

时间异步无线传感器网络的目标跟踪动态成簇算法^①

薛 皓^② 吴银锋* 万江文* 吴佳灵*

(北京邮电大学自动化学院 北京 100876)

(*北京航空航天大学仪器科学与光电工程学院 北京 100083)

摘 要 为了解决时间异步无线传感器网络在目标跟踪时的节点协作管理和跟踪时间配准问题,提出了一种适用于时间异步条件下目标跟踪的动态成簇算法。该方法通过分析目标的无线信号强度和各节点至目标的距离来动态组建跟踪簇,然后依据目标及簇头的通信距离对簇头射频信号的覆盖区域进行功能划分,实现节点对目标的协作跟踪,同时以簇为跟踪时间的计算单元,通过簇内计时和簇间贯序传递的方法实现跟踪时间的配准。仿真实验表明,该算法进行目标跟踪时能有效均衡网络能耗,且具有较好的跟踪精度和系统鲁棒性。

关键词 无线传感器网络(WSN),目标跟踪,时间异步,成簇,事件驱动

0 引 言

无线传感器网络(wireless sensor networks, WSN)由大量具有感知、通信和数据处理能力的传感器节点以多跳自组织方式构成,应用于目标跟踪时具有部署方便、成本低廉、跟踪可靠和精度高的特点。能量受限是 WSN 的重要特征之一^[1],如何降低网络能耗是设计目标跟踪系统时必需考虑和解决的关键问题。基于无线传感器网络的目标跟踪实质上是一个节点间的协作跟踪过程,目前已提出了一些时间同步条件下的跟踪算法,如二元检测(binary-detection)协作跟踪^[2]、信息驱动协作跟踪(information driven sensory query, IDSQ)^[3,4]和动态传送树协作跟踪(dynamic covey tree-based collaboration, DCTC)^[5,6]。然而时间同步过程要求节点间周期性地交换时间参考分组,这样就不可避免地增大了跟踪系统的通信能耗。因为单位信息的通信功耗比计算功耗高出 2~3 个数量级^[7],而时间异步条件下的无线传感器网络目标跟踪技术恰好能够以有限增加的计算量来减小通信开销,从而达到降低网络总体能耗的目的。

目前,在时间异步无线传感器网络目标跟踪研究方面,Lin^[8]、Li^[9]、Rafati^[10,11]和 Vemula^[12]等对测量建模和轨道估计方法进行了探索性的研究,并取

得了一些理论成果,但针对节点如何有效协同工作以及处理跟踪时间问题的研究尚少,为此本文提出了一种时间异步条件下的无线传感器网络目标跟踪动态成簇算法,并对算法的性能进行了仿真验证和讨论。

1 问题描述

二维区域中的单目标跟踪由随机分布的多个节点协作实现,且各节点的本地时钟为异步。被跟踪目标在移动过程中周期发送射频信号(以下简称目标信号),处于活动状态的传感器节点接收到目标信号后,触发簇构建机制,形成动态跟踪簇。簇头节点(cluster header, CH)单跳通信范围内的所有传感器节点被唤醒,部分节点开始对目标进行跟踪,其余节点则进入监听状态(伪跟踪状态),为簇结构调整做准备。跟踪事件的起始时间由网内首个簇头 CH₀标定。簇内的跟踪时间由 CH 计算,CH 间进行跟踪时间的传递和更新。

目标跟踪过程如图 1 所示。目标信号的有效传输半径为 R_T ,传感器节点的通信半径为 R_N ,且满足 $R_T \leq R_N/2$,这样就保证了所有能探测到目标的传感器节点,都在彼此的通信范围之内,可以直接进行相互通信。目标沿着轨迹在跟踪区域行进时,轨

① 北京市教育委员会共建项目专项资助。

② 男,1976 年生,博士;研究方向:无线传感器网络;联系人, E-mail: xkue@163.com
(收稿日期:2008-07-07)

迹附近区域的传感器节点被唤醒,并形成动态跟踪簇,其余节点则仍处于休眠状态。CH 在目标移动轨迹附近距离目标最近的节点中产生并依次传递(如图中空心箭头所示)。

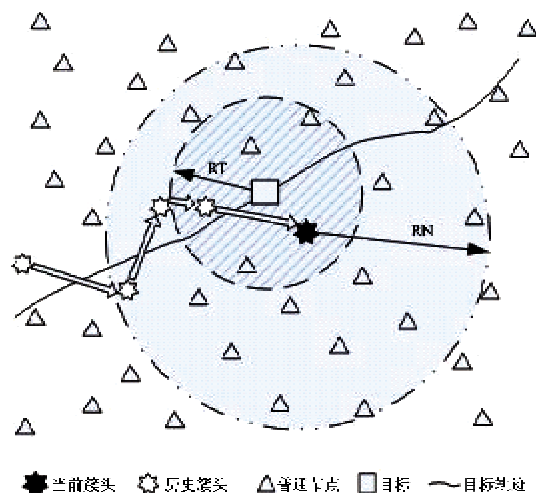


图1 目标跟踪过程示意图

2 面向跟踪的动态成簇算法

动态成簇算法的簇构建、簇结构调整等机制均由目标运动事件触发。网内节点有以下几种状态:空闲状态、休眠状态和工作状态,其中工作状态按角色又可分为簇头状态、候选状态和成员状态。节点按所处区域可分为跟踪节点、伪跟踪节点和空闲节点。图2所示为动态成簇算法的基本工作流程。算法包括的关键步骤有CH选择、跟踪簇形成、簇结构调整和CH转移等。

2.1 跟踪簇的构建

2.1.1 CH选举及初始跟踪时间标定

本文采取一种基于无线信号强度(radio signal strength, RSS)的簇头选择机制^[13]。网络初始化完成后,所有节点都处于空闲状态。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,每个节点根据接收到的目标信号RSS值,启动一个选举定时器(RSS值越大,定时器越短),并进入候选状态,同时设置自身状态为CH。如节点在定时器超时前未收到任何跟踪时间标定通告,则广播一个跟踪时间标定通告。否则,如果节点在定时器超时前收到其他节点的跟踪时间标定通告信息,它将中止自己的选举定时器,并进入成员状态,成为子节点。

CH₀发布跟踪时间标定信息同时,记录当前时

刻的本地时钟,即标定目标跟踪事件从此时刻开始计时,此后的跟踪时间序列在此基础上进行更新。

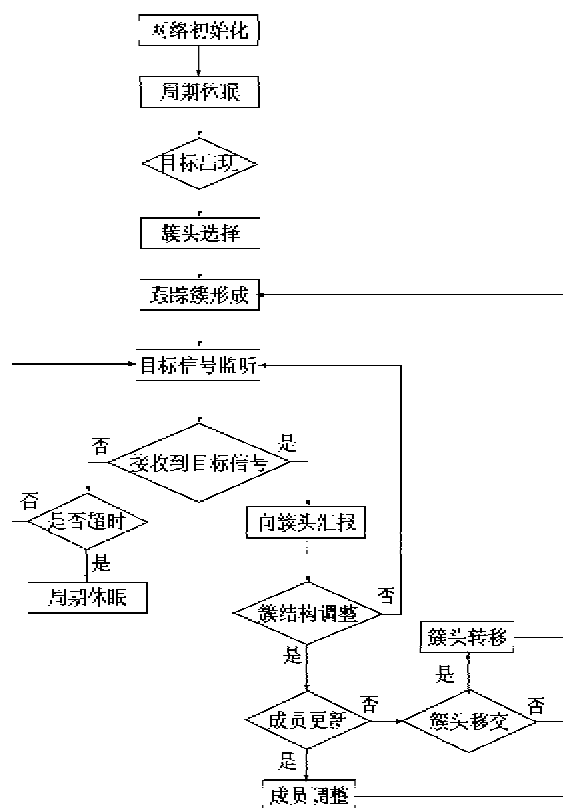


图2 动态成簇算法基本工作流程

2.1.2 成员招募和伪跟踪区域生成

首先做如下定义:

跟踪区域:同时处于CH和目标的有效通信范围的区域。

伪跟踪区域:CH有效通信范围内除去跟踪区域的其他区域。

设 D_i 为节点 i 到目标的欧氏距离。 $D_i < R_T$,则表示节点能正确接收到目标信号,即节点(该类节点以下简称为跟踪节点)处于跟踪区域; $R_T < D_i < R_N$,则表示节点不能接收到目标信号,但能接收到CH的信号,也就是说节点(该类节点以下简称为伪跟踪节点)处于伪跟踪区域。伪跟踪节点接收到CH跟踪时间标定通告后,启动伪跟踪定时器(定时长度比选举定时器长)。

收到CH的跟踪时间标定通告的节点 i 将进行以下判断和操作:

(a) 如果处于候选状态,则退出该状态,同时停止选举定时器,立刻进入成员状态。

(b) 如果处于空闲状态,则转入伪跟踪状态,启

动伪跟踪定时器开始监听目标信号。

(c) 如果没有接收到跟踪时间标定通告,直接进入休眠状态。

进入成员状态的节点启动跟踪定时器,开始监听目标信号。一旦收到目标信号,子节点将 D_i 、节点 ID 及跟踪定时器当前值发送至 CH,在 CH 处进行目标位置估计和时间标定。时间标定方法将在 2.3 节讨论。

2.2 簇结构调整

2.2.1 成员状态调整

子节点状态将随着目标的移动而动态变化。部分跟踪节点接收到的目标信号越来越弱,甚至信号消失;部分伪跟踪节点则开始能够接收到目标信号,且信号越来越强。跟踪节点每接收到一个目标信号,立刻启动跟踪定时器,如果定时器超时前仍未接收到下一个目标信号,则认为自身已经在目标的有效通信范围之外。如果连续一段时间(例如 5 个跟踪定时器周期)接收不到目标信号,该节点转化为伪跟踪节点,并向 CH 发送退出通告。CH 接收到节点的退出通告,将在跟踪成员列表中将其删除。伪跟踪节点在伪跟踪定时器超时前接收到目标信号,则转化为跟踪节点。CH 接收到新加入的跟踪节点信息,则将其 ID 加入跟踪成员列表。

CH 移交时跟踪簇的范围随之改变。伪跟踪节点在伪跟踪定时器超时前未能接收到目标信号,则进入周期休眠状态。部分处于周期休眠状态的节点,收到了 CH 的目标跟踪时间标定通告后,则转入伪跟踪状态。

2.2.2 簇头移交

随着目标的移动,CH 接收到的目标信号强度逐渐变弱,直到无法收到目标信号。本算法中,选择 CH 的原则是倾向于靠近目标的节点,这样,成员节点就最大可能分布在 CH 周围。 D_{CH}^i 为第 i 个 CH 到目标的欧氏距离,CH 转移条件为

$$D_{CH}^i \geq \epsilon R_N, i \in N \quad (1)$$

其中 ϵ 为簇头移交因子,取值在 0.3~0.5 之间。

每次收到被跟踪目标的信号,CH 计算当前时刻自身与目标之间的距离,如果满足转移条件,则指定当前离目标最近的子节点为下一个跟踪簇的 CH,并广播 CH 转移通告信息,该信息包含被指定节点的 ID 和最新的当前目标跟踪时间信息。接收到指定节点的应答信息后,CH 退出簇头状态,转入成员状态。

CH 移交过程如下:CH 移交机制启动后,指定数

据库中 D_{MEM}^i 最小的节点作为下一个 CH,并广播 CH 移交通告信息。接收到簇头移交消息的成员节点,如果 ID 号匹配,该该成员节点进入 CH 状态,并记录目标跟踪时间,开始周期性地广播跟踪时间标定通告;否则将转入选举状态。

2.3 目标跟踪时间标定、计算及传递

时间异步的无线传感器网络中,没有时间同步机制的支持,各节点的本地时间各不相同。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,以下分别进行描述。

2.3.1 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。 CH_0 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。

2.3.2 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 CH。在 CH 处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。

2.3.3 簇间跟踪时间贯序传递

CH 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 CH_0 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 CH 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:

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

其中 $T_0 = P$ 。

Sink 节点的目标跟踪时间配准按式

$$T_r = T_i + D \quad (3)$$

进行,其中 T_r 为修正过的目标跟踪绝对时间。

3 算法仿真及讨论

建立二维机动目标运动模型

$$\mathbf{x}_k = \Phi \mathbf{x}_{k-1} + \Gamma \omega_k \quad (4)$$

这里 $\mathbf{x}_k = (x, \dot{x}, y, \dot{y})^T \in \mathbf{R}^4$, 为目标状态向量, 其中 $[x, y]^T \in \mathbf{R}^2$ 为目标的位置, $[\dot{x}, \dot{y}]^T \in \mathbf{R}^2$ 为目标在时刻 t 的速度向量。 Φ 为状态转移矩阵, Γ 为噪声输入矩阵, 其值分别为

$$\Phi = \begin{bmatrix} 1 & 1 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 1 \\ 0 & 0 & 0 & 1 \end{bmatrix}, \quad \Gamma = \begin{bmatrix} 0.5 & 0 \\ 1 & 0 \\ 0 & 0.5 \\ 0 & 1 \end{bmatrix}$$

$\omega_k = (\omega_x, \omega_y)^T$ 为零均值高斯白噪声, 协方差为 $\mathbf{Q}$: $E[\omega_k, \omega_j^T] = \mathbf{Q} \delta_{j,k}$, 其中 $\mathbf{Q} = q \mathbf{I}_{2 \times 2}$ 是 2×2 单位矩阵, $\sqrt{q} = 0.1$ 。被跟踪目标的初时状态为 $\mathbf{x}_0 = (10, 0.5, 90, -0.5)^T$ 。表 1 为仿真使用的网络参数值。

表 1 仿真参数

参数	描述	取值
N	监控区域节点数	100, 150, 200
R_N	节点通信半径	30m, 35m, 40m
R_T	目标通信半径	15m
ϵ	簇头移交因子	0.3~0.5
T	仿真时间	100s
t	采样周期	1s

图 3 给出了 $N = 150, R_N = 35, \epsilon = 0.4, T = 50$ 时跟踪区域内的目标跟踪情况。此时跟踪区域内共有 43 个活动节点, 其中跟踪节点 11 个, 伪跟踪节点 31 个。图 4 为利用本实验室研究的加权最小二乘定位算法^[14]对目标位置进行估计的结果。当节点数由 100 增加到 200 时, 定位误差显著减小, 由 9% 降低到 5%。这是由于节点数增加, 使 CH 收到了更

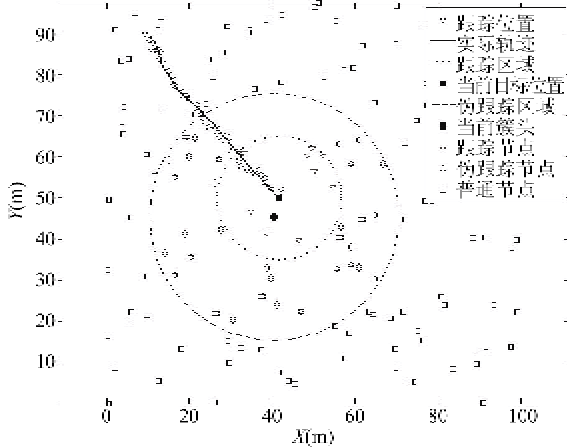


图 3 跟踪过程仿真

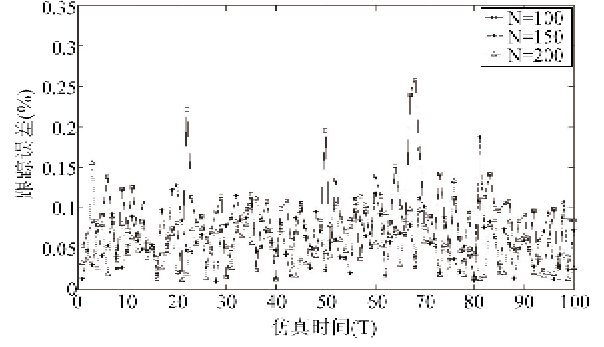


图 4 跟踪误差

多的有效距离信息。但值得注意的是, 更大的网络节点密度意味着更多的冗余节点及更严重的数据冲突威胁, 因此在系统设计时应权衡跟踪精度需求与节点密度, 优化网络部署方案。

跟踪过程中, 处于跟踪状态的节点的多少对跟踪效率和跟踪精度有着重要影响。如果冗余节点过多, 则容易在 CH 处造成严重的数据冲突, 增加丢包率; 如果跟踪节点过少, 则会影响定位精度, 甚至造成定位失败。图 5 给出了 $R_N = 35, \epsilon = 0.4$ 时跟踪节点的平均个数。不同节点密度条件下, 簇内处于跟踪状态的节点数约占节点总数的 7%。这样的数量分布, 在满足跟踪精度的同时, 保证了有个别节点发生故障时仍能对目标进行有效跟踪, 且不致引起严重的数据冲突。

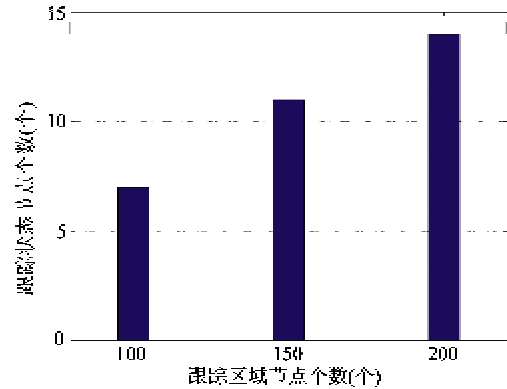


图 5 跟踪状态节点数

簇头转换次数是影响算法稳定性的关键因素之一。频繁的簇头转换会导致通信量增加, 算法不稳定, 甚至跟踪失败。监测区域内节点密度和 CH 移交的条件阈值是 CH 数量的决定因素。图 6 给出了不同节点布撒数量和簇头移交因子条件下的簇头转

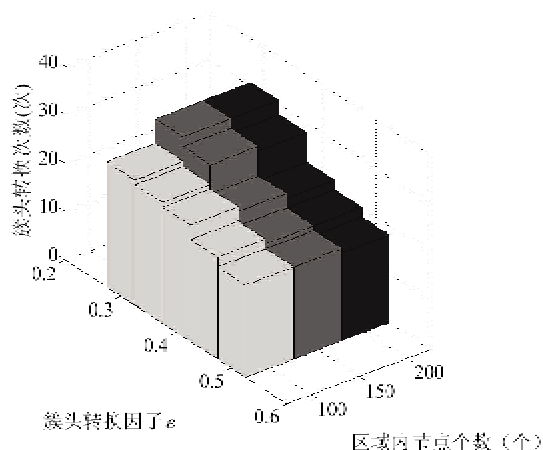
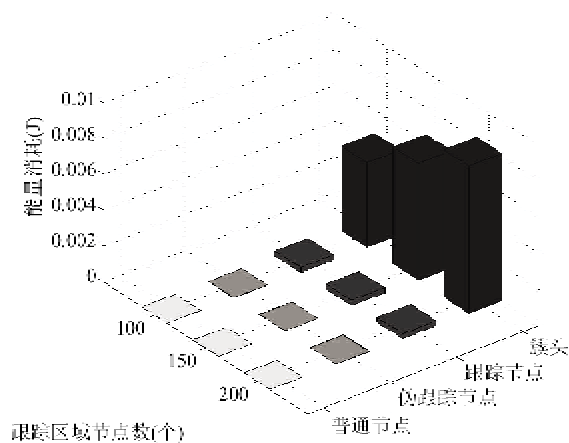
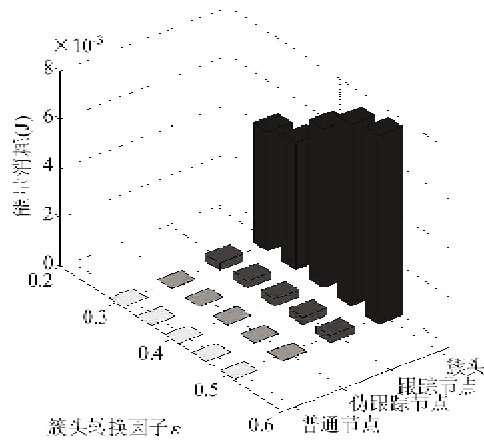


图6 CH转换次数

移次数。随着 ϵ 的增大,三种节点布撒数量条件下的CH转换次数均趋于减少,且簇头移交因子较小



(a) 节点数与节点能量消耗关系



(b) 簇头转换因子与节点能量消耗关系

图7 节点能量消耗

4 结论

结合无线传感器网络在目标跟踪中的应用需求,提出了一种时间异步条件下的目标跟踪动态成簇算法。该方法通过高效的节点协同工作以及跟踪簇重构机制实现了对目标的有效跟踪,同时较好地平衡了网络节点的负载,延长了跟踪传感器网络的生命期。仿真实现了目标跟踪的过程,在此基础上分析了算法的性能,结果表明,该算法具有较好的跟踪精度和系统鲁棒性,可满足目标跟踪的应用需求。

参考文献

[1] Estrin D, Govindan R, Heidemann J, et al. Next century

(阈值小)时,节点数为100的情况下的CH转移次数明显少于其他两种条件下的转换次数。值得注意的是, $\epsilon > 0.4$ 时,三种节点密度条件下CH转移次数基本相等。由此可见,选择较大的 ϵ 值,有利于降低CH转换次数,增加系统稳定性和鲁棒性。

图7给出了跟踪过程中节点的平均能量消耗。如图所示,由于CH承担了主要的通信、计算任务,能量消耗最大,且随着节点密度和 ϵ 值的增大而增加。跟踪节点负责接收目标信息及向CH报告目标信息,因此能量消耗较少。处于伪跟踪状态的节点只进行短时间的不间断监听,能量消耗很少。空闲节点处于休眠状态,能量消耗几乎可以忽略。虽然簇头能量消耗较多,但高能耗状态持续时间短,且由不同节点依次承担,这样就从总体上平衡了网络的能耗,有效延长了网络的生存期。

challenges: scalable coordination in sensor networks. In: Proceedings of the ACM/IEEE International Conference on Mobile Computing and Networking, Seattle, USA, 1999. 263-270

- [2] Mechito K, Sundresh S, Kwon Y, et al. Cooperative tracking with binary-detection sensor networks. In: Proceedings of the 1st International Conference on Embedded Networked Sensor Systems (SensSys'03), Los Angeles, USA, 2003. 5-7
- [3] Zhao F, Shin J, Reich J. Information-driven dynamic sensor collaboration. *IEEE Signal Processing Magazine*, 2002, 19(2): 61-72
- [4] Kumar S, Zhao F, Shepherd D. Collaborative signal and information processing in micro-sensor networks. *IEEE Signal Processing Magazine*, 2002, 19(1): 13-14
- [5] Zhang W S, Cao G H. An energy efficient framework for mo-

- hile target tracking in sensor networks. In: Proceedings of the IEEE Military Communications Conference, Monterey, USA, 2003. 597-602
- [6] Zhang W S, Cao G H. DCTC: Dynamic convoy tree based collaboration for target tracking in sensor networks. *IEEE Transactions on Wireless Communication*, 2004, 3(5): 1689-1701
- [7] Raghunathan V, Schurgers C, Park S, et al. Energy-aware wireless microsensor networks. *IEEE Signal Processing Magazine*, 2002, 19: 40-50
- [8] Lin X D, Bar-Shalom Y, and Kirubarajan T. Multisensor-multitarget bias estimation for general asynchronous sensors. *IEEE Transactions on Aerospace and Electronic Systems*, 2005, 41(3): 899-921
- [9] Li T, Ekpenyong A, Huang Y F. A location system using asynchronous distributed sensors. In: Proceedings of IEEE Asia-Pacific Conference on Circuits and Systems, Taiwan, China, 2004. 2. xvii-xvii
- [10] Rafati A, Moshiri B, Salahshoor K, et al. Asynchronous sensor bias estimation in multisensor-multitarget systems. In: Proceedings of the IEEE International Conference on Multi-sensor Fusion and Integration for Intelligent Systems, Heidelberg, Germany, 2006. 402-407
- [11] Rafati A, Moshiri B, Rezaei J. A new algorithm for general asynchronous sensor bias estimation in multisensor-multitarget systems. In: Proceedings of the 10th International Conference on Information Fusion, Quebec, Canada, 2007. 1-8
- [12] Vemula M, Miguez J, Artes-Rodriguez A. A sequential monte carlo method for target tracking in an asynchronous wireless sensor network. *Positioning, Navigation and Communication*, 2007: 49-54
- [13] 申兴发,李鸿斌,赵军等.面向目标跟踪的传感器网络分布式组管理机制.仪器仪表学报,2007,28(6):966-972
- [14] 于宁,万江文,冯仁剑.无线传感器网络定位求精方法.高技术通讯,2008,18(10):1017-1022

A dynamic clustering algorithm for target tracking in asynchronous wireless sensor networks

Xue Hao, Wu Yinfeng^{*}, Wan Jiangwen^{*}, Wu Jialing^{*}

(Automatization School, Beijing University of Posts and Telecommunications, Beijing 100876)

(* School of Instrument Science and Opto-electronics Engineering,

Beijing University of Aeronautics and Astronautics, Beijing 100191)

Abstract

In order to solve the problems of node cooperation and tracking time registration in target tracking of wireless sensor networks, a dynamic clustering algorithm on the condition of time asynchronous is proposed. The algorithm dynamically establishes tracking-clusters by analyzing the wireless signal strength of the target and distances from target to nodes, and then restructures the covered area of the RF signal for functional partition by the cluster header according to the communication distance between the target and the cluster header. The algorithm uses the cluster as the unit for calculation of tracking time, and the tracking time is registered by timing in cluster and sequence transferring between different clusters. The simulation results show that the algorithm can proportion the energy consumption of network effectively, and has the good performance in tracking accuracy and system robustness.

Key words: wireless sensor network (WSN), target tracking, asynchronous, clustering, event-driven