



最小化延迟的无线传感器网络节点调度算法^{*}

李陶深, 汤 珺, 梁俊斌, 兰红星

(广西大学计算机与电子信息学院 南宁 530004)

摘 要: 如何实现节能和减小延迟一直是无线传感器网络中研究的热点及难点问题。提出一种新的最小化延迟的能量均衡的节点调度(MDS)算法, 算法通过对能量以及延迟的分析计算, 计算出节点的能量判定阈值, 并在选择下一跳的时候, 总是选择节点剩余能量在限定范围内且延迟最小的节点, 以此实现最小化网络延迟的目的。理论分析及实验结果表明, MDS 算法能够较好地控制网络的生命周期在可接受范围内, 同时可以解决在能耗限定条件下的最小化延迟的问题。

关键词: 无线传感器网络; 工作周期; 能量均衡; 最小化网络延迟

doi: 10.3969/j.issn.1000-0801.2013.12.007

Minimum Delay Node Scheduling Algorithm for Wireless Sensor Network

Li Taoshen, Tang Jun, Liang Junbin, Lan Hongxing

(School of Computer, Electronics and Information, Guangxi University, Nanning 530004, China)

Abstract: In wireless sensor networks, saving energy and reducing delay are always the focus of research and difficult problems. In order to solve that problem, a new minimum delay scheduling (MDS) algorithm was proposed. Through the analysis and calculation of energy and delay of nodes, this algorithm may calculate the node energy threshold. To realize minimum network delay and it always selects the node whose remaining energy was in the limited range and had the minimum delay to be next top node. Theoretical analysis and simulation results show that MDS algorithm can control the lifetime of network within an acceptable range while effectively minimizing network delay comparing with previous algorithms.

Key words: wireless sensor network, duty-cycle, energy balance, minimum network delay

1 引言

在无线传感器网络(wireless sensor network, WSN)中, 常使用工作周期(duty-cycle)方式来节省能量^[1-3]。然而由于传感器节点的通信半径有限, 需要通过多跳的方式传输到目的节点(sink 节点), 工作周期和多跳共存的方式可能会使网络产生巨大的延迟。因此, 能耗和延迟一直是 WSN 中研究的热点与难点问题^[4,5]。休眠时间长, 节点可以节省更

多的能量, 但同时可能会使网络延迟增大, 因为当节点有数据发送时, 其下一跳节点不一定处于工作状态; 反之, 节点工作时间长, 能够及时接收与发送数据, 但需要消耗很大的能量。因此, 能耗和延迟是一对相互权衡的重要性能指标, 用增大能耗的方法来减少延迟或不充分考虑延迟的情况下延长网络生命周期都是不可行的。

近几年, 人们提出了许多在无线传感器网络中如何选择下一跳的节点调度方案, 有的方案在延迟的约束下侧重

^{*} 国家自然科学基金资助项目(No.61363067, No. 61103245)



考虑网络节能^[6],有的则在能耗约束下重点考虑节点延迟^[7]。参考文献[8]使用马尔可夫随机域在限制延迟的条件下优化网络的能耗,这种节能方法是随机网络中基于图的派生计算。在此基础上,参考文献[9]提出了一种基于近似斯泰纳树(Steiner-tree approximation)的最小化能耗方案。这两种方案都是在延迟约束条件下最小化能耗的节点调度方案。如果在能耗约束条件下重点考虑网络节点延迟时,人们通常选择离 sink 节点最近的节点作为下一跳节点。

为了选择那些节点选择剩余能量在平均范围内且延迟最小的节点作为下一跳的节点,本文提出一种最小化网络延迟且能量均衡的节点调度算法。算法以最小化网络延迟为目标,在选择下一跳节点时,首先通过一个阈值来判定在某个状态是否进行数据的传输。该阈值的计算综合考虑了网络的能耗和延迟,希望在考虑到节点剩余能耗的平均值的约束条件下,最小化网络的每跳延迟。

2 网络模型与问题描述

2.1 网络模型

假设有 N 个传感器节点 $v_i(i=1,2,\dots,N)$,服从泊松分布且随机分布在一个面积为 $C \text{ m} \times C \text{ m}$ 的正方形区域 A 内,每个节点的发送半径为 R 。

在很多传感器网络应用中,接收和转发数据的时间都非常短,而用于空闲侦听的时间却比较长。考虑到在现实环境中,被监视事件发生的频率大多较低,所以没有必要让节点一直处于侦听状态。为了节省能耗,本文采用节点工作一段时间休眠一段时间的异步工作周期工作方式。由于本文研究的是由工作周期引起的延迟,所以忽略传送过程中各种因素引发的延迟,即当节点 i 有数据需要发送给它的邻居节点 j 时, i 可以在 j 醒来时时刻立即发送。不失一般性,定义工作休眠的周期为 T 。

在 $t=0$ 的时候,网络中的全部节点将自己产生的数据分组传输到 sink 节点。本文采用数据完全汇聚的流量模型^[10],即每个接收节点将它所接收到的多个大小为 L bit 的数据分组与自己产生的 L bit 的数据进行汇聚,最终发送出去的数据分组大小也为 L bit。

当节点 i 有数据需要传送时,它在其中继区域内根据法则选择下一跳。中继区域的定义见下文。

2.2 相关的定义

为方便说明,首先给出以下几个定义。

定义 1 (邻居节点)若已知节点 i 与节点 j 的距离小

于节点的通信半径 r ,即 $|z_i - z_j| < r$,则称节点 i 与 j 是邻居节点,用 N_i 表示节点 i 的邻居节点数。

定义 2 (中继区域)节点 i 的中继区域定义为它的邻居节点中比 i 更靠近 sink 节点 $j(j=1,2,\dots)$ 的 K 个节点所在的区域。

定义 3 (轮)指数据分组从源节点传送到 sink 节点的一次过程,期间不考虑它所持续的时间是多少,用 $R_i(i=1,2,\dots)$ 表示。

定义 4 (网络生命周期)网络中第一个死亡(失去作用)的节点存活的轮数为网络的生命周期。

定义 5 (等待时间($W_i(t)$))假设节点在 $kT+T_i(k=0,1,2,\dots)$ 时刻醒来工作,让 $T_i(i=1,2,\dots,N-1)$ 在 $[0,T]$ 随机取得, T 为网络的工作周期,节点 i 在 t 时刻的等待时间定义为:

$$W_i(t) = \inf\{kT+T_i \geq t: k \geq 0\} - t$$

2.3 问题的描述

当无线传感器网络中的节点 i 有数据需要传输时,需要同时考虑节点的能量和网络的延迟。当节点 i 把数据传给第一个醒来的中继区域内的邻居节点 j 时(方案 F1),可以减小网络的等待耗时,但也可能热点节点 j 多次使用导致失效(死亡);反之,若节点等待中继区域内的节点全部醒来再传给这些节点中剩余能量最大的节点(方案 F2),达到网络均衡的同时可能产生很大的网络延迟。

不失一般性,定义 $T=1, r=1, p_K$ 表示在 K 个中继节点中选择其中一个作为下一跳的概率,它的条件期望可定义为 E_K 。

让节点的初始能量 e_i 在 $[E_a, E_b]$ 随机取得,其中的 E_a 、 E_b 为根据应用定义的初始值。本文采用参考文献[11]中的一阶无线电模型(first order radio model)。在这种模型中,节点发送 L bit 消息到距离它 D 个距离的节点需要消耗的能量为:

$$E_{Tx}(k, D) = E_{elec} \times L + E_{amp} \times k \times D^2 \quad (1)$$

接收这个消息的节点消耗的能量为:

$$E_{Rx}(k) = E_{elec} \times L \quad (2)$$

其中, E_{elec} 为 50 nJ/bit, E_{amp} 为 100 pJ/(bit·m²),根据第 2.2 节给出的网络流量模型易知:

$$e = e_i - R \times (E_{Tx} + E_{Rx}) \quad (3)$$

其中, R 为网络进行的轮数。

本文的研究要实现在网络均衡的条件下最小化每跳的延迟。令 $E[D^F]$ 表示在方案 F 下的平均每跳延迟, $E[E^F]$ 表

示第 R_i 轮中继节点的平均剩余能量,则有:

$$\min E[D^F] \quad (E[E^F] \geq e) \quad (4)$$

从式(4)可以看出,要选出剩余能量最大的下一跳,就必须等待最长的时间,产生最大的延迟。为方便说明,可把式(4)化简为:

$$\min E[D^F] - \beta \times E[E^F] \quad (5)$$

定理 1 假设 R 轮中,对于给定的一个能量 e ,存在一个 β_e ,使得 $E[E] = e$ 为式(5)的最优解,把这个方案设定为 F 最优。

证明

因为 F 最优是 R 轮中式(3)的最优解,则有: $E[D^F \text{ 最优}] - \beta_e \times E[E^F \text{ 最优}] \leq E[D^F] - \beta_e \times E[E^F]$ 。

即 $E[D^F \text{ 最优}] \leq E[D^F] - \beta_e \times (E[E^F] - e)$,所以,对任意一个 $E[E^F] \geq e$ 的方案 F ,都可以得到: $E[D^F \text{ 最优}] \leq E[D^F]$ 。

得证。

下面,将重点解决式(5)中的问题。

3 本文算法的描述

为了解决式(5)中的问题,本文使用参考文献[6]中的马尔可夫决策过程 (markov decision process, MDP) 框架。假设在节点 i 的中继区域内的节点 $i_1, \dots, i_K$ 的等待时间为: $W_1(t) \leq W_2(t) \leq \dots \leq W_K(t)$, 对应的剩余能量为 $E_1, E_2, \dots, E_K$, 令 $U_1 = W_1(t)$, $U_k = W_k(t) - W_{k-1}(t)$, $2 \leq k \leq K$ 。令 (w_k, e_k) 表示中继节点 k 醒来时刻的最佳状态 (i_k 点阶段), 其中, $e_k = \max\{E_1, E_2, \dots, E_k\}$, w_k 取决于上阶段 (i_{k-1} 点阶段) 的 U_{k-1} 及 w_{k-1} 。

令 i_1 点阶段 $s_1 = (U_1, E_1)$, $C = \{0, 1\}$ 为控制参数, 0 表示节点 i 需要等待更多的邻居节点醒来, 1 表示节点可以将数据发送给根据方案 F 得出的最佳的下一跳。让 $\{\mu_k^F: 0 \leq k \leq K\}$ 为每一阶段的映射, 即选择下一跳的判定参数, 其中, $\mu_0^F: \{(0, 0)\} \rightarrow \{0\}$; 当 $k \geq 1$ 时, $\mu_k^F: [0, T] \times [0, r] \rightarrow \{0, 1\}$ (定义 $T=1, r=1$), ψ 表示终止阶段。

若在 k 点阶段, $1 \leq k \leq K$, 令 $a_k \in C$, 则下一个阶段 (i_{k+1} 阶段) 可表示为:

$$s_{k+1} = \begin{cases} (w_k + U_{k+1}, \max\{E_{k+1}, e_k\}), & a_k = 0 \\ \psi, & a_k = 1 \end{cases} \quad (6)$$

根据式(5), 可以得到一跳的代价函数为:

$$g_k = ((w_k, e_k), a_k) = \begin{cases} U_{k+1}, & a_k = 0 \\ -\beta a_k, & a_k = 1 \end{cases} \quad (7)$$

如果 i_k 阶段为终止阶段, 那么在 i_{k+1} 阶段, 不论 a_k 的值是多少, $s_{k+1} = \psi$, $g_k(\psi, a_k) = 0$ 。同理, 如果 s_K 阶段为最终阶段, 它的代价函数 $g_K(s_K)$ 为:

$$g_K(s_K) = \begin{cases} 0, & s_K = \psi \\ -\beta e_k, & \text{其他} \end{cases} \quad (8)$$

所以方案 F 的平均代价为:

$$J_F(0) = E_K[\sum_{k=0}^{K-1} g_k(s_k, \mu_k^F(s_k)) + g_K(s_K)] \quad (9)$$

从式(9)易知, 它是 $\{(U_k, E_k): 1 \leq k \leq K\}$ 的联合分布, 且:

$$J_F(0) = E[D^F] - \beta E[E^F] \quad (10)$$

所以最优方案的代价为:

$$J^* = \min_F J_F(0) = J^{F \text{ 最优}(\beta)}(0) \quad (11)$$

令 $J_k(w, e)$ 为 i_k 阶段的最优代价, 当处于 i_k 阶段的时候, 全部中继区域内的邻居阶段处于工作状态, 则有:

$$J_k(w, e) = -\eta e = \eta \max\{e, \varphi_k(w, e)\} \quad (12)$$

其中, 定义对于全部 (w, e) , $\varphi_k(w, e) = 0$ 。当又有一个节点从休眠状态变换到工作状态的时候 (如 i_{k-1} 点状态), 那么 $a_{K-1} = 0$ 与 $a_{K-1} = 1$ 都有可能发生, 且定义条件期望的表达式为:

$$E_{(W_1=w_k)}[*] = E_K[* | W_k = w_k] \quad (13)$$

则有:

$$J_{K-1}(w, e) = \min\{-\eta e, E_{(W_{K-1}=w)}[U_K + J_K(w + U_K, \max\{e, E_K\})]\} \quad (14)$$

将式(14)带入式(13), 得:

$$J_{K-1}(w, e) = \min\{-\eta e, E_{(W_{K-1}=w)}[U_K - \eta \max\{e, E_K, \varphi_K(w + U_K, \max\{e, E_K\})\}]\} = -\eta \max\{e, \varphi_{K-1}(w, e)\} \quad (15)$$

其中, $\varphi_{K-1}(w, e)$ 的表达式为:

$$\varphi_{K-1}(w, e) = E_{(W_{K-1}=w)}[\max\{e, E_K, \varphi_K(w + U_K, \max\{e, E_K\})\} - \frac{U_K}{\beta}] \quad (16)$$

综上, 可以得出定理 2。

定理 2 当 $1 \leq k \leq K-1$ 时, 以下等式恒成立:

$$J_k(w, e) = -\eta \max\{e, \varphi_k(w, e)\} \quad (17)$$

其中, $\varphi_k(w, e)$ 的表达式为:

$$\varphi_k(w, e) = E_{(W_k=w)}[\max\{e, E_{K+1}, \varphi_{K+1}(w + U_{K+1}, \max\{e, E_{K+1}\})\} - \frac{U_{K+1}}{\beta}] \quad (18)$$

综上, 可得出消耗最优方案 F 为:

$$\mu_k^{F \text{ 最优}(\beta)}(w, e) = \begin{cases} 1, & e \geq \varphi(w, e) \\ 0, & \text{其他} \end{cases} \quad (19)$$

其中, $\varphi_k(w, e)$ 的值是利用式(18)计算得出的。



根据以上的分析和定理 2, MDS 算法的设计思路是节点选择下一跳时, 在考虑节点剩余能量的前提下, 选择能够获得延迟最小的节点作为下一跳。这就是说, MDS 算法在选择下一跳节点的时候, 设置一个判定参数 (即判定阈值), 当中继区域内节点的这个参数满足条件时即可成为下一跳, 反之, 则等待满足条件的节点出现。算法中按照以下原则选择下一跳节点:

- 当 $\mu_k^{F_{\text{最优}}(\beta)}=1$ 的时候, 将数据传给醒来节点中的最优节点;
- 当 $\mu_k^{F_{\text{最优}}(\beta)}=0$ 的时候, 需要等待下一个节点醒来再进行比较, 直到 $\mu_k^{F_{\text{最优}}(\beta)}=1$ 的时候选出最优的下一跳节点。

4 算法性能分析与实验验证

4.1 算法的消息复杂度

根据 MDS 算法, 假设无线传感器网络中每个节点都产生数据, 数据通过多跳的方式中继到目的节点 (sink 节点), 则易知它的时间复杂度是 $O(1)$ 。下面计算 MDS 算法的消息复杂度, 节点 i 需要收集 n 个邻居的数据时, 算法的消息复杂度为 $O(\log n)$, 由 MDS 算法的性质可知, 当每个节点都有数据需要发送的时候, 节点 i 要比较其邻居节点的判定参数大小, 此时, 它的消息复杂度为 $O(\log n)$, 所以包含 n 个节点的网路的消息复杂度为 $n O(\log n)$ 。

4.2 算法实验与性能

为了验证本文算法的有效性和可行性, 在 MATLAB 2010a 平台上进行了仿真实验。实验中, 假设 300 个节点随机分布在面积为 $50 \text{ m} \times 50 \text{ m}$ 的区域内, 每个节点的初始能量在 $[1 \text{ J}, 1.5 \text{ J}]$ 的能量区间内随机取得, sink 节点位于区域的中心。实验的内容是将 MDS 算法与参考文献[12]中的 BF 算法进行比较。BF 算法的策略是选择一个离 sink 节点更接近的节点作为下一跳节点, 并以此最小化网络的延迟。实验时, 对比两个算法在不同网络节点数下的平均延迟与生命周期, 实验的结果均是运行 10 次后的平均结果。实验对比结果如图 1~图 4 所示。

从图 1 可以看出, 在不同 β 参数情况下, 两个算法的每跳延迟会随着 β 值的增大而增加, 到 $\beta=1000$ 的时候逐渐趋于稳定, 两个算法的差别很小。在不同的网络节点数下, 网络的平均延迟会随着节点数的增加而减小, BF 算法的平均网络延迟稍优于 MDS 算法 (如图 2 所示)。在不同

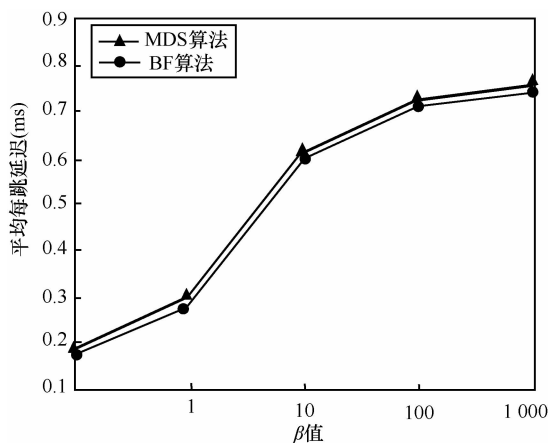


图 1 算法 MDS 和 BF 在不同 β 值下的每跳延迟

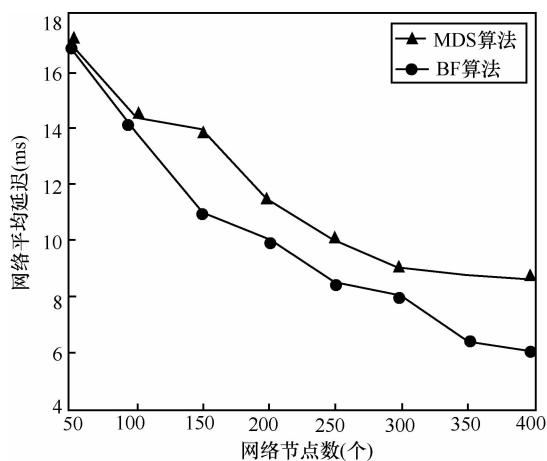


图 2 算法 MDS 和 BF 在不同节点数下的网络平均延迟

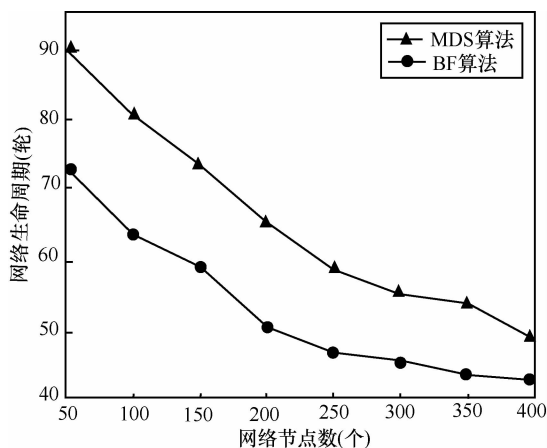


图 3 算法 MDS 和 BF 在不同节点密度下的生命周期

节点密度下, 两个算法的网络生命周期随着节点数的增大而减小, 但 MDS 算法的生命周期大于 BF 算法。如图 3 所示, 当节点数分别为 50、100、150、200、250、300、350、400 时, MDS 算法的生命周期分别优于 BF 算法 23.3%、28.6%、25.4%、30%、27.5%、21.9%、24.2% 和 14.8%。当取不同 K 值

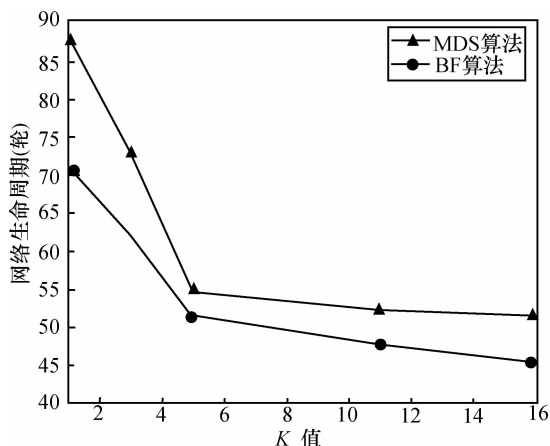


图4 算法 MDS 和 BF 在不同 K 值下的网络生命周期

时,网络的生命周期会随着 K 值的增大而减小,在 K 值分别为 1、3、5、15 的时候,MDS 算法的生命周期分别优于 BF 算法 25.4%、17.7%、28%、13.6%。

实验结果分析表明,与 BF 算法相比,MDS 算法只花费了较小的延迟代价就可获得更大的网络生命周期,总体上说 MDS 算法优于 BF 算法。

5 结束语

本文提出了一种新的最小化延迟的能量均衡的节点调度算法 MDS,在选择下一跳的时候,定义一个判定值,当判定值为 1 时,表示节点能够根据方案将数据传给它选中的下一跳;为 0 时,表示需要等待新的节点醒来继续比较。判定值的选择与节点的剩余能量和网络的延迟相关。选择下一跳节点时,在考虑选择邻居节点中剩余能量不小于限定值的同时,还要考虑其延迟是最小的。不同的 K 值与 β 值得出的结果各不相同,可以根据不同的网络应用设置其初始值。理论分析及实验结果表明,与现有的算法相比,本文的 MDS 算法可以把能耗控制在一个可接受的范围,同时能最小化网络的生命周期。

参考文献

- 1 Song W Z, Huang R, Xu M, *et al.* Design and deployment of sensor network for real-time high-fidelity volcano monitoring. *IEEE Transactions on Parallel and Distributed Systems*, 2010, 21(11):1658~1674
- 2 Anandkumar A, Swami A, Yukich J E, *et al.* Energy scaling laws for distributed inference in random fusion networks. *IEEE Journal on Selected Areas in Communications*, 2009, 27(7): 1203~1217
- 3 Liu S, Fan K W, Sinha P. CMAC: an energy-efficient MAC layer

protocol using convergent packet forwarding for wireless sensor networks. *ACM Transactions on Sensor Networks*, 2009(4)

- 4 Wang X, Huang W, Wang S, *et al.* Delay and capacity tradeoff analysis for motioncast. *IEEE/ACM Transactions on Networking*, 2011, 19(5): 1354~1367
- 5 Liu Y, Zhu Y, Ni L M, *et al.* A reliability-oriented transmission service in wireless sensor networks. *IEEE Transactions on Parallel and Distributed Systems*, 2011, 22(12): 2100~2107
- 6 Becchetti L, Marchetti-Spaccamela A, Vitaletti A, *et al.* Latency-constrained aggregation in sensor networks. *ACM Transactions on Algorithms*, 2009, 6(1)
- 7 Anandkumar A, Tong L, Swami A, *et al.* Minimum cost data aggregation with localized processing for statistical inference. *Proceedings of INFOCOM 2008: 27th IEEE Communications Society Conference on Computer Communications*, Phoenix, USA, 2008: 780~788
- 8 Bertsekas D P. *Dynamic Programming and Optimal Control* (3rd edition). Massachusetts: Athena Scientific, 2005
- 9 Anandkumar A, Yukich J, Tong L, *et al.* Energy scaling laws for distributed inference in random networks. *IEEE Journal of Select Area Communication*, 2009, 27(7): 1203~1217
- 10 Wang X, Fu L, Tian X, *et al.* Converge cast: on the capacity and delay tradeoffs. *IEEE Transactions on Mobile Computing*, 2012, 11(6): 970~982
- 11 Madden S, Franklin M J, Hellerstein J M, *et al.* TAG: a tiny aggregation service for Ad Hoc sensor networks. *ACM SIGOPS Operating Systems Review*, 2002, 36(SI): 131~146
- 12 Naveen K P, Kumar A. Tunable locally-optimal geographical forwarding in wireless sensor networks with sleep-wake cycling nodes. *Proceeding of INFOCOM*, San Diego, CA, USA, 2010: 920~928

[作者简介]



李陶深,男,博士,广西大学计算机与电子信息学院教授,主要研究方向为云计算、分布式数据库、无线 mesh 网络、网络与信息安全。

汤琨,女,广西大学计算机与电子信息学院硕士研究生,主要研究方向为无线传感器网络。

梁俊斌,男,博士,广西大学计算机与电子信息学院副教授,主要研究方向为无线传感器网络、物联网技术。

兰红星,男,博士,广西大学计算机与电子信息学院研究员,主要研究方向为网络与信息安全。

(收稿日期:2013-07-08)