

抗丢包的传感网容积卡尔曼滤波目标跟踪算法

方旭明, 陈 英

(金陵科技学院软件工程学院, 江苏 南京 211169)

摘 要:卡尔曼滤波是高斯噪声无线传感网中最常用的目标追踪技术之一。由于无线传感网常被随机部署在一些复杂的环境,所以节点之间的通信会出现丢包的现象。对于传统的卡尔曼滤波器而言,丢包会大幅降低其目标追踪的精度和鲁棒性。针对这个问题,将描述丢包的 Bernoulli 过程模型融合进容积卡尔曼滤波器,提出了一种抗丢包的无线传感网目标跟踪算法。仿真结果表明:在不同的丢包率下新提出的算法比现有的卡尔曼滤波目标跟踪算法的精确性和鲁棒性都要好。

关键词:无线传感网;目标跟踪;丢包容忍;容积卡尔曼滤波;Bernoulli 过程

中图分类号:TP212;TN92

文献标识码:A

文章编号:1672-755X(2018)01-0010-05

An Anti-Packet-Loss Cubature-Kalman-Filtering Target Tracking Algorithm for Sensor Networks

FANG Xu-ming, CHEN Ying

(Jingling Institute of Technology, Nanjing 211169, China)

Abstract: Kalman filtering is one of the most commonly used target tracking techniques in Gauss-noise Wireless Sensor Networks. Because wireless sensor networks are of the randomly deployed in a number of complex environments, the communication between nodes will appear the phenomenon of packet loss. For the traditional Kalman filter, packet loss will greatly reduce the accuracy and robustness of the target tracking. To solve this problem, we fuse the Bernoulli process model describing packet loss into a cubature Kalman filter and propose an anti-packet-loss target tracking algorithm for Wireless Sensor Networks. The simulation results show that the proposed algorithm is more accurate and robust than the existing Kalman filtering target tracking algorithm at different packet loss rates.

Key words: wireless sensor network; target tracking; packet loss tolerance; cubature Kalman filtering; Bernoulli process

为了在丢包情况下可靠地跟踪无线传感器网络中移动目标的位置,能够容忍数据丢失的鲁棒卡尔曼滤波器被广泛地研究^[1-2]。最近,具有容忍数据丢失的鲁棒卡尔曼滤波器的一般情况已由 Sinopoli 等^[3]进行了研究,他们利用 Bernoulli 过程对包的随机丢失过程进行建模,推导出了一个丢包的随机过程模型。由于传统的卡尔曼滤波器只能用于跟踪直线运动的目标,跟踪非线性运动的目标会偏离真实的轨迹。Kluge S 等^[4]将随机丢包过程模型引入能跟踪非线性运动目标的扩展卡尔曼滤波器(Extended Kalman Filter, EKF),使扩展卡尔曼滤波器具备了容忍丢包的能力。Xu 等^[5]将容忍丢包的扩展卡尔曼滤波器应

收稿日期:2018-01-28

基金项目:金陵科技学院高层次人才科研启动基金(jit-b-201723);江苏高校品牌专业建设工程资助项目(PPZY2015B140)

作者简介:方旭明(1981—),男,江苏扬州人,讲师,博士,主要从事无线传感网定位与优化研究。

用于无线传感网移动目标的跟踪,并通过仿真分析了目标跟踪的性能。为减少 EKF 在跟踪非线性运动目标过程中所引起的线性化误差, Li 等^[6]将随机丢包过程模型嵌入到无迹卡尔曼滤波器(Unscented Kalman Filter, UKF),使无迹卡尔曼滤波器获得了容忍丢包的能力。虽然 UKF 已经被用于提高鲁棒跟踪算法的精度,但是当移动目标的状态向量包含太多的状态分量时,估计结果会变得不准确。由 Arasaratnam 等^[7]推导出来的容积卡尔曼滤波器(Cubature Kalman Filter, CKF)可以被用来解决这个问题。为了使跟踪算法更准确和更鲁棒,本试验将随机丢包过程模型引入进容积卡尔曼滤波器。因此,基于容积卡尔曼滤波的目标跟踪算法便具有了抗丢包的能力。

1 算法描述

1.1 测量与过程模型

在 WSN 中,移动目标的接收信号强度指示(Received Signal Strength Indicator, RSSI)值由锚节点测量。同时,利用测量的 RSSI 来估计移动目标与锚节点之间的距离。最后,鲁棒的容积卡尔曼滤波器对移动目标在二维平面上的坐标进行估计。

在时间步 k ,由鲁棒容积卡尔曼滤波器基于 RSSI 值估计的移动节点坐标被表示为 $\mathbf{x}_k = [x_k, y_k]^T$ 。锚节点的坐标可以写成向量 $\mathbf{a} = [a_1, a_2, \dots, a_n]^T$,其中 $\mathbf{a}_i = [x_i, y_i]^T$ 。第 i 个锚节点和移动节点之间的距离可以表示为:

$$d_k^i = \sqrt{(x_i - x_k)^2 + (y_i - y_k)^2} \quad (1)$$

目前,许多模型将无线信道的信号衰减表示为信号传播距离的函数。其中,对数正态路径损耗模型在实际无线传感器网络实现中得到广泛采用。从发射机接收的 d 距离处的 RSSI 值如下所示:

$$r = r_0 - 10\alpha \log_{10}(d/d_0) \quad (2)$$

其中, r_0 是在距离 d_0 处测量的以分贝为单位的平均功率, α 是路径损耗指数。

表示移动目标运动的测量方程如下所示:

$$\mathbf{z}_k = h(\mathbf{x}_k) + \mathbf{v}_k \quad (3)$$

其中, \mathbf{z}_k 是在时间步 k 的测量向量, h 是用于观测状态 \mathbf{x}_k 的测量函数, \mathbf{v} 表示测量噪声向量,它是由服从正态分布的随机变量组成的。

测量向量里的分量来自于锚节点接收的目标节点 RSSI 值,它被定义为:

$$\mathbf{z}_k = [r_k^1, \dots, r_k^L]^T \quad (4)$$

其中, L 表示锚节点的数量。

基于 RSSI 距离模型建立的测量函数定义如下:

$$h(\mathbf{x}_k) = \begin{bmatrix} r_0 - 10\alpha \log_{10}(d_k^1/d_0) \\ \vdots \\ r_0 - 10\alpha \log_{10}(d_k^L/d_0) \end{bmatrix} \quad (5)$$

测量噪声服从于均值为 0 的正态分布 $N(0, \mathbf{R})$,其中协方差矩阵表示如下:

$$\mathbf{R} = \text{diag}(\sigma_1^2, \dots, \sigma_L^2) \quad (6)$$

描述移动目标运动规律的状态方程如下所示:

$$\mathbf{x}_k = f(\mathbf{x}_{k-1}) + \mathbf{w}_{k-1} \quad (7)$$

其中, \mathbf{x}_k 表示在时间步 k 的状态向量, f 是状态转移函数,它被用来预测下一个时间步的状态, \mathbf{w} 表示服从于正态分布 $N(0, \mathbf{Q})$ 的过程噪声。

考虑一个典型的跟踪场景,其中移动目标在一个未知速率下执行机动转弯。移动目标的运动学特征可以用协同转弯(Coordinated Turn, CT)来建模^[5]。运动目标的状态方程如下:

$$\mathbf{x}_k = \begin{bmatrix} 1 & \frac{\sin(\omega_{k-1} \Delta t)}{\omega_{k-1}} & 0 & \frac{\cos(\omega_{k-1} \Delta t) - 1}{\omega_{k-1}} & 0 \\ 0 & \cos(\omega_{k-1} \Delta t) & 0 & -\sin(\omega_{k-1} \Delta t) & 0 \\ 0 & \frac{1 - \cos(\omega_{k-1} \Delta t)}{\omega_{k-1}} & 1 & \frac{\sin(\omega_{k-1} \Delta t)}{\omega_{k-1}} & 0 \\ 0 & \sin(\omega_{k-1} \Delta t) & 0 & \cos(\omega_{k-1} \Delta t) & 0 \\ 0 & 0 & 0 & 0 & 1 \end{bmatrix} \mathbf{x}_{k-1} + \mathbf{w}_{k-1} \quad (8)$$

其中,目标的状态向量表示为 $\mathbf{x} = [x \ \dot{x} \ y \ \dot{y} \ \omega]^T$, x 和 y 表示目标的二维坐标, \dot{x} 和 \dot{y} 分别表示目标在 x 方向和 y 方向上的速度, ω 表示目标的转弯角速度, Δt 是两个连续时间步之间的时间间隔。

过程噪声 \mathbf{w} 服从于均值为 0、协方差为 \mathbf{Q} 的正态分布 $N(0, \mathbf{Q})$, 协方差矩阵如下所示:

$$\mathbf{Q} = \begin{bmatrix} \frac{\Delta t^3 q_1}{3} & \frac{\Delta t^2 q_1}{2} & 0 & 0 & 0 \\ \frac{\Delta t^2 q_1}{2} & \Delta t q_1 & 0 & 0 & 0 \\ 0 & 0 & \frac{\Delta t^3 q_2}{3} & \frac{\Delta t^2 q_2}{2} & 0 \\ 0 & 0 & \frac{\Delta t^2 q_2}{2} & \Delta t q_2 & 0 \\ 0 & 0 & 0 & 0 & \Delta t q_3 \end{bmatrix} \quad (9)$$

其中, q_1 、 q_2 和 q_3 表示与过程噪声强度有关的变量。

1.2 算法步骤

由于在无线信道上传输的数据包的随机丢失可以用 Bernoulli 过程来建模^[4], 将在时间步 k 测量的到达定义为一个二进制随机变量 γ_k 。如果一个测量在时间步 k 到达, 那么 γ_k 被设置为 1; 否则, γ_k 被设置为 0。

在丢包的情况下, 在时间步 k 的测量噪声方差被定义如下^[6]:

$$p(\mathbf{v}_k | \gamma_k) = \begin{cases} N(0, \mathbf{R}), & \gamma_k = 1 \\ N(0, \sigma^2 \mathbf{I}), & \gamma_k = 0 \end{cases} \quad (10)$$

如果 $\gamma_k = 1$ 时, 在时间步 k 的测量噪声协方差则为 \mathbf{R} , 否则为某个 $\sigma^2 \mathbf{I}$ 。当没有测量到达时, σ 趋向于无穷。

将测量到达的 Bernoulli 过程融合进容积卡尔曼滤波器, 以至于重新设计的鲁棒 CKF 算法能可靠地在丢包无线传感网中跟踪移动目标的状态演化。

鲁棒 CKF 算法的步骤如下:

步骤 1: 时间更新

1) 因式分解在时间步 $k-1$ 的后验状态估计误差协方差

$$\mathbf{P}_{k-1|k-1} = \mathbf{S}_{k-1|k-1} \mathbf{S}_{k-1|k-1}^T \quad (11)$$

2) 计算容积点

$$\mathbf{X}_{i,k-1|k-1} = \mathbf{S}_{k-1|k-1} \boldsymbol{\xi}_i + \mathbf{x}_{k-1|k-1} \quad (i=1, 2, \dots, m) \quad (12)$$

其中, $\boldsymbol{\xi}_i = \sqrt{\frac{m}{2}} [1]_i$, $m=2n$, n 是状态向量的维数, 符号 $[1]_i$ 被用来表示完全对称点集 $[1]$ 的第 i 个点, $[1]$ 是以所有可能的方式通过置换和改变 n 维单位向量符号得到的。

3) 计算传播的容积点

$$\mathbf{X}_{i,k|k-1}^* = f(\mathbf{X}_{i,k-1|k-1}) \quad (i=1, 2, \dots, m) \quad (13)$$

4) 估计预测的状态

$$\hat{\mathbf{x}}_{k|k-1} = \frac{1}{m} \sum_{i=1}^m \mathbf{X}_{i,k|k-1}^* \quad (14)$$

5) 预测状态估计误差的协方差

$$\mathbf{P}_{k|k-1} = \frac{1}{m} \sum_{i=1}^m \mathbf{X}_{i,k|k-1}^* \mathbf{X}_{i,k|k-1}^{*\top} - \hat{\mathbf{x}}_{k|k-1} \hat{\mathbf{x}}_{k|k-1}^\top + \mathbf{Q}_{k-1} \quad (15)$$

步骤 2: 测量更新

1) 因式分解预测的状态估计误差协方差

$$\mathbf{P}_{k|k-1} = \mathbf{S}_{k|k-1} \mathbf{S}_{k|k-1}^\top \quad (16)$$

2) 计算容积点

$$\mathbf{X}_{i,k|k-1} = \mathbf{S}_{k|k-1} \boldsymbol{\xi}_i + \hat{\mathbf{x}}_{k|k-1} \quad (i=1, 2, \dots, m) \quad (17)$$

3) 计算传播的容积点

$$\mathbf{Z}_{i,k|k-1} = h(\mathbf{X}_{i,k|k-1}) \quad (i=1, 2, \dots, m) \quad (18)$$

4) 估计预测的测量

$$\hat{\mathbf{z}}_{k|k-1} = \frac{1}{m} \sum_{i=1}^m \mathbf{Z}_{i,k|k-1} \quad (19)$$

5) 估计测量误差协方差

$$\mathbf{P}_{z,z,k|k-1} = \frac{1}{m} \sum_{i=1}^m \mathbf{Z}_{i,k|k-1} \mathbf{Z}_{i,k|k-1}^\top - \hat{\mathbf{z}}_{k|k-1} \hat{\mathbf{z}}_{k|k-1}^\top + \gamma_k \mathbf{R}_k + (1 - \gamma_k) \sigma^2 \mathbf{I} \quad (20)$$

6) 估计互协方差

$$\mathbf{P}_{x,z,k|k-1} = \frac{1}{m} \sum_{i=1}^m \mathbf{X}_{i,k|k-1} \mathbf{Z}_{i,k|k-1}^\top - \hat{\mathbf{x}}_{k|k-1} \hat{\mathbf{z}}_{k|k-1}^\top \quad (21)$$

7) 计算卡尔曼增益

$$\mathbf{K}_k = \mathbf{P}_{x,z,k|k-1} \left[\frac{1}{m} \sum_{i=1}^m \mathbf{Z}_{i,k|k-1} \mathbf{Z}_{i,k|k-1}^\top - \hat{\mathbf{z}}_{k|k-1} \hat{\mathbf{z}}_{k|k-1}^\top + \gamma_k \mathbf{R}_k + (1 - \gamma_k) \sigma^2 \mathbf{I} \right]^{-1} \quad (22)$$

8) 估计更新的状态

$$\hat{\mathbf{x}}_{k|k} = \hat{\mathbf{x}}_{k|k-1} + \mathbf{P}_{x,z,k|k-1} \left[\frac{1}{m} \sum_{i=1}^m \mathbf{Z}_{i,k|k-1} \mathbf{Z}_{i,k|k-1}^\top - \hat{\mathbf{z}}_{k|k-1} \hat{\mathbf{z}}_{k|k-1}^\top + \gamma_k \mathbf{R}_k + (1 - \gamma_k) \sigma^2 \mathbf{I} \right]^{-1} (\mathbf{z}_k - \hat{\mathbf{z}}_{k|k-1}) \quad (23)$$

9) 估计更新的状态, 估计误差协方差

$$\mathbf{P}_{k|k} = \mathbf{P}_{k|k-1} - \mathbf{P}_{x,z,k|k-1} \left[\frac{1}{m} \sum_{i=1}^m \mathbf{Z}_{i,k|k-1} \mathbf{Z}_{i,k|k-1}^\top - \hat{\mathbf{z}}_{k|k-1} \hat{\mathbf{z}}_{k|k-1}^\top + \gamma_k \mathbf{R}_k + (1 - \gamma_k) \sigma^2 \mathbf{I} \right]^{-1} \mathbf{P}_{z,z,k|k-1} \quad (24)$$

步骤 3: 对于下一个测量重复步骤 1 和 2。

2 仿真实验及性能分析

2.1 仿真环境设置

为了评估算法的跟踪性能, 我们在一个 $60 \text{ m} \times 60 \text{ m}$ 监视区域进行了仿真实验。9 个位置分别为 $(0, 60)$ 、 $(60, 60)$ 、 $(0, 0)$ 、 $(60, 0)$ 、 $(30, 30)$ 、 $(0, 30)$ 、 $(60, 30)$ 、 $(30, 60)$ 、 $(30, 0)$ 的锚节点被部署在该区域。RS-SI 测量误差的方差被设置为 0.25。通过 500 次蒙特卡洛仿真实验, 移动目标的跟踪性能被评估。移动节点从 $(28, 40)$ 开始运动, 2 个连续时间步之间的时间间隔为 1 s。在 $0 \sim 15$ 、 $20 \sim 35$ 、 $40 \sim 65$ 、 $70 \sim 85$ 、 $90 \sim 100$ 时间步期间, 移动目标进行匀速运动, 其速度为 $1 \text{ m} \cdot \text{s}^{-1}$ 。在 $15 \sim 20$ 、 $35 \sim 40$ 、 $65 \sim 70$ 、 $85 \sim 90$ 时间步期间, 移动目标按顺时针方向以 $0.262 \text{ rad} \cdot \text{s}^{-1}$ 进行机动转弯。协同转弯机动的过程噪声协方差被设置为 $0.01 * \mathbf{I}_{5 \times 5}$ 。无线传感网中数据包丢失的概率分别被设置为 0.1、0.2 和 0.3。

2.2 仿真结果与分析

图 1 比较了 3 种丢包率 EKF、UKF 和 CKF 在不同时间步的估计精度。从图 1 可以看出, 丢包时所有 3 种算法都会产生相对更大的估计误差。然而, CKF 具有比其他 2 种算法更小的估计误差。由此可见, 在丢包时 CKF 的鲁棒性和精确性比 EKF 和 UKF 更好。

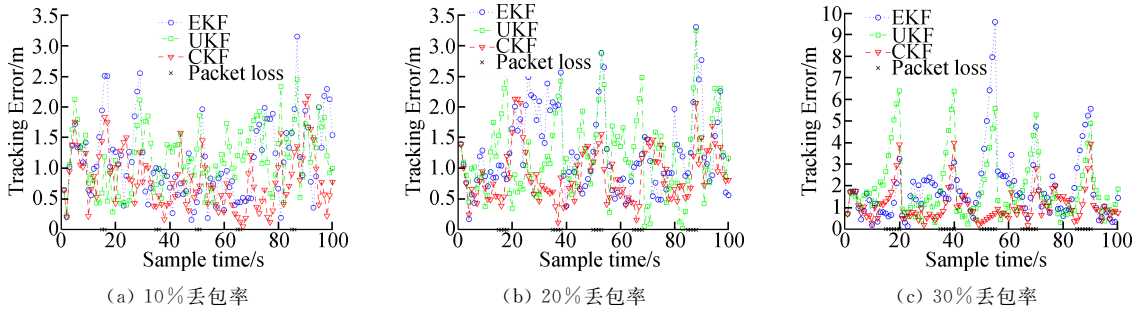


图 1 EKF、UKF 和 CKF 不同时间步的估计误差

图 2 显示了在不同数据包丢失概率情况下的 EKF、UKF 和 CKF 估计结果。如图 2 所示,在不同的数据包丢失概率下 EKF 和 UKF 的估计误差总大于 CKF 的估计误差,这意味着 CKF 总可以更准确地跟踪移动目标。

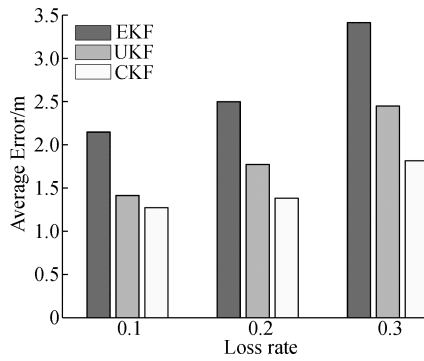


图 2 EKF、UKF 和 CKF 在不同丢包率下的平均估计误差

3 结 语

本文根据无线传感器网络使用卡尔曼滤波器跟踪移动目标过程容易受到数据包丢失的影响,将能够描述丢包过程的 Bernoulli 过程模型引入到容积卡尔曼滤波器,解决了目前基于卡尔曼滤波器的无线传感器网络目标跟踪算法在丢包时精度和鲁棒性变差的问题。大量仿真实验的结果表明,在不同数据包丢失率的情况下,提出的抗丢包容积卡尔曼滤波目标跟踪算法比现有卡尔曼滤波目标跟踪算法的精确性和鲁棒性都更好。

参考文献:

- [1] Bhuiyan M Z, Wang G, Vasilakos A V. Local area prediction-based mobile target tracking in wireless sensor networks [J]. IEEE Transactions on Computers, 2015, 64(7): 1968 - 1982
- [2] Mahfouz S, Mourad-Chehade F, Honeine P, et al. Target tracking using machine learning and Kalman filter in wireless sensor networks [J]. IEEE Sensors Journal, 2014, 14(10): 3715 - 3725
- [3] Sinopoli B, Schenato L, Franceschetti M, et al. Kalman filtering with intermittent observations [J]. IEEE Transactions on Automatic Control, 2004, 49(9): 1453 - 1464
- [4] Kluge S, Reif K, Brokate M. Stochastic stability of the extended kalman filter with intermittent observations [J]. IEEE Transactions on Automatic Control, 2010, 55(2): 514 - 518
- [5] Xu C. Filtering with uncertain observations and its application in WSN [J]. Computer Applications and Software, 2014, 31(4): 133 - 137
- [6] Li L, Xia Y. Stochastic stability of the unscented Kalman filter with intermittent observations [J]. Automatica, 2012, 48(5): 978 - 981
- [7] Arasaratnam I, Haykin S. Cubature Kalman filters [J]. IEEE Transactions on Automatic Control, 2009, 54(6): 1254 - 1269