

无线传感器网络拓扑控制算法的改进

李华¹, 陈超², 卢令²

(1. 解放军第四五二医院信息科, 成都 610036; 2. 四川理工学院自动化与电子信息学院, 四川 自贡 643000)

摘要:由于传感器节点能量有限,需要设计一个高效的拓扑控制算法来节约能量,延长整个网络生命周期。LEACH 算法是典型的层次型拓扑控制算法,在深入分析 LEACH 算法的基础上,基于其存在的缺点,提出一种改进算法,通过计算节点综合性能权值来选举簇头,性能越好的节点越容易成为簇头;同时,当簇头节点剩余能量小于规定阈值时,才进行簇头的更新;簇头更新采用簇内局部更新的策略,以减少簇头选举带来的开销。通过仿真分析和比较,改进算法所构建的拓扑结构在总体性能上更优于 LEACH 算法。

关键词:无线传感器网络;LEACH 算法;拓扑控制;生命周期

中图分类号:TP393

文献标志码:A

引言

近年来,无线传感器网络^[1]作为一种全新类型的智能网络信息系统,在信息领域迅速发展。相比于其他短距离无线通信技术,无线传感器网络具有易部署、成本低、规模大、自组织等许多优点,使得无线传感器网络的研究与应用得到了越来越多的重视。但是无线传感器网络中,节点通常使用电池供电,而其应用环境又使得电池的更换非常困难。所以,想要充分利用节点有限的计算和存储能力,就必须有一个好的拓扑控制机制来优化网络的拓扑结构,以达到节能和延长网络生命周期的目的。

根据目前国内外的研究成果,可将无线传感器网络的拓扑控制分为功率控制和层次控制两种。功率控制中,要求节点必须具有较强的存储和计算能力来维护路由信息,因而只适用于规模较小的无线传感器网络。而层次控制可有效解决这个问题,网络结构具有较好的可扩展性,可用于大规模部署的传感器网络。LEACH 算法(Low Energy Adaptive Clustering Hierarchy)^[2]是最早

被提出来的较成熟且具有典型性的层次控制算法,该算法通过选举簇头来构成骨干网^[3],但是簇头的选择具有随机性,会导致簇头的分布不均匀,不能有效均衡能耗和延长网络生命周期,对于此,本文将在簇头选举和更新方面对 LEACH 算法进行改进。

1 经典的 LEACH 算法

LEACH 算法通过等概率随机选择簇头,算法的运行过程是周期性的,因此定义了“轮(round)”这个概念,每一次轮循环包括建簇和数据传输两个阶段^[4],每一次轮循环结束后要重新选举簇头节点,执行建簇和数据传输以保持网络节点的能耗均衡。

1.1 建簇阶段

LEACH 算法中各节点轮流担任簇头,每一轮选举簇头时,各节点在 $0 \sim 1$ 之间随机产生一个数 T_i ,同时设定阈值 T_n ,若 $T_i < T_n$,则簇头由节点 i 担任^[5]。对于已经担任过簇头的节点,设置 $T_n = 0$,于是该节点在以后各轮中都作为普通节点。随着轮数增加,节点的阈值 T_n 将

收稿日期:2014-04-22

基金项目:人工智能四川省重点实验室开放基金(2012RZY21);四川省科技厅创新团队项目(2012Z068);四川理工学院研究生创新基金(y2012009)

作者简介:李华(1974-),女,四川雅安人,工程师,主要从事信息管理与信息技术研发方面的研究,(E-mail)236537194@qq.com

随着已担任过簇头的节点数目的增大而增大,则未担任过簇头的节点被选为簇头的概率也就会增大。如果只有一个节点未担任过簇头,其 $T_n = 1$, 意味着该节点一定当选。阈值 T_n 由以下公式确定:

$$T_n = \begin{cases} \frac{p_0}{1 - p_0 \times [\text{rmod}(1/p_0)]} & n \in G \\ 0 & \text{其他} \end{cases}$$

式中, p_0 是簇头数占总节点数的百分比, r 是当前轮数, $\text{rmod}(1/p_0)$ 是目前为止节点集合中已当选过簇头的节点个数, G 是目前为止节点集合中没有当选过簇头的节点的集合。

选举完簇头后,簇头节点便向全网广播簇头消息,消息包含簇头节点 ID 和消息类型的头部。网内其他节点接收到簇头消息后,根据消息强弱来选择要加入的簇。若节点 i 接收到来自某个簇头节点消息的信号越强,说明节点 i 离该簇头节点越近,通信代价也就越小。如果接收到的消息信号强度相当,则节点可以随机选择要加入的簇。

若节点 i 决定要加入簇 j 后,需通知簇头 j ,即向簇头 j 发送自身 ID 和簇头 j 的 ID。

当簇头节点接收到每个节点的入簇请求消息后,会建立一个 TDMA 调度时间表并将其发送给每一个簇成员节点。当所有非簇头节点都收到时间表后,簇的建立就完成了,随后便是数据传输阶段。

1.2 稳定的数据传输阶段

该阶段主要是簇内节点按 TDMA 调度时间表完成数据传输任务。该阶段被划分成为若干个时间长度相等的帧,每个帧又包含若干个时间长度相等的时隙。

所有非簇头节点都只在自己所在时隙内发送数据,其余时间关闭自己的无线通信模块以减少能耗。簇头节点要随时准备接收成员节点发来的数据,因此要时刻保持活跃状态,簇头节点收集所有簇内节点发来的信息后,对相似信号和噪音信号进行处理和融合,并发送给汇聚节点。网络持续工作一个固定的时间段后,将会重新进行簇建立阶段和稳定的数据传输阶段^[6]。

1.3 LEACH 算法的缺点

LEACH 算法通过构建骨干网的方式来进行数据多跳传输,网络的性能得到了很大的提升,但是在某些方面算法还存在一定的局限性。

首先,LEACH 算法是基于初始能量相同且均匀分布的网络模型研究的。而实际情况很难达到这样的理想状态,节点的能量和地理位置都将会存在差异,因此 LEACH 算法无法保证簇头的均匀分布,从而不能有效

地提高网络整体性能和生命周期,簇头节点负载也不均衡。其次,LEACH 算法中,簇头是随机选择的,并未考虑到剩余能量和其他问题,而对于剩余能量不同的节点,成为簇头的代价也不同。另外,LEACH 算法中,簇头节点承担任务重,能耗相对较大,因此簇头节点容易失效并造成簇头频繁更新,而全网的簇头选举会带来较高的额外开销,这样的开销将会降低节点的能量利用率。

2 基于组合加权的改进算法

2.1 改进算法(M-LEACH)的设计思想

借鉴 Ad Hoc 网络中的按需加权分簇算法(AOW)的思想^[7-8],对无线传感器网络的 LEACH 算法进行改进。在 LEACH 分簇算法中,簇头的合理选择对网络的拓扑结构至关重要,簇头的选择是否合理决定了整个无线传感器网络性能的好坏。传统的 LEACH 算法通过随机的方式选择簇头,M-LEACH 算法主要对簇头的选举和簇的更新方式进行改进。

改进算法的主要设计思想是:在簇头的选举阶段,把节点的剩余能量、节点度数、节点的移动性以及节点与其邻居节点之间的平均距离作为簇头选举所考虑的因素;簇头的更新只在簇内进行,并不在全网范围内进行更新,局部调整簇头可以减少全网选举簇头带来的开销;簇头的更新使用按需更新簇头策略,即当簇头节点的剩余能量小于某个阈值时进行簇头的重新选举。

2.2 改进算法描述

在簇头选举初始阶段,为每一个节点分配一个权值 W ,该权值衡量了节点适合充当簇头的程度,权值 W 越小的节点越适合充当簇头。节点 i 的权值 $W(i)$ 由以下公式计算可得:

$$W(i) = a1 \times E(i) + a2 \times D(i) + a3 \times M(i) + a4 \times P(i)$$

其中, $E(i)$ 表示节点 i 到目前为止已消耗掉的能量, $D(i)$ 表示节点 i 当前的节点度与网络理想节点度差值的绝对值, $M(i)$ 是衡量节点 i 移动性的参数, $P(i)$ 表示节点 i 到所有邻居节点的平均距离, $a1$ 、 $a2$ 、 $a3$ 、 $a4$ 分别为四个参数的权重。算法的执行过程可描述如下:

(1) 建簇阶段

①首先,根据网络具体情况确定簇头概率 p ,所有节点通过周期性地交换“Hello”消息,可统计出邻居节点数目以及与所有邻居节点的距离。

②每个节点 i 计算自己已耗能量 $E(i)$ 。假设所有节点拥有相同的初始能量,由于节点接收和发送数据所消耗能量远比感知数据时所消耗能量大,所以各节点的已

消耗能量由节点收发数据时消耗能量来计算。

③每个节点 i 计算自己的节点度 D_i 与网络理想节点度 D_0 之差,即

$$D(i) = |D_i - D_0|$$

④用每个节点 i 的平均移动速度代表节点的移动性 $M(i)$ 。

⑤每个节点 i 统计其到自己所有邻居节点的距离总和 P_i (总),并除以邻居节点个数,得到节点 i 到所有邻居节点之间的平均距离 $P(i)$ 。

⑥对每个节点 i 计算它的组合权重 $W(i)$,其中权重因子 a_1 、 a_2 、 a_3 、 a_4 满足 $a_1 + a_2 + a_3 + a_4 = 1$,当某个参数越重要,其权重因子的值越大。每个节点依据自己的权值 $W(i)$ 大小设置一个门限时间,门限时间的大小和 $W(i)$ 的大小成正比,权值越小的节点,为其设置越短的门限时间。

⑦门限时间到的节点自动升为簇头,这意味着权值越小的节点就优先成为了簇头,即节点综合性能越好,就越容易成为簇头。若存在节点权值相等的情况,即门限时间相同,则选择 ID 号较小的那个节点作为簇头。

⑧当选为簇头的节点在全网范围内广播簇头消息,若在各自的门限时间内,节点收到了簇头广播消息,则根据广播消息的信号强弱选择要加入的簇并向所选簇的簇头发送入簇消息,同时取消门限时间。簇头节点收到所有入簇消息后,建立一个 TDMA 调度时间表,并发送给簇内每个节点。当所有节点都收到 TDMA 时间表后,进入稳定的数据传输阶段。

(2) 稳定的数据传输阶段

簇成员节点按照 TDMA 时间表将采集数据发送给簇头节点,簇头节点收集所有成员节点发来的信息,对其进行融合处理并转发给汇聚节点。簇内节点在 TDMA 时间表分配给自己的时隙之外关闭其通信模块。

(3) 簇的更新

①当簇头节点的能量小于某个阈值 E_m 时,接收簇内节点最后一帧数据,簇成员节点会将自己当前的权值信息和 ID 号联合最后一帧数据发送给簇头节点。簇头节点比较每个成员节点的权值,权值最小的节点将当选为下一轮的簇头,簇头节点将在簇内广播下一轮簇头的信息,并把自己设置为非簇头节点。

②簇内节点收到新簇头节点信息后,把自己的 ID 号与新簇头的 ID 号进行对比,如果相等则将自己设置为簇头。

值得注意的是,第一轮簇头选举时,由于所有节点有着相同的初始能量,所以计算权值时不考虑已消耗能量。

3 仿真分析

3.1 仿真环境参数设置

仿真区域为一矩形区域:500 × 100 m,随机放置 100 个节点,基站坐标为(50 m,50 m)。节点的通信距离范围为:1 ~ 100 m。其余参数设置如下:

(1)整个网络是连通的,即不存在独立部分或者是孤立的节点。

(2)所有节点随机分布且每个节点都有一个唯一的 ID 号。

(3)部分节点具有移动性,即部分节点在一定时间以后会改变自身坐标。

(4)节点通过广播“hello”消息来计算邻节点数及其与邻节点的距离。

(5)节点的天线采用全向天线且所有节点都能接收到来自邻节点的消息。

(6)节点初始能量为 0.5 J,簇头概率为 0.05,控制包长度为 100 字节,数据包长度为 4000 字节。节点能量等于 0 时,标记为死亡,并不再进行数据的传输。

(7)节点的权重因子 a_1 、 a_2 、 a_3 、 a_4 满足 $a_1 + a_2 + a_3 + a_4 = 1$,且考虑到每个影响因素在簇头选举过程中的作用程度不同,分别取值为:0.5、0.2、0.2、0.1。同时,只有当簇头的能量下降至充当簇头时能量的百分之七十时,才重新选举簇头。

3.2 仿真结果分析

使用 MATLAB 对 M-LEACH 算法进行仿真,并与 LEACH 算法进行比较分析。

(1)生命周期。也就是网络维持正常工作所持续的时间,本文中将其定义为网络从开始工作到节点存活个数降至初始节点数的 50% 时所持续的时间。算法的生命周期仿真结果如图 1 所示。

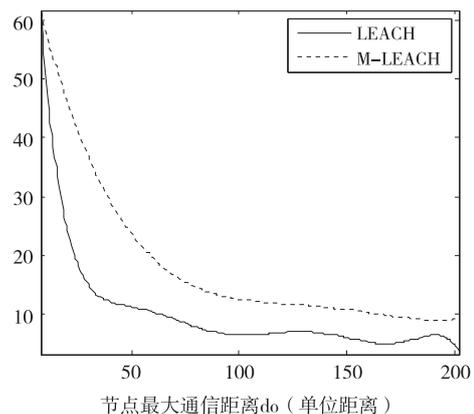


图 1 网络生命周期

由图1可知,随着节点的最大通信距离的增大,改进算法与LEACH算法的网络生命周期都在减小,并且在整个过程中,改进算法的生命周期都优于原算法。其原因是:相比于LEACH算法,改进算法选出的簇头节点的综合性能更好,能始终让剩余能量较多的、节点移动性较小的、节点度与理想节点度之差较小的、与邻节点的平均距离较优的节点来担任簇头,使得能量的有效性得到了提高,节点的负载也更均匀,延长了节点存活的时间,因此,网络生命周期也 longer。

(2)节点平均剩余能量。即网络正常工作中,所有节点的平均剩余能量。节点平均剩余能量仿真结果如图2所示。

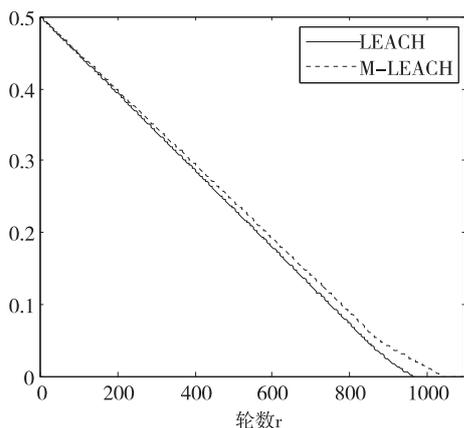


图2 节点平均剩余能量比较

从图2可以看出,改进算法的剩余能量要高于LEACH算法,并且随着轮数的增加,剩余能量最终都趋于0,即节点死亡。这是因为,改进算法采用了新的簇头更新机制,减少了簇头的更新次数,从而降低了选举簇头带来的开销,节约了一定的能耗,同时也有助于延长网络的生命周期。

(3)节点充当簇头的公平性指数(HFI)。该性能指标用来衡量节点充当簇头的公平性程度,也就是节点充当簇头的时间相比于所有节点充当簇头的平均时间的偏离程度。HFI由以下公式计算可得:

$$HFI = E \{ |t_i - E(t_i)| \}, i \in V$$

其中, t_i 为节点 i 充当簇头的时间, $E(t_i) = \sum_{i \in V} t_i / N$ 为节点作为簇头的平均时间, N 为全体节点数, V 为网络中的节点集合。因此, HFI 值越小, 节点在充当簇头方面就越公平。在最好的时候, $t_i = E(t_i)$, 这时 $HFI = 0$ 。一般情况下, 簇头的能耗远比普通节点大, HFI 值越小, 说明所有节点越能较均匀地分担能耗, 也就是所有节点可以较公平地充当簇头, 从而可以有效地延长网络生存

时间。HFI 值越小表示网络工作越长的时间才会出现节点的分区, 因此网络的整体性能可以得到更好的维护。节点充当簇头的公平性指数的仿真结果如图3所示。

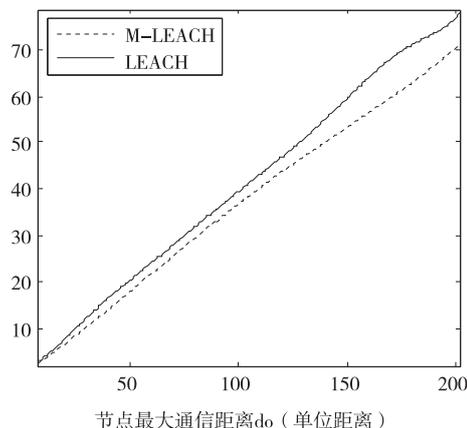


图3 节点充当簇头的公平性指数比较

从图3可以看出,两算法的HFI值都随节点的最大通信距离 do 的增大而增大, 而改进算法的HFI值在整个区间内都比LEACH算法的HFI值要小, 说明改进后的算法使得节点能够更公平的充当簇头。随着节点的最大通信距离增大, 造成了节点之间的冲突域, 数据传输的碰撞率增大, 能量损耗变大, 节点充当簇头的时间变短, 以至于频繁的分簇, 簇头公平性降低, 所以HFI值呈上升趋势。而本算法对簇头选举进行了改进, 簇头分布更为合理, 能耗也更均匀地分配到所有节点上, 因此, 节点充当簇头的公平性更好。

(4)网络负载平衡因子(LBF)。网络的负载表现为簇头处理负载的能力, 即簇头能够支持的簇的大小。分簇结构中, 簇头将承担较多的任务, 能耗较大, 因此拓扑控制算法应使网络的负载尽可能均匀地分配给选出的每个簇头。某簇头处理负载的能力可由该簇的大小近似衡量, 因此, 网络负载平衡因子可由簇成员节点个数的方差的倒数来计算:

$$LBF = n_h / \sum_{n=1}^{n_h} (x_n - u)^2$$

其中, n_h 为网络中簇头总数, x_n 为簇头 n 所在簇的成员节点个数, $u = (N - n_h) / n_h$ 为网络内所有簇的成员节点的平均个数, N 为网络中总节点数。若LBF越大, 说明网络的负载越能均匀地分配给每个簇头, 在负载平衡程度最好时, $x_n = u$, 这时LBF趋于无穷大^[9]。网络负载平衡因子仿真结果如图4所示。

由图4可以看出, 在整个循环过程中, 改进算法的负载平衡因子明显高于LEACH算法。在循环开始时, 两算法的负载平衡因子都较大, 随着轮数的增加, 都呈

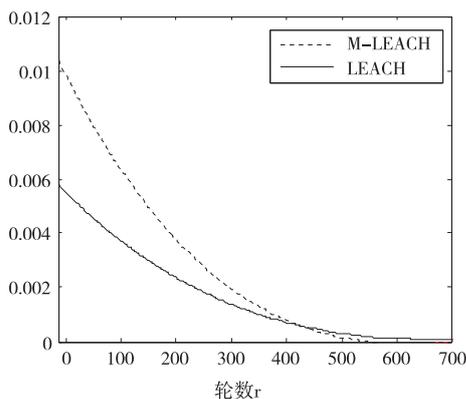


图 4 负载平衡因子比较

递减趋势,在轮数 $r = 600$ 后 LBF 均趋于 0。改进算法的 LBF 较好的原因在于:算法在簇头选举时综合考虑了节点的多种影响因素,使选出来的簇头不管在节点度方面,还是在簇头与其邻居节点的平均距离方面,都优于 LEACH 算法,因此,簇头分布更均匀,负载的分布也越平衡,从而避免了瓶颈节点的出现以及网络拥塞的情况。

4 结束语

本文提出了一种基于组合加权的改进 LEACH 算法,并对其簇更新方式进行了改进。改进算法在生命周期、节点平均剩余能量、 HFI 、 LBF 上都优于 LEACH 算法。另外,本文中仅对分簇算法进行了改进,后续研究可对分簇算法与功率控制相结合的拓扑控制机制做一定的尝试和研究。

参考文献:

- [1] Estrin D, Govindan R, Heidemann J S, et al. Next century challenges: scalable coordination in sensor networks[C]// Proc of the 5th International Conference on Mobile Computing and Networking (MobiCom), Seattle, Washington, August 15-20, 1999: 263-270.
- [2] Heinzelman W R, Chandrakasan A, Balakrishnan H. An application-special protocol architecture for wireless microsensor networks[J]. IEEE Transactions on wireless communications, 2002, 1(4): 660-670.
- [3] Gerla M, Tsai J T C. Multicluster, mobile, multimedia radio network[J]. Wireless Networks, 1995, 1(3): 255-265.
- [4] 刘晓文, 闫静杰, 苗锦, 等. 矿井无线传感器网络 LEACH 协议的改进[J]. 煤炭科学技术, 2009, 37(4): 46-49.
- [5] 陈晓娟, 王卓, 吴浩. 一种基于 LEACH 的改进 WSN 路由算法[J]. 传感技术学报, 2013, 26(1): 116-121.
- [6] 肖刘军, 邓平. 一种基于位置和能量的 WSN 改进分簇协议[J]. 通信技术, 2010, 43(8): 43-45.
- [7] 杜国勇, 束永安. 基于链接率的 Ad hoc 自适应按需加权分簇算法[J]. 计算机技术与发展, 2014, 24(1): 93-97.
- [8] 林要华, 胡华平. 基于轨道预测自适应 Ad Hoc 分簇算法[J]. 计算机工程与科学, 2012, 32(2): 27-30.
- [9] 郑少仁, 王海涛, 赵志峰, 等. Ad hoc 网络技术[M]. 北京: 人民邮电出版社, 2005.

Improvement of Topology Control Algorithm for Wireless Sensor Networks

LI Hua¹, CHEN Chao², LU Ling²

(1. Department of Information, No. 452 Hospital of PLA, Chengdu 610036, China;

2. School of Automation & Electronic Information, Sichuan University of Science & Engineering, Zigong 643000, China)

Abstract: Because the energy of the sensor node is limited, it is necessary to design an efficient topology control algorithm to save energy and to prolong the lifecycle of whole network. LEACH algorithm is a typical topology control algorithm of hierarchical type. Based on deep analysis of LEACH algorithm and its existed disadvantages, an improved algorithm is put forward. The improved algorithm selects the cluster head by calculating the integrated performance weight of node, the node whose performance is better is easier to become the cluster head; meanwhile, only when the surplus energy of cluster node is less than the specified threshold can the cluster head be updated; the cluster head is updated by the strategy that partially update in the cluster, which can reduce the header expenditure bring from the selection of cluster head. Through the simulation analyze and comparison, it is known that the overall performance of topology structure built by improved algorithm is better than that of LEACH algorithm.

Key words: wireless sensor networks; LEACH algorithm; topology control; lifecycle