

# 能量高效的成簇路由优化算法

侯 华<sup>1</sup>, 宋 彬<sup>1</sup>, 周武旻<sup>2</sup>

(1 河北工程大学 移动通信研究室, 河北 邯郸 056038;

2 中国科学技术大学 无线信息网络实验室, 安徽 合肥 230027)

**摘 要:** 为了提高无线传感器网络的能量利用效率, 通过对分簇网络的研究, 提出了一种能量高效的成簇路由优化算法. 该算法对 LEACH 算法的簇头选举策略进行了优化, 在簇头选举时考虑了节点的能量因素; 在数据传输阶段, 借助直线方程建立了多跳路由算法. 仿真结果表明, 该算法能有效地提高网络的能量利用效率.

**关键词:** 无线传感器网络; 能量利用效率; 多跳路由; 分簇

中图分类号: TN929

文献标识码: A

文章编号: 1000-7180(2015)07-0121-04

## Energy Efficient Clustering Routing Optimization Algorithm

HOU Hua<sup>1</sup>, SONG Bin<sup>1</sup>, ZHOU Wu-yang<sup>2</sup>

(1 Mobile Communications Research Laboratory, Hebei University of Engineering, Handan 056038, China;

2 Wireless Information Network Lab, USTC, Hefei 230027, China)

**Abstract:** In order to improve the energy utilization of wireless sensor network, a kind of energy efficient clustering routing optimization algorithm is put forward, which is based on the research of clustering network. The algorithm optimizes the cluster head election strategy of LEACH, and it also consider the node energy factor in the cluster head election stage. In data transmission stage, the algorithm establishes multi-hop routing algorithm with the help of the linear equation. The simulation results show that the proposed algorithm can effectively improve the network energy utilization.

**Key words:** wireless sensor network; energy utilization; multi-hop routing; clustering

### 1 引言

无线传感器网络被广泛应用于军事、环境、智能家居、工业等领域<sup>[1]</sup>. 但由于传感器节点携带的能量有限, 其部署之后能量不能得到补充, 因此使传感器节点有限的能量得到高效利用是延长网络生存周期的主要方法之一<sup>[2-3]</sup>.

基于分簇的层次型路由算法是将无线传感器网络中的节点划分为簇头和簇内节点两部分. 簇内节点负责采集监测区域内的数据并将其发送给簇头, 簇头负责收集簇内节点的数据, 融合处理后以单跳的通信方式转发给基站(BS), 如 LEACH、EAERP 等算法<sup>[4-6]</sup>. 但该类算法在簇头选举时没有考虑节点的能量因素, 低能量的节点也会被当选为簇头, 这会

加速该类节点的能量消耗, 导致能量空洞现象的出现. 同时这类协议采用的都是单跳路由模型, 因此在大规模网络中距离基站比较远的节点能量消耗过快, 容易致使这类节点过早死亡.

本文基于能量高效利用的思想设计了一种能量高效的成簇路由优化算法. 该算法通过选举高能量的节点当选为簇头, 使各节点的能量消耗达到均衡; 同时借助直线方程构建多跳路由来均衡节点的能量消耗.

### 2 算法设计

#### 2.1 网络模型

为了便于研究, 网络中的每个节点都可以用坐标表示, 并作出如下假设:

收稿日期: 2014-09-27; 修回日期: 2014-11-06

基金项目: 河北省高等学校科学技术研究重点项目“能量有效的无线传感器网络跨层优化技术研究”(ZH2011222)

(1)  $N$  个传感器节点随机的分布在一个巨大的正方形区域, 网络中的第  $i$  个传感器节点用  $s_i$  表示, 其中  $i = \{1, 2, \dots, N\}$ , 基站位于正方形区域外的一侧。

(2) 传感器节点一旦部署之后不可再移动, 且每个传感器节点在部署时都会被分配一个唯一的标识符。

(3) 若已知发送节点的发射功率, 传感器节点可以根据接收信号的强度计算出发送节点距其的近似距离。

### 2.2 能量消耗模型

本文采用与文献[4]中相同的能量消耗模型, 以发送者和接收者之间的距离作为参考依据, 同时考虑了自由空间模型和多径衰落模型。传感器节点将  $l$  bit 的数据发送  $d$  距离远时的能量消耗为

$$E_{TX}(l, d) = \begin{cases} lE_{elec} + \epsilon_{fs}d^2 & , d < d_0 \\ lE_{elec} + \epsilon_{mp}d^4 & , d \geq d_0 \end{cases} \quad (1)$$

节点接收  $l$  bit 数据包需要消耗的能量为

$$E_{RX}(l) = lE_{elec} \quad (2)$$

式中,  $E_{elec}$  代表收发电路的损耗能量,  $\epsilon_{fs}$  和  $\epsilon_{mp}$  为放大器系数,  $d_0$  表示临界距离, 当数据传输距离小于临界距离  $d_0$  时, 就采用自由空间模型, 此时的能量消耗和距离的 2 次方成正比; 当数据传输距离大于临界距离  $d_0$  时, 采用多径衰落模型, 此时的能量消耗和距离的 4 次方成正比。

### 2.3 簇头选举阈值

在 LEACH 协议中, 网络中的每个节点首先产生一个随机数, 若该随机数大于簇头的选举阈值  $T(s_i)$ , 则该节点就当选为簇头, 否则不当选。每个节点参加簇头选举的阈值可由式(3)确定:

$$T(s_i) = \begin{cases} \frac{P}{1 - P * (r \bmod \frac{1}{P})} & , \text{if } s_i \in G \\ 0 & , \text{otherwise} \end{cases} \quad (3)$$

式中,  $P$  为节点当选为簇头的概率,  $r$  为选举轮数,  $r \bmod \frac{1}{P}$  为这一轮循环中当选过簇头的节点个数,  $G$  是这一轮循环中未当选过簇头的节点集合。

网络运行一段时间之后, 各节点的能量是不相同的, 为了提高网络的能量利用效率, 应该让能量大的节点当选为簇头, 能量小的节点当选为簇内节点。在 LEACH 协议中, 每个节点参加簇头选举的概率都为  $P$ , 这会降低节点的能量利用效率。针对这个问

题, 本文对 LEACH 协议中的簇头选举概率  $P$  做了进一步改进, 改进后的簇头选举概率  $P_{s_i, CH}$  如式(4)所示。

$$P_{s_i, CH} = \frac{E_{s_i}}{E_{GlobalMax}} \cdot P, \quad (4)$$

$$E_{GlobalMax} = \text{Max}\{E_{s_i} \mid s_i \in \{1, 2, \dots, N\}\} \quad (5)$$

式中,  $P_{s_i, CH}$  为网络中的任意节点  $s_i$  参加簇头选举的概率,  $E_{s_i}$  为节点的当前能量,  $E_{GlobalMax}$  为网络所有节点中的最大能量。

经过上述改进后, 簇头的选举阈值可表示为

$$T(s_i) = \begin{cases} \frac{P_{s_i, CH}}{1 - P_{s_i, CH} * (r \bmod \frac{1}{P_{s_i, CH}})} & , \text{if } s_i \in G \\ 0 & , \text{otherwise} \end{cases} \quad (6)$$

簇头选举结束后, 依 LEACH 算法的成簇策略建簇, 之后进入数据传输阶段。

### 2.4 路由

在路由选择阶段, 为了消除大规模网络中距离基站比较远的节点能量消耗过快的现象, 同时使网络具有良好的连通性, 本文借助直线方程设计了能量高效的多跳路由协议。

如图 1 所示,  $CH_{s_i}$ ,  $CH_{s_j}$  和  $CH_{s_m}$  分别为簇头节点,  $R_{CH_{s_i}}$  为簇头  $CH_{s_i}$  的最大通信范围。首先, 簇头  $CH_{s_i}$  根据式(7)求得自己的路由表  $\text{RoutTable}_{CH_{s_i}}$ , 然后簇头  $CH_{s_i}$  按下文提出的路由协议在路由表  $\text{RoutTable}_{CH_{s_i}}$  中选取下一跳中继节点。首先, 利用斜率公式, 如式(8), 得到由簇头  $CH_{s_i}$  和  $BS$  这两点所确定的直线  $l_{CH_{s_i}, BS}$  的斜率  $K_{CH_{s_i}, BS}$ , 其中  $(x_{CH_{s_i}}, y_{CH_{s_i}})$  和  $(x_{BS}, y_{BS})$  分别为簇头  $CH_{s_i}$  和  $BS$  的坐标。

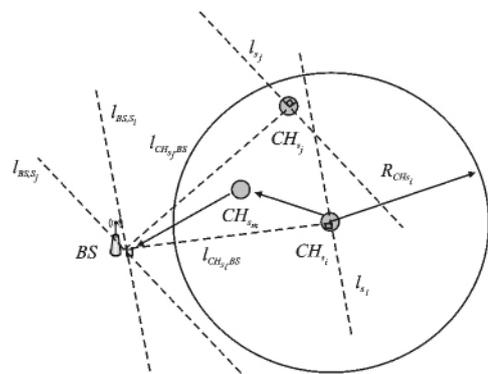


图 1 路由

$$\text{RoutTable}_{CH_{s_i}} = \{CH_{s_j} \mid d_{CH_{s_i},CH_{s_j}} < R_{CH_{s_i}}\}, \quad (7)$$

$$K_{CH_{s_i},BS} = \frac{y_{CH_{s_i}} - y_{BS}}{x_{CH_{s_i}} - x_{BS}}. \quad (8)$$

过  $BS$  做一条垂直于直线  $l_{CH_{s_i},BS}$  的垂线  $l_{BS,s_i}$ , 由直线互相垂直的判定定理可以计算出直线  $l_{BS,s_i}$  的斜率  $K_{BS,s_i}$ . 同样, 过簇头  $CH_{s_i}$  做一条垂直于直线  $l_{CH_{s_i},BS}$  的垂线  $l_{s_i}$ , 由直线互相垂直的判定定理可以计算出直线  $l_{s_i}$  的斜率  $K_{CH_{s_i},s_i}$ . 直线互相垂直的判定定理如式(9)所示, 其中  $K_1$  和  $K_2$  为两条互相垂直的直线的斜率:

$$K_1 \cdot K_2 = -1. \quad (9)$$

确定了直线  $l_{BS,s_i}$  和直线  $l_{CH_{s_i},BS}$  的斜率之后, 根据直线的点斜式方程求解出直线  $l_{BS,s_i}$  和直线  $l_{CH_{s_i},BS}$  的直线方程. 直线的点斜式方程如式(10)所示, 其中  $k$  为直线的斜率,  $b$  为直线在  $y$  轴上的截距:

$$y = kx + b. \quad (10)$$

直线  $l_{BS,s_i}$  的直线方程确定之后, 根据点到直线的距离公式, 求得簇头  $CH_{s_i}$  和其路由表  $\text{RoutTable}_{s_i}$  中的任意簇头  $CH_{s_j}$  到直线  $l_{BS,s_i}$  的欧式距离  $d_{CH_{s_i},l_{CH_{s_i},s_j}}$  和  $d_{CH_{s_j},l_{CH_{s_i},s_j}}$ , 同理得  $d_{CH_{s_i},l_{CH_{s_i},s_j}}$  和  $d_{CH_{s_j},l_{CH_{s_i},s_j}}$ . 然后根据式(12)求得簇头  $CH_{s_i}$  的伪路由表  $\text{PseudoRoutTable}_{s_i}$ . 点到直线的距离公式如式(11)所示, 其中  $Ax + By + C = 0$  为直线方程的一般式,  $A, B, C$  分别为常数,  $d$  为点  $(x, y)$  到直线方程  $Ax + By + C = 0$  的欧式距离:

$$d = \frac{|Ax + By + C|}{\sqrt{A^2 + B^2}}. \quad (11)$$

$$\text{PseudoRoutTable}_{s_i} = \{s_j \mid d_{CH_{s_j},l_{CH_{s_i},s_j}} < d_{CH_{s_i},l_{CH_{s_i},s_j}}, d_{CH_{s_i},l_{CH_{s_i},s_j}} > d_{CH_{s_j},l_{CH_{s_i},s_j}}\} \quad (12)$$

伪路由表  $\text{PseudoRoutTable}_{s_i}$  确定之后, 簇头  $CH_{s_i}$  依概率  $P_{CH_{s_i},\text{route}}$  在  $\text{PseudoRoutTable}_{s_i}$  中选取下一跳节点. 概率  $P_{CH_{s_i},\text{route}}$  可由式(13)确定.

$$P_{CH_{s_i},\text{route}} = \frac{E_{CH_{s_j}}}{d_{CH_{s_i},CH_{s_j}}^2 + d_{CH_{s_i},BS}^2}, \quad (13)$$

式中,  $E_{CH_{s_j}}$  为簇头  $CH_{s_j}$  的剩余能量,  $d_{CH_{s_i},CH_{s_j}}^2 + d_{CH_{s_i},BS}^2$  为簇头  $CH_{s_i}$  选择簇头  $CH_{s_j}$  作为其下一跳节点的转发代价.

簇头  $CH_{s_i}$  的剩余能量越大, 簇头  $CH_{s_i}$  选择簇头  $CH_{s_j}$  作为其下一跳路由的转发代价越小, 簇头

$CH_{s_j}$  当选为簇头  $CH_{s_i}$  的下一跳路由的概率就越大. 优化后的路由由协议使路由的分配更加合理, 提高了簇头的能量利用效率, 增强了网络的可靠性.

### 3 性能评估

本文通过与 LEACH 算法和 EAERP 算法的仿真对比来评估所提算法的有效性和可行性. 仿真环境采用 omnetpp 和 matlab. 在仿真实验中, 节点分布在一个  $200 \text{ m} \times 200 \text{ m}$  的正方形区域, 节点总数为 400 个, 基站位于正方形区域外的  $(100, 250) \text{ m}$  位置. 其他仿真参数如表 1 所示.

表 1 仿真参数

参数	值
初始能量	0.5 J
$E_{\text{elec}}$	50 nJ/bit
$\epsilon_{\text{fs}}$	$10 \text{ pJ} \cdot \text{bit}^{-1} \cdot \text{m}^{-2}$
$\epsilon_{\text{amp}}$	$0.0013 \text{ pJ} \cdot \text{bit}^{-1} \cdot \text{m}^{-4}$
$d_0$	87 m
数据包大小	4 000 bit
控制包大小	200 bit
数据融合能量	$5 \text{ nJ} \cdot \text{bit}^{-1} \cdot \text{signal}^{-1}$

图 2 所示为所提算法与 LEACH 和 EAERP 两种算法生存周期的对比. 其中 LEACH, EAERP 和所提算法的第一个节点死亡时的仿真轮数分别为 245 轮, 634 轮和 906 轮, 可见所提算法的网络生存周期比 EAERP 算法提高了 43%, 比 LEACH 的提高了 270%. 原因在于, 在 LEACH 算法中, 簇头选举时没有考虑节点的能量因素, 低能量的节点可能会被当选为簇头, 使节点过早死亡; 同时, 与基站单跳的数据交换模式, 使得距离基站较远的簇头能量消耗较大, 这也会导致节点的过早死亡. EAERP 算法虽然也是单跳通信模式, 但是其应用了遗传算法进行分簇, 使得形成的分簇网络要优于较 LEACH 算法, 节点的能量利用效率也较高. 而所提算法采用多跳的数据交换模式与基站通信, 延长了网络的生存周期; 此外, 所提算法在簇头选举时从全局的角度考虑了节点的剩余能量, 所以所提算法的网络生存周期比 EAERP 算法要长.

在分簇网络中, 簇头的能量消耗要大于簇内节点, 簇头的能量消耗是否均衡, 直接影响着网络的生存周期. 本文随机选取了 20 轮簇头的能量消耗并进行了对比, 如图 3 所示. 其中, LEACH 算法中的簇

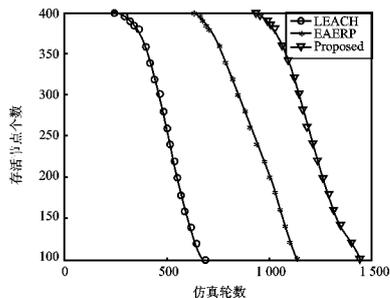


图2 网络生存周期

头能量消耗波动较大,EAERP 算法和所提算法的簇头能量消耗波动较小.这说明 EAERP 算法和所提算法能更好地平衡簇头之间的能耗.由图 3 还可以看出,所提算法簇头的能耗最低,说明所提算法的簇头能量利用效率最高.

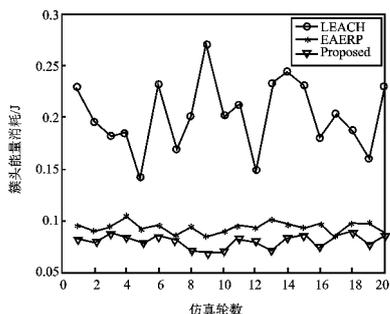


图3 簇头的能耗

当网络的第一个节点死亡时,网络的剩余能量越少,网络的能量利用效率就越高.同时,降低第一个节点死亡时网络的剩余能量也有利于提高网络的稳定周期.图 4 中记录了第一个节点死亡时网络的剩余能量与初始能量的比值,其中 LEACH 算法为 64%,EAERP 算法为 47%,所提算法为 40%.由图中的数据可以得出结论,所提算法的能量利用效率较 LEACH 和 EAERP 算法有了很大的提高,网络的稳定性也有所改善.

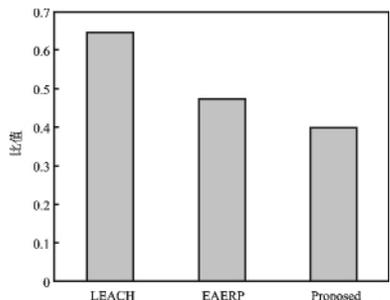


图4 第一个节点死亡时网络的剩余能量

#### 4 结束语

本文提出了一种能量高效的成簇路由优化算

法.该算法对 LEACH 算法的簇头选举阈值做了优化,在簇头选举时从全局的角度考虑了节点的剩余能量,延长了网络第一个节点的死亡时间;同时该算法利用直线方程寻找高效连通的多跳路由,在选择下一跳中继节点时,考虑了簇头的剩余能量和转发代价,使选举出的下一跳路由更适合担当数据转发的任务.但本文提出的算法在成簇阶段只考虑了网络的全局能量消耗,使得选出的簇头不是最优的,而且成簇的收敛速度也有待提高.下一步的工作是考虑局部能量消耗对网络的影响.同时为了降低成簇难度,可考虑将遗传算法应用到多跳分簇网络的优化中.

#### 参考文献:

- [1] Jennifer Yick, Biswanath Mukherjee, Dipak Ghosal. Wireless sensor network survey[J]. Computer Networks, 2008,52(12):2292-2330.
- [2] Giuseppe Anastasi, Marco Conti, Mario Di Francesco, Andrea Passarella " Energy conservation in wireless sensor networks: A survey" [J]. Ad Hoc Networks, 2009,7(3):537-568.
- [3] Hanghang Sun, Yuning Qian, Ruqiang Yan, Design and realization of an intelligent sensor node with its application in energy-aware WSNs[C]// Instrumentation and Measurement Technology Conference (I2MTC), 2012 IEEE International, Graz; IEEE, 2012:941-946.
- [4] Heinzelman W B, Chandrakasan A P, Balakrishnan H. An application-specific protocol architecture for wireless microsensor networks[J]. Wireless Communications IEEE Transactions on, 2002,1(4):660-670.
- [5] Enan A Khalil, Bara'a A. Attea Energy-aware evolutionary routing protocol for dynamic clustering of wireless sensor networks[J]. Swarm and Evolutionary Computation, 2011,1(4):195-203.
- [6] Geetha V, Kallapur P V, Sushma Tellajeera. Clustering in wireless sensor networks: performance comparison of LEACH & LEACH-C protocols using NS2[J]. Procedia Technology, 2012(4):163-170.

#### 作者简介:

侯 华 女,(1983-),博士,副教授,研究生导师.研究方向为移动通信、无线通信、认知无线电技术.  
 宋 彬 男,(1987-),研究生.研究方向为无线传感器网络. E-mail:happysongbin@163.com.  
 周武旻 男,(1972-),博士,教授,博士生导师.研究方向为多载波技术、信道编解码技术、跨层协议处理.