

基于 LEACH 的改进型无线传感器网络路由算法研究

杜风雷

(浙江警官职业学院, 浙江 杭州 310018)

摘 要: 针对经典分簇路由算法 LEACH 在选取簇头时没有充分考虑节点剩余能量的问题, 在 LEACH 协议的基础上, 改进了簇首选择、最优簇首数和簇形成过程, 提出一个自适应分布式算法, 即 ADCP-LEACH 算法, 该算法可以应用于周期性的采集数据, 根据每个节点竞选簇头时的阈值不同, 让剩余能量大的节点当选簇头的概率更大。NS2 仿真实验表明, 改进的协议有效延长了网络的存活时间, 性能优于 LEACH 协议。

关键词: 无线传感器网络; LEACH; 路由算法; 生存时间

中图分类号: TP31

文献标识码: A

文章编号: 1000-7180(2010)08-0128-04

Research of Improved Wireless Sensor Network Routing Algorithm Based on LEACH

DU Feng-lei

(Zhejiang Police Vocational Academy, Hangzhou 310018, China)

Abstract: Classical clustering routing algorithm for LEACH cluster head is selected without full consideration of the remaining energy of node problem, based on LEACH protocol to improve the cluster's first choice, the optimal cluster number and cluster formation of the first proposed an adaptive distribution type algorithm, which ADCP-LEACH algorithm can be applied to periodic data collection, each cluster head node election when the threshold is different for the remaining energy of a large cluster head node election probability greater. NS2 simulation results show improved agreement effectively extended the survival time of the network, better performance than LEACH protocol.

Key words: wireless sensor networks; LEACH; dual-layer router algorithm; lifetime

1 引言

无线传感器网络路由的一个重要问题是如何有效地均衡整个网络的能量消耗水平, 避免形成能量空洞, 从而导致整个网络过早死亡^[1]。由于无线传感器网络受到严格而近乎苛刻的能源和带宽限制, 且传感器节点经常是工作在恶劣环境中, 采用微型电池供电, 一般很难进行充电或替代, 网络的能量资源非常有限。因此在现有的能量资源条件下, 延长网络的生命周期, 研究和设计满足能量有效性的通信路由协议是无线传感器网络技术迫切需要解决的问题。对节点分簇是无线传感器网络中节约能源的一种有效方案。LEACH 协议采用随机选举簇头的方

式来轮换簇头, 避免了簇头过分消耗能量, 然而 LEACH 协议采用连续数据发送模式和单跳的路径选择模式, 即需要假设网络中的所有节点都能直接与簇头节点和汇聚节点通信。因此如果监测范围比较大, LEACH 算法并不太适用, 即使是在小规模网络中, 离汇聚节点较远的节点由于采用大功率通信也会导致生存时间过短的问题出现, 而且簇头轮换引起的动态分簇也带来了拓扑变换和大量广播这样的额外开销, 耗费过多的能量^[2]。

针对 LEACH 算法的存在的的问题, 本文分别对簇首选择和最优簇首数的确定方式做出改进, 提出一个自适应分布式算法, 即 ADCP-LEACH 算法。该算法可以应用于周期性的采集数据, 根据每个节

点竞选簇头时的阈值不同,让剩余能量大的节点当选簇头的概率更大,延长了网络的生存时间.实验结果表明,ADCP-LEACH 性能明显优于 LEACH 协议,在均匀网络和非均匀网络中网络生存时间分别延长了 20% 以上,有效地证明了该算法的可行性.

2 系统模型

2.1 网络模型

无线传感器网络模型作如下合理的假设:

(1) 将节点分成三类:基站 BS 节点、簇头 CH、普通传感器节点 SN,它们随机分布在区域中,其中 BS 和 SN 的位置是固定的;

(2) 基站 BS 的能量和存储能力是不受限制的,它主要负责收集和处理传感器节点发送来的数据及管理整个网络,在这里假定基站是安全的;

(3) 传感器节点间消耗的能量根据传输距离进行相应的调整^[3];

(4) 节点的能量可异构,但不能补充,即不要求所有节点的初始能量具有相同的值;

(5) 各节点的通信半径为 R_{probe} ,节点的无线发射功率可控,即节点可根据距离来调整发射功率的大小;

(6) 节点周期性地收集到的数据传回基站.

2.2 能量模型

设定一个初始阈值 d_0 ,当 $d < d_0$ 时,即为近距离传输,节点能耗 $E_T(l, d)$ 与 d 的平方成正比,即 $n = 2$,此时 $E_T(l, d)$ 可表示为

$$E_T(l, d) = E_{\text{Tx-elec}}(l) + E_{\text{Tx-amp}}(l, d) = l \times E_{\text{elec}} + l \times \epsilon_{\text{fs}} d^2 = l_i + u_i \times d^n \quad (1)$$

当为远距离即 $d \geq d_0$ 时,能量损耗随着距离的增大急剧增加,节点能耗 $E_T(l, d)$ 与 d 的 4 次方成正比,即 $n = 4$,此时 $E_T(l, d)$ 可表示为

$$E_T(l, d) = E_{\text{Tx-elec}}(l) + E_{\text{Tx-amp}}(l, d) = l \times E_{\text{elec}} + l \times \epsilon_{\text{mp}} d^4 = l_i + u_i \cdot d^n \quad (2)$$

式(1)和式(2)中 $\epsilon_{\text{fs}}, \epsilon_{\text{mp}}$ 分别为这两种模型中功率放大所需的能量.

另外,节点接收 l 比特的数据消耗的能量为

$$E_r(l) = l \times E_{\text{elec}} \quad (3)$$

E_{elec} 为单位比特数据的电路耗能, E_{elec} 的大小与发射端的编解码器、滤波器、调制解调等因素有关.在可接受的容错率下,射频放大器的能量消耗,依赖于传输的距离^[4].由式(3)可知,接受信息也需较大能量消耗,故协议不但要试图减少传输距离,也

要减少接收和传输信息的数量,故节点处的数据融合是必要的,数据融合也消耗一定的能量.在式(1)和式(2)中,数据为簇内传输时,取 $n = 2$;簇头与 sink 节点通信时,取 $n = 4$.无线传输能量消耗模型如图 1 所示^[5].

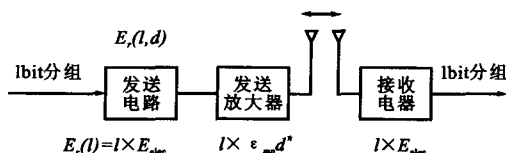


图1 无线传输能量消耗模型

传感器网络的生命周期是由传感器网络所提供的服务来决定的^[6],定义为最后一个节点的能耗为零结束.

3 ADCP 算法具体描述

由于大多数无线传感器路由算法简单地采用传统 LEACH 算法中的簇首选择机制,没有将传感器节点的实际分布情况考虑进去,因此选择出来的簇头就会出现问題,从而导致传感器网络生命周期的提前结束,因此密度探求阶段是必要的.为了使网络以较低的能耗得到若干个比较均匀的簇,现将传感器网络分成 K_{opt} 个簇,选出 K_{opt} 个簇首,每个节点以 R_{probe} (密度的探测半径) 为半径的范围内向所有节点发送 hello 消息来感知其周围节点的密度^[7].

3.1 探求节点密度

以 N_i 表示传感器网络中的第 i 个节点, $\text{Dist}(N_i, N_j)$ 表示传感器网络中任意两点之间的直线距,当 $\text{Dist}(N_i, N_j)$ 小于 R_{probe} 时, N_i 与 N_j 就为邻居节点.当某个节点的邻居节点很多,就表示以该节点为圆心, R_{probe} 为半径的圆形区域平均分布的节点比较多,单位区域的密度比较高.在探求节点密度的阶段中,各节点向以自己为圆心, R_{probe} 为半径向周围节点发送消息,从而能够计算出传感器网络中的第 i 个节点周围节点的个数.

3.2 簇首选择

在簇首选择阶段中将选出 K_{opt} 个簇首.首先设定相应的阈值 $T(n)$,然后将每个节点在簇首选择的过程中产生的随机数 a 与阈值 $T(n)$ 进行比较:

如果随机数 a 小于阈值 $T(n)$,则 PLAIN 节点将成为候选节点.

如果随机数 a 大于阈值 $T(n)$,则 PLAIN 节点就为非簇首节点.

在阈值进行设定的时候,引入节点的剩余能量

E_{remain} . 候选节点将广播消息 COMPETE_MSG 来表达其意愿, 消息中有 id 号和剩余能量 E_{remain} , 同时设定一个定时器值 T_{out} , T_{out} 为最大竞争时间. 现设 x 为剩余能耗与初始能耗之比, 当 $x > 50\%$ 时, 阈值计算如下:

$$T(n) = \frac{p}{1 - p \times [r \bmod (1/p)]} \quad (4)$$

式中, p 为簇首在所有节点中所占的百分比, r 为当前的选举轮数. 当 $x < 0.5$ 时, 即剩余能量小于初始能量的 50% 时, 阈值计算如下:

$$T(n) = 2P * E_{\text{remain}} / E_{\text{initial}} \quad (5)$$

式中, E_{remain} 为节点的剩余能量, E_{initial} 为节点的初始能量.

每个候选节点在 T_{out} 的限定的时间内是否有能量更高的候选节点, 当找到一个能量更高的候选节点时, 它将立即放弃竞争; 如果候选节点在竞争范围内没有比它能量更高的节点时, 它将最终被选为簇首节点, 这样在每个节点的竞争范围内将有一个簇首节点产生, 并且在稠密区域中的候选节点竞争范围小一些, 所以稠密区域中的候选节点比其他区域的候选节点更有可能竞争成功^[8]. 以下为簇首选择算法:

$\forall n_i \in N$;

state = PLAIN; // 初始化

$a = \text{Random}(0, 1)$; // 产生一个随机数

if ($x < 0.5$)

$$T(n) = \frac{p}{1 - p \times [r \bmod (1/p)]};$$

else if ($x > 0.5$)

$$T(n) = 2P * E_{\text{remain}} / E_{\text{initial}};$$

if ($a < T(n)$)

state = $N_{\text{candidate}}$;

else

state = PLAIN;

broadcast COMPETE_MSG(id, E); // 广播消息

do{

msg = receive COMPETE_MSG;

$n_j = \text{msg.getId}()$;

If ($d_{ij} < R_{\text{competes-i}}$)

If ($(n_i \cdot E_{\text{residual}} < n_j \cdot E_{\text{residual}}) \parallel (n_i \cdot E_{\text{residual}} = n_j \cdot E_{\text{residual}} \parallel n_i > n_j)$)

state = PLAIN;

}While ($T < T_{\text{out}}$);

if (state == $N_{\text{candidate}}$)

state = HEAD;

3.3 簇的形成阶段

在簇的形成阶段, 每个簇首在网络中广播消息,

其他普通节点接收消息并根据自己与簇首的距离和簇首负载是否均衡来选择加入哪个簇. 在图 2 中, 如果某个普通节点没有落在任何一个簇首的 $R_{\text{competes-i}}$ 半径范围内, 为了减少传输时的能量消耗, 它将选择最近的簇加入; 如果某个普通节点落在一个或多个簇首的 $R_{\text{competes-i}}$ 半径范围内, 为了平衡每个簇的负载, 它将选择一个最小的簇加入.

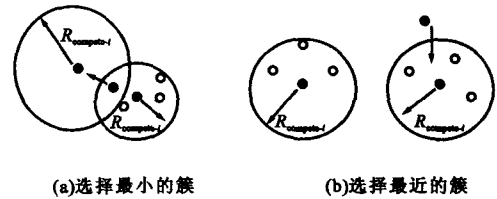


图 2 节点选择簇并加入

传感器网络被平均分成 K_{opt} 个簇, K_{opt} 个簇能够覆盖整个网络^[9]. 满足如下:

$$E(R) = \sqrt{\frac{A}{\pi K_{\text{opt}}}} \quad (6)$$

使得

$$R_{\text{probe}} = E(R) \quad (7)$$

$$R_{\text{competes-i}} =$$

$$R_{\text{probe}} \cdot \sqrt{1 + \frac{E(\text{Num}_{\text{neighbors}}) - \text{Num}_{\text{neighbors-i}}}{E(\text{Num}_{\text{neighbors}})}} \quad (8)$$

邻居节点的数学期望值为

$$E(\text{Num}_{\text{neighbors}}) = \frac{N}{K_{\text{opt}}} \quad (9)$$

设 N_i 邻居节点的个数为 N_{num} , 如果 N_{num} 大于 $\frac{N}{K_{\text{opt}}}$, 那么该节点处在密度较大的区域里; 如果 N_{num} 小于 $\frac{N}{K_{\text{opt}}}$, 那么该节点处在稀疏区域里.

4 仿真实验

采用 NS-2.27 环境下仿真实现 ADCP-LEACH 协议并对其性能进行评价, 仿真过程中生命周期被定义为传感器网络从部署开始到网络中最后一个节点能耗为零结束.

从图 3 中可以看出, 由于 ADCP-LEACH 协议多数情况下选出的簇首个数更加逼近 K_{opt} 的值, 所以, 无论在均匀场景还是在非均匀场景, ADCP-LEACH 性能均好于 LEACH.

从表 1 可以看出, 当初始能量分别为 2J、5J 和 10J 时, 网络生命周期大幅提高, 相比 LEACH 算法, ADCP-LEACH 算法在 FND、HND 和 LND 上均提高 20% 以上, 有效地延长了网络的生命周期.

表 1 两种算法在网络生存时间方面的比较

协议	初始 能量/J	FND/s	HND/s	LND/s
LEACH		399	536	658
ADCP-LEACH	2	488	652	840
提高倍数		22.3%	21.6%	27.6%
LEACH		1 102	1 389	1 756
ADCP-LEACH	5	1 349	1 754	2 189
提高倍数		22.4%	26.3%	24.7%
LEACH		2 196	2 543	3 238
ADCP-LEACH	10	2 687	3 396	4 452
提高倍数		22.3%	33.5%	37.4%

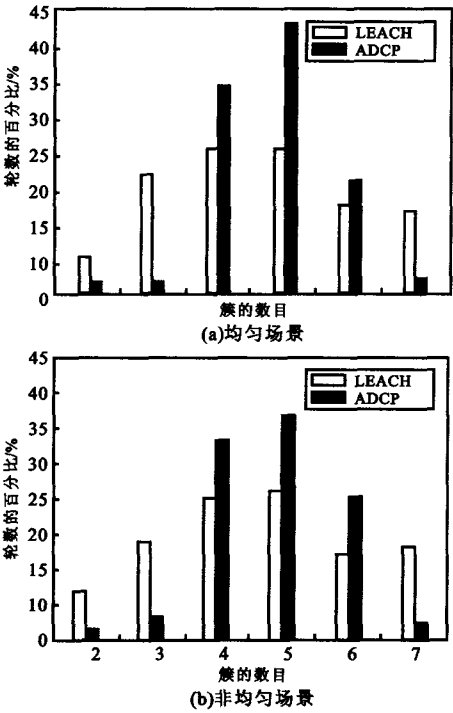


图 3 簇首数目比较

5 结束语

本文以 LEACH 协议为基础,提出了一种新的

分布式的、能量高效的自适应分布式算法 ADCP-LEACH,该算法主要从簇首选择和最优簇首数的设置两方面做了改进。仿真结果显示 ADCP-LEACH 性能明显优于 LEACH 协议,可以显著提高整个网络的能量有效性,延长网络寿命,在均匀网络和非均匀网络中网络生存时间分别延长了 20% 以上,有效地证明了该算法的可行性。

参考文献:

[1] Heinzelman W. Application - specific protocol architectures for wireless networks[R]. Boston: Massachusetts Institute of Technology, 2000.

[2] 彭静,刘光祜,谢世欢.无线传感器网络路由协议研究现状与趋势[J].计算机应用研究,2007(2):4-9.

[3] 任丰原,黄海宁,林闯.无线传感器网络[J].软件学报,2003(7):1282-1291.

[4] 耿晓义,柴乔林,张擎.一种能量均衡的无线传感器网络分簇算法[J].计算机工程与应用,2007(33):141-143.

[5] 姜华,袁晓兵,沈杰,等.无线传感器网络中信道接入分簇算法的研究[J].计算机工程与应用,2006(7):22-27.

[6] 王稼.一种基于分簇的无线传感器网络路由协议的设计[J].微纳电子技术,2007,44(7):490-493.

[7] 陈丹,郑增威,李际军.无线传感器网络研究综述[J].计算机测量与控制,2004,12(8):701-704.

[8] 郑增威,吴朝晖.若干无线传感器网络路由协议比较研究[J].计算机工程与设计,2003,24(9):28-31.

[9] 张德跃.无线传感器网络分簇路由算法研究[D].山东:山东师范大学,2008.

作者简介:

杜风雷 男,硕士,讲师.研究方向为计算机技术应用.