

基于多变量 Laplace 分布的非线性系统分布式鲁棒状态估计

王国庆^{1,2}, 杨春雨^{1,2}, 马 磊^{1,2}

(1. 中国矿业大学信息与控制工程学院, 江苏徐州 221116; 2. 地下空间智能控制教育部工程研究中心, 江苏徐州 221116)

摘 要: 本文考虑非高斯噪声下传感器网络的状态估计问题. 在机动目标跟踪、室内定位、水声导航等应用中, 传感器的非高斯噪声会造成针对高斯噪声设计的常规状态估计算法精度下降. 在现有针对单传感器系统的基于多变量 Laplace (Multivariate Laplace, ML) 鲁棒状态估计 (Robust State Estimation based on ML, RSE-ML) 算法基础上, 本文借助信息滤波的特点, 推导了针对多传感器系统的集中式 RSE-ML (Centralized RSE-ML, CRSE-ML) 算法, 进一步利用一致性平均得到分布式 RSE-ML (Distributed RSE-ML, DRSE-ML) 算法. 本文提出的 DRSE-ML 算法中利用 ML 建模非高斯噪声, 借助变分贝叶斯方法估计噪声和状态参数, 采用一致性算法进行分布式信息交互, 克服了集中式算法通信和计算负担重的缺点, 且具有自由参数少、估计精度高的特点. 仿真结果表明, 所提出的 DRSE-ML 算法估计精度优于现有相关算法, 且能逼近集中式 CRSE-ML 算法的估计精度.

关键词: 非高斯噪声; 鲁棒状态估计; 分布式状态估计; 多变量 Laplace 分布; 卡尔曼滤波; 变分贝叶斯

中图分类号: TP24

文献标识码: A

文章编号: 0372-2112(2022)11-2610-09

电子学报 URL: <http://www.ejournal.org.cn>

DOI: 10.12263/DZXB.20220141

Distributed Robust State Estimation for Nonlinear Systems Based on Multivariate Laplace Distribution

WANG Guo-qing^{1,2}, YANG Chun-yu^{1,2}, MA Lei^{1,2}

(1. School of Information and Control Engineering, China University of Mining and Technology, Xuzhou, Jiangsu 221116, China;

2. Engineering Research Center of Intelligent Control for Underground Space, Ministry of Education, Xuzhou, Jiangsu 221116, China)

Abstract: This paper considers the state estimation problem of sensor networks with non-Gaussian noise. In many applications, such as maneuvering target tracking, indoor positioning and underwater acoustic navigation, the non-Gaussian noise of the sensors may reduce the accuracy of the estimation algorithm designed for Gaussian noise. The centralized robust state estimation based on multivariate Laplace (CRSE-ML) algorithm is derived according to the characteristics of information filtering on the basis of existing RSE-ML, which is designed for single-sensor systems. Following that, the distributed RSE-ML (DRSE-ML) algorithm is then obtained with the introduction of consensus average. Within the DRSE-ML algorithm, the ML distribution is used to model non-Gaussian noise, the variational Bayesian method is applied to estimate noise and state parameters, and the consensus algorithm is adopted for distributed information exchange. The distributed algorithm overcomes the shortcomings of the centralized one, which has heavy communication and computational burden, and has few free parameters and high estimation accuracy. Simulation results show that the estimation accuracy of the proposed DRSE-ML is better than the existing related algorithms, and can approach the estimation accuracy of the centralized CRSE-ML algorithm.

Key words: non-Gaussian noise; robust state estimation; distributed state estimation; multivariate Laplace distribution; Kalman filter; variational Bayesian

1 引言

以卡尔曼滤波(Kalman Filter, KF)为标志的状态估计算法是信息融合的主要手段之一,在目标跟踪、组合导航、自动控制、协同定位等领域得到广泛应用^[1,2]. 经典 KF 是针对线性系统设计的算法,随后又扩展到非线性系统,其代表性算法包括扩展卡尔曼滤波(Extended Kalman Filter, EKF), 无迹卡尔曼滤波(Unscented Kalman Filter, UKF)和容积卡尔曼滤波(Cubature Kalman Filter, CKF)^[3],其中 CKF 从形式上是 UKF 的一种特例. 以上算法均是在高斯白噪声的假设下得到的,因此当噪声分布不再服从高斯分布时,其估计性能均会下降,甚至会出现算法发散.

在机动目标跟踪中,受到反射、折射等造成的多路径以及杂波干扰,传感器噪声通常表现出非高斯特性^[4]. 此外,在水下协同导航、电网状态估计、室内定位等应用中^[5-7],传感器严重的非高斯噪声也会引起融合估计精度下降. 近年来,针对非高斯噪声系统状态估计研究主要包括两类^[8],即基于度量的策略和基于分布的策略. 由于针对高斯噪声设计的 KF 可以借助二次型代价函数推导得到,因此,对于非高斯噪声系统,可以借助其他类型的代价函数来替代二次型代价函数来得到更好的估计效果^[9]. 基于 M 函数定义代价函数可以得到 Huber 卡尔曼滤波(Huber Kalman Filter, HKF)^[10,11],基于极大熵定义的代价函数可以得到极大熵卡尔曼滤波(Maximum Correntropy Kalman Filter, MCKF)^[9,12-14]. 两类算法对于非高斯噪声的具体分布没有限制,对于多种非高斯噪声均具有较好的抑制作用,但是两者都有一个关键的自由参数需要设定. 另一类基于分布的策略,采用各种分布对常见非高斯噪声进行近似,进而设计相应的估计算法,常用的有 t 分布^[6,15,16]和多变量 Laplace (Multivariate Laplace, ML) 分布^[8]. t 分布的自由度参数对估计结果影响显著,选取不当会恶化估计效果,最近文献^[8]提出采用 ML 分布建模厚尾非高斯噪声,得到基于 ML 分布建模的鲁棒状态估计方法(Robust State Estimation based on Multivariate Laplace, RSE-ML). 与基于 t 分布的相关算法相比,该算法避免 t 分布自由参数选取的问题,也具有较高的估计精度. 但是该算法仅考虑了一个传感器情况下量测噪声服从联合 ML 分布的情景,不易扩展到更为一般的多传感器系统.

随着传感器、计算机、信号处理等相关技术的发展,以传感器网络为代表的多传感器系统能够实现以低成本传感器信息融合来替代高精度传感器,且具有去中心化、鲁棒性强、易于扩展等优点^[17]. 由于 RSE-ML 算法针对单传感器系统设计,因此,对于多传感器系统,假设所有传感器噪声服从联合 ML 不再合理. 本文针对带非高斯噪声非线性系统的多传感器融合估计问

题,首先推导了针对多传感器系统的集中式 RSE-ML (Centralized RSE-ML, CRSE-ML) 算法. 在此基础上,借助一致性方法得到针对多传感器系统的分布式 RSE-ML (Distributed RSE-ML, DRSE-ML) 算法. 基于传感器网络的机器人跟踪的仿真实验表明,本文所提出的分布式鲁棒状态估计算法能够实现对于带非高斯噪声的多传感器系统状态的精确估计,精度优于现有几种相关的算法.

2 系统模型

考虑如下的多传感器系统,即利用 $k=1, \dots, K$ 个传感器对系统状态进行观测,其中系统方程表示为

$$\mathbf{x}_i = f_i(\mathbf{x}_{i-1}) + \mathbf{w}_i \quad (1)$$

式(1)中 \mathbf{x}_i 表示系统真实的状态变量,维数为 m , $f_i(\cdot)$ 表示系统状态转移矩阵, \mathbf{w}_i 表示均值为零方差为 \mathbf{Q}_i 的高斯白噪声,且与量测噪声不相关,下标 i 表示采样时间.

传感器量测方程表示为

$$\mathbf{z}_i^k = h_i^k(\mathbf{x}_i) + \mathbf{v}_i^k \quad (2)$$

式(2)中 \mathbf{z}_i^k 表示第 k 个传感器在 i 时刻的量测数据,且维数为 n_k , $h_i^k(\cdot)$ 表示传感器量测方程, \mathbf{v}_i^k 表示传感器量测噪声. 本文将非高斯的传感器量测噪声建模为零均值 ML 分布进行处理,且假设不同传感器之间的量测噪声不相关,不同时刻的量测噪声不相关,即为白噪声(如果有有色噪声可以通过状态方程扩维的方式进行白化).

3 RSE-ML 算法

首先考虑仅有一个传感器 k 的情形. RSE-ML 算法将厚尾类型的非高斯噪声建模为 ML 分布,随后借助变分贝叶斯学习的方法估计出噪声参数和系统状态^[8]. 为便于理解,本节 RSE-ML 算法中的非线性积分项采用容积变换(一种特殊的无迹变换)方法^[18]进行求解. 算法包括时间更新和量测更新两个步骤.

3.1 时间更新

时间更新与常规非线性估计方法一致,若采用容积变换计算非线性积分,则状态一步预测 $\hat{\mathbf{x}}_{i|i-1}^k$ 及其协方差矩阵 $\mathbf{P}_{i|i-1}^k$ 可以表示为^[18]

$$\hat{\mathbf{x}}_{i|i-1}^k = \sum_{s=1}^{2m} \frac{1}{2m} f_i(\boldsymbol{\chi}_{i-1|j-1}^{k,s}) \quad (3)$$

$$\mathbf{P}_{i|i-1}^k = \sum_{s=1}^{2m} \frac{1}{2m} f_i(\boldsymbol{\chi}_{i|j-1}^{k,s}) \left(f_i(\boldsymbol{\chi}_{i|j-1}^{k,s}) \right)^T - \hat{\mathbf{x}}_{i|j-1}^k \left(\hat{\mathbf{x}}_{i|j-1}^k \right)^T + \mathbf{Q}_i \quad (4)$$

式(3)和式(4)中上标 T 表示矩阵转置,容积点 $\boldsymbol{\chi}_{i-1|j-1}^{k,s}$ 可利用上一时刻的估计值 $\hat{\mathbf{x}}_{i-1|j-1}^k$ 和估计误差协

方差矩阵 $\mathbf{P}_{i-1|i-1}^k$ 根据容积规则计算得到:

$$\chi_{i-1|i-1}^{k,s} = \begin{cases} \hat{\mathbf{x}}_{i-1|i-1}^k + \left(\sqrt{m\mathbf{P}_{i-1|i-1}^k} \right)_s, & s = 1, \dots, m \\ \hat{\mathbf{x}}_{i-1|i-1}^k - \left(\sqrt{m\mathbf{P}_{i-1|i-1}^k} \right)_{s-m}, & s = m+1, \dots, 2m \end{cases} \quad (5)$$

其中, $\left(\sqrt{m\mathbf{P}_{i-1|i-1}^k} \right)_s$ 表示对 $m\mathbf{P}_{i-1|i-1}^k$ 进行 Cholesky 分解后取第 s 列.

3.2 量测更新

与常规非线性状态估计方法不同, RSE-ML 算法采用变分贝叶斯学习的方法来估计噪声参数, 中间需要 L 次固定点迭代. 在第 l 次固定点迭代中, 状态后验估计值 $\hat{\mathbf{x}}_{ii}^{k,(l+1)}$ 及其误差协方差矩阵 $\mathbf{P}_{ii}^{k,(l+1)}$ 分别由式(6)和式(7)计算得到:

$$\hat{\mathbf{x}}_{ii}^{k,(l+1)} = \hat{\mathbf{x}}_{ii-1}^k + \mathbf{K}_i^{k,(l+1)} \left(\mathbf{z}_i^k - \hat{\mathbf{z}}_{ii-1}^k \right) \quad (6)$$

$$\mathbf{P}_{ii}^{k,(l+1)} = \mathbf{P}_{ii-1}^k - \mathbf{K}_i^{k,(l+1)} \mathbf{P}_{zz}^{k,(l+1)} \left(\mathbf{K}_i^{k,(l+1)} \right)^T \quad (7)$$

式(6)中的量测的预测值 $\hat{\mathbf{z}}_{ii-1}^k$:

$$\hat{\mathbf{z}}_{ii-1}^k = \sum_{s=1}^{2m} \frac{1}{2m} h_i^k \left(\chi_{ii-1}^{k,s} \right) \quad (8)$$

其中, $\chi_{ii-1}^{k,s}$ 是利用一步预测及其方差信息得到:

$$\chi_{ii-1}^{k,s} = \begin{cases} \hat{\mathbf{x}}_{ii-1}^k + \left(\sqrt{m\mathbf{P}_{ii-1}^k} \right)_s, & s = 1, \dots, m \\ \hat{\mathbf{x}}_{ii-1}^k - \left(\sqrt{m\mathbf{P}_{ii-1}^k} \right)_{s-m}, & s = m+1, \dots, 2m \end{cases} \quad (9)$$

式(6)和式(7)中量测更新的滤波增益 $\mathbf{K}_i^{k,(l+1)}$ 由式(10)求得:

$$\mathbf{K}_i^{k,(l+1)} = \mathbf{P}_{xz}^{k,(l+1)} \left(\mathbf{P}_{zz}^{k,(l+1)} \right)^{-1} \quad (10)$$

可以看到 $\mathbf{K}_i^{k,(l+1)}$ 包含两部分, 其中 $\mathbf{P}_{xz}^{k,(l+1)}$ 与常规非线性算法一致:

$$\mathbf{P}_{xz}^{k,(l+1)} = \sum_{s=1}^{2m} \frac{1}{2m} \chi_{ii-1}^{k,s} \left(h_i^k \left(\chi_{ii-1}^{k,s} \right) \right)^T - \hat{\mathbf{x}}_{ii-1}^k \left(\hat{\mathbf{z}}_{ii-1}^k \right)^T \quad (11)$$

$\mathbf{P}_{zz}^{k,(l+1)}$ 与常规方法相比增加了 $w_i^{k,(l)}$ 和 $\mathbf{R}_i^{k,(l)}$ 两项^[8]:

$$\mathbf{P}_{zz}^{k,(l+1)} = \sum_{s=1}^{2m} \frac{1}{2m} h_i^k \left(\chi_{ii-1}^{k,s} \right) \left(h_i^k \left(\chi_{ii-1}^{k,s} \right) \right)^T - \hat{\mathbf{z}}_{ii-1}^k \left(\hat{\mathbf{z}}_{ii-1}^k \right)^T + w_i^{k,(l)} \mathbf{R}_i^{k,(l)} \quad (12)$$

根据文献[8], 式(12)中 $w_i^{k,(l)}$ 计算方法如下:

$$w_i^{k,(l)} = \left(E^0 \left((w_i^k)^{-1} \right) \right)^{-1} \quad (13)$$

其中, $E^0 \left((w_i^k)^{-1} \right)$ 由式(14)计算得到:

$$E^0 \left((w_i^k)^{-1} \right) = \left(\frac{\hat{\omega}_i^{k,(l)}}{\hat{\eta}_i^{k,(l)}} \right)^{-\frac{1}{2}} \frac{K_{\delta_i^k, 0-1} \left(2\sqrt{\hat{\eta}_i^{k,(l)} \hat{\omega}_i^{k,(l)}} \right)}{K_{\delta_i^k, 0} \left(2\sqrt{\hat{\eta}_i^{k,(l)} \hat{\omega}_i^{k,(l)}} \right)} \quad (14)$$

式中, $K_\delta(v)$ 表示 δ 阶第二类修正 Bessel 函数, $E(\cdot)$ 表示

期望.

式(14)中的 $\hat{\delta}_{ii}^{k,(l)}$ 、 $\hat{\omega}_i^{k,(l)}$ 和 $\hat{\eta}_i^{k,(l)}$ 分别由下式得到:

$$\hat{\delta}_{ii}^{k,(l)} = 1 - \frac{n_z^k}{2} \quad (15)$$

$$\hat{\omega}_i^{k,(l)} = \frac{1}{2} \text{tr} \left(\mathbf{B}_i^{k,(l)} E^0 \left(\left(\mathbf{R}_i^k \right)^{-1} \right) \right) \quad (16)$$

$$\hat{\eta}_i^{k,(l)} = \lambda_0 \quad (17)$$

其中, $\text{tr}(\cdot)$ 表示矩阵的迹, 对于 ML 分布 $\lambda_0 = 1$.

式(16)中的 $\mathbf{B}_i^{k,(l)}$ 可以由式(18)得到:

$$\mathbf{B}_i^{k,(l)} = \mathbf{z}_i^k \left(\mathbf{z}_i^k \right)^T - \sum_{s=1}^{2m} \frac{1}{2m} h_i^k \left(\chi_{ii}^{k,(l),s} \right) \left(h_i^k \left(\chi_{ii}^{k,(l),s} \right) \right)^T \quad (18)$$

式(18)中 $\chi_{ii}^{k,(l),s}$ 根据容积变化规则由 $\hat{\mathbf{x}}_{ii}^{k,(l)}$ 和 $\mathbf{P}_{ii}^{k,(l)}$ 生成.

式(12)中 $\mathbf{R}_i^{k,(l)}$ 可由式(19)得到:

$$\mathbf{R}_i^{k,(l)} = \left(E^0 \left(\left(\mathbf{R}_i^k \right)^{-1} \right) \right)^{-1} = \frac{\hat{\mathbf{V}}_{ii}^{k,(l)}}{\hat{\mathbf{v}}_{ii}^{k,(l)}} \quad (19)$$

其中,

$$\hat{\mathbf{V}}_{ii}^{k,(l+1)} = E^0 \left(\left(w_i^k \right)^{-1} \right) \mathbf{B}_i^{k,(l)} + \mathbf{V}_0^k \quad (20)$$

$$\hat{\mathbf{v}}_{ii}^{k,(l+1)} = v_0^k + 1 \quad (21)$$

v_0^k 和 \mathbf{V}_0^k 分别表示逆 Wishart 分布的超参数. 根据文献[8], 两者可以设为 $v_0^k = \rho^k$ 和 $\mathbf{V}_0^k = \rho^k \bar{\mathbf{R}}^k$, 其中 $\rho^k > n_k + 1$, $\bar{\mathbf{R}}^k$ 为量测噪声主导方差矩阵.

为便于理解, 算法 1 中给出 RSE-ML 算法在一个采样周期内的伪代码.

算法 1 RSE-ML 算法

输入: $\hat{\mathbf{x}}_{i-1|i-1}^k, \mathbf{P}_{i-1|i-1}^k, \mathbf{z}_i^k, f_i(\cdot), h_i^k(\cdot), \mathbf{Q}_i, L, \mathbf{V}_0^k, v_0^k$

时间更新:

根据式(3)和式(4)计算 $\hat{\mathbf{x}}_{ii-1}^k$ 和 \mathbf{P}_{ii-1}^k ;

量测更新:

初始化: $E^0 \left((w_i^k)^{-1} \right) = 1, E^0 \left(\left(\mathbf{R}_i^k \right)^{-1} \right) = \bar{\mathbf{R}}^{-1}$;

变分贝叶斯迭代:

FOR $l = 0, 1, \dots, L-1$

根据式(19)计算 $E^0 \left(\left(\mathbf{R}_i^k \right)^{-1} \right)$;

根据式(14)计算 $E^0 \left((w_i^k)^{-1} \right)$;

根据式(6)和式(7)计算 $\hat{\mathbf{x}}_{ii}^{k,(l+1)}$ 及 $\mathbf{P}_{ii}^{k,(l+1)}$;

END FOR

输出: $\hat{\mathbf{x}}_{ii}^k = \hat{\mathbf{x}}_{ii}^{k,(L)}, \mathbf{P}_{ii}^k = \mathbf{P}_{ii}^{k,(L)}$

与文献[18]相比, RSE-ML 算法考虑量测噪声建模为 ML 分布的情况, 通过变分贝叶斯迭代求解后验分布. 特别地, 当 $w_i^k = 1$ 且 \mathbf{R}_i^k 已知, 则 RSE-ML 退化为文献[18]中经典的非线性滤波算法. 即文献[18]是 RSE-ML 算法的一个特例, 在 RSE-ML 算法中借助文献[18]中容积变换规则计算非线性积分.

现有 RSE-ML 是针对单传感器系统设计的, 由于各个传感器的噪声通常不相关, 常用的借助扩维方式扩

展到多传感器系统的方法不再适用. 接下来将借助信息滤波的特性, 将现有的 RSE-ML 扩展到多传感器系统, 得到集中式 CRSE-ML 算法, 进而得到针对多传感器系统的分布式 DRSE-ML 算法.

4 多传感器系统的 CRSE-ML 和 DRSE-ML 算法

4.1 CRSE-ML 算法

对于多传感器系统, 采用分布式处理方法能够增强系统的容错性, 降低融合中心通信和计算负担. 因此将集中式的 RSE-ML 扩展到适合多传感器的分布式系统中尤为必要. 现有的分布式估计算法大都是基于信息滤波形式得到的^[19,20], 借助信息滤波中信息矩阵和信息向量的特点容易表示出 CRSE-ML, 进而得到分布式 DRSE-ML 算法.

要得到信息滤波形式, 关键在于表示出量测更新中的信息矩阵和信息向量. 首先借助于统计线性化方法, 将非线性的量测函数线性化. 线性化后的量测矩阵表示为^[21,22]

$$\mathbf{H}_i^{k,(l+1)} = (\mathbf{P}_{zz}^{k,(l+1)})^{-1} (\mathbf{P}_{ii-1}^k)^{-1} \quad (22)$$

式中 $\mathbf{P}_{zz}^{k,(l+1)}$ 可以根据式 (11) 计算.

利用线性化的量测矩阵, 可以得到与常规量测更新式 (6) 和式 (7) 等价的信息形式:

$$(\mathbf{P}_{ii}^{k,(l+1)})^{-1} = (\mathbf{P}_{ii-1}^k)^{-1} + (\mathbf{H}_i^{k,(l+1)})^T (\mathbf{W}_i^{k,(l)} \mathbf{R}_i^{k,(l)})^{-1} (\mathbf{H}_i^{k,(l+1)}) \quad (23)$$

$$\begin{aligned} (\mathbf{P}_{ii}^{k,(l+1)})^{-1} \hat{\mathbf{x}}_{ii}^{k,(l+1)} &= (\mathbf{P}_{ii-1}^k)^{-1} \hat{\mathbf{x}}_{ii-1}^k \\ &+ (\mathbf{H}_i^{k,(l+1)})^T (\mathbf{W}_i^{k,(l)} \mathbf{R}_i^{k,(l)})^{-1} (\mathbf{z}_i^k - \hat{\mathbf{z}}_{ii-1}^k + \mathbf{H}_i^{k,(l+1)} \hat{\mathbf{x}}_{ii-1}^k) \end{aligned} \quad (24)$$

证明 将式 (22) 带入式 (10) 中 $\mathbf{K}_i^{k,(l+1)}$ 可以得到:

$$\mathbf{K}_i^{k,(l+1)} = \mathbf{P}_{ii-1}^k (\mathbf{H}_i^{k,(l+1)})^T (\mathbf{P}_{zz}^{k,(l+1)})^{-1} \quad (25)$$

将式 (25) 带入式 (7) 中, 则

$$\mathbf{P}_{ii}^{k,(l+1)} = \mathbf{P}_{ii-1}^k - \mathbf{P}_{ii-1}^k (\mathbf{H}_i^{k,(l+1)})^T (\mathbf{P}_{zz}^{k,(l+1)})^{-1} \mathbf{H}_i^{k,(l+1)} (\mathbf{P}_{ii-1}^k)^T \quad (26)$$

将式 (26) 带入矩阵求逆公式 $(\mathbf{A} - \mathbf{B}\mathbf{D}^{-1}\mathbf{C})^{-1} = \mathbf{A}^{-1} + \mathbf{A}^{-1}\mathbf{B}(\mathbf{D} - \mathbf{C}\mathbf{A}^{-1}\mathbf{B})^{-1}\mathbf{C}\mathbf{A}^{-1}$ ^[19,21], 可以得到

$$\begin{aligned} (\mathbf{P}_{ii}^{k,(l+1)})^{-1} &= (\mathbf{P}_{ii-1}^k)^{-1} + (\mathbf{H}_i^{k,(l+1)})^T \\ &\times \left(\mathbf{P}_{zz}^{k,(l+1)} - \mathbf{H}_i^{k,(l+1)} \mathbf{P}_{ii-1}^k (\mathbf{H}_i^{k,(l+1)})^T \right)^{-1} \mathbf{H}_i^{k,(l+1)} \\ &= (\mathbf{P}_{ii-1}^k)^{-1} + (\mathbf{H}_i^{k,(l+1)})^T (\mathbf{W}_i^{k,(l)} \mathbf{R}_i^{k,(l)})^{-1} \mathbf{H}_i^{k,(l+1)} \end{aligned} \quad (27)$$

即为式 (23).

利用式 (23) 进一步对估计增益 $\mathbf{K}_i^{k,(l+1)}$ 进行等价

变换:

$$\begin{aligned} \mathbf{K}_i^{k,(l+1)} &= \mathbf{P}_{ii}^{k,(l+1)} (\mathbf{P}_{ii}^{k,(l+1)})^{-1} \mathbf{P}_{ii-1}^k (\mathbf{H}_i^{k,(l+1)})^T (\mathbf{P}_{zz}^{k,(l+1)})^{-1} \\ &= \mathbf{P}_{ii}^{k,(l+1)} \left((\mathbf{P}_{ii-1}^k)^{-1} + (\mathbf{H}_i^{k,(l+1)})^T (\mathbf{W}_i^{k,(l)} \mathbf{R}_i^{k,(l)})^{-1} \right. \\ &\quad \times \left. (\mathbf{H}_i^{k,(l+1)}) \right) \mathbf{P}_{ii-1}^k (\mathbf{H}_i^{k,(l+1)})^T (\mathbf{P}_{zz}^{k,(l+1)})^{-1} \\ &= \mathbf{P}_{ii}^{k,(l+1)} \left((\mathbf{H}_i^{k,(l+1)})^T + (\mathbf{H}_i^{k,(l+1)})^T (\mathbf{W}_i^{k,(l)} \mathbf{R}_i^{k,(l)})^{-1} \right. \\ &\quad \times \left. (\mathbf{H}_i^{k,(l+1)}) \mathbf{P}_{ii-1}^k (\mathbf{H}_i^{k,(l+1)})^T \right) (\mathbf{P}_{zz}^{k,(l+1)})^{-1} \\ &= \mathbf{P}_{ii}^{k,(l+1)} (\mathbf{H}_i^{k,(l+1)})^T (\mathbf{W}_i^{k,(l)} \mathbf{R}_i^{k,(l)})^{-1} \end{aligned} \quad (28)$$

利用式 (27) 和式 (28), 可以得到 $(\mathbf{P}_{ii}^{k,(l+1)})^{-1} \hat{\mathbf{x}}_{ii}^{k,(l+1)}$ 的等价形式:

$$\begin{aligned} &(\mathbf{P}_{ii}^{k,(l+1)})^{-1} \hat{\mathbf{x}}_{ii}^{k,(l+1)} \\ &= (\mathbf{P}_{ii}^{k,(l+1)})^{-1} \left(\hat{\mathbf{x}}_{ii-1}^k + \mathbf{K}_i^{k,(l+1)} (\mathbf{z}_i^k - \hat{\mathbf{z}}_{ii-1}^k) \right) \\ &= (\mathbf{P}_{ii}^{k,(l+1)})^{-1} \hat{\mathbf{x}}_{ii-1}^k + (\mathbf{H}_i^{k,(l+1)})^T (\mathbf{W}_i^{k,(l)} \mathbf{R}_i^{k,(l)})^{-1} (\mathbf{z}_i^k - \hat{\mathbf{z}}_{ii-1}^k) \\ &= \left((\mathbf{P}_{ii-1}^k)^{-1} + (\mathbf{H}_i^{k,(l+1)})^T (\mathbf{W}_i^{k,(l)} \mathbf{R}_i^{k,(l)})^{-1} (\mathbf{H}_i^{k,(l+1)}) \right) \hat{\mathbf{x}}_{ii-1}^k \\ &\quad + (\mathbf{H}_i^{k,(l+1)})^T (\mathbf{W}_i^{k,(l)} \mathbf{R}_i^{k,(l)})^{-1} (\mathbf{z}_i^k - \hat{\mathbf{z}}_{ii-1}^k) \\ &= (\mathbf{P}_{ii-1}^k)^{-1} \hat{\mathbf{x}}_{ii-1}^k + (\mathbf{H}_i^{k,(l+1)})^T (\mathbf{W}_i^{k,(l)} \mathbf{R}_i^{k,(l)})^{-1} \\ &\quad \times (\mathbf{z}_i^k - \hat{\mathbf{z}}_{ii-1}^k + \mathbf{H}_i^{k,(l+1)} \hat{\mathbf{x}}_{ii-1}^k) \end{aligned} \quad (29)$$

即为式 (24). 证毕.

每个传感器利用自身传感器量测数据运行变分贝叶斯迭代, 利用式 (23) 和式 (24) 式迭代更新线性化后的量测矩阵 $\mathbf{H}_i^{k,(l+1)}$ 和相应的 $\mathbf{W}_i^{k,(l)}$ 和 $\mathbf{R}_i^{k,(l)}$.

进一步根据线性系统集中式估计算法中信息矩阵和信息向量与每个传感器单独运行估计算法的信息矩阵和信息向量之间满足代数和关系^[19,21], 可以得到 CRSE-ML 算法中状态估计值和方差矩阵的计算公式:

$$(\mathbf{P}_{ii}^c)^{-1} = (\mathbf{P}_{ii-1}^c)^{-1} + \sum_{k=1}^K (\mathbf{H}_i^{k,(L)})^T (\mathbf{W}_i^{k,(L-1)} \mathbf{R}_i^{k,(L-1)})^{-1} \mathbf{H}_i^{k,(L)} \quad (30)$$

$$\begin{aligned} (\mathbf{P}_{ii}^c)^{-1} \hat{\mathbf{x}}_{ii}^c &= (\mathbf{P}_{ii-1}^c)^{-1} \hat{\mathbf{x}}_{ii-1}^c \\ &+ \sum_{k=1}^K (\mathbf{H}_i^{k,(L)})^T (\mathbf{W}_i^{k,(L-1)} \mathbf{R}_i^{k,(L-1)})^{-1} (\mathbf{z}_i^k - \hat{\mathbf{z}}_{ii-1}^k + \mathbf{H}_i^{k,(L)} \hat{\mathbf{x}}_{ii-1}^k) \end{aligned} \quad (31)$$

其中, 上标 c 表示集中式算法对应的变量, 其余参数的计算与单个传感器时计算方法类似, 其中相应的容积分点选取是利用集中式算法相应的估计值及其协方差矩阵得到的. CRSE-ML 完整的伪代码在算法 2 中给出.

可以看到, 在 CRSE-ML 算法中存在求和项

算法 2 CRSE-ML 算法

输入: $\hat{\mathbf{x}}_{i-1|l-1}^c, \mathbf{P}_{i-1|l-1}^c, \mathbf{z}_i^k, f_i(\cdot), h_i^k(\cdot), \mathbf{Q}_i, L, \mathbf{V}_0^k, \mathbf{v}_0^k$

时间更新:

根据式(3)和式(4)利用 $\hat{\mathbf{x}}_{i-1|l-1}^c$ 和 $\mathbf{P}_{i-1|l-1}^c$ 计算 $\hat{\mathbf{x}}_{ii-1}^c$ 和 \mathbf{P}_{ii-1}^c ;

量测更新:

初始化: $\hat{\mathbf{x}}_{ii-1}^k = \hat{\mathbf{x}}_{ii-1}^c, \mathbf{P}_{ii-1}^k = \mathbf{P}_{ii-1}^c$;

初始化: $\mathbf{E}^{(0)}((w_i^k)^{-1}) = 1, \mathbf{E}^{(0)}((\mathbf{R}_i^k)^{-1}) = \bar{\mathbf{R}}^{-1}$;

变分贝叶斯迭代:

FOR $l = 0, 1, \dots, L-1$

根据式(19)计算 $\mathbf{E}^{(l)}((\mathbf{R}_i^k)^{-1})$;

根据式(14)计算 $\mathbf{E}^{(l)}((w_i^k)^{-1})$;

根据式(22)计算 $\mathbf{H}_i^{k(l+1)}$;

根据式(23)和式(24)计算 $\hat{\mathbf{x}}_{ii}^{k(l+1)}$ 及 $\mathbf{P}_{ii}^{k(l+1)}$;

END FOR

根据式(25)和式(26)计算 $\hat{\mathbf{x}}_{ii}^c$ 和 \mathbf{P}_{ii}^c ;

输出: $\hat{\mathbf{x}}_{ii}^c$ 和 \mathbf{P}_{ii}^c

$\sum_{k=1}^K (\mathbf{H}_i^{k(L)})^T (\mathbf{w}_i^{k(L-1)} \mathbf{R}_i^{k(L-1)})^{-1} (\mathbf{z}_i^k - \hat{\mathbf{z}}_{ii-1}^k + \mathbf{H}_i^{k(L)} \hat{\mathbf{x}}_{ii-1}^k)$ 和

$\sum_{k=1}^K (\mathbf{H}_i^{k(L)})^T (\mathbf{w}_i^{k(L-1)} \mathbf{R}_i^{k(L-1)})^{-1} \mathbf{H}_i^{k(L)}$. 如果能采用分布式

的方法得到求和项,则可以得到每个传感器仅与局部传感器进行信息交互的分布式状态估计算法. 分布式 DRSE-ML 算法将在下一节给出.

4.2 DRSE-ML 算法

在分布式算法中,假设每个传感器仅与相邻的传感器进行通信,通信拓扑为连接的无向图 $G=(V, E_v)$, 其中 V 表示传感器的集合, E_v 表示边的合集. 其中每个传感器的邻域 N_k 包含传感器自身的节点. 为降低算法的复杂度,采用每个传感器先利用自身传感器数据获得局部估值,随后进行一致性迭代的方式进行处理.

每个传感器首先利用上一步的状态估计值 $\hat{\mathbf{x}}_{i-1|l-1}^{k,d}$ 和 $\mathbf{P}_{i-1|l-1}^{k,d}$ 计算状态估计值 $\hat{\mathbf{x}}_{ii-1}^{k,t}$ 及其协方差矩阵 $\mathbf{P}_{ii-1}^{k,t}$:

$$\hat{\mathbf{x}}_{ii-1}^{k,t} = \sum_{s=1}^{2m} \frac{1}{2m} f_i(\bar{\mathbf{x}}_{i-1|l-1}^{k,s}) \quad (32)$$

$$\mathbf{P}_{ii-1}^{k,t} = \sum_{s=1}^{2m} \frac{1}{2m} f_i(\bar{\mathbf{x}}_{ii-1}^{k,s}) \left(f_i(\bar{\mathbf{x}}_{ii-1}^{k,s}) \right)^T - \hat{\mathbf{x}}_{ii-1}^{k,t} \left(\hat{\mathbf{x}}_{ii-1}^{k,t} \right)^T + \mathbf{Q}_i \quad (33)$$

其中, $\bar{\mathbf{x}}_{i-1|l-1}^{k,s}$ 是由 $\hat{\mathbf{x}}_{i-1|l-1}^{k,d}$ 和 $\mathbf{P}_{i-1|l-1}^{k,d}$ 生成,上标 d 和 t 分别表示分布式算法对应的参数和中间变量.

随后每个传感器利用自身的量测数据进行 L 次变分贝叶斯迭代:

$$\left(\mathbf{P}_{ii}^{k,t,(l+1)} \right)^{-1} = \left(\mathbf{P}_{ii-1}^{k,t} \right)^{-1} + \left(\bar{\mathbf{H}}_i^{k,(l+1)} \right)^T \left(\bar{\mathbf{w}}_i^{k,(l)} \bar{\mathbf{R}}_i^{k,(l)} \right)^{-1} \bar{\mathbf{H}}_i^{k,(l+1)} \quad (34)$$

$$\left(\mathbf{P}_{ii}^{k,t,(l+1)} \right)^{-1} \hat{\mathbf{x}}_{ii}^{k,t,(l+1)} = \left(\mathbf{P}_{ii-1}^{k,t} \right)^{-1} \hat{\mathbf{x}}_{ii-1}^{k,t} + \left(\bar{\mathbf{H}}_i^{k,(l+1)} \right)^T \left(\bar{\mathbf{w}}_i^{k,(l)} \bar{\mathbf{R}}_i^{k,(l)} \right)^{-1} \left(\mathbf{z}_i^k - \hat{\mathbf{z}}_{ii-1}^k + \bar{\mathbf{H}}_i^{k,(l+1)} \hat{\mathbf{x}}_{ii-1}^{k,t} \right) \quad (35)$$

其中,

$$\bar{\mathbf{H}}_i^{k,(l+1)} = \left(\bar{\mathbf{P}}_{xz}^{k,(l+1)} \right)^T \left(\mathbf{P}_{ii-1}^{k,t} \right)^{-1} \quad (36)$$

式(36)中的 $\bar{\mathbf{P}}_{xz}^{k,(l+1)}$ 、 $\bar{\mathbf{w}}_i^{k,(l)}$ 、 $\hat{\mathbf{x}}_{ii-1}^{k,t}$ 、 $\bar{\mathbf{R}}_i^{k,(l)}$ 的计算与第3节类似,不再给出详细步骤.

每个传感器利用自身传感器数据进行迭代后,得到量测矩阵和噪声参数的估计结果. 在此基础上,进行一致性平均,利用局部信息交互来逼近利用所有传感器信息的集中式算法.

在一致性平均过程中,首先进行初始化:

$$\mathbf{Y}_i^k(0) = \left(\bar{\mathbf{H}}_i^{k,(L)} \right)^T \left(\bar{\mathbf{w}}_i^{k,(L-1)} \bar{\mathbf{R}}_i^{k,(L-1)} \right)^{-1} \bar{\mathbf{H}}_i^{k,(L)} \quad (37)$$

$$\mathbf{y}_i^k(0) = \left(\bar{\mathbf{H}}_i^{k,(L)} \right)^T \left(\bar{\mathbf{w}}_i^{k,(L-1)} \bar{\mathbf{R}}_i^{k,(L-1)} \right)^{-1} \times \left(\mathbf{z}_i^k - \hat{\mathbf{z}}_{ii-1}^k + \bar{\mathbf{H}}_i^{k,(L)} \hat{\mathbf{x}}_{ii-1}^{k,t} \right) \quad (38)$$

随后每个传感器仅与相邻传感器 $j \in N_k$ 进行 $t = 1, \dots, M$ 次迭代:

$$\mathbf{Y}_i^k(t) = \sum_{j \in N_k} \pi_{k,j} \mathbf{Y}_i^j(t-1) \quad (39)$$

$$\mathbf{y}_i^k(t) = \sum_{j \in N_k} \pi_{k,j} \mathbf{y}_i^j(t-1) \quad (40)$$

其中, $\pi_{k,j} \geq 0$ 是传感器 k 与 j 之间的加权系数,且满足 $\sum_{j \in N_k} \pi_{k,j} = 1, \mathbf{\Pi} = [\pi_{k,j}]_{K \times K}$ 为素矩阵. $\pi_{k,j}$ 可以选择常用的

Metropolis 系数^[21,22]. 可以证明^[19],当 $M \rightarrow \infty$ 时,有

$$\mathbf{Y}_i^k(t) = \frac{1}{K} \sum_{k=1}^K \mathbf{Y}_i^k(0) \quad (41)$$

$$\mathbf{y}_i^k(t) = \frac{1}{K} \sum_{k=1}^K \mathbf{y}_i^k(0) \quad (42)$$

完成一致性迭代之后,最终的估计结果由式(43)和式(44)求得:

$$\left(\mathbf{P}_{ii}^{k,d} \right)^{-1} = \left(\mathbf{P}_{ii-1}^{k,t} \right)^{-1} + K_M \mathbf{Y}_i^k(M) \quad (43)$$

$$\left(\mathbf{P}_{ii}^{k,d} \right)^{-1} \hat{\mathbf{x}}_{ii}^{k,d} = \left(\mathbf{P}_{ii-1}^{k,t} \right)^{-1} \hat{\mathbf{x}}_{ii-1}^{k,t} + K_M \mathbf{y}_i^k(M) \quad (44)$$

当 $M \geq 2$ 时,一致性迭代近似收敛,此时根据式(41)和式(42)的结果,令 $K_M = K$,则可以恢复出式(30)和式(31)中的估计结果. 实际执行中,仅进行几次一致性迭代即可得到较好的估计效果,后面的数值仿真也验证了该结论. 特别地,当 $M=1$ 时,相当于每个传感器仅利用相邻传感器的信息,只进行了局部信息扩散,因此 $K_M=1$. DRSE-ML 完整的流程在算法3中给出.

文献[19]中基于相关熵代价函数抑制非高斯噪声的特点,推导基于极大相关熵的分布式状态估计算法,对于一般的非高斯噪声有较好的抑制能力. 其抑制非高斯噪声能力与核宽度密切相关,但是缺乏自适应更新核宽度的策略,因此其估计精度提升有限. 本节提出的 DRSE-ML 分布式算法采用 ML 分布建模常见的厚尾分布,利用变分迭代更新分布参数,仿真结果可以看到

算法 3 DRSE-ML 算法

输入: $\hat{x}_{i-1|j-1}^{k,d}, \mathbf{P}_{i-1|j-1}^{k,d}, \mathbf{z}_i^k, f_i(\cdot), h_i^k(\cdot), \mathbf{Q}_i, L, M, \mathbf{V}_0^k, \mathbf{v}_0^k$
 时间更新:
 根据式(32)和式(33)计算 $\hat{x}_{ii-1}^{k,t}$ 和 $\mathbf{P}_{ii-1}^{k,t}$;
 量测更新:
 初始化: $\mathbf{E}^{(0)}((\bar{w}_i^k)^{-1}) = 1, \mathbf{E}^{(0)}((\bar{\mathbf{R}}_i^k)^{-1}) = \bar{\mathbf{R}}^{-1}$;
 变分贝叶斯迭代:
 FOR $l = 0, 1, \dots, L-1$
 根据式(36)计算 $\bar{\mathbf{H}}_i^{k,l+1}$;
 根据式(34)和式(35)计算 $\hat{x}_{ii}^{k,t(l+1)}$ 及 $\mathbf{P}_{ii}^{k,t(l+1)}$;
 END FOR
 一致性迭代:
 根据式(37)和式(38)初始化 $\mathbf{Y}_i^k(0)$ 和 $\mathbf{y}_i^k(0)$;
 FOR $t = 1, \dots, M$
 根据式(39)和式(40)计算 $\mathbf{Y}_i^k(t)$ 和 $\mathbf{y}_i^k(t)$;
 END FOR
 根据式(43)和式(44)计算 $\hat{x}_{ii}^{k,d}$ 和 $\mathbf{P}_{ii}^{k,d}$;
 输出: $\hat{x}_{ii}^{k,d}$ 和 $\mathbf{P}_{ii}^{k,d}$

DRSE-ML 具有更好的估计精度。

5 算法仿真

为验证本文提出算法的有效性,以二维平面内基于传感器网络的移动机器人跟踪为例^[20](示意图如图 1 所示),比较本文所提算法与同类算法的估计精度。根据运动变量之间的关系,可以得到系统的状态方程^[8,19,20]:

$$\mathbf{x}_i = \begin{bmatrix} \mathbf{I}_2 & T\mathbf{I}_2 \\ 0_2 & \mathbf{I}_2 \end{bmatrix} \mathbf{x}_{i-1} + \mathbf{w}_i \quad (45)$$

其中, $T=1$ s 表示采样时间, \mathbf{I}_2 表示 2 维单位矩阵,系统的状态量 $\mathbf{x}_i = [x_i \ y_i \ \dot{x}_i \ \dot{y}_i]^T$ 分别为两个方向的位置和速度。过程噪声 \mathbf{w}_i 为零均值的高斯白噪声,其噪声方差表示为

$$\mathbf{Q}_i = q \begin{bmatrix} \frac{T^3}{3} \mathbf{I}_2 & \frac{T^2}{2} \mathbf{I}_2 \\ \frac{T^2}{2} \mathbf{I}_2 & T\mathbf{I}_2 \end{bmatrix} \quad (46)$$

其中,仿真中取 $q=0.5$ 。

传感器量测信息为目标与传感器的距离,可以表示为^[19,20]

$$\mathbf{z}_i^k = \sqrt{(x_i - x_0^k)^2 + (y_i - y_0^k)^2} + \mathbf{v}_i^k, \quad k = 1, \dots, 10 \quad (47)$$

其中, (x_0^k, y_0^k) 表示第 k 个传感器的位置, \mathbf{v}_i^k 表示传感器噪声。仿真中 10 个传感器之间的网络拓扑如图 1 所示,每个传感器的位置参考文献^[19]来设置(单位替换为 cm)。显然依靠单个传感器的测距信息不足以得到机器人运动的位置和速度信息,但是,融合多个传感器的信息则可以实现对于机器人运动信息的精确估计。

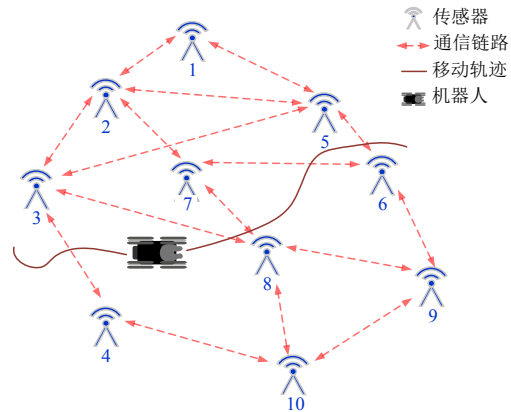


图 1 基于传感器网络的机器人目标跟踪示意图

仿真中对比了以下算法:4.1 节的集中式 CRSE-ML 算法,文献^[14]中的集中式 CMCUKF 算法,文献^[18]中集中式的 UKF 算法(CUKF),4.2 节的分布式 DRSE-ML 算法(分别考虑了一致性迭代次数为 3 和 5 的情形),文献^[19]中的分布式 DMCUKF,文献^[21,23]中的分布式 HCMCI-UKF 算法。仿真进行了 100 次蒙特卡洛仿真,仿真时间取 100 s,算法中的非线性积分项按照容积变换的规则选取采样点,变分贝叶斯迭代次数 L 设为 5,极大熵滤波算法(CMCUKF 和 DMCUKF)中的核宽度设为 10,分布式算法(DMCUKF、HCMCI-UKF 和 DRSE-ML)算法中的加权系数选用 Metropolis 系数。CRMSE-ML 和 DRMSE-ML 算法中取 $\rho^k = 4$ 和 $\bar{\mathbf{R}}^k = \bar{\mathbf{R}}$ 。

仿真中,所有估计算法的初始值均设为 $\bar{\mathbf{x}}_0 = [-4000 \text{ cm}, 1000 \text{ cm}, 10 \text{ cm/s}, 10 \text{ cm/s}]^T$,与之相对应,为保证比较公平,每种算法初始误差方差矩阵均设为 $\mathbf{P}_0 = \text{diag}(4000 \text{ cm}^2, 4000 \text{ cm}^2, 1 \text{ cm}^2/\text{s}^2, 1 \text{ cm}^2/\text{s}^2)^T$ 。

为便于比较算法估计精度,选用速度和位置的平均均方误差(Average Root Mean Square Error, ARMSE)^[21]来比较每种算法的估计误差。仿真使用 MATLAB R2020B 软件,使用的台式机处理器为英特尔酷睿 i9-10900,主频为 2.80 GHz,内存为 16.0 GB。

仿真考虑两种非高斯噪声,用来模拟受到遮挡、多路径、干扰等引起的传感器测距出现大量尖峰脉冲干扰的情况。

情况 1: 量测噪声服从高斯混合分布:

$$\mathbf{v}_i^k \sim \begin{cases} N(\mathbf{0}, \bar{\mathbf{R}}), & \text{概率 } 0.9 \\ N(\mathbf{0}, 100\bar{\mathbf{R}}), & \text{概率 } 0.1 \end{cases} \quad (48)$$

其中 $N(\mathbf{0}, \bar{\mathbf{R}})$ 表示均值为零方差为 $\bar{\mathbf{R}}$ 的高斯分布。即在任意时刻量测噪声以 0.9 的概率按照 $N(\mathbf{0}, \bar{\mathbf{R}})$ 生成,以 0.1 的概率按照 $N(\mathbf{0}, 100\bar{\mathbf{R}})$ 生成。此时 \mathbf{v}_i^k 不再服从高斯分布,而是一种典型的厚尾非高斯分布,其概率密度函数如图 2 所示。可以看到此时的概率函数比高斯分布

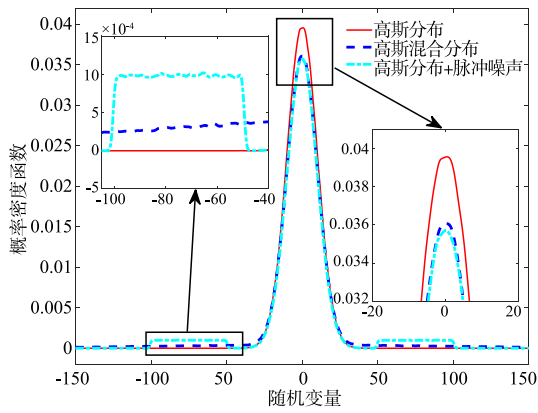


图2 一维变量服从高斯分布、高斯混合分布和高斯分布+脉冲噪声时的概率密度函数

的尾部更厚,能够建模这种存在少量异常值的噪声.

情况2: 量测噪声为高斯分布+脉冲噪声:

$$v_i^k \sim \begin{cases} N(0, \bar{R}), & \text{概率} 0.9 \\ \pm U(50, 100), & \text{概率} 0.1 \end{cases} \quad (49)$$

其中 $U(50, 100)$ 表示区间为 $[50, 100]$ 的整数离散均匀分布. 即在任意时刻量测噪声以 0.9 的概率按照 $N(0, \bar{R})$ 生成, 以 0.1 的概率取在区间 $[-100, -50]$ 或 $[50, 100]$ 的整数均匀分布的值. 其概率密度函数在图2中给出, 可以看到此时的概率密度函数存在更厚的尾部.

各种算法在两种情况下的仿真结果在图3~6中给出, 为便于比较不同算法的计算复杂度, 相应总的计算时间在表1中给出.

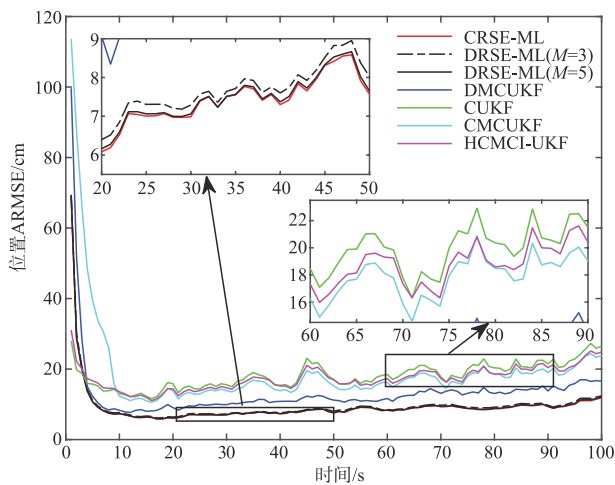


图3 情况1下不同算法的位置 ARMSEs

可以看到, 当传感器噪声不再服从高斯分布时, 在两种噪声条件下针对高斯噪声系统设计的集中式估计算法 CUKF 和分布式估计算法 HCMCI-UKF 估计效果较差. 尽管 CMCUKF 考虑了非高斯噪声的影响, 由于原始算法只考虑单一传感器的情景, 直接扩展到多传感器系统由于不同传感器的噪声不相关, 因此其估计精

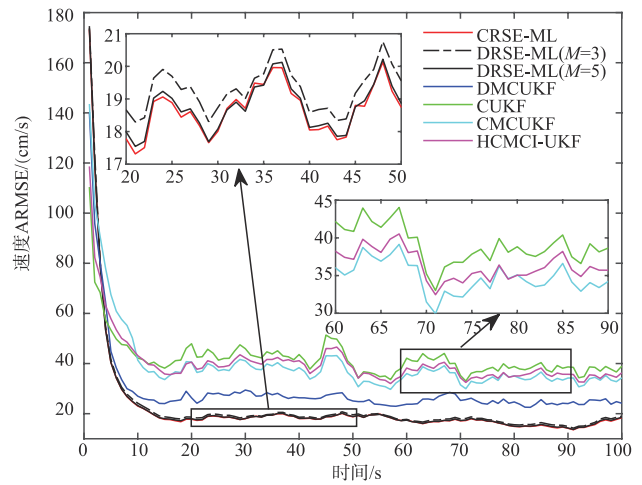


图4 情况1下不同算法的速度 ARMSEs

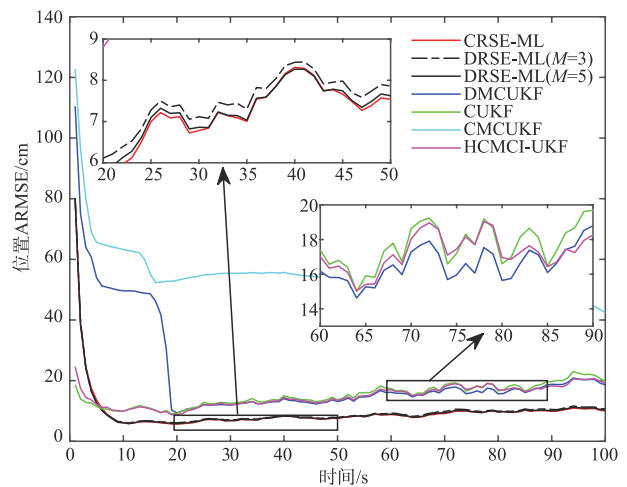


图5 情况2下不同算法的位置 ARMSEs

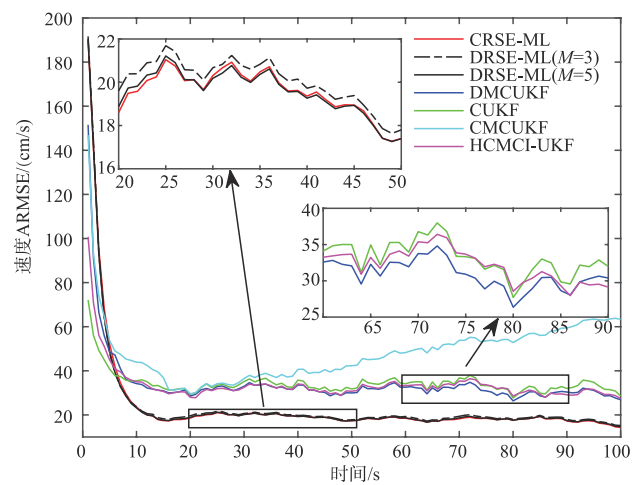


图6 情况2下不同算法的速度 ARMSEs

度较差. 对应的分布式 DMCUKF 算法针对多传感器系统设计, 算法的估计精度高于 CMCUKF 和常规估计算法. 本文提出的 CRSE-ML 算法考虑了每个传感器之间

表 1 不同情况下各算法运行时间 单位:s

算法名称	情况 1	情况 2
CRSE-ML	25.4230	24.3340
DRSE-ML($M=3$)	24.9070	24.4400
DRSE-ML($M=5$)	25.2910	26.3010
DMCUKF	6.4230	5.3240
CUKF	1.7460	1.7030
CMCUKF	2.0310	2.0150
HCMCI-UKF	5.0360	5.4110

的差异,且融合了所有传感器的信息,在两种情况下使用相同的参数均能够得到最好的估计效果. 本文提出的分布式估计方法 DRSE-ML 进行 5 次一致性迭代时可以逼近集中式 CRSE-ML 算法估计精度,进行 3 次一致性迭代时估计精度依然较高,优于现有 DMCUKF 算法.

从表 1 中的计算时间上来看,本文所提出的 CRSE-ML 和 DRSE-ML 算法由于进行了 5 次变分贝叶斯迭代,计算时间大约是同类 DMCUKF 和 HCMCI-UKF 算法计算时间的 5 倍,处于同一量级,但是其估计精度和同一参数对于不同类型噪声的鲁棒性均有明显提升. 本文提出的集中式 CRSE-ML 和 DRSE-ML 总计算时间相当,但是分布式算法所需要的计算时间是 10 个传感器节点总的计算耗时,即 DRSE-ML 算法可以实现每个传感器仅需要处理较低的计算任务和局部通信却获得逼近集中式算法的估计精度,体现了分布式算法的优势.

6 结论

针对非高斯噪声下的多传感器系统状态问题,在现有 RSE-ML 算法基础上,基于信息滤波的特点,推导了集中式 CRSE-ML 算法,进而得到相应分布式的 DRSE-ML 算法. 所提出的 DRSE-ML 算法避免了现有方法对所有传感器噪声服从联合 ML 分布的假设,保留了 RSE-ML 自由参数少的特点,实现了每个传感器仅与相邻传感器通信的去中心化结构. 仿真结果表明,所提出的 DRSE-ML 算法能够逼近集中式的 CRSE-ML 算法,本文提出两种算法的估计精度优于几种现有的鲁棒状态估计方法,且计算耗时略高于同类分布式算法. 未来该算法可以拓展到带随机延时^[6]等网络诱导下系统的以及考虑模型切换^[9]等复杂系统.

参考文献

- [1] 潘泉, 胡玉梅, 兰华, 等. 信息融合理论研究进展: 基于变分贝叶斯的联合优化[J]. 自动化学报, 2019, 45(7): 1207-1223.
PAN Quan, HU Yumei, LAN Hua, et al. Information fusion progress: Joint optimization based on variational

- Bayesian theory[J]. Acta Automatica Sinica, 2019, 45(7): 1207-1223. (in Chinese)
- [2] 刘剑锋, 孙力帆, 普杰信, 等. 基于刚性约束的双移动机器人协同定位[J]. 电子学报, 2020, 48(9): 1777-1785.
LIU Jian-feng, SUN Li-fan, PU Jie-xin, et al. Cooperative localization in a team of two mobile robots based on rigid constraints[J]. Acta Electronica Sinica, 2020, 48(9): 1777-1785. (in Chinese)
- [3] JIA B, XIN M, CHENG Y. Relations between sparse-grid quadrature rule and spherical-radial cubature rule in nonlinear Gaussian estimation[J]. IEEE Transactions on Automatic Control, 2015, 60(1): 199-204.
- [4] KULIKOV G Y, KULIKOVA M V. Estimation of maneuvering target in the presence of non-Gaussian noise: A coordinated turn case study[J]. Signal Processing, 2018, 145: 241-257.
- [5] DANG L, CHEN B, WANG S, et al. Robust power system state estimation with minimum error entropy unscented Kalman filter[J]. IEEE Transactions on Instrumentation and Measurement, 2020, 69(11): 8797-8808.
- [6] 卢春光, 张永顺, 李志汇, 等. 量测随机延迟下带厚尾噪声的鲁棒 Student's t 随机容积卡尔曼滤波器[J]. 电子学报, 2019, 47(9): 1830-1840.
LU Chun-guang, ZHANG Yong-shun, LI Zhi-hui, et al. Robust student's t-based stochastic cubature Kalman filter with heavy-tailed noises and randomly delayed measurements[J]. Acta Electronica Sinica, 2019, 47(9): 1830-1840. (in Chinese)
- [7] BAI M, HUANG Y, ZHANG Y, et al. A novel heavy-tailed mixture distribution based robust Kalman filter for cooperative localization[J]. IEEE Transactions on Industrial Informatics, 2021, 17(5): 3671-3681.
- [8] WANG G, YANG C, MA X. A novel robust nonlinear Kalman filter based on multivariate Laplace distribution[J]. IEEE Transactions on Circuits and Systems II: Express Briefs, 2021, 68(7): 2705-2709.
- [9] 沈忱, MIHAYLOVA Lyudmila. 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.
SHEN Chen, MIHAYLOVA Lyudmila. Maximum mixture correntropy state estimation for jump Markov systems [J]. Acta Electronica Sinica, 2021, 49(11): 2225-2233. (in Chinese)
- [10] CHANG L, LI K. Unified form for the robust Gaussian information filtering based on m-estimate[J]. IEEE Signal Processing Letters, 2017, 24(4): 412-416.

- [11] WU H, CHEN S, YANG B, et al. Robust derivative-free cubature Kalman filter for bearings-only tracking[J]. Journal of Guidance, Control, and Dynamics, 2016, 39(8): 1866-1871.
- [12] LIU X, REN Z, LYU H, et al. Linear and nonlinear regression-based maximum correntropy extended Kalman filtering[J]. IEEE Transactions on Systems, Man, Cybernetics: Systems, 2021, 51(5): 3093-3102.
- [13] WANG G, ZHANG Y, WANG X. Maximum correntropy Rauch-Tung-Striebel smoother for nonlinear and non-Gaussian Systems[J]. IEEE Transactions on Automatic Control, 2021, 66(3): 1270-1277.
- [14] WANG G, LI N, ZHANG Y. Maximum correntropy unscented Kalman and information filters for non-Gaussian measurement noise[J]. Journal of the Franklin Institute, 2017, 354(18): 8659-8677.
- [15] HUANG Y, ZHANG Y, LI N, et al. Robust student's t based nonlinear filter and smoother[J]. IEEE Transactions on Aerospace and Electronic Systems, 2016, 52(5): 2586-2596.
- [16] HUANG Y, ZHANG Y, 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.
- [17] HE S, SHIN H, Xu S, et al. Distributed estimation over a low-cost sensor network: A review of state-of-the-art[J]. Information Fusion, 2020, 54: 21-43.
- [18] ARASARATNAM I, HAYKIN S. Cubature Kalman filters[J]. IEEE Transactions on Automatic Control, 2009, 54(6): 1254-1269.
- [19] WANG G, LI N, ZHANG Y. Distributed maximum correntropy linear and nonlinear filters for systems with non-Gaussian noises[J]. Signal Processing, 2021, 182: 107937.
- [20] YANG X, ZHANG W, YU L, et al. Sequential Gaussian approximation filter for target tracking with nonsynchronous measurements[J]. IEEE Transactions on Aerospace and Electronic Systems, 2019, 55(1): 407-18.
- [21] WANG G, LI N, ZHANG Y. Hybrid consensus sigma point approximation nonlinear filter using statistical linearization[J]. Transactions of the Institute of Measurement and Control, 2018, 40(8): 2517-2525.
- [22] WANG G, LI N, ZHANG Y. Diffusion nonlinear Kalman filter with intermittent observations[J]. Proceedings of the Institution of Mechanical Engineers, Part G: Journal of Aerospace Engineering, 2018, 232(15): 2775-2783.
- [23] BATTISTELLI G, CHISCI L, MUGNAI G, et al. Con-

sensus-based linear and nonlinear filtering[J]. IEEE Transactions on Automatic Control, 2015, 60(5): 1410-1415.

作者简介



王国庆 男,1990年6月出生于山东省滕州市. 现为中国矿业大学信息与控制工程学院副教授、硕士生导师. 从事状态估计、信息融合和机器人导航方面的研究.

E-mail: wangguoqing0632@163.com



杨春雨(通讯作者) 男,1979年9月出生于辽宁省朝阳市. 现为中国矿业大学信息与控制工程学院教授、博士生导师. 从事智能系统控制技术、奇异摄动系统优化控制等方面的研究.

E-mail: chunyu yang@cumt.edu.cn



马磊 男,1989年6月出生于江苏省徐州市. 现为中国矿业大学信息与控制工程学院助理研究员. 从事奇异摄动系统的分析与设计、网络化控制系统的控制与滤波、信息物理系统的安全估计与控制的研究.

E-mail: malei@cumt.edu.cn