

文章编号: 1001-0920(2011)03-0393-04

基于IDSQ的自适应动态协同自组织算法

陈延军, 潘 泉, 梁 彦, 魏雅川

(西北工业大学 自动化学院, 西安 710072)

摘 要: 在无线传感器网络中传感器节点相互协同完成感知任务, 以传感器量测的信息效用与获取量测的能量消耗来折中地选择参与节点, 但用户对服务质量(QoS)的需求并没有在节点选择时得以体现. 为此, 提出一种自适应动态协同自组织算法(A-DCS). 该算法能根据用户给定的精度需求, 自适应地选择簇首, 确定参与感知任务的簇成员顺序和个数, 并计算相应的能量消耗. 以目标跟踪为应用背景的仿真结果表明, 在跟踪精度和能量消耗 2 个指标下, 该算法优于信息驱动传感器查询(IDSQ)和动态协同自组织(DCS).

关键词: 无线传感器网络; 信息效用; 自组织; 目标跟踪

中图分类号: TP393

文献标识码: A

Adaptively dynamic collaborative self-organization algorithm based on IDSQ

CHEN Yan-jun, PAN Quan, LIANG Yan, WEI Ya-chuan

(College of Automation, Northwestern Polytechnical University, Xi'an 710072, China. Correspondent: CHEN Yan-jun, E-mail: chenyanjun@mail.nwpu.edu.cn)

Abstract: Sensors should be selected elaborately to fulfil the sensing task in wireless sensor networks, sensor selection is based on the tradeoff between information utility achieved by the sensor measure and energy overhead to fetch the measure. But the users' requirements for quality of service(QoS) aren't yet involved in the process of sensor collaboration. Therefore, an algorithm, adaptively dynamic collaborative self-organization(A-DCS), is proposed. According to the given accuracy, the cluster header can be select and the sequence and number of cluster member can be made adaptively, and the corresponding energy consumption is computed. This algorithm is then adapted to the specific application of target tracking. Simulation results show that this algorithm is superior to information driven sensor querying(IDSQ) and dynamic collaborative self-organization(DCS) based on the tracking accuracy and the energy consumption.

Key words: wireless sensor networks; information utility; self-organization; target tracking

1 引 言

无线传感器网络^[1-2]中节点的固有特性要求节点之间必须协同完成某种感知功能, 节点在感知区域的无规则散布决定了量测的空间分布特性, 在未知节点量测情况下必须选择一个最优的节点量测集合有序地参与状态估计. Zhao 等人^[3-4]提出了 IDSQ 算法, 该算法以信息效用函数和代价函数为基础建立目标函数, 以目标函数最优为原则选择节点参与状态估计; Zuo 等人^[5-6]则提出以状态估计误差的 CRLB 作为节点的选择标准. 然而, 以上算法均未涉及给定精度阈值时的节点协同问题. 为此, 文献 [7-9] 提出了 DCS 算

法, 该算法强调代价函数在节点选择中的重要性, 通过改变参与状态估计的节点数目使估计的状态达到预先设定的精度. 由于该算法中用于判断待估计量精度的参数与采用的信息效用函数有关, 且算法没有给出具体的设定指标, 每一拍在选择簇成员的同时又重新判断簇首是否切换, 大大地增加了算法的复杂度, 使得该算法很难适用于无线传感器网络中的状态估计问题. 因此, 在用户设定精度阈值的情况下, 迫切需要一个轻量级的协同算法, 以解决节点选择问题和状态估计问题.

本文基于 IDSQ 提出一种自适应动态协同自组

收稿日期: 2009-12-07; 修回日期: 2010-04-09.

基金项目: 国家自然科学基金重点项目(60634030); 国家自然科学基金项目(60702066); 航天科技创新基金项目(CASC0214).

作者简介: 陈延军(1977-), 男, 博士生, 从事传感器网络、统计信号处理等研究; 潘泉(1961-), 男, 教授, 博士生导师, 从事多目标跟踪、信息融合等研究.

织算法 (A-DCS). 该算法首先判断上一时刻的簇首节点是否切换, 如需切换, 则在其成员节点中选择新的簇首节点, 并基于此簇首生成该时刻新的候选成员节点, 根据设定的估计精度自适应选择参与状态估计的节点顺序和数目. 该算法弥补了 IDSQ 算法缺乏控制状态估计精度的不足, 而且解决了 DCS 算法的参数敏感性和高复杂度问题, 能够满足系统实时性的要求, 适用于传感器网络中的分布式动态估计问题.

2 算法介绍

该算法采用的目标函数为

$$\Theta_{\text{obj}}(i, j) = \gamma \varphi_{\text{utility}}(p(x^{(t+1)} | \bar{Z}^{(t)}, z_j^{(t+1)})) - (1 - \gamma) \phi_{\text{cost}}(\lambda_i, \lambda_j). \quad (1)$$

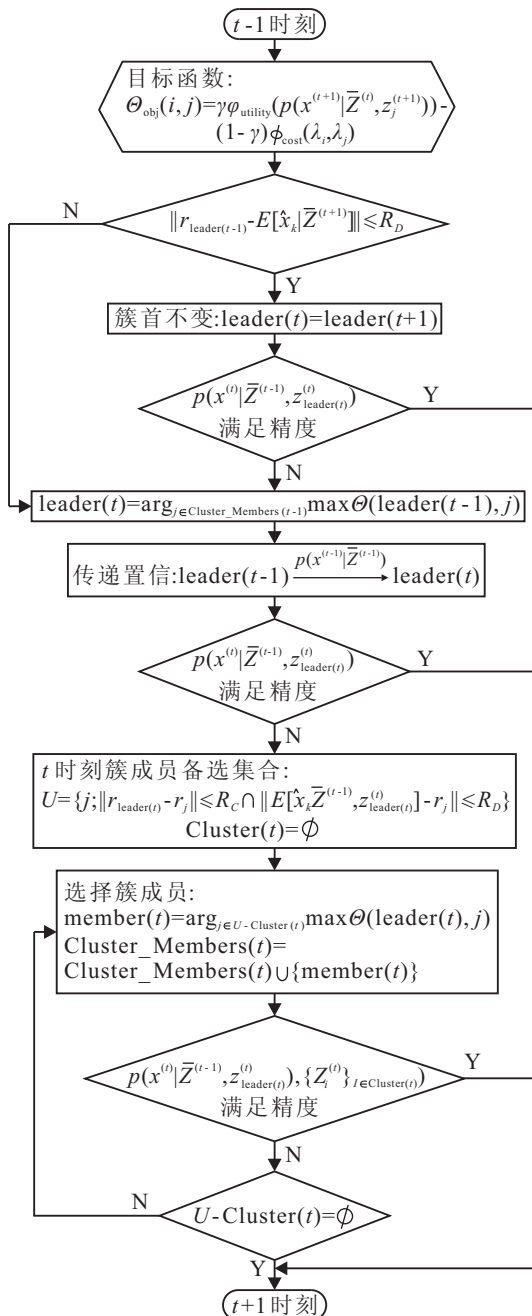


图 1 A-DCS 流程图

式中: λ_i, λ_j 表示节点 i, j 的节点特性, 例如节点的位置以及封装在节点的传感器类型等; $\bar{Z}^{(t)}$ 表示时刻 t 以前的量测集合; $\varphi_{\text{utility}}(p(x^{(t+1)} | \bar{Z}^{(t)}, z_j^{(t+1)}))$ 为利用 j 节点在 $t+1$ 时刻的量测给该时刻的状态估计带来的信息效用; $\phi_{\text{cost}}(\lambda_i, \lambda_j)$ 表示 $t+1$ 时刻 i 节点为了得到 j 节点的量测所带来的系统开销, 例如网络带宽和能量消耗等; γ 表示信息效用和代价的相对权重.

算法的具体流程如图 1 所示. 其中: $\text{Leader}(t)$ 为 t 时刻的簇首节点, $\text{Cluster_members}(t)$ 为 t 时刻的簇成员节点集合, r_i 为节点 i 的空间坐标, R_C 和 R_D 表示节点的通讯半径和感知半径.

3 算法性能分析

本文以目标跟踪为应用背景, 设定封装在节点传感器板上的传感器采用声音传感器和红外传感器.

3.1 传感器模型的确定

3.1.1 声音传感器

声波的传播实质上是能量的传递, 在各向同性的介质中, 声音幅度衰减模型为

$$z = \frac{A_0}{r^\alpha}. \quad (2)$$

式中: A_0 为距声源 1 m 处声波的振幅; r 为声波的传输距离; α 为振幅衰减系数, 理想情况下 $\alpha = 1$, 在工程实际中考虑到外界环境的影响, $\alpha = 0.5 \sim 1.5$. 由此, 声音传感器的传输模型为

$$z_i(t) = g_i \frac{A(t - t_{ni})}{\|\mathbf{x}_s(t - t_{ni}) - \mathbf{x}_i\|^\alpha} + v_i(t). \quad (3)$$

式中: $z_i(t)$ 为传感器 i 在时刻 t 测得的声音振幅, g_i 为传感器的增益, \mathbf{x}_i 为传感器的空间位置, t_{ni} 为声波的传输时间, \mathbf{x}_s 为声源的空间位置, $v_i(t)$ 为量测噪声 (通常为高斯白噪声). 本课题组通过实验验证了该模型的合理性, 并求得 $\alpha = 0.896$. 假设目标的初始声音振幅 A_0 服从区间为 $[50, 100]$ 的均匀分布.

3.1.2 红外传感器

IR 传感器工作时不发射电磁波, 它接受目标自身产生或反射其他能源的热辐射而得到量测信号, 主要提供目标角度数据和红外辐射特性信息. 由此, IR 传感器模型为

$$\theta_i(t) = \arctan\left(\frac{y_s - y_i}{x_s - x_i}\right) + v_i(t). \quad (4)$$

式中: $\theta_i(t)$ 为传感器 i 在时刻 t 测得的方位角; (x_s, y_s) 为目标在时刻 t 的坐标, 传感器 i 的坐标为 (x_i, y_i) ; $v_i(t)$ 为量测噪声, 通常为高斯白噪声.

3.2 算法参数的选取

节点动态自组织算法中信息效用函数为

$$\varphi_{\text{Utility}}(p(x^{(t)} | \bar{Z}^{(t-1)}, z_j^{(t)})) = (x_{sj} - \hat{x}^{(t|t-1)})' \text{Cov}^{-1}(\hat{x}^{(t|t-1)})(x_{sj} - \hat{x}^{(t|t-1)}). \quad (5)$$

式中: $x^{(t)}$, $\bar{z}^{(t-1)}$ 和 $z_j^{(t)}$ 含义如前文所述, x_{s_j} 为传感器的空间坐标, $\hat{x}^{(t|t-1)}$ 为目标位置在 t 时刻的预测. 因为均在节点的通讯半径内选取节点, 所以假定每个节点的代价相同. 节点传感器的探测半径 $R_D = 25$ m, 节点的通信半径 $R_C = 50$.

3.3 评价指标

1) 跟踪精度指标. 设目标状态真值为 X , 估计值为 \hat{X} . $X_{j,k}$, $\hat{X}_{j,k}$ 分别代表第 j 次仿真中目标在 k 时刻的状态, 用均方根误差衡量目标的定位精度, 即

$$\text{RMSE} = \frac{1}{N_1} \sum_k \left[\frac{1}{N_s} \sum_{j=1}^{N_s} d(X_{k,j}, \hat{X}_{k,j}) \right]. \quad (6)$$

2) 能量指标. 传感器网络中的能量消耗包括节点信息感知、数据处理以及节点之间的数据通讯(数据包的发送与接收). 本文所用的能量模型基于文献[10]. 如果传感器节点 j_1 感知并处理 b 字节的信息, 则所消耗的能量为

$$E_s(j_1, b) = e_s b, \quad (7)$$

式中 e_s 由传感器节点 j_1 的规格决定. 如果节点 j_1 给一跳邻居节点 j_2 发送 b 字节大小的信息包, 则发送端消耗的能量为

$$E_t(j_1, j_2, b) = (e_t + e_d r_{j_1, j_2}^\alpha) \times b. \quad (8)$$

式中: e_t 和 e_d 由节点 j_1 发射模块参数决定, 节点 j_1 与 j_2 之间的欧氏距离为 r_{j_1, j_2} , α 由信道的特征决定. 接收端的能量消耗为

$$E_r(j_2, b) = e_r b, \quad (9)$$

式中 e_r 由节点 j_2 的接收模块决定.

综上所述, 整个感知、处理、通信过程所消耗的总能量为

$$\begin{aligned} E_a(j_1, j_2, b) = \\ E_s(j_1, b) + E_t(j_1, j_2, b) + E_r(j_2, b) = \\ e_0 + e_1 r_{j_1, j_2}^\alpha. \end{aligned} \quad (10)$$

式中: $e_0 = (e_s + e_t + e_r)b$, $e_1 = e_d b$.

为处理问题方便, 忽略了节点处理空闲时所消耗的能量. 仿真中: $\alpha = 2$, $e_s = 50 \times 10^{-6}$, $e_t = 45 \times 10^{-6}$, $e_r = 135 \times 10^{-6}$, 以上各物理量单位均为 J/bit; $e_d = 10 \times 10^{-9}$ mJ/(bit·m²), $b = 1024$. 因此, $e_0 = 0.23552$ mJ, $e_1 = 1.024 \times 10^{-4}$ mJ/m².

3.4 仿真场景

感知区域大小为 $800 \text{ m} \times 800 \text{ m}$, 在此区域内随机撒布 500 个节点, 在每个节点加载声音传感器和红外传感器.

t 时刻的状态为

$$x^{(t)} \triangleq [x^{(t)}, v_x^{(t)}, y^{(t)}, v_y^{(t)}]^T,$$

目标作匀速直线运动, 其状态方程为

$$x^{(t+1)} = Ax^{(t)} + v^{(t)}. \quad (11)$$

式中: A 为状态转移矩阵, $v^{(t)}$ 服从协方差矩阵为 Q 的高斯白噪声.

$$A = \begin{bmatrix} 1 & T & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & T \\ 0 & 0 & 0 & 1 \end{bmatrix},$$

$$Q = q \times \text{diag} \left(\begin{bmatrix} T^3/3 & T^2/2 \\ T^2/2 & T \end{bmatrix}, \begin{bmatrix} T^3/3 & T^2/2 \\ T^2/2 & T \end{bmatrix} \right).$$

其中: T 为采样间隔, q 决定过程噪声的强度. 目标初始状态为 $[-300, 10, -300, 10]^T$, 状态估计的滤波算法采用 EKF, 以每一时刻的簇首为融合中心.

3.5 结果分析

IDSQ 算法在每个时刻选择单个节点作为融合中心, 接收驻留在上一时刻融合中心中目标的状态置信并用其量测更新(见图2), 簇首频繁切换而且没有利用邻居节点提供的量测信息, 因此状态估计精度较差. 同时, 由于只选择了一个节点, 算法缺乏鲁棒性. 相比 IDSQ 算法, A-DCS 算法能根据用户自定义的跟踪精度自适应动态选择簇首和簇成员, 簇首的切换次数相比 IDSQ 要小很多, 既可以避免 IDSQ 在选择节点时的“乒乓现象”, 也能减小由于传递目标状态所消耗的能量(见图3). A-DCS 簇首切换次数以及每一时刻簇成员的数目由用户设定的跟踪精度阈值决定, 为

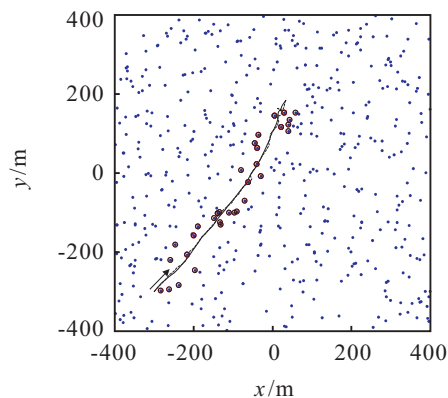


图2 节点协同目标跟踪(IDSQ)

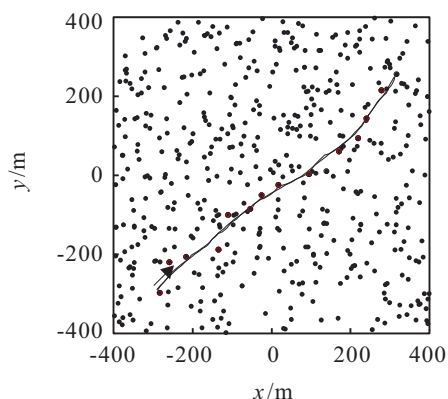


图3 节点协同目标跟踪(A-DCS)

表 1 簇首更新次数和簇成员平均个数对比

协同策略	设定跟踪精度 Accuracy	簇首切换次数	平均簇成员数
A-DCS	0.5	41	1.50
A-DCS	1.0	28	0.90
A-DCS	2.0	23	0.17
A-DCS	4.0	18	0.08
A-DCS	6.0	16	0.05
A-DCS	8.0	15	0.02
A-DCS	10.0	15	0.00
DCS	不能直观的设定	45	2.10
IDSQ	无此功能	60	0.00

验证本文算法, 仿真 1000 次, 统计簇首切换次数和平均的簇成员数如表 1 所示。

从表 1 可以看出: 本文算法的簇首切换次数以及平均簇成员数随着用户设定精度发生变化, 给定精度越高, 簇首切换越频繁, 平均的簇成员数越多, 反之亦然。IDSQ 在每个时刻都进行簇首切换, 并没有使用邻居节点的信息, 因此簇首的切换次数明显大于 A-DCS 和 DCS, 而且不具有用户设定跟踪精度的功能。DCS 需要提供的参数很多, 参数与所采用信息效用函数的形式有关, 且缺乏鲁棒性, 若设置不合适, 则程序可能陷入死循环。仿真中

$$[M_0, M_1, M_2, d_0, d_1, d] = [100, 500, 1000, 50, 50, 50].$$

DCS 的跟踪效果与 A-DCS 用户设定 Accuracy = 6 时的跟踪情形相当 (见图 4), 但该算法的簇首切换次数、平均簇成员个数明显大于 A-DCS 算法 Accuracy = 6 时的簇首切换次数和平均簇成员个数。

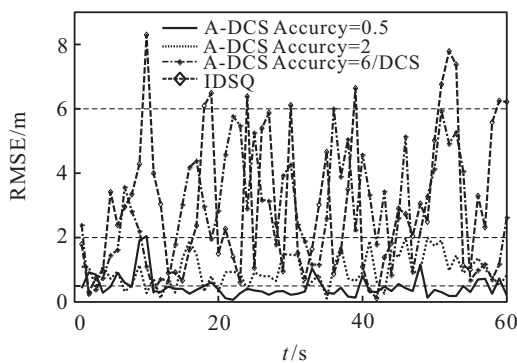


图 4 跟踪精度 RMSE 对比

图 4 中, DCS 与 A-DCS 算法 Accuracy = 6 的跟踪精度相当, 因而只给出了一条 RMSE 曲线。结合图 4 和图 5 可以看出: A-DCS 的跟踪精度 RMSE 随着用户设定的精度而变化, 用户设定的跟踪精度阈值越小, 算法的跟踪精度越高, 相应的能量消耗越大; IDSQ 和 DCS 的跟踪精度与用户设定 Accuracy = 6 的跟踪精度相当, 但其所消耗的能量却与 Accuracy = 0.5 时 A-DCS 所消耗的能量相当, 因此 A-DCS 既可以提高目标跟踪的精度, 也可以适当放松精度需求, 使能量消耗较小。

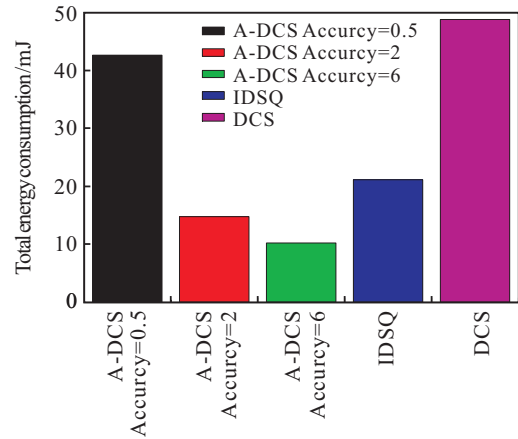


图 5 能量消耗对比

4 结论与展望

综上所述, A-DCS 可根据用户对目标跟踪精度的需求自适应地选择簇首, 决定簇成员参与状态估计的顺序和簇成员个数, 最大限度地利用邻居节点所提供的信息, 同时可避免 IDSQ 在选择节点时的“乒乓现象”。与 DCS 算法相比, 该算法在满足相同状态估计精度的同时, 具有较小的计算复杂度, 用户可根据对 QoS 的需求, 在估计精度和能量消耗的双重指标下寻求折中。该算法适用于无线传感器网络背景下的其他状态估计问题。

A-DCS 算法尽管可根据用户设定的精度动态地选择簇首和簇成员, 但选择节点时没有考虑节点的剩余能量以及节点之间的空间相关性。同时, 为提高待估计量的精度, 该算法仅以增加簇成员数来序贯地更新估计, 没有考虑通过提高采样率来提高待估计量的精度, 也没有利用节点量测之间的时间相关性。因此, 下一步研究应将节点的剩余能量、节点之间的空间相关性作为节点选择的影响因素, 利用单个节点量测之间的时间相关性, 有效地调节节点的采样率。

A-DCS 算法是集中式, 状态驻留在簇首中, 缺乏鲁棒性。应设计一种强鲁棒轻量级的分布式估计算法, 以适应无线传感器网络节点低功耗、低计算能力、低通讯能力的特点。

参考文献(References)

- [1] Akyildiz I F, Su W L, Sankarasubramaniam Y. A survey on sensor networks[J]. IEEE Communications Magazine, 2002, 40(8): 102-114.
- [2] Akyildiz I F, Su W L, Sankarasubramaniam Y. Wireless sensor networks: A survey[J]. Computer Networks — The Int J of Computer and Telecommunications Networking, 2002, 38(4): 393-422.
- [3] Feng Z, Jaewon S, Reich J. Information-driven dynamic sensor collaboration[J]. IEEE Signal Processing Magazine, 2002, 19(2): 61-72.