

能量有效的扇区划分 LEACH 改进算法

刘衍珩¹, 任安¹, 孙大洋², 王爱民¹, 钟辉³

(1. 吉林大学 计算机科学与技术学院, 长春 130012; 2. 吉林大学 通信工程学院, 长春 130012;
3. 吉林大学 党委组织部, 长春 130012)

摘要: 针对无线传感器网络生存期和能量有效性问题, 提出了一种基于多跳传输和节点 ID 扇形分布的低功耗自适应集簇分层型协议 (LEACH) 改进算法 MHS-LEACH. 该算法根据节点的剩余能量调整簇头, 并按虚拟扇区分配簇头节点 ID, 使得某簇头节点在同一扇区内寻找其父节点, 经过多次迭代与基站建立树形链路, 实现了簇头节点与基站间的多跳通信. 仿真结果表明, 与 LEACH 算法和 ID-LEACH 算法相比, MHS-LEACH 算法不仅有效地延长了无线传感器网络生存时间, 而且使整个网络的能量消耗更加均衡.

关键词: 无线传感器网络; 低功耗自适应集簇分层型协议 (LEACH); 多跳通信; 能量管理; 扇区划分

中图分类号: TP 393

文献标志码: A

文章编号: 0254 - 0037(2012)11 - 1683 - 06

Improved LEACH of the Energy Efficiency and Sector Distribution

LIU Yan-heng¹, REN An¹, SUN Da-yang², WANG Ai-min¹, ZHONG Hui³

(1. College of Computer Science and Technology, Jilin University, Changchun 130012, China;
2. School of Communication Engineering, Jilin University, Changchun 130012, China;
3. Organization Department, Jilin University, Changchun 130012, China)

Abstract: To solve the problem of the lifetime and the energy efficiency in wireless sensor networks, an energy efficient approach to improve the low energy adaptive clustering hierarchy (LEACH) named MHS-LEACH was proposed. The algorithm achieved the goal of energy efficiency by dividing sensing area into sectors and distributing cluster head IDs according to these sectors. The algorithm adjusted cluster heads according to the remaining energy of nodes, and distributed cluster head IDs according to virtual sectors, and looked for the father node of some cluster node among the cluster nodes in the same sector to build the tree link to the sink in order to achieve the multi-hops communication between the cluster nodes and the sink node. Experimental results indicate that compared with LEACH and ID-LEACH, MHS-LEACH not only prolongs the life time of the wireless sensor network efficiently, but also balances the energy consumption per round.

Key words: wireless sensor networks; low energy adaptive clustering hierarchy (LEACH); multi-hops communication; energy management; sectors distribution

低功耗自适应集簇分层型协议 (LEACH) 算法是无线传感器网络的经典分簇算法, 该算法由 Heinzelman 等^[1] 在 2000 年提出, 它具有较强的数据

融合能力, 并结合了基于分簇的能量有效路由、MAC 协议算法, 在延长网络生存期^[2] 方面有较好的性能. 然而, LEACH 算法进行分簇时, 所分的簇并

收稿日期: 2011-04-25.

基金项目: 国家自然科学基金资助项目 (61073164); 国家科技部国际合作与交流资助项目 (2008DFA12140).

作者简介: 刘衍珩 (1958—), 男, 教授, 博士生导师, 主要从事网络安全与管理、网络体系结构、无线传感器网络方面的研究,
E-mail: lyh_lb_lk@yahoo.com.cn.

不均衡,即每次分簇不能保证每个簇都含有相同的节点个数,并且不能保证被选簇头节点具有较大的剩余能量,其规模在动态随机地变化.在实际研究中发现,当网络传感器节点数量较多、簇头距离网络的汇聚节点较远时,将迅速地消耗能量,影响了整个网络的生存时间及能量消耗的均衡性.

近几年,研究者们提出了很多关于 LEACH 的改进算法,在不同方面改进了其性能. Gou 等^[3]在 2009 年提出了一种优先考虑节点当前能量和其能量接收率的 LEACH 改进算法,并且网络中的每个节点均可利用太阳能充电.该算法存在的问题是:由于每个节点的地理位置不同,每次充电不可能获得等量的能量,这样会造成能量的浪费. Thein 等^[4]在 2010 年提出的基于能量有效性^[5]的簇头选举算法虽然可以很好地平衡负载和延长网络生存期,但每次执行均要改变簇头的位置以分配所有节点的负载,算法执行比较复杂.文献^[6]提出了一种新的位置感知分簇算法(LACA),节点根据其地理位置决定是否作为簇头,有效抑制了孤立节点产生.响应式分布分簇算法(RDCA)^[7]根据局部拓扑信息进行分布式簇头选举,簇的划分依据代价函数,但是 RDCA 中的节点必须收集等待邻居节点信息,这样增大了网络延时,增加网络控制传输开销,从而降低了算法的整体性能.利用多跳通信的 LEACH 改进算法还有 Lindsey 等^[8]在 2002 年提出的 PEGASIS 算法,利用所有节点构成的链路进行数据发送和融合,最后将融合的数据发送给基站,但链路数量过多,开销过大.文献^[9]根据节点剩余能量的大小及存活情况选择单跳或者双跳通信,但算法中的阈值即判定参数需要进一步优化才能更明显地提高网络性能. Xu 等^[10]也提出了一种簇头间多跳通信的 LEACH 改进算法.通过计算簇头节点间的距离选择相互距离最近的簇头节点建立多跳通信传输链路,虽然在提高网络生存期方面优势明显,但计算量较大,复杂度较高.

从以上文献可以看出,簇头间的多跳通信的确可以解决远距离簇头节点与基站直接通信能耗过大、导致节点过早死亡的问题.文献^[11]提出了一种基于 ID 的无线传感器网络路由算法——ID-LEACH 算法,利用二进制编码建立多跳传输的树形链路进行数据传输和融合.

本文在 LEACH 算法以及 ID-LEACH 算法的基础上,提出了一种基于多跳传输和节点 ID 扇形分布的 LEACH 改进算法 MHS-LEACH,根据剩余能量调

整簇头节点,并根据簇头节点所在的扇区分配其扇形 ID,建立树形传输链路.仿真结果表明,根据簇头节点的扇形 ID 所建立的多跳传输链路能更有效地延长整个网络的生命周期,使其能量消耗更加均衡,有效地提高了无线传感器网络的能量有效性.

1 MHS-LEACH

1.1 网络模型

本算法利用扇区对簇头节点间传输链路的建立进行限制,其网络模型描述如下:

- 1) 网络存在一个基站 BS,位于二维平面坐标原点上.
- 2) 所有节点同构,静态随机分布在二维平面上,并且自身位置坐标已知.
- 3) 所有节点初始能量相同,都具备数据融合的功能.
- 4) 所有节点都能直接与 BS 进行通信.
- 5) 基站能获取各节点的剩余能量信息.

1.2 能量模型

本文采用文献^[8]的能量模型.假设发送端的能耗分为信号处理和功率放大 2 部分,接收端的能耗仅用于信号处理.当 2 个节点间距离小于 d_0 时采用自由空间路径损耗模型,距离较小,减少了彼此间的干扰;大于等于 d_0 时采用多径衰落模型^[12],发射功率大约与距离 4 次方成正比,能耗远比自由空间模型大.因此,当 2 个距离为 d 的节点之间发送和接收数据 l 时,发送端能耗为

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

接收端耗能为

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

式中: d_0 为区分自由空间路径损耗模型和多径衰落模型的距离阈值; l 为数据长度; E_{elec} 为电路发射、接收 1 bit 数据所消耗的能量; d 为 2 节点间的通信距离; ε_{fs} 、 ε_{mp} 分别为 2 种模型功率放大所需的能量.

节点间数据融合也需要一定能耗,用 E_{DA} 表示融合 1 bit 数据所消耗的能量,则发送端所消耗能量改为

$$E(l, d) = \begin{cases} l(E_{\text{elec}} + E_{\text{DA}}) + l\varepsilon_{\text{fs}}d^2, & d < d_0 \\ l(E_{\text{elec}} + E_{\text{DA}}) + l\varepsilon_{\text{mp}}d^4, & d \geq d_0 \end{cases} \quad (3)$$

本算法中采用式(3)作为发送端能量模型.

1.3 相关定义

1.3.1 节点的无效链路

节点 u 与节点 v 之间的链路 L 为无效链路当且

仅当存在节点 w , 节点 u, w 的欧氏距离 $d(u, w)$ 小于节点 u, v 的欧氏距离 $d(u, v)$.

节点若选择无效链路进行通信, 根据能量模型, 通信距离 d 越大, 能耗 $E_{TX}(l, d)$ 越大, 不利于网络节能.

1.3.2 平衡因子

平衡因子 ε 可用如下公式计算:

$$\varepsilon = \frac{E_a}{E_i} \quad (4)$$

式中: E_a 为网络某部分区域节点当前剩余能量的均值; E_i 为网络中所有节点共同的初始能量. 平衡因子反映了网络当前的能量状态.

1.3.3 节点信息

节点信息用于标识节点的状态信息, 具体信息如下: ID 表示节点的物理标识; Xd 表示节点横坐标; Yd 节点表示纵坐标; sector_id 表示节点所在扇形编号; fid 表示父节点物理标识; distance 表示节点与其父节点的通信距离; E_r 表示节点剩余能量; E_i 表示节点初始能量; E_{th} 表示节点能量阈值.

1.4 算法描述

本算法以轮数 r 为单位执行, 每轮主要分为成簇阶段、簇头 ID 分配及树形链路建立阶段和数据传输阶段. 首先根据节点剩余能量成簇并调整簇头, 然后根据簇头节点的位置确定其所在的扇区编号, 为该节点分配扇形 ID 并建立树形传输链路, 最后簇头节点对簇内节点进行数据融合, 与簇间节点进行多跳通信, 最终将数据传送到基站.

1.4.1 成簇阶段

1) 利用 LEACH 算法成簇.

在簇头选择的过程中, 每个节点都要产生一个 $0 \sim 1$ 的随机数, 与相应的阈值 $T(n)$ 进行比较, 若该随机数小于阈值 $T(n)$, 则当选为簇头, 反之则为非簇头节点. $T(n)$ 为

$$T(n) = \begin{cases} \frac{p}{1 - p \times \left(r \bmod \frac{1}{p} \right)}, & n \in G \\ 0, & n \notin G \end{cases} \quad (5)$$

式中: p 为簇头占总节点数的百分比; r 为当前轮数; G 为第 r 轮未被选为簇头的节点集.

2) 根据簇内剩余能量调整簇头.

① 计算簇内节点剩余能量的平均值 E_a 与平衡因子 ε , 其计算公式为

$$E_a = \frac{\sum_{i=1}^N E_{i,r}}{N} \quad (6)$$

式中: $E_{i,r}$ 为节点 i 的剩余能量; N 为网络中满足如下条件的节点总数.

a. 节点必须为存活节点.

b. 节点的剩余能量 E_i 不小于其能量阈值 E_{th} . 将 E_a 带入式(4)计算平衡因子 ε .

② 计算簇内每个节点剩余能量占初始能量的比例 x , 其计算公式为

$$x = \frac{E_r}{E_i} \quad (7)$$

③ 比较簇头节点的 x 与 ε .

a. 若簇头节点的 x 小于 ε , 则对该簇内其他节点的 x 进行升序排列, 将该序列所有的 x 依次与 ε 比较, 直到找到第 1 个 x 大于 ε 的节点, 用该节点替换当前簇头, 则此次簇头调整完毕.

b. 若簇头节点的 x 大于 ε , 则无需进行调整.

以上过程结束后, 成簇阶段完毕.

1.4.2 簇头 ID 分配以及树形链路建立阶段

1) 簇头 ID 分配.

① 将网络区域划分成 M 个扇区并编号.

② 遍历每个簇头, 根据簇头位置判断其所在扇形编号, 并将该扇形编号作为该簇头节点的扇形 ID, 即 sector_id, 直到所有簇头遍历结束, 则簇头 ID 分配完毕.

2) 树形链路建立.

① 簇头选择其父节点的规则如下.

a. 簇头节点 sector_id 与其父节点 sector_id 相同.

b. 簇头节点与基站的通信距离大于其父节点与基站的通信距离.

c. 簇头节点与其父节点间的 distance 小于该簇头节点与其任意邻居簇头节点的通信距离.

若满足此条件的节点有多个, 则选择节点坐标与该簇头节点坐标的角度差最小的簇头节点. 若角度差仍然相等, 选择节点 ID 最小的簇头节点作为父节点.

② 按节点 ID 遍历所有簇头节点, 按以上规则为每个簇头节点确定其父节点, 直到所有簇头节点遍历结束, 则此时树形链路建立完毕, 如图 1 所示.

1.4.3 数据传输阶段

1) 每个簇内的非簇头节点与其所在簇的簇头节点直接通信, 将需要传送的数据直接传送给簇头节点.

2) 簇头节点对收集到的数据进行融合. 某个簇头节点将融合后的数据传送给其所在树形链路的

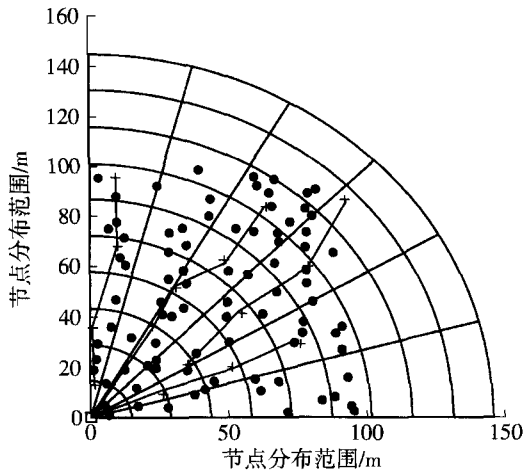


图1 MHS-LEACH网络节点分布
Fig.1 MHS-LEACH nodes distribution

父节点.

3) 父节点首先将本簇内的非簇头节点传送的数据进行融合,再继续向上层节点传送所有需要转发的数据,直到需要传输的数据包到达根节点(即基站),此次数据传输阶段完毕.

2 仿真结果与分析

为评价 MHS-LEACH 算法的性能,在 $100\text{ m} \times 100\text{ m}$ 的矩形区域内随机部署 100 个节点,如图 2 所示.

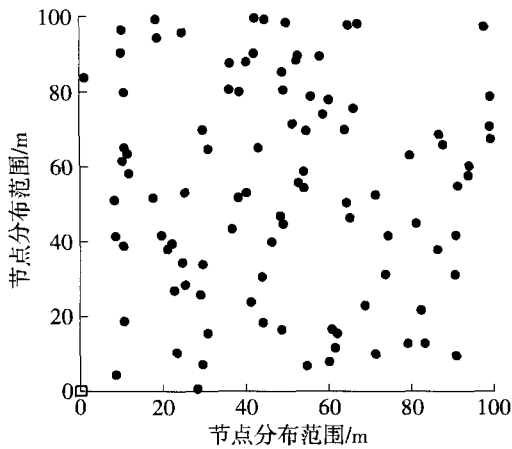


图2 初始网络节点分布
Fig.2 Initial nodes distribution

在仿真实验中,分别就第 1 个节点死亡时间 FND 和半数节点死亡时间 HND 对 LEACH 算法、ID-LEACH 算法和 MHS-LEACH 算法进行对比评价.具体实验参数如下.

数据包长度为 2 000 bit,节点的初始能量 E_i 为

0.5 J ,节点分布范围为 $100\text{ m} \times 100\text{ m}$,BS 坐标为 $(0,0)$,节点数为 100,电路消耗能量 E_c 为 $5.0 \times 10^{-8}\text{ J/bit}$,数据融合能耗 E_{DA} 为 $5.0 \times 10^{-9}\text{ J/bit}$,信道传播模型参数 ϵ_{fs} 为 $1.0 \times 10^{-11}\text{ J}/(\text{bit} \cdot \text{m}^2)$, ϵ_{mp} 为 $1.3 \times 10^{-15}\text{ J}/(\text{bit} \cdot \text{m}^4)$.

2.1 成簇方式与生存时间关系分析

MHS-LEACH 算法在成簇的过程中,引入剩余能量,对簇头节点进行调整,使得剩余能量较高的节点被选为簇头节点,推迟了低能量节点的死亡时间.而 LEACH 算法成簇时,由于簇头选择的随机性,剩余能量低的节点也可以当选为簇头,会加速能量低的节点死亡.如图 3 所示,LEACH 算法和 ID-LEACH 算法的 FND 分别为 846 轮和 1 292 轮, MHS-LEACH 算法的 FND 为 2 356 轮,比 LEACH 算法延长了 1 510 轮,比 ID-LEACH 算法延长了 1 064 轮,而 LEACH 算法和 ID-LEACH 算法的 HND 分别为 1 075 轮和 2 062 轮, MHS-LEACH 算法的 HND 为 2 577 轮,比 LEACH 算法延长了 1 502 轮,比 ID-LEACH 算法延长了 515 轮.由此可见, MHS-LEACH 算法大幅度提升了网络的生存时间,显著优于 LEACH 算法和 ID-LEACH 算法.

LEACH 算法和 ID-LEACH 算法中,网络运行到第 2 100 轮和第 2 610 轮节点全部死亡,从第 1 个节点死亡到最后 1 个节点死亡分别经历了 1 254 轮和 1 318 轮. MHS-LEACH 网络在第 2 750 轮节点全部死亡,节点死亡经历了 394 轮.可以看出,采用 MHS-LEACH 算法,节点陆续死亡的时间较 LEACH 算法和 ID-LEACH 算法少得多,网络能量下降均衡.

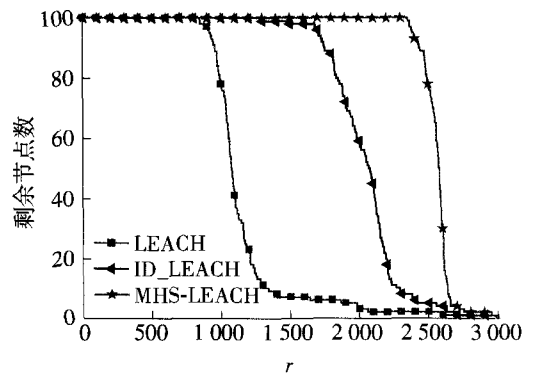


图3 剩余节点的数量随轮数 r 的变化
Fig.3 Number of the remaining nodes changing with round r

2.2 通信方式与能量消耗关系分析

MHS-LEACH 算法根据簇头节点的位置确定其所在的扇区,并为该节点分配扇形 ID 的优势在于:

如果按其他形状的区域分配 ID, 例如环形, 则树形链路建立的过程中有可能产生无效链路, 根据能耗模型, 会造成该通信链路上能耗偏大的情况, 采用扇区可以避免无效链路的产生, 有效地改善了网络能耗的均衡性。

簇头扇形 ID 分配过程结束后, 会出现某些扇区不存在簇头的情况。此情况不会影响算法的运行, 因为该算法中扇区的存在是为了避免簇头节点的无效链路产生, 所以在簇头节点间建立树形链路的过程中, 扇区起到了限制作用, 即使某些扇区中不存在簇头, 也不会对数据通信情况及网络能耗均衡性造成影响。

MHS-LEACH 算法中树形链路的建立, 将簇头节点与基站直接通信的方式改为多跳通信方式, 并且保证每条通信链路在备选链路中最短, 避免了远距离簇头与基站通信时能耗过大的情况, 均衡了簇头节点的能量消耗。

另外, 如果单纯根据节点位置已知的前提寻找与该节点距离最近的邻居节点进行数据转发, 缺少 MHS-LEACH 算法中树形传输链路的限制。当网络规模较大时, 会造成数据转发跳数过多, 传输延迟过大, 影响整个网络中各节点与基站之间的数据传输效率。

如图 4 所示, MHS-LEACH 算法每轮消耗的能量比 LEACH 算法和 ID-LEACH 算法更加均衡。ID-LEACH 算法虽然同样采用多跳通信的方式, 但由于在建立通信链路时, 利用二进制编码形成树形结构, 没有考虑通信距离对能量消耗造成的影响, 仍然存在某些较长的通信链路上能耗过大的问题。

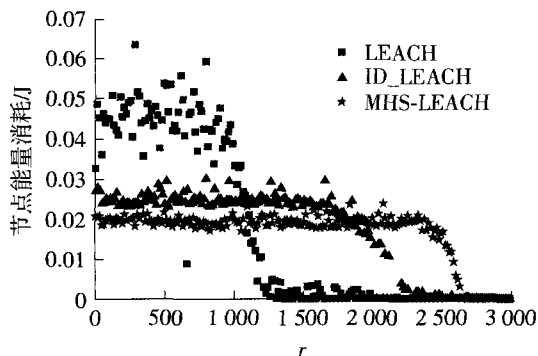


图 4 所有节点在每轮的能量消耗

Fig. 4 Energy consumption of all nodes in each round

由此可见, MHS-LEACH 算法根据簇头节点位置为其分配扇形 ID, 从而建立多跳树形传输链路, 虽然提高了一定的计算复杂度和时间, 但是从总体

上延长了单个节点和整个网络的生存时间, 均衡了整个网络的能量消耗, 因此这样做是值得的。

此外, LEACH 算法和 ID-LEACH 算法的网络能量分别在 1 250 余轮和 2 200 余轮的时间内逐渐耗尽, 采用 MHS-LEACH 算法 (见图 5), 网络的能量在 2 700 余轮的时间内逐渐耗尽, 较 LEACH 算法和 ID-LEACH 算法持续时间更长, 能量消耗更均衡, 节点能量得到有效利用。

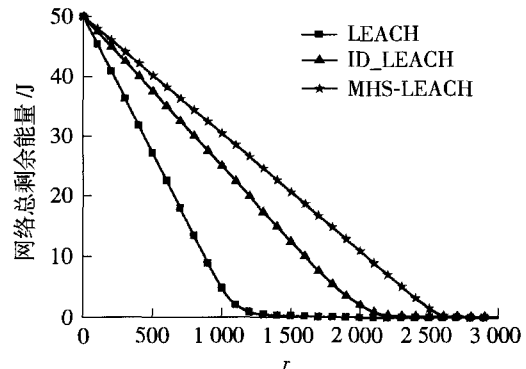


图 5 整个网络总的剩余能量随轮数 r 的变化

Fig. 5 Remaining energy of the entire network changing with round r

3 结论

1) MHS-LEACH 算法实现简单, 计算量较小, 复杂度低。算法在划分扇区以及分配扇形 ID 时只需根据几何定理, 依照节点坐标所处的角度进行分配, 算法可读性及应用性较强。

2) MHS-LEACH 算法均衡了网络的通信能耗。平衡因子的引入对节点能量实现了动态管理。在算法执行的每轮单位时间里, 均进行一次利用平衡因子检测节点剩余能量的工作, 动态调整簇头, 使网络能耗均衡, 有效延长网络的生存期。

3) 树形链路的建立过程相比其他算法, 在解决远距离节点能耗过快问题上更加简单, 使节点能耗得到进一步有效利用, 更能提高网络能量的有效性。

4) 利用 MHS-LEACH 算法的网络整体生存时间大幅提高。

参考文献:

- [1] HEINZELMAN W R, CHANDRAKASAN A, BALAKRISHNAN H. Energy-efficient communication protocol for wireless micro sensor networks [C] // Proceedings of the 33rd Hawaii International Conference on System Sciences. Los Alamitos: IEEE Computer Society,

- 2000: 3005-3014.
- [2] WANG H, AGOULMINE N, MA M, et al. Network lifetime optimization in wireless sensor networks[J]. IEEE Journal on Selected Areas in Communications, 2010, 9(28): 1127-1137.
- [3] GOU Hao-song, YOO Youngh-wan, ZENG Hong-qing. A partition-based LEACH algorithm for wireless sensor networks[C] // Ninth IEEE International Conference on Computer and Information Technology. Daejeon: ACM, 2009: 40-45.
- [4] THEIN M C M, THEIN T. An energy efficient cluster-head selection for wireless sensor networks [C] // 2010 International Conference on Intelligent Systems, Modeling and Simulation. Liverpool: IEEE, 2010: 287-291.
- [5] ALIPPI C, ANASTASI G, FRANCESCO M D, et al. An adaptive sampling algorithm for effective energy management in wireless sensor networks with energy-hungry sensors [J]. IEEE Trans on Instrumentation and Measurement, 2010, 2(59): 335-344.
- [6] 田炜, 杨震. 新的位置感知分簇算法[J]. 通信学报, 2010, 7(3): 25-30.
TIAN Wei, YANG Zhen. New location aware clustering algorithm[J]. Journal on Communications, 2010, 7(3): 25-30. (in Chinese)
- [7] 胡静, 沈连丰, 宋铁成, 等. 新的无线传感器网络分簇算法[J]. 通信学报, 2008, 29(7): 20-26.
HU Jing, SHEN Lian-feng, SONG Tie-cheng, et al. New clustering algorithm for wireless sensor networks [J]. Journal on Communications, 2008, 29(7): 20-26. (in Chinese)
- [8] LINDSEY S, RAGHAVENDRA C S. PEGASIS: power efficient gathering in sensor information systems[C] // Proc of the IEEE Aerospace Conf. Montana: IEEE Aerospace and Electronic Systems Society, 2002: 1125-1130.
- [9] GUO Li-qing, XIE Yi, YANG Chen-hui, et al. Improvement on LEACH by combining adaptive cluster head election and two-hop transmission [C] // 2010 International Conference on Machine Learning and Cybernetics. Qingdao: IEEE, 2010: 1678-1683.
- [10] XU Long-long, ZHANG Jian-jun. Improved LEACH cluster head multi-hops algorithm in wireless sensor networks[C] // 2010 Ninth International Symposium on Distributed Computing and Applications to Business Engineering and Science. Hong Kong: IEEE, 2010: 263-267.
- [11] TORKZABAN V, RAHMANI S, DEHGHAN M. An ID-based routing protocol for WSN [C] // First Asian Himalayas International Conference on Internet. Kathmandu: IEEE, 2009: 1-6.
- [12] HEINZELMAN W R, CHANDRAKASAN A, BALAKRISHNAN H. An application-specific protocol architecture for wireless micro sensor networks [C] // Proceedings of IEEE Transactions on Wireless Communications. Portugal: IEEE, 2002: 660-670.

(责任编辑 梁洁)