

时间异步无线传感器网络的分布式目标跟踪^①

万江文^② 薛 皓* 于 宁 陈 斌*

(北京航空航天大学仪器科学与光电工程学院 北京 100191)

(* 北京邮电大学计算机学院 北京 100876)

摘 要 为了降低无线传感器网络在目标跟踪过程中的网络能耗,提出了一种时间异步条件下的分布式目标跟踪方法。首先,依据节点到目标的距离进行动态成簇,以跟踪簇为时间的计算单元,由簇头完成簇内跟踪时间计算及簇间贯序传递,然后引入并行粒子滤波(PPF)算法将粒子集分为多个子集,在子节点处并行采样、计算权重和重采样,最后,簇头节点收集各子节点上传的结果并完成目标的局部状态估计。仿真结果表明,PPF 算法具有较好的跟踪精度,且相比于集中式粒子滤波(CPF)算法,可降低约 38% 的通信量。

关键词 无线传感器网络(WSN), 目标跟踪, 时间异步, 并行粒子滤波(PPF)

0 引 言

无线传感器网络(wireless sensor network, WSN)是一种按需求完成指定任务的“智能”系统,目标跟踪是其典型应用之一。跟踪过程中,传感器节点通过相互协作和数据共享,在资源受限的条件下实现对目标的跟踪。粒子滤波(particle filter, PF)^[1]是一种非线性滤波技术,在处理非线性非高斯问题时具有良好性能,被广泛应用于视频和雷达跟踪等领域,但其不足之处在于算法的计算量庞大。为了解决该问题,Coates^[2]利用似然因子分解和参数驯化将粒子滤波分布式地在多个节点上计算。Brooks^[3]提出将信号和目标分类的分布式跟踪策略。Bolic^[4]提出了重采样比例分配(resampling proportional allocation, RPA)和重采样非比例分配(resampling non-proportional allocation, RNA)算法,将重采样操作分布地在多个处理单元执行。相对于扩展卡尔曼滤波(extended Kalman filter, EKF)^[5]、无迹卡尔曼滤波(unsecented Kalman filter, UKF)^[6]等非线性滤波方法,现有分布式粒子滤波计算量仍然较大,因此如何充分利用计算资源,在有效实现目标跟踪的同时尽可能降低能耗,是今后无线传感器网络目标跟踪研究的重要方向之一。另一方面,为了实现测量数据配准,现有无线传感器网络目标跟踪方法通常需要预先进行时间同步。节点间的时间参考分组交换操作^[7-9]将不可

避免地增大网络的通信能耗,这对于一些实时性和跟踪精度要求不高,但能量严格受限的应用场合(如野生动物活动监测)往往是得不偿失的。

针对以上问题,本文提出了一种无需时间同步协议支持的分布式无线传感器网络目标跟踪方法,最后介绍了时间异步条件下,无线传感器网络目标跟踪过程中的传感器节点管理算法和并行粒子滤波(parallel particle filter, PPF)的状态估计算法,并进行了仿真实验。

1 无线传感器网络的目标跟踪

无线传感器网络目标跟踪是由多节点相互协作实现的。特定区域内的节点通过协作式的信号处理实现目标初步定位,向局部数据处理中心发送目标位置信息,并在该处执行相对复杂的目标状态估计算法以得到局部的目标运动速度等状态参数,据此推测目标将要到达的区域,通过节点激活协议,启动预测区域内的节点参与目标跟踪。本文提出的时间异步条件下实现无线传感器网络目标跟踪的方法,包括动态成簇算法和并行式信息滤波算法。

1.1 动态成簇算法

设网内所有节点的探测和通信能力相同。根据分布式跟踪的思想,将无线传感器网络节点动态组织成跟踪簇。网内节点状态分为空闲状态、休眠状态和工作状态,其中工作状态可分为伪跟踪状态、跟

① 国家自然科学基金(60873240), 863 计划(2009AA01Z201)和北京市教育委员会共建资助项目。

② 男,1963 年生,博士,教授;研究方向:传感系统与仪器,传感网络与信息融合,定位与跟踪技术;联系人, E-mail: jwwan@buaa.edu.cn (收稿日期:2008-10-15)

踪状态。处于工作状态的节点按其在跟踪簇内角色的不同,又可分为簇头(cluster header, CH)状态、候选状态和成员状态。动态成簇算法的关键机制有簇构建、簇调整和跟踪时间管理,以下分别进行描述。

(1) 簇构建。目标靠近或进入网络时,目标附近的多个传感器节点同时检测到目标信号。节点通过抢占机制选择距目标最近的节点担当簇头。簇头产生后设定自身为簇头状态,并发布通告,唤醒其一跳通讯范围内的其它节点。接到唤醒通告的节点发出加入申请,并设定自己为成员状态,并转入跟踪状态。跟踪簇外的节点除路由节点外,均转入休眠状态。目标在跟踪簇内移动时,各子节点不断对目标状态进行捕捉,并通过与簇头间的信息交换,在簇头完成局部的目标状态估计。

(2) 簇调整。随着目标的移动,跟踪簇的成员组成或簇头将发生变化。通过距离判断,动态决定跟踪簇成员的添加、删除以及簇头转换。成员更新后,需要调整簇头处的成员列表。簇头发生移交时需要将前一次滤波参数传递给新簇头,并利用新的测量值更新后验概率。

(3) 跟踪时间管理。在没有时间同步协议支持的无线传感器网络中,各节点的本地时钟始终异步运行。网络初始化阶段,网内节点除了进行自身定位外,还分别计算本地时钟与汇聚节点(Sink)上标准时钟的时间差。本文所涉及目标跟踪时间的处理包括目标跟踪起始时间标定、簇内跟踪时间计算及簇间跟踪时间贯序传递。

a. 跟踪起始时间标定。目标逐渐靠近监测区域过程中,一旦探测到目标信号的节点数达到一定数量(如3个),则确认目标出现,并开始构建第一个跟踪簇。第一个簇头(CH₀)产生后记录的当前本地时间即被标定为整个目标跟踪事件的起始时间。

b. 簇内跟踪时间计算。跟踪节点接收到跟踪时间标定通告后,启动跟踪计时器。接收到目标信号时,记录当前计时器的读数,并把该值写入数据包发送至簇头。簇头计算多个计时器读数的均值,并将结果换算成时间,与最新的目标跟踪时间进行累加。

c. 簇间跟踪时间贯序传递。簇头发生移交时,需要进行跟踪时间的贯序传递,最终 Sink 节点完成跟踪时间的配准。设初始化完成后 CH₀ 的本地时间为 P_0 ,与汇聚节点的时间差为 D (可为正值或者负值);目标跟踪的起始时间为 P ; T_i ($i = 1, 2, 3, \dots$) 为第 i 个簇头纪录的目标跟踪时间, t_i ($i = 1,$

$2, 3, \dots$) 为在第 i 个跟踪簇内进行的跟踪时间长度。忽略各节点晶体的频率漂移和通信延时误差,目标跟踪时间贯序递推计算如下:

$$T_i = T_{i-1} + t_i \quad (i = 1, 2, 3, \dots) \quad (1)$$

其中 $T_0 = P$ 。

Sink 节点进行目标跟踪时间配准的方法为

$$T_r = T_i + D \quad (2)$$

其中 T_r 为修正过的目标跟踪绝对时间。

1.2 并行式信息滤波

1.2.1 目标状态模型及测量模型

设跟踪簇由 N 个传感器节点组成,无线传感器网络中目标跟踪问题可以表示为

$$x_k = f_k(x_{k-1}, v_{k-1}) \quad (3)$$

$$z_k^i = h_k^i(x_k, n_k^i) \quad (4)$$

式(3)为目标状态方程: x_k 是 k 时刻目标的状态向量, f_k 是非线性状态转移函数, v_{k-1} 是独立同分布的过程噪声序列。式(4)为量测方程: z_k^i 是传感器节点 i 在 k 时刻的观测向量, h_k^i 是该节点的非线性测量函数, n_k^i 为独立同分布的观测噪声序列,且传感器节点的观测噪声互不相关。

1.2.2 粒子滤波

粒子滤波是一种基于 Monte Carlo 仿真的最优回归贝叶斯滤波方法,它将所关心的状态矢量表示为一组带有相关权值的随机样本,并基于这些样本和权值估计系统状态。与 EKF、UKF 等常用的非线性滤波算法相比,粒子滤波不受线性化误差或高斯噪声假定的限制,且精度高、适用性强。采样重要重采样(sampling importance resampling, SIR)是粒子滤波的核心算法,主要步骤如下:

a. 采样。根据重要函数 $p(x_k | x_{k-1})$ 采样,产生粒子集 $\{x_k^i, w_k^i\}$, $i = 1, \dots, N$ 。

b. 权重计算及归一化。计算每个粒子的权重值:

$$w_k^i = p(z_k^i | x_k^i) w_{k-1}^i \quad (5)$$

$$\tilde{w}_k^i = w_k^i / \sum_{i=1}^N w_k^i \quad (6)$$

c. 重采样。对粒子集 $\{x_k^i, \tilde{w}_k^i\}_{i=1}^N$ 进行重采样,产生新的粒子集 $\{\tilde{x}_k^i\}_{i=1}^N$ 。

1.2.3 并行式粒子滤波

传感器节点的能量受限且数据处理能力相对较弱,如果采用集中式粒子滤波(center particle filter, CPF)算法,即节点将所有目标状态观测量发送到处理中心(如 Sink 节点)进行处理,不仅导致跟踪系统

实时性下降,更增大了网络通讯能耗。本文采用的并行粒子滤波(PPF)算法将滤波所需的粒子集分划成多个小的子集,分配到跟踪簇中的每个子节点上进行并行计算,这样就将计算量分散于多个节点,从而减少簇头的计算负荷,均衡网络能耗。

$\{x_k^{i,j}, w_k^{i,j}\}$ 表示运行在节点 j 上的第 i 个粒子,设滤波所需粒子数为 M ,具体算法实现过程如下:

(1) 初始化。在初始化时刻 ($k = 0$) 产生跟踪簇,为节点 j 上分配 n_j 个粒子,且 $\sum_{j=1}^N n_j = M$ 。根据先验概率 $p(x_0 | z_0)$ 进行采样,均匀分布初始化粒子集,每个粒子具有相等的权重值。

(2) 子节点的粒子采样。采样时刻 k , 已知前一采样时刻的粒子集 $\{x_{k-1}^{i,j}, w_{k-1}^{i,j}\}$, 通过状态方程(3)预测状态,并采样新的粒子:

$$x_k^{i,j} \sim q(x_k | x_{k-1}^{i,j}, z_k) = p(x_k | x_{k-1}^{i,j}) \quad (7)$$

(3) 子节点的更新和聚合参数计算。当子节点收到当前观测量后,计算粒子权重:

$$w_k^{i,j} \propto w_{k-1}^{i,j} \frac{p(z_k | x_k^{i,j}) p(x_k^{i,j} | x_{k-1}^{i,j})}{q(x_{k-1}^{i,j} | x_k^{i,j}, z_k)} \quad (8)$$

聚合参数计算:

$$S_k^j = \sum_{i=1}^{n_j} w_k^{i,j} \quad (9)$$

$$X_k^j = \sum_{i=1}^{n_j} x_k^{i,j} \cdot w_k^{i,j} \quad (10)$$

$$G_k^j = \sum_{i=1}^{n_j} (w_k^{i,j})^2 \quad (11)$$

$$P_k^j = \sum_{i=1}^{n_j} w_k^{i,j} \cdot x_k^{i,j} \cdot (x_k^{i,j})^T \quad (12)$$

其中, X_k^j 是未归一化的本地状态估计值, S_k^j 是与之相对应的未归一化权重值, G_k^j 和 P_k^j 分别用于计算状态估计误差和控制粒子退化现象的产生。

(4) 上载通信。将各子节点的聚合参数 ($S_k^j, X_k^j, G_k^j, P_k^j$) 传送到簇头。

(5) 状态估计。在簇头和子节点同时形成状态估计。簇头对子节点的权重进行求和:

$$C_k = \sum_{j=1}^N S_k^j \quad (13)$$

计算全局估计和方差,并传送到汇聚节点:

$$\hat{x}_k = \sum_{j=1}^N \sum_{i=1}^{n_j} w_k^{i,j} \cdot x_k^{i,j} = \sum_{j=1}^N X_k^j / C_k \quad (14)$$

$$P_k = \sum_{j=1}^N P_k^j / C_k - \hat{x}_k \cdot \hat{x}_k^T \quad (15)$$

子节点根据簇头估计结果,计算状态:

$$p(x_k | z_{1:k}) \propto \hat{x}_k \quad (16)$$

(6) 重采样。如果权值小的粒子数低于指定门限值,则重采样粒子集,置位重采样标志 f , 并将其传送到各子节点,各子节点独立进行采样过程:

$$\{x_k^{i,j}, w_k^{i,j}\} \rightarrow \{x_k^{i,j}, 1/n_j\} \quad (17)$$

为了保持粒子集的一致性,需要定期进行全局采样:

$$\{x_k^{i,j}, w_k^{i,j}\} \rightarrow \{x_k^{i,j}, 1/\sum_{j=1}^N n_j\}, k = mC \quad (18)$$

(7) 本地估计。计算本地状态估计和方差:

$$\hat{x}_k = E[x_k | z_{1:k}] \approx \sum_{j=1}^M w_k^j \cdot x_k^j \quad (19)$$

$$P_k = \sum_{i=1}^N P_k^i - \hat{x}_k \cdot \hat{x}_k^T \quad (20)$$

(8) 粒子调节。各子节点根据全局重采样结果,进行粒子的调节,使簇中的所有粒子具有相同的权重。

(9) 粒子交换。跟踪簇转换时,需要将粒子集从旧簇头传送到新簇头。为了减少通讯数据量,旧簇头利用高斯混合模型 (Gaussian mixture model, GMM) 对粒子集进行参数近似,新簇头利用近似参数重建粒子集。粒子集参数近似方法如下:

$$\pi_{t,k}^n \approx \hat{\pi}_{t,k}^n = \sum_{m=1}^c \hat{\chi}_{t,k}^m \cdot N(\hat{\mu}_{t,k}^m, \hat{\sigma}_{t,k}^m) \quad (21)$$

其中, $\hat{\chi}_{t,k}^m$ 为 GMM 权值参数; $\hat{\mu}_{t,k}^m$ 为 GMM 均值参数; $\hat{\sigma}_{t,k}^m$ 为 GMM 方差参数; c 为混合器个数; $\hat{\pi}_{t,k}^n$ 为 $\pi_{t,k}^n$ 的近似表示; N 表示高斯分布。通过 GMM 近似后的系统均值和协方差为

$$E_{t,k}(\hat{x}_{t,k}) = \sum_{m=1}^c \hat{\chi}_{t,k}^m \cdot \hat{\mu}_{t,k}^m \quad (22)$$

$$P_{t,k}(\hat{x}_{t,k}) = \sum_{m=1}^c \hat{\chi}_{t,k}^m [\hat{\sigma}_{t,k}^m + (\hat{\mu}_{t,k}^m - x_{t,k})(\hat{\mu}_{t,k}^m - x_{t,k})^T] \quad (23)$$

2 仿真验证及分析

假设 250 个传感器节点随机分布于 $100\text{m} \times 250\text{m}$ 的长方形监测区域内。所有节点具有相同的功能和性能,有效通信半径为 30m,传感器有效探测距离为 15m。节点间平均连通度为 20,通信成功或失败的概率固定。已完成网络初始化,节点位置已知。

采用如下目标运动模型^[10]:

$$x_k = Ax_{k-1} + q_{k-1} \quad (24)$$

这里 $x_k = (x, \dot{x}, y, \dot{y})^T \in \mathbf{R}^4$, 为目标状态向量, 其中 $[x, y]^T \in \mathbf{R}^2$ 和 $[\dot{x}, \dot{y}]^T \in \mathbf{R}^2$ 分别为目标在时刻 k 的位置和速度向量。 t 为传感器节点的采样间隔, 仿真中取值为 1s 。 A 为状态转移矩阵, q_{k-1} 为高斯过程噪声, 其方差为

$$E(q_{k-1}q_{k-1}^T) = \begin{pmatrix} \frac{t^3}{3} & \frac{t^2}{2} & 0 & 0 \\ \frac{t^2}{2} & t & 0 & 0 \\ 0 & 0 & \frac{t^3}{3} & \frac{t^2}{2} \\ 0 & 0 & \frac{t^2}{2} & t \end{pmatrix} q \quad (25)$$

q 是噪声的功率谱密度, 仿真中取值为 $q = 0.1$ 。图 1 给出了仿真场景以及目标在场景中的实际运动轨迹。

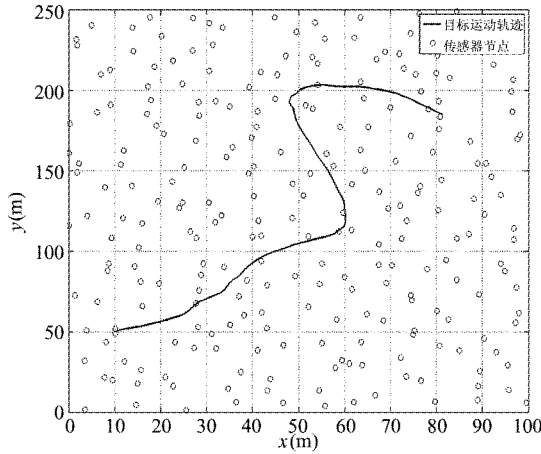


图 1 跟踪场景及目标实际运动轨迹

传感器节点的观测值为目标的方位角 θ , 统一的观测方程如下:

$$\theta_k^i = \arctan\left(\frac{y_k - s_y^i}{x_k - s_x^i}\right) + r_k^i \quad (26)$$

其中 (s_x^i, s_y^i) 是传感器节点 i 的位置, $r_k^i \sim N(0, \sigma^2)$ 为高斯白噪声且标准差 $\sigma = 0.05$ 。目标初始状态为 $x_0 = [10 \ 0.5 \ 50 \ 0.5]^T$ 。滤波器初始状态 $\bar{x}_0 = [5 \ 0.5 \ 45 \ 0.5]^T$, 状态的先验概率分布为 $x_0 \sim N(0, P_0)$, 其中:

$$P_0 = 0.05 \times \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix}$$

在相同条件下对 CPF 和 PPF(粒子数 $M = 500$)

进行跟踪效果对比实验, 经过 100 次蒙特卡洛仿真, 两种滤波器的轨迹跟踪效果与实际轨迹的对比见图 2。图 3 为两种算法对目标位置估计的根均方差 (root mean square error, RMSE) 比较。从跟踪效果来看, CPF 和 PPF 滤波效果接近, 都能较好地实现目标的状态跟踪和轨迹预测。在跟踪精度方面, PPF 相对于 CPF 略有下降, 这也能从图 3 中看出, PPF 和 CPF 的 RMSE 值分别为 2.154 和 1.685。这是由于在簇头转换中, 为了减少数据通信量, 使用了 GMM 方法对状态参数近似和重建机制, 这在一定程度上增加了过程噪声。

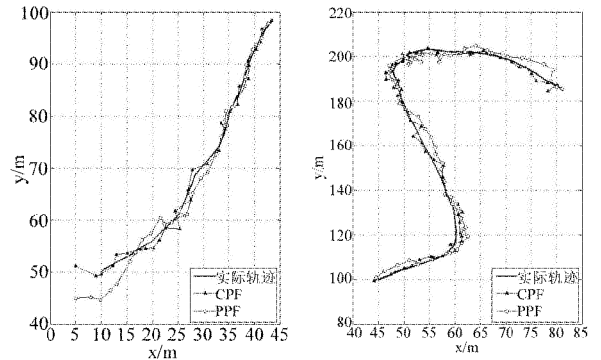


图 2 CPF 和 PPF 跟踪效果对比

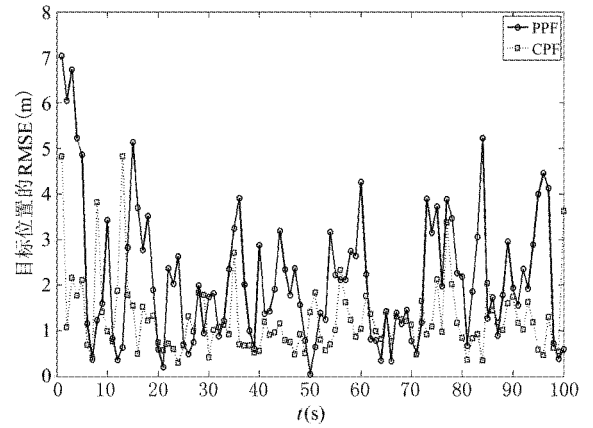


图 3 RMSE 比较

图 4 为每 10s 对 CPF 和 PPF 的平均数据通信量进行统计的结果。从图中可以看出, 由于 PPF 方法不需要将每次的观测数据发送到数据处理中心, 而是在进行局部化处理, 这样就大为降低了网络通信量。虽然由于跟踪簇成员数目不同, 相应数据通信量会略有变化, 但总体看来, PPF 的通信量比 CPF 减少约 38%, 这对于能量受限的无线传感器网络来说尤其重要。

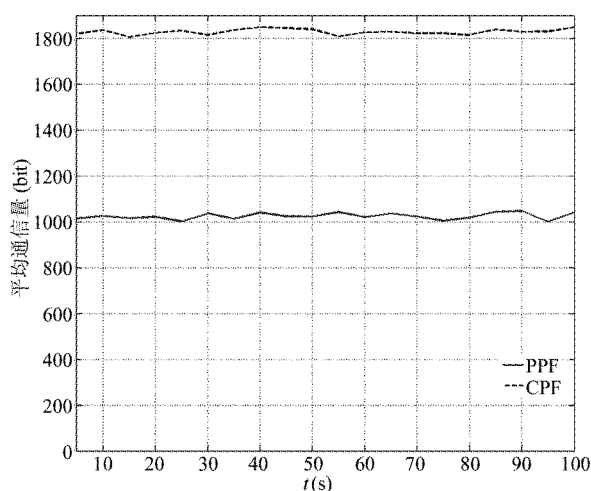


图 4 通信量比较

3 结 论

在保证跟踪精度的同时,如何降低网络能耗以延长其生存期,是无线传感器网络目标跟踪算法设计时所必需解决的关键问题之一。

本文提出了一种无需时间同步协议支持的分布式无线传感器网络目标跟踪方法。该方法通过高效的局部节点协同机制,实现跟踪过程中对节点和跟踪时间的有效管理;利用并行粒子滤波算法,将粒子滤波分散于多个节点上并行计算,从而提高计算效率,平衡网络节点能耗。仿真结果显示,该方法用于目标跟踪时,具有较好的跟踪精度,且能有效减少跟踪过程中的数据通信量,进而降低节点能耗。

参考文献

- [1] Gordon N J, Salmond D J, Smith A F M. Novel approach to nonlinear/non-gaussian bayesian state estimation. *Radar and Signal Processing IEEE Proceedings-F*, 1993, 140(2): 107-113
- [2] Coates M. Distributed particle filters for sensor networks. In: *Proceedings of the Information Processing in Sensor Networks*, Berkeley, California, USA, 2004. 99-107
- [3] Brooks R R, Ramanathan P, Sayeed A M. Distributed target classification and tracking in sensor networks. *IEEE Proceedings on Sensor Networks and Applications*, 2003, 91(8): 1163-1171
- [4] Bolic, M, Djuric, P M, Hong S J. Resampling algorithms and architectures for distributed particle filters. *IEEE Transactions on Signal Processing*, 2005, 53(7): 2442-2450
- [5] Michael K, Pal L Y. Covariance control for multisensor systems. *IEEE Transactions on AES (S0018925)*, 2002, 38(4): 1138-1157
- [6] Simon J, Julier, Jeffrey K. Uhlmann. A new extension of the kalman filter to nonlinear systems. In: *Proceedings of the International Symposium on Aerospace/Defense Signal Processing, Sensor Fusion, and Target Recognition*, Orlando, FL, USA, 1997. 182-193
- [7] Dai H, Han R. TSync: a lightweight bidirectional time synchronization service for wireless sensor networks. *ACM Mobile Computing and Communication Review*, 2004, 8(1): 125-139
- [8] Ganerival S, Kumar R, Srivastava M B. Timing-sync protocol for sensor networks. In: *Proceedings of the 1st ACM Conference on Embedded Networked Sensor Systems (SenSys)*, Los Angeles, California, USA, 2003. 138-149
- [9] Miklos M, Branislav K, Gyula S, et al. The flooding time synchronization protocol. In: *Proceedings of the 2nd Conference on Embedded Networked Sensor Systems (SenSys)*, Baltimore, Maryland, USA, 2004. 39-49
- [10] Bar-Shalom Y, Li X R, Kirubarajan T. *Estimation with Applications to Tracking and Navigation: Theory, Algorithms, and Software*. New York: John Wiley & Sons, 2001. 373-374

Distributed target tracking in asynchronous wireless sensor networks

Wan Jiangwen, Xue Hao*, Yu Ning, Chen bin*

(School of Instrument Science and Opto-electronics Engineering, Beijing University of Aeronautics and Astronautics, Beijing 100191)

(* School of Computer Science and Technology, Beijing University of Posts and Telecommunications, Beijing 100876)

Abstract

To reduce the energy consumption of wireless sensor networks during their target tracking, a distributed target tracking method suited to the condition of time asynchronous was proposed. Firstly, dynamic clusters were established according to the distance between each node and the target. The tracking cluster was used as the calculation unit of time. The tracking time calculation in one cluster and sequential transferring among different clusters were implemented by cluster headers. Then, the particle set was separated into some subsets by the parallel particle filter (PPF) algorithm, which were sampled, weighed and resampled in several nodes. Finally, the estimation of local states was implemented by cluster headers through gathering the results uploaded from nodes. The simulation results show that the PPF algorithm has a good tracking accuracy and can reduce communication traffic about 38% compared with the center particle filter (CPF) algorithm.

Key words: wireless sensor network (WSN), target tracking, asynchronous, parallel particle filter (PPF)