯项权值小于提取阈值, 则该目标会被遗漏。在多目标跟踪过程中, 如果存在一些特殊的场景, 比如目标航迹交叉或者目标运动比较接近, 就会出现多个目标量测较为密集的情况。此时如果某个目标距其真实量测较远, 而距其他目标的量测较近, 更新结束后可能会出现该目标高斯项的权值较小的情况, 导致漏检现象发生。在这种情况下, 会产生很多短小的航迹, 导致这些航迹被误认为杂波而被舍去, 从而出现航迹不连续的问题。

2 带标签 GM-PHD 改进算法

概率假设密度滤波算法的本质是根据前一时刻的目标状态 X_{k-1} 与当前得到的量测信息 Z_k 来估计目标状态 X_k 。在更新过程中会产生 $J_{k|k-1} \times N_k$ 个高

斯项,估计目标状态时一些高斯项权值会低于目标提取阈值,导致目标没有被估计到。这种目标突然消失的原因可以分为 2 种,一种是目标真实消亡,另一种则是滤波器没有跟踪到目标,也就是漏检。假设有 2 个目标 A 、 B ,在 $k-1$ 时刻它们均存在且相距较近,此外假设观测环境中杂波密度非常小。若 A 的量测 Z_A 距离 B 较近,同时 B 的量测 Z_B 距 B 也较近,更新结束后 B 的权重会比较大,相反 A 的权重则会很小。且 A 的权值可能小于状态提取阈值导致目标 A 被漏检,故当目标较为接近时,出现的漏检现象可考虑是目标较为接近所导致。

为了解决在航迹交叉或者目标运动较为接近的情况下出现的目标漏检现象,本文提出了带标签 GM-PHD 改进算法。首先对 k 时刻估计出的目标标签与 $k-1$ 时刻的标签进行比较,查看是否发生了目标消亡。若有目标在上一时刻存在而该时刻消失,则可通过计算上一时刻该标签对应高斯项的状态与其他标签对应高斯项状态之间的欧式距离,来判断是否出现目标航迹较为接近的情况。若目标较为接近,则认为是因其而导致的漏检现象,需要额外处理,否则认为是正常的航迹终止。

改进算法主要分为以下 5 个步骤:1)根据上一时刻目标状态和标签预测得到该时刻的目标状态和高斯项标签;2)结合该时刻得到的量测集合,对该时刻的预测状态进行更新;3)对高斯项进行裁剪合并,估计该时刻的目标状态和对应标签;4)对估计出的标签进行管理,判断是否发生由目标较为接近导致的漏检现象,如发生则对高斯项更新状态的权值重置,并返回到第 3)步;5)根据估计出的标签进行航迹更新和管理。下面对算法步骤进行详细介绍。

步骤 1 预测。假设 $k-1$ 时刻的后验 PHD 可表示为 J_{k-1} 个高斯项和的形式:

$$v_{k-1}(\mathbf{x}) = \sum_{i=1}^{J_{k-1}} w_{k-1}^i N(\mathbf{x}; m_{k-1}^i, \mathbf{P}_{k-1}^i) \quad (10)$$

则 k 时刻的预测强度函数可表示为:

$$v_{k|k-1}(\mathbf{x}) = \sum_{i=1}^{J_{k|k-1}} w_{k|k-1}^i N(\mathbf{x}; m_{k|k-1}^i, \mathbf{P}_{k|k-1}^i) = v_{S,k|k-1}(\mathbf{x}) + v_{\beta,k|k-1}(\mathbf{x}) + \gamma_k(\mathbf{x}) \quad (11)$$

其中:

$$v_{S,k|k-1}(\mathbf{x}) = p_{S,k} \sum_{j=1}^{J_{k-1}} w_{k-1}^j N(\mathbf{x}; m_{S,k|k-1}^j, \mathbf{P}_{S,k|k-1}^j) \quad (12)$$

$$m_{S,k|k-1}^j = \mathbf{F}_{S,k-1} m_{S,k-1}^j \quad (13)$$

$$\mathbf{P}_{S,k|k-1}^j = \mathbf{Q}_{S,k-1} + \mathbf{F}_{S,k-1} \mathbf{P}_{S,k-1}^j \mathbf{F}_{S,k-1}^T \quad (14)$$

$$v_{\beta,k|k-1}(\mathbf{x}) = \sum_{i=1}^{J_{k-1}} \sum_{j=1}^{J_{\beta,k}} w_{\beta,k-1}^i w_{\beta,k}^j N(\mathbf{x}; m_{\beta,k|k-1}^{ij}, \mathbf{P}_{\beta,k|k-1}^{ij}) \quad (15)$$

$$m_{\beta,k|k-1}^{ij} = \mathbf{F}_{\beta,k-1}^j m_{\beta,k-1}^i \quad (16)$$

$$\mathbf{P}_{\beta,k|k-1}^{ij} = \mathbf{Q}_{\beta,k-1}^j + \mathbf{F}_{\beta,k-1}^j \mathbf{P}_{\beta,k-1}^i (\mathbf{F}_{\beta,k-1}^j)^T \quad (17)$$

$$\gamma_k(\mathbf{x}) = \sum_{i=1}^{J_{\gamma,k}} w_{\gamma,k}^i N(\mathbf{x}; m_{\gamma,k}^i, \mathbf{P}_{\gamma,k}^i) \quad (18)$$

$$J_{k|k-1} = J_{k-1} + J_{\beta,k} + J_{\gamma,k} \quad (19)$$

其中, $v_{S,k|k-1}$ 、 $v_{\beta,k|k-1}$ 、 γ_k 分别表示存活目标、衍生目标和新生目标概率假设密度函数, w^i 表示第 i 个高斯项的权值, $\mathbf{P}_{S,k}$ 是 $k-1$ 时刻的高斯项在 k 时刻仍然存在的概率。

步骤 2 更新。带标签 GM-PHD 滤波算法的更新方程由 2 个部分组成,分别为没有检测到量测的目标概率假设密度函数和有检测到量测的目标概率假设密度函数:

$$v_k(\mathbf{x}) = (1 - p_{D,k}) v_{k|k-1}(\mathbf{x}) + \sum_{z \in Z_k} v_{D,k}(\mathbf{x}; z) \quad (20)$$

其中:

$$v_{D,k}(\mathbf{x}; z) = \sum_{i=1}^{J_{k|k-1}} w_{k|k-1}^i N(\mathbf{x}; m_{k|k-1}^i, \mathbf{P}_{k|k-1}^i) \quad (21)$$

$$w_{k|k-1}^i = \frac{p_{D,k} w_{k|k-1}^i q_k^i(z)}{\kappa_k(z) + p_{D,k} \sum_{l=1}^{J_{k|k-1}} w_{k|k-1}^l q_k^l(z)} \quad (22)$$

$$m_{k|k-1}^i = m_{k|k-1}^i + \mathbf{K}_k^i (z - \mathbf{H}_k m_{k|k-1}^i) \quad (23)$$

$$\mathbf{K}_k^i = \mathbf{P}_{k|k-1}^i \mathbf{H}_k^T (\mathbf{H}_k \mathbf{P}_{k|k-1}^i \mathbf{H}_k^T + \mathbf{R}_k) \quad (24)$$

$$\mathbf{P}_{k|k-1}^i = [\mathbf{I} - \mathbf{K}_k^i \mathbf{H}_k] \mathbf{P}_{k|k-1}^i \quad (25)$$

$$q_k^i(z) = N(z; \mathbf{H}_k m_{k|k-1}^i, \mathbf{R}_k + \mathbf{H}_k \mathbf{P}_{k|k-1}^i \mathbf{H}_k^T) \quad (26)$$

步骤 3 裁剪合并方法与 GM-PHD 滤波算法类似,估计得到的目标状态随机集表示为:

$$\hat{X}_k = \{m_k^i; w_k^i > w_{th}\} \quad (27)$$

步骤 4 漏检处理。在估计出目标状态后,通过式(28)判断该时刻是否发生了漏检。

$$I_l = g(\text{unique}(\hat{X}_{k-1}), \text{unique}(\hat{X}_k)) \quad (28)$$

其中, $\text{unique}(\cdot)$ 函数用来去除集合中的重复元素, $g(A, B)$ 函数表示找出存在于集合 A 而不存在于集合 B 的元素。如果 I_l 为空,则表示没有发生漏检,可直接进行航迹管理;否则,可能发生了目标漏检。如果发生了目标漏检,则需要判断漏检目标与其他目标的位置是否较为接近。通过使用 $k-1$ 时刻算法估计得到的目标状态来判断目标是否较为接近。如果位置不是很接近,则使用文献[17]中的航迹管理解决该问题。

若存在目标漏检且出现了目标较为接近的情况,则根据与漏检目标较为接近的目标所属标签是否存在多个高斯项大于 w_{th} 来判断是否出现该情况。如果与漏检目标 t^l 较为接近的标签 t^i 在该时刻存在多个状态,则在这些高斯项中挑选距预测均值最为接近的一个作为该时刻标签 t^i 的航迹值,然后将距漏检目标 t^l 的预测值最为接近的高斯项标签 t^i 改为漏检目标的标签。如果不存在重复标签,则使用如下方法对更新后的高斯项权值进行重新赋值。

$$(t^{i*}, j^*) = \underset{t^i \in I_l, \forall j=1:N_k}{\operatorname{argmax}} (w_k^j(t^i)) \quad (29)$$

$$w_{\max} = w_k^{j^*}(t^{i*}) \quad (30)$$

$$w_k^{i*}(t^{i*}) = \hat{\alpha}_{k-1}(t^{i*}) \mathbf{P}_{S,k} \frac{1 + e^{-b/a}}{1 + e^{-(T-b)/a}} \quad (31)$$

$$w_k^{j*}(t^i) = \frac{w_k^{j*}(t^{i*})(1 - w_k^{j*}(t^{i*}))}{1 - w_{\max}} \quad (32)$$

其中, N_k 表示 k 时刻的量测个数, $\hat{w}_{k-1}(t^{i*})$ 表示标签为 t^{i*} 的高斯项在 $k-1$ 时刻的权值, α 是缩放因子, $T=1$ 是采样间隔, a, b 是衰减函数 $\frac{1+e^{-b/a}}{1+e^{-(T-b)/a}}$ 的参数。通过式(29)可找到漏检目标最大权值高斯项所对应的量测,并使用式(31)中的衰减函数与前一时刻该高斯项的权值为该高斯项进行权值重置。虽然与前一时刻相比该高斯项权值有所减少,但相对于被漏检的高斯项权值,该权值更接近上一时刻的目标权值。而上一时刻目标是被估计出的,证明该目标权值较大,因此权值重置后的高斯项被检测到的可能性更大。通过上述公式可以解决由无效量测导致交叉目标中目标漏检的问题。

步骤5 使用标签管理机制对估计出的目标状态进行航迹管理。

3 实验设置与结果分析

目标的运动模型和观测模型可表示为:

$$\mathbf{x}_k = \mathbf{F}_k \mathbf{x}_{k-1} + w \tilde{\mathbf{w}}_{k-1} \quad (33)$$

$$\mathbf{z}_k = \mathbf{H}_k \mathbf{x}_k + \tilde{\mathbf{v}}_k \quad (34)$$

其中, $\mathbf{F}_k = \begin{bmatrix} \mathbf{I}_2 & T\mathbf{I}_2 \\ \mathbf{0}_2 & \mathbf{I}_2 \end{bmatrix}$, $\mathbf{H}_k = [\mathbf{I}_2 \quad \mathbf{0}_2]$, 过程噪声 $\tilde{\mathbf{w}}_k$ 和量

测噪声 $\tilde{\mathbf{v}}_k$ 服从相互独立的零均值高斯分布, 标准差分

别为 $\sigma_w = 5$, $\sigma_v = 10$, $\mathbf{Q}_{k-1} = \begin{bmatrix} 0.25T^4\mathbf{I}_2 & 0.5T^3\mathbf{I}_2 \\ 0.5T^3\mathbf{I}_2 & T^2\mathbf{I}_2 \end{bmatrix} \sigma_w^2$ 和

$\mathbf{R}_k = \mathbf{I}_2 \sigma_v^2$ 为协方差阵, 时间间隔为 $T=1$, $\mathbf{I}_2$ 和 $\mathbf{0}_2$ 分别表示二维的单位阵和零矩阵。目标的存活概率为 $p_{S,k} = 0.99$, 检测概率 $p_{D,k} = 0.98$ 。观测场景中的杂波强度函数为 $\kappa_k(\mathbf{z}) = \lambda Vu(\mathbf{z})$, $\lambda = 12.5 \times 10^{-6}/\text{m}^2$ 为杂波强度, 表示单位面积杂波点的平均数目, 检测区域面积为 $V = 4 \times 10^6 \text{m}^2$, $u(\cdot)$ 表示杂波在观测区域内是服从均匀分布的。剪枝阈值 $T_{\text{th}} = 10^{-5}$, 合并门限 $U=4$, 允许高斯分布个数最多为 $J_{\max} = 200$, 目标状态提取阈值 $w_{\text{th}} = 0.5$ 。在本次实验中使用的衰减函数参数为 $a=0.5$, $b=4$, $\alpha=0.8$ 。

本次实验的检测区域为 $[-1000, 1000] \text{m} \times [-1000, 1000] \text{m}$, 会有3个目标先后出现。目标1和目标2都是在 $k=0$ 时刻出现, 而目标3则是目标1在 $k=66$ 时刻衍生出现的。目标1与目标2的初始状态分别为: $\mathbf{m}_1^1 = [250, 250, 0, 0]^T$, $\mathbf{m}_2^1 = [-500, 300, 0, 0]^T$, 它们会在 $k=55$ 时刻相交。

目标新生时刻的概率假设密度和衍生目标的概率假设密度为:

$$\gamma_k(\mathbf{x}) = 0.1N(\mathbf{x}; \mathbf{m}_\gamma^1, \mathbf{P}_\gamma) + 0.1N(\mathbf{x}; \mathbf{m}_\gamma^2, \mathbf{P}_\gamma) \quad (35)$$

$$\beta_{k|k-1}(\mathbf{x}|\zeta) = 0.05N(\mathbf{x}; \zeta, \mathbf{P}_\beta) \quad (36)$$

其中, 新生目标的权值为 0.1, 衍生目标的权值为 0.05, 新生目标 PHD 和衍生目标 PHD 的方差分别为 $\mathbf{P}_\gamma = \text{diag}([100, 100, 25, 25]^T)$, $\mathbf{P}_\beta = \text{diag}([100, 100, 400, 400]^T)$ 。本次实验采用最优子模式分配 (Optimal Sub Pattern Allocation, OSPA) 距离对算法进行评价, 参数设置为 $p=1$, $c=200$ 。

图1是传感器接收到的量测信息图, 从图中可以看出在第50~第60时刻之间, 无论是在 X 坐标还是 Y 坐标目标量测都比较靠近。在第55时刻, 目标1和目标2发生了交叉。图2是目标数目估计示意图。从中可以看出, 无论是 GM-PHD 滤波算法还是原有带标签 GM-PHD 滤波算法都出现了漏估计现象, 尤其是 GM-PHD 滤波算法由于不断出现短小航迹而被标签管理机制认为是虚假目标而被舍去, 导致出现航迹缺失现象。而本文算法解决了这一问题, 并体现了算法的稳定性。

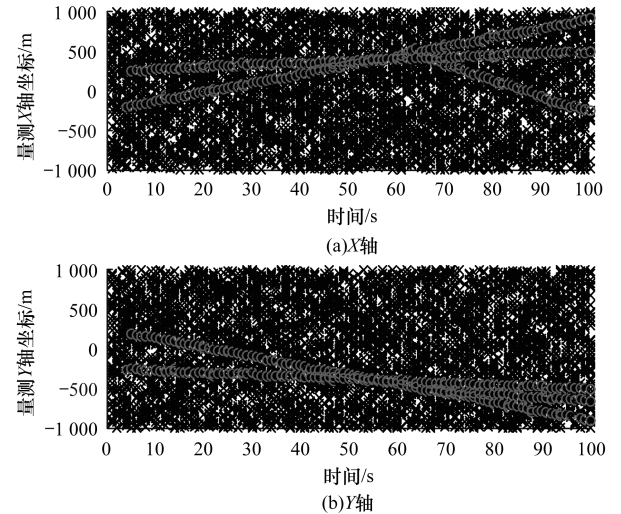


图1 量测信息

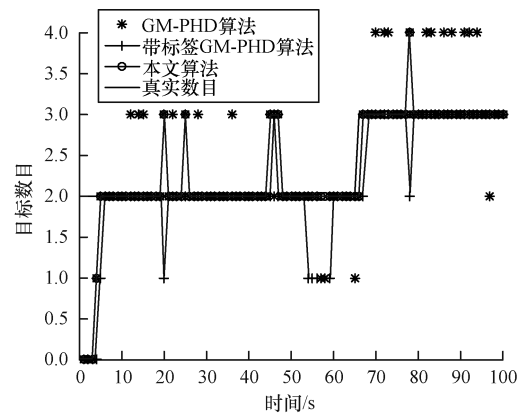


图2 目标数目估计

图3是 GM-PHD 滤波算法、带标签 GM-PHD 滤波算法与本文改进算法的 OSPA 距离比较, OSPA 距

离越小表示该算法对于目标的估计越准确。从图中可以看出,带标签 GM-PHD 滤波算法和改进算法都比 GM-PHD 滤波算法要稳定并且估计更精确,而在第 50~第 60 时刻带标签 GM-PHD 出现较为明显的波动时,改进算法依然很稳定并且 OSPA 值较低。在第 65~第 67 时刻,OSPA 距离都比较大,原因是衍生目标刚出现时权值较低,因此从其产生到确认航迹需要一定的时间。

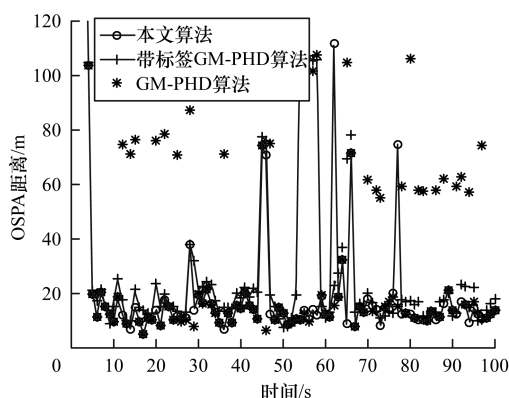


图 3 OSPA 距离

对 GM-PHD 滤波算法和带标签 GM-PHD 滤波算法以及本文改进算法进行 50 次蒙特卡罗实验得到每一次滤波平均 OSPA 距离,如表 1 所示。从表中可以看出,随着杂波密度的增加,各算法的性能均有所下降(OSPA 距离增加),但本文改进算法的估计性能均高于带标签 GM-PHD 滤波算法与 GM-PHD 滤波算法。

表 1 不同杂波密度下的算法 OSPA 距离比较 m

杂波密度	GM-PHD 算法	带标签 GM-PHD 算法	改进算法
1.25×10^{-6}	19.234 0	41.221 9	15.718 3
4.50×10^{-6}	24.370 3	43.347 8	17.928 7
7.50×10^{-6}	27.913 2	52.300 6	18.042 6
1.25×10^{-5}	29.213 9	54.060 0	20.555 1

对改进算法进行 50 次蒙特卡罗实验得到每一次滤波平均花费时间为 1.67 s,而带标签 GM-PHD 滤波算法的平均运行时间为 1.10 s。造成该现象的主要原因是改进算法中若出现漏检现象需要重复进行裁减合并及状态提取步骤,因此时间复杂度相对带标签 GM-PHD 滤波算法要高。

4 结束语

针对目标交叉或目标运动较为接近情况下带标签 GM-PHD 滤波算法对目标估计不精确的问题,本文提出了改进算法。通过标签管理方法和权值重置方案,使得对目标航迹较为接近情况下的漏检目标问题得到了缓解,并且仿真实验证明了算法的有效性。虽然相比改进前的算法,本文算法计算量有所增加,但是可以稳定地估计出目标的数目和状态,并形成航迹,对于多目标跟踪具有现实意义。

参考文献

- [1] BEARD M, REUTER S, VO B N. Multiple extended target tracking with labeled random finite sets [J]. IEEE Transactions on Signal Processing, 2016, 64(7): 1638-1653.
- [2] MAHLER R. Multi-target Bayes filtering via first-order multi-target moments [J]. IEEE Transactions on Aerospace and Electronic Systems, 2003, 39(4): 1152-1178.
- [3] MAHLER R. PHD filters of higher order in target number [J]. IEEE Transactions on Aerospace and Electronic Systems, 2007, 43(4): 1523-1543.
- [4] VO B T, VO B N. The cardinality balanced multi-target Bernoulli filter and its implementations [J]. IEEE Transactions on Signal Processing, 2009, 57(2): 409-423.
- [5] VO B N, MA W. The Gaussian mixture probability hypothesis density filter [J]. IEEE Transactions on Signal Processing, 2006, 54(11): 4091-4104.
- [6] VO B N, SINGH S, DOUCET A. Sequential Monte Carlo methods for multi-target filtering with random finite sets [J]. IEEE Transactions on Aerospace and Electronic Systems, 2005, 41(4): 1224-1245.
- [7] 吴俊辰, 周峰. 点目标和扩展目标联合跟踪算法 [J]. 计算机应用研究, 2017, 34(9): 1-7.
- [8] PANTA K, VO B N, SINGH S. Improved probability hypothesis density filter for multitarget tracking [C]// Proceedings of the 3rd International Conference on Intelligent Sensing and Information Processing. Berlin, Germany: Springer, 2005: 213-218.
- [9] LIU Yi, WANG Ping. Data association combined with the probability hypothesis density filter for multi-target tracking [C]// Proceedings of International Conference on Wireless Communications & Signal Processing. Washington D. C., USA: IEEE Press, 2015: 1-6.
- [10] CLARK D, PANTA K, VO B N. The GM-PHD filter multiple target tracker [C]// Proceedings of the 9th International Conference on Information Fusion. Berlin, Germany: Springer, 2006: 1-8.
- [11] PANTA K, CLARK D, VO B N. Data association and track management for Gaussian mixture probability hypothesis density filter [J]. IEEE Transactions on Aerospace and Electronic Systems, 2009, 45(3): 1003-1016.
- [12] ZHU Hongyan, HAN Chongzhao, LIN Yan. Particle labeling PHD filter for multi-target track-valued estimates [C]// Proceedings of the 14th International Conference on Information Fusion. Washington D. C., USA: IEEE Press, 2011: 1-8.
- [13] REUTER S, VO B T, VO B N, et al. The labeled multi-Bernoulli filter [J]. IEEE Transactions on Signal Processing, 2014, 62(12): 3246-3260.
- [14] 赵一峰. 基于权重约束 GM-PHD 滤波的多目标跟踪方法 [J]. 计算机工程, 2017, 43(3): 282-288.
- [15] ZHANG Huanqing, YONG Jinlong, GE Hongwei, et al. An improved GM-PHD tracker with tracker management for multiple target tracking [C]// Proceedings of International Conference on Control, Automation and Information Science. Washington D. C., USA: IEEE Press, 2015: 185-190.
- [16] CLARK D E, BELL J. Multi-target state estimation and track continuity for the particle PHD filter [J]. IEEE Transactions on Aerospace and Electronic Systems, 2007, 43(4): 1441-1453.
- [17] 陈金广, 赵甜甜, 马丽丽, 等. 可提取衍生目标的带标签 GM-PHD 算法 [J]. 光电工程, 2016, 43(12): 79-84.

编辑 顾逸斐