

无线传感器网络能量均衡的非均匀分簇算法

卢先领 王莹莹 王洪斌 徐保国

(江南大学物联网工程学院 无锡 214122)

摘要 无线传感器网络节点随机分布,针对均匀分簇容易造成网络中能耗不均的问题,提出一种能量均衡的非均匀分簇算法 EBUCA(Energy-Balanced Unequal Clustering Algorithm)。该算法在簇头选举阶段,根据节点的剩余能量、节点所在区域稀疏程度来保证簇头的均匀分布;同时结合各簇头所在区域的节点密度与距 sink 节点的距离来构造大小不等的簇半径,使节点所在区域密度大或距离基站较近的簇半径较小,平衡了簇内和簇间的通信能耗。仿真结果表明与 LEACH、DBCP、EEUC 算法相比,EBUCA 算法能够有效地均衡节点能耗,延长网络生命周期。

关键词 无线传感器网络,节点密度,能耗均衡,非均匀分簇,生命周期

中图分类号 TP393 **文献标识码** A

Energy-balanced Unequal Clustering Algorithm in Wireless Sensor Network

LU Xian-ling WANG Ying-ying WANG Hong-bin XU Bao-guo

(School of Internet of Things Engineering, Jiangnan University, Wuxi 214122, China)

Abstract The distribution of nodes in wireless sensor network (WSN) is random and equal clustering algorithm arouses energy consumption unbalance. So this paper presented an energy-balanced unequal clustering algorithm (EBUCA). Based on residual energy and density, it selects cluster heads, then forms different size of cluster by density of the cluster heads and the distance to sink, making the clusters with higher density and closer to the base station to have smaller size to achieve the purposes of balancing energy and loads. The simulation results show that compared with LEACH, DBCP, EEUC, the improved algorithm can balance the energy consumption of nodes, and prolong the life-time of the network.

Keywords Wireless sensor network, Density, Energy-balanced, Unequal clustering, Network lifetime

1 引言

无线传感器网络(Wireless Sensor Network, WSN)是一种新型的以数据为中心的无线网络,它由具有传感、计算和通信能力的节点组成。该网络能够协作地实时监测、采集和处理节点分布区域内的各种环境或监测对象的信息,并将处理后的数据传送到网络中的特定位置节点;它已广泛应用于军事、环境、医疗、家庭、商业等许多方面。

网络路由算法的主要目的是选择合适的路径并沿着该路径将数据从源节点发送到目的节点。网络层算法负责路由发现和维护,关系到网络设计的成功与否。由于传感器节点的电源能量、计算和存储能力都十分有限,因此,设计能量高效的路由算法,尽可能延长网络的寿命是 WSN 路由算法设计首要考虑的问题。本文针对分簇路由算法中的“热区”问题,在 LEACH(Low Energy Adaptive Clustering Hierarchy)、DBCP(Density-based Clustering Protocol)、EEUC(Energy-Efficient Unequal Clustering)等算法的基础上提出了一种基于

节点密度的能耗均衡的非均匀分簇算法——EBUCA。

2 相关工作

2.1 LEACH 及其算法分析

LEACH^[1]是 WSN 中最早提出的分簇路由算法。在 LEACH 算法中,每个节点在 0~1 之间随机选择一个数,如果选择的数小于阈值 $T(n)$,则该节点就充当簇头节点。其中 $T(n)$ 的计算如下:

$$T(n) = \begin{cases} \frac{p}{1 - p(r \bmod \frac{1}{p})}, & n \in G \\ 0, & n \notin G \end{cases} \quad (1)$$

式中, p 是节点当选为簇头的概率, r 是目前循环进行的轮数; G 是最近 $1/p$ 轮未当选为簇头的节点集合。由式(1)可知,当选过簇头的节点在接下来的 $1/p$ 轮循环中将不能成为簇头。

针对整个网络中节点的能耗不均问题,文献[2,3]利用节点的剩余能量来改变节点成为簇头的概率,以避免低能量节点成为簇头。文献[4]提出一种改进的多跳均匀分簇路由

到稿日期:2012-07-02 返修日期:2012-11-23 本文受中央高校基本科研业务费专项资金(JUSRP21129),国家自然科学基金项目(30971689)资助。

卢先领(1972-),男,博士,副教授,主要研究方向为无线传感器网络、体域网, E-mail: jnluxl@gmail.com; 王莹莹(1988-),女,硕士生,主要研究方向为无线传感器网络; 王洪斌(1987-),男,硕士生,主要研究方向为无线传感器网络、体域网情景感知; 徐保国(1951-),男,教授,博士生导师,主要研究方向为过程控制、智能仪表及现场总线网络等。

算法,即通过候选簇头的竞选半径和节点剩余能量来确定分布相对均匀的簇头,簇头之间采用以簇头节点剩余能量和链路传输代价的权值建立多跳路由算法。LEACH-CS^[5]以跨区域距离的约束来自定义合适的多跳路由方案。通过均匀分簇、多跳路由在一定程度上有效地平衡了簇头的能量消耗及数据传输的整体能量消耗,提高了能量效率。但是均匀分簇仍然会导致簇负载的不平衡,且距离 sink 节点近的簇头会因转发大量数据而能耗较大。

由此可见,在均匀分簇的网络中无论采用单跳或多跳方式,均存在簇间能耗不均衡的问题。针对这个问题,很多研究人员采用非均匀分簇策略来平衡簇头能耗。EEUC^[6]将非均匀分簇和簇间多跳相结合,利用非均匀的竞争半径,使靠近基站的簇范围较小,从而簇头能够节约能量以供簇间数据转发使用,达到均衡簇头能量消耗的目的^[7]。

ACOU^[8]继承了 EEUC 的非均匀分簇结构,在簇首选举阶段,采用首轮所有节点参与竞选、后续轮簇内调整的方法替代 EEUC 随机激活的周期性簇首选举策略。在簇首间通信阶段,引入链路可靠性和实时性参数,在利用定向扩散的蚁群优化算法传输数据的同时,对路由进行动态维护和性能优化。关于非均匀分簇的簇头选择问题:DBCP^[9]根据节点的密度来控制节点成为簇头的概率,DTUC^[10]从概率的角度进行分层和分簇^[11],利用最优簇头个数和剩余能量来控制节点成为簇头的概率,以上算法通过控制簇头的选择,来达到平衡各簇内节点数目,及平衡网络能耗的目的。

2.2 存在问题分析

已有算法存在以下不足:首先,不能保证每个节点的初始能量是相同的,每个节点(包括能量低的)成为簇头的几率是相同的,这样会导致低能量节点能量很快耗尽,造成网络中节点能量不均衡;其次,无线传感器网络中节点分布具有随机性,而均匀分簇可能导致有些区域产生的簇头数过多或过少;最后,簇头采用单跳的方式与 sink 节点通信,这样离 sink 节点较远的簇头会因消耗的能量过大而过早死亡。并且整个网节点在两跳范围内,不符合大规模网络的要求。图 1 给出了 LEACH、EEUC、DBCP、EBUCA 等算法的成簇策略。

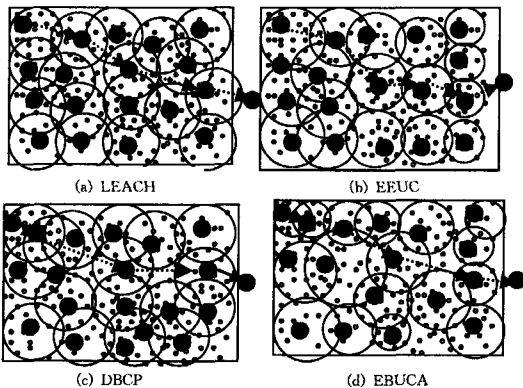


图 1 成簇策略图

如图 1(a)所示,利用 LEACH 算法划分簇,有的簇中节点很多,有的则很少,划分的簇负载不均匀,很容易造成某些节点因能耗过大而过早死亡;利用 EEUC 中所提算法来划分簇半径,如图 1(b)所示,在远离基站且密度较大的区域,由于簇半径较大,簇内节点过多,会造成该区域内簇头节点因能量

耗尽而过早死亡;DBCP 根据节点的密度来选择簇头,在节点密集区增加簇头,在节点稀疏区减少簇头,所有簇采用相同的簇半径,如图 1(c)所示,则会出现各个簇之间重叠部分很大,在一定程度上造成能量浪费;本文提出了一种能量均衡的非均匀分簇算法——EBUCA,如图 1(d)所示,在簇头选择阶段,根据节点的剩余能量和密度控制簇头的产生,在节点密度大的区域增大簇头产生的概率,在节点密度小的区域则相应地减少簇头产生的概率。在簇的形成阶段,利用簇头所在区域的密度和簇头与 sink 节点的距离共同决定簇半径的大小。在簇头多的区域相应地减小簇半径,达到均衡各簇能耗的目的。

3 EBUCA 算法描述

3.1 簇头选择

LEACH 算法将单个节点的状态作为簇头选择的度量标准,没有考虑到周围节点的影响,这样会由于网络节点稀疏程度不同造成簇头节点的分布不均。为了在节点密集区域增加簇头的个数,需要增大对应节点成为簇头的概率,对于节点稀疏区域,减少其中节点成为簇头的概率即可。

同时,还要考虑节点的剩余能量平衡问题,避免能量低的节点成为簇头。重新定义 $T(n)$ 如下:

$$T(n) = \begin{cases} \frac{p}{1 - p(r * \text{mod } \frac{1}{p})} \frac{E_n \text{ current}}{E_n \text{ max}} (1 + \alpha_n), & n \in G \\ 0, & n \notin G \end{cases} \quad (2)$$

$$\alpha_n = \frac{\text{Neighbor}(n) \text{ alive} - 1}{\text{Network alive}} \quad (3)$$

式中, α_n 表示节点 n 的密度, $\text{Neighbor}(n) \text{ alive}$ 和 Network alive 分别表示节点 n 的邻居节点以及整个网络中存活节点的数目,即剩余能量大于 0 的数目。 $E_n \text{ max}$ 和 $E_n \text{ current}$ 分别表示节点的初始能量和当前能量。

3.2 簇的建立

簇头一般通过多跳的方式将数据发送到 sink 节点,距离 sink 节点较近的簇头会作为中继节点来转发其他簇头的数据,从而消耗更多的能量。根据各簇头距 sink 节点的距离来构造各簇大小不等的簇半径,可以平衡网络能耗,达到进一步延长网络生命周期的目的。同时,当簇头所在区域密度较大时相应地减少簇半径,使簇半径随着节点密度的增大而减少,从而平衡各簇内的节点数目。

簇头节点选定后,非簇头节点进入睡眠状态, sink 节点用一个给定的发送功率向网络内广播一个信号,各簇头节点根据接收信号的强度计算它到 sink 节点的近似距离 $d(H_i, \text{sink})$, 并确定其簇半径 R_{H_i} , 令簇头的最大簇半径为 R_0 , 每个簇的半径计算公式如下:

$$R_{H_i} = \left[(1 - \alpha_{H_i}) \frac{d_{\text{max}} - d_{H_i}}{d_{\text{max}} - d_{\text{min}}} \right] R_0 \quad (4)$$

式中, α_{H_i} 为簇头节点的密度,在 0~1 之间取值, d_{max} 和 d_{min} 分别表示网络中的簇头到 sink 节点的最大和最小距离, d_{H_i} 表示簇头节点与 sink 节点的距离。

由式(4)可以看出,距离 sink 节点最近的簇首节点的簇

半径为 $(1-\alpha_{H_i})R_0$,距离 sink 节点最远的簇头所对应的簇半径为 R_0 ,簇半径与簇首到 sink 节点的距离呈线性递减关系。当两个簇头节点密度相同时,距 sink 节点距离越近,簇半径越小,使簇头节点节省能量用于簇间数据转发。当两个簇头到 sink 节点的距离相等时,密度大的半径变小,达到平衡各簇负载的目的。

各个簇的半径确立后,处于睡眠状态的节点被唤醒,簇头节点以固定的功率广播其各自的簇半径 R_{H_i} ,非簇头节点根据 R_{H_i} 和 $d(H_i, sink)$ 来选择簇头,选择过程如下:

步骤 1 节点选择与自己距离最近的簇头 H_i 。

步骤 2 计算节点与簇头 H_i 的距离 $d(i, H_i)$, 并比较 $d(i, H_i)$ 与簇半径 R_{H_i} 的大小,如果 $d(i, H_i) \leq R_{H_i}$, 则加入该簇。如果 $d(i, H_i) > R_{H_i}$, 则返回步骤 1。

步骤 3 如果与节点较近的簇头 H_i 不存在,或者不存在符合 $d(i, H_i) \leq R_{H_i}$ 的簇头,则节点加入以邻居节点为簇头的簇,或以邻居节点为中继加入簇,从而成为簇成员节点。

当数据到达各簇头后,簇头根据在簇建立阶段得到的信息 $d(H_i, H_j)$ 和 $d(H_j, sink)$ 选择离自己最近的簇头,以多跳的方式将最终的融合数据传送到 sink 节点。

4 仿真实验与结果分析

本文采用 NS2 进行仿真实验,在网络的生存周期、能量效率、能量均衡性 3 个方面与 LEACH、EEUC 和 DBCP 算法进行比较。

4.1 仿真环境与参数设置

网络的仿真区域为 $200\text{m} \times 200\text{m}$, 100 个节点随机分布在该区域中,基站坐标为 $(100, 200)$ 。每轮簇头个数占网络中存活节点的 5% ($P=0.05$)。数据传输模型采用一阶无线电模型^[12]。其中实验中涉及到的参数值如表 1 所列。

表 1 实验参数

实验参数	参数值
E_n_max	2J
CSThresh	$1e-9$ J/bit
RXThresh	$6e-9$ J/bit
Efriss_amp	$9.6741659015025702e-12$ J/bit/m ²
Etwo_ray_amp	$1.3037037037037037e-15$ J/bit/m ⁴
Ebf	$5e-9$ J/bit/signal

4.2 仿真结果分析

在簇头选择阶段,根据节点密度和剩余能量来选择簇头,在节点分布稠密的区域增加簇头数量,在节点分布稀疏的区域减少簇头数量,网络中节点分布如图 2 所示,簇头根据周围节点的数目均匀地分布在网络中。

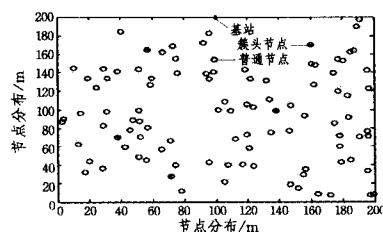


图 2 网络中节点分布图

网络的生命周期是衡量网络质量的重要指标,在仿真实验中,选择网络中节点的存活个数与网络的运行轮数,对 4 种

算法进行比较,如图 3 所示,从第 1 个节点失效的时间对比可知,EBUCA 比 LEACH 提高了 140% 的网络生命周期,较 DBCP 提高 62%,较 EEUC 提高 20%。LEACH 和 EEUC 分别在 951 轮和 1207 轮时,所有节点死亡,而 EBUCA 到 1500 轮还有节点存活,由此可见,本文所提算法有效地延长了网络生命周期。

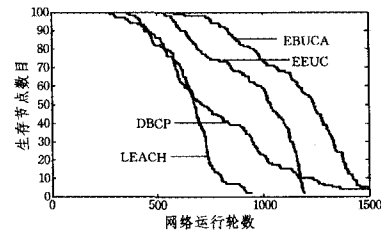


图 3 网络生命周期对比

图 4 对比了 4 种算法的网络中剩余总能量随仿真周期的变化曲线。较小的坡度显示较慢的能量消耗速度和较长的生存时间,EBUCA 算法的坡度明显小于其他 3 种。LEACH 在网络运行到 700 轮时,节点全部死亡,网络剩余总能量为 0,EEUC 与 DBCP 大约运行到 1100 轮时,网络中能量耗尽。通过对比可知,EBUCA 比其他算法更能有效地节约能量。

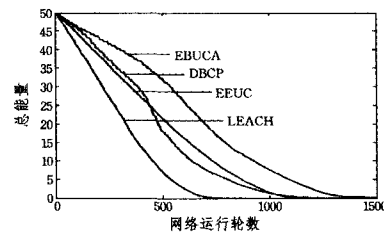


图 4 网络中剩余总能量对比

每轮所消耗的能量显示节点能耗的均衡性,只有采用合理的簇首选择机制、路由策略、簇重构机制,各节点能耗才能保持在一个均衡的状态,才能延长网络的生命周期。图 5 显示了 4 种算法每轮的能耗,由图 5 可知,EBUCA 和 EEUC 比 LEACH 和 DBCP 具有更好的能耗均衡性,主要表现为相对波动性较小,因此在均衡全网能量方面表现良好。

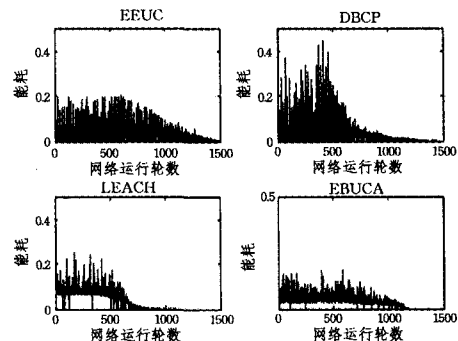


图 5 各轮能耗对比

通过比较分析可以看出,本文算法可以显著地延长网络中节点的死亡时间,即随着网络运行时间的增加,使用改进算法的网络与使用 LEACH、EEUC、DBCP 算法的网络相比,不仅可以使簇首均匀地分布在网络中,优化网络的拓扑结构,同时使能量消耗均匀地分布在各节点上,达到平衡网络负载的目的,使网络的能量得到均衡的利用,而且本文算法能够有效地提高能量利用率,显著地延长网络寿命。

结束语 通过对已有算法及其存在问题的分析,结合非均匀分簇的思想,提出了一种能量均衡的非均匀分簇路由算法——EBUCA。改进算法利用节点密度和剩余能量设置簇头的选择阈值,在节点密集区增加簇头数量并在簇的建立阶段利用簇头密度及距 sink 节点的距离建立不同的簇半径,达到平衡全网簇规模的目的。最后通过仿真实验验证算法的优越性,即通过与 LEACH、EEUC 和 DBCP 路由算法进行对比,证明 EBUCA 路由算法能有效平衡各节点能耗,延长网络生命周期。

参 考 文 献

[1] Heinzelman W, Chandrakasan A, Balakrishnan H. Energy-Efficient Communication Protocol for Wireless Microsensor Networks[C]//Proceedings of the 33rd Hawaii International Conference on System Science, Hawaii, 2000; 3005-3014

[2] Qing Li, Zhu Qing-xin, Wang Ming-wen. Design of a distributed energy-efficient clustering algorithm for heterogeneous wireless sensor networks[C]//Computer Communications, 2006; 2230-2237

[3] Handy M J, Haase M, Timmermann D. Low Energy Adaptive Clustering Hierarchy with Deterministic Cluster-Head Selection [C]//Proc of the 4th IEEE Conf on Mobile and Wireless Communications Networks. IEEE Communications Society, 2002; 368—372

[4] 徐久强,毕伟伟,朱剑. WSN 中多跳均匀分簇路由算法的设计与

仿真[J]. 系统仿真学报, 2011, 23(5): 992-997

[5] 顾跃跃,白光伟,陶金晶. LEACH-CS:一种自定义的 WSN 跨区多跳路由机制[J]. 计算机科学, 2011, 38(1): 78-82

[6] Li Cheng-fa, Ye Mao, Chen Gui-hai. An energy-efficient unequal clustering mechanism for wireless sensor networks[C]//Proc of the 2nd IEEE International Conference on Mobile Ad hoc and Sensor Systems (MASS), 2005

[7] 蒋畅江,石为人,王平. 能量均衡的无线传感器网非均匀分簇路由协议[J]. 软件学报, 2012, 23(5): 1222-123

[8] Du Jiang, Wang Liang. Uneven Clustering Routing Algorithm for Wireless Sensor Networks Based on Ant Colony Optimization[C]//International Conference on Computer Research and Development (ICCRD), 2011; 67-71

[9] 乔俊峰,刘三阳,曹祥宇. 无线传感器网络中基于节点密度的簇算法[J]. 计算机科学, 2009, 36(12): 46-48

[10] Sungryoul L, Han C, Byoungchang P. LUCA: An Energy-efficient Unequal Clustering Algorithm Using Location Information for Wireless Sensor Networks[J]. Wireless Personal Communications, 2011, 56(4): 715-731

[11] Thein M C M, Thein T. An Energy Efficient Cluster-Head Selection for Wireless Sensor Networks[C]//International Conference on Intelligent Systems Modelling and Simulation (ISMS), 2010; 287-291

[12] Rappaport T. Wireless Communications: Principles & Practice [M]. Englewood Cliffs, NJ: Prentice-Hall, 1996

(上接第 61 页)

发展。然而,移动设备自身计算能力、存储能力、电池续航能力等不足;加之无线网络传输带宽有限且资费昂贵;与此同时,现有的大多数云服务并不关心移动客户端,这些都严重影响了移动云服务的市场发展。因此,本文研究了移动计算环境的特点,设计并实现了一种基于代理的移动云服务访问框架,即通过代理服务器对云服务做传输协议、数据格式等方面的桥接转换,使其在无线网络带宽占用和移动设备资源消耗上变得更适宜移动设备访问;代理服务器上的服务 Mashup 既为移动设备访问个性化服务提供了方便,又节省了移动设备计算、带宽等资源。在未来的研究中,要考虑到移动设备应支持更多的操作系统(如 iOS、Windows Mobile 等操作系统)。而代理服务器上的框架应考虑到随着移动用户的增加,其可能会成为瓶颈,应将这部分的框架移植到计算能力更强的云基础设施上(如 Amazon、Google 等)。

参 考 文 献

[1] Kyung M. Mobile Cloud Computing Challenges[OL]. <http://www2.alcatel-lucent.com/blogs/techzine/2010/mobile-cloud-computing-challenges>, 2010

[2] Chen Y, et al. EXPedite: A System for Encoded XML Processing [C]//Proc of the ACM 13th Conf on Information and Knowledge Management, 2004

[3] Ioana G, Oriana R, Dejan J, et al. Calling the cloud: Enabling mobile phones as interfaces to cloud applications. Lecture Notes of the Institute for Computer Sciences[J]. Social Informatics and Telecommunications Engineering, 2010, 48(4): 161-174

[4] Jan S, Oriana R, Gustavo A. AlfredO: an architecture for flexible interaction with electronic devices[C]//Middleware'08 Proceedings of the 9th ACM/IFIP/USENIX International Conference on Middleware, 2008

[5] Jan S, Gustavo A, Timothy R. R-OSGi: Distributed applications through software modularization [C] // MIDDLEWARE2007 Proceedings of the 8th ACM/IFIP/USENIX international conference on Middleware, 2007

[6] Xin wen Z, Sangoh J, Anugeetha K, et al. Towards an Elastic Application Model for Augmenting Computing Capabilities of Mobile Platforms[C]//Middleware'09 Proceedings of the 10th ACM/IFIP/USENIX International Conference on Middleware, 2009

[7] Jason H. Using RESTful Web-Services and Cloud Computing to Create Next Generation Mobile Applications[C]//OOPSLA, 2009

[8] Mahadev S, Paramvir B, Ramón C, et al. The Case for VM-Based Cloudlets in mobile computing[J]. IEEE Pervasive Computing, 2009, 8(4): 14-23

[9] Qian W. Mobile Cloud Computing [D]. Master of Science, The Department of Computer Science in University of Saskatchewan, Saskatoon, 2011

[10] Roy T. Architectural Styles and the Design of Network-based Software Architectures [D]. Doctor of Philosophy, Dept. of Computer Science, Univ. of California, Irvine, 2000

[11] PowerTutor [OL]. <http://ziyang.eecs.umich.edu/projects/power tutor/>, 2011