

面向数据收集的转发树构建与链路调度算法

刘 洋

(乌鲁木齐职业大学 信息工程学院, 乌鲁木齐 830002)

摘 要:多数基于压缩感知的数据收集方法假设网络无干扰或能够采取有效的冲突避免措施,当网络存在干扰或噪声时,难以同时兼顾效率和可靠性。为此,以压缩感知理论为基础,研究物理干扰模型下的数据收集问题,将其建模为转发树构建和链路调度联合问题,并设计可构建转发树的分布式求解算法,确定一组转发树并在调度后于最短调度周期内将测量数据发往汇点,实现传输延时和采集能效间的平衡。仿真结果表明,该算法能有效降低数据传输延时,提高数据采集能效。

关键词:无线传感器网络;物理干扰模型;数据收集;压缩感知;转发树;链路调度

中文引用格式:刘 洋. 面向数据收集的转发树构建与链路调度算法[J]. 计算机工程,2017,43(10):82-90.

英文引用格式:LIU Yang. Forwarding-tree Construction and Link Scheduling Algorithm for Data Collection[J]. Computer Engineering,2017,43(10):82-90.

Forwarding-tree Construction and Link Scheduling Algorithm for Data Collection

LIU Yang

(College of Information Engineering, Urumqi Vocational University, Urumqi 830002, China)

[Abstract] Most data collection methods based on Compressive Sensing (CS) usually assume that the network has no interference or it can take effective conflict avoidance measures. When these methods are applied to the network with interference or noise, they cannot balance the energy efficiency and reliability. Therefore, based on the compression theory, the data collection problem under the physical interference model is studied, which is modeled as the combined problem of forwarding-tree construction and link scheduling combined problem, and a distributed algorithm for forwarding-tree construction and link scheduling is proposed to determine a set of forwarding-tree and send the measured data to the sink after forwarding-tree scheduling in the shortest scheduling cycle, so as to achieve balance between the transmission delay and collection energy efficiency. Simulation results show that the proposed algorithm can effectively reduce the data transmission delay and improve energy efficiency of data collection.

[Key words] Wireless Sensor Network (WSN); physical interference model; data collection; Compressive Sensing (CS); forwarding-tree; link scheduling

DOI:10.3969/j.issn.1000-3428.2017.10.015

0 概述

无线传感器网络(Wireless Sensor Network, WSN)^[1-3]具有多种功能,受到了人们的大量关注,其在健康监测、环境监测、交通监测和重要基础设施及建筑监测等领域获得广泛应用。其中大部分应用领域需要传感器周期性地测量数据,并通过多跳路径等方法将数据发往汇点(或称 Sink),然后对数据进行处理。但这些传感器的能量有限,一旦部署便很少维护,因此,如何提高数据采集的能效对于延长无线传感器网络的寿命具有重要意义。

基于压缩感知(Compressive Sensing, CS)理论

的数据收集方法^[4-7]能在不造成重大计算或传输开销的前提下降低全局通信成本,现已成为人们关注的焦点。该方法在给定某条路由后可将通信成本分配给所有传感器,进而节约大量能耗,延长网络寿命,实现负载均衡^[8]。为了实现网络数据的高能效采集,这些方法大多将数据收集问题建模为以汇点为根的转发树构建问题,并设计集中式和分布式算法来解决该问题。数据收集树构建完毕后,需要进行链路调度,以便相邻传输互相之间不会产生干扰,避免经过压缩的测量数据受到损坏,同时发挥空间复用的最大效用^[9]。然而,上述数据收集方法存在不足。上述方法假设网络无干扰或能够采取有效的

基金项目:国家自然科学基金(61273072)。

作者简介:刘 洋(1979—),男,副教授、硕士,主研方向为无线传感器网络、数据库技术。

收稿日期:2016-07-29 **修回日期:**2016-10-11 **E-mail:**110853290@qq.com

冲突避免措施来进行数据收集,未考虑物理干扰对数据收集过程的影响,当被应用到网络存在干扰或噪声的真实环境中时,将无法同时兼顾数据收集的高能效性和数据收集的可靠性^[10-11]。此外,为保证数据收集的可靠性,上述方法在转发树构建和链路调度的过程中主要以满足数据收集的精度为前提,不能同时达到数据收集的低能耗和低延时这一要求。

针对现有方法的不足,本文在物理干扰模型下研究转发树构建和链路调度之间的关系,从而确定合适的转发树,在汇点处收集测量数据。文献[12]指出,在物理干扰模型下的无线传感器链路调度问题属于 NP 难题。为此,本文以降低数据收集延时和传输能耗为优化目标,提出一种分布式求解算法,从而解决树构建和链路调度问题,实现物理干扰模型下的高能效数据收集。

1 系统模型和问题定义

1.1 系统模型

本文将无线传感器网络看成一个有向图 $G=(V,E)$,其中, V 表示随机部署在某个区域上 n 个节点构成的集合; E 表示互相位于对方通信半径内任意 2 个传感器节点之间的链路集合。通过改变节点的传输功率便可调节网络密度。出于简便考虑,假设所有节点的传输功率固定且均为 P ,因此,生成的图形为连通图。假设在每一轮(周期)内,各个节点获得一个数据读数 x_i (如速度、密度或温度),该节点需要将读数发往可能位于网络中某一位置的基站(汇点)。因此,汇点在每一轮期间需要从网络中的所有节点处收集到总体规模为 n 的数据向量 $\mathbf{X}=[x_1,x_2,\dots,x_n]^T$ 。因为并不是所有节点与汇点之间存在直接链路,所以传感器可能需要多跳路径来传输数据。本文采用基于时分多址(Time Division Multiple Access, TDMA)的 MAC 访问机制,将时间划分为多个等长时隙,将可在同一时隙内同时处于活跃状态的链路集合看成一个配置。由于一个配置由来自多个转发树并可同时处于活跃状态的链路/传输数据构成,因此对于树中的某个母节点来说,直到它从子节点接收到传输数据时才会对该母节点进行传输调度。设 d_{ij} 表示节点 i 和 j 间的欧氏距离, G_{ij} 表示从发送节点 i 到接收节点 j 的信道增益(如 $G_{ij}=d_{ij}^{-\alpha}$,其中 α 表示路径损失指数)。因此,本文采用文献[13]中的物理干扰模型,发生并行传输时,如果 j 的信号与干扰噪声比(Signal to Interference plus Noise Ratio, SINR)大于阈值 β ,则接收节点 j 可成功接收到节点 i 的传输数据,即:

$$\text{SINR}_{(i,j)} = \frac{P \cdot G_{ij}}{\eta + \sum_{\forall (h,k) \in E; h \neq i} P \cdot G_{hj}} \geq \beta \quad (1)$$

$\forall (i,j) \in E$

其中, η 表示背景噪声。本文将为收集所有经过编码的测量数据而需要对所有转发树的链路进行调度的时隙总数视作一个周期。一个周期的规模决定了测量数据的采集延时。本文进一步假设所有报文(每个报文携带一个经过编码的测量数据)的尺寸相同。

1.2 压缩数据采集

根据压缩感知理论可知,只要原始读数在某一稀疏域可变换或可压缩,那么只需极少量的样本测量数据,便可在汇点高效恢复 n 个传感器的读数。假设原始数据 $\mathbf{X}=[x_1,x_2,\dots,x_n]^T$ 采用某一合适矩阵 Ψ 后获得一个 k 重稀疏表示,其中 Ψ 是尺寸为 $n \times n$ 的傅里叶变换矩阵。也就是说, $\mathbf{X}=\Psi\mathbf{S}$, \mathbf{S} 是 \mathbf{X} 的 k 重稀疏列向量表示, \mathbf{S} 中只有 k 个系数不为 0 且 $k \ll n$ 。根据压缩感知理论中的受限等距映射属性(Restricted Isometry Property, RIP)^[4],汇点可以只接收 $m=O(k \lg n)$ 个测量数据而不是 n 个读数,且 $m \ll n$ 。也就是说, $\mathbf{Z}=\Phi\Psi\mathbf{S}=\Phi\mathbf{X}$, \mathbf{Z} 是维度为 $m \times 1$ 的样本测量数据的列向量, Φ 是维度为 $m \times n$ 的随机样本测量数据矩阵。换言之,汇点接收到 $\mathbf{Z}=[z_1,z_2,\dots,z_m]^T$ 后便可完整复原原始数据 \mathbf{X} ,其中 $z_i=\sum_{j=1}^n \phi_{ij}x_j, i=1,2,\dots,m$ 。每个 z_i 表示 Φ 中某一行的系数不为 0 时网络中对应节点测量数据的加权和。本文将这些节点称为兴趣节点,从兴趣节点汇聚的数据称为一个投影。矩阵 Φ 有 m 行,每一行对应一个加权和(投影),同时有 n 列,每一列对应一个传感器节点。

获得 m 个测量数据 \mathbf{Z} 后,汇点需要结合随机样本矩阵 Φ 和傅里叶变换矩阵 Ψ ,并求解以下凸优化问题来恢复数据的稀疏表示 $\tilde{\mathbf{S}}$ (非原始数据)。

$$\min_{\tilde{\mathbf{S}}} \|\tilde{\mathbf{S}}\|_2, \text{ s. t. } \mathbf{Z} = \Phi\Psi\tilde{\mathbf{S}} = \mathbf{A}\tilde{\mathbf{S}} \quad (2)$$

恢复稀疏向量 $\tilde{\mathbf{S}}$ 后,通过设置 $\mathbf{X}=\Psi\tilde{\mathbf{S}}$ 便可获得原始数据 \mathbf{X} 。根据 RIP 理论可知,如果投影数量 m 或投影稀疏性略低于最小要求,那么只要系数 \mathbf{X} 呈现幂律衰减,汇点便可重建出原始读数向量 \mathbf{X} 。在本文中,为了将非 0 系数更均匀地分布在矩阵 Φ 中且使每个投影尽量稀疏,可将矩阵 Φ 每一行中的非 0 系数数量确定为 $\left\lfloor \frac{n}{m} \right\rfloor$,以便保证 Φ 不会出现某一行元素全为 0 的情况。由于投影的稀疏性和数量 m 取决于传感器读数的 k 重稀疏傅里叶变换表示,因

此本文给出的随机样本矩阵 Φ 满足基于压缩传感的数据恢复条件, 汇点可以利用压缩感知技术完全恢复出原始数据读数。

1.3 问题定义

假设网络中的数据传输互相之间可能发生干扰, 应采用访问机制对数据传输进行协调, 给出如下的定义:

定义 1 已知由 n 个节点构成的连通图 G 、1 个汇点和 1 个稀疏矩阵 Φ , 基于投影的压缩数据采集问题需要确定 m 个转发树, 每个转发树负责从部分节点采集经过编码的测量数据, 然后通过路由传输以能效最高的方式将数据发往汇点(上述过程采集数据时的部分节点是指矩阵 Φ 相应行中系数不为 0 的节点, 并将这些节点称为兴趣节点)。

每个树 $t_i, i=1, 2, \dots, m$ 对应 1 个投影, 汇点通过这个投影便可采集到兴趣节点测量数据的加权和 z_i 。本文的目标是通过构建这些转发树, 使网络中的总体传输次数最小化。每个路由树的数据采集操作如下所示: 对每个兴趣节点 ($j, j \in I_i$ 其中 I_i 表示树 t_i 中兴趣节点的集合), 当采集到自身的测量数据时, 将其读数 x_j 与随机系数 ϕ_{ij} 相乘, 并将数据 $\phi_{ij}x_j$ 与其子节点(如果有的话)发来的数据相融合, 然后在一个报文中将获得的加权(编码)和发往母节点。最后, 作为所有树的根节点的汇点将接收到来自各个

兴趣节点集合(即树)的加权和 $z_i = \sum_{j=1}^n \phi_{ij}x_j$ 。因此, 汇点接收到的报文数量是 m 而不是 n , 且 $m \ll n$ 。

定义 2 已知一组转发树 $t_i, i=1, 2, \dots, m$, 调度问题的主要内容就是确定规模最大的集合(此处集合是指一个时隙内同时处于活跃状态的链路构成的配置), 以便使生成的调度长度最小化。该问题可保证经过压缩的测量数据在最小延期内发送到汇点。

定义 3 转发树构建和调度(Forwarding-tree Construction and Scheduling, FTCS)联合问题, 由定义 1 与定义 2 所述的问题构成。

下面结合图 1 中的网络示例来阐述 FTCS 的运行过程。其中, M 表示希望收集所有节点数据的汇点。深灰色的圆圈表示投影的汇聚树。每组兴趣节点用同一颜色表示。箭头旁的数字表示树中链路处于活跃状态时的时隙。将联合 FTCS 方法与将树构建和调度单独对待的另一方法进行比较, 如图 1 所示。该示例演示了汇点如何利用 3 个投影从所有传感器采集数据。可以看出, 2 种方法数据采集时的数据传输次数(链路)相同。然而, 图 1 表明利用联合 FTCS 方法构建而得的树只需 8 个时隙便可调度完毕, 而另一方法虽然传输次数相同但采集测量数据时共需要 9 个时隙, 这是因为该方法在构建树时未考虑降低调度长度这一要求。

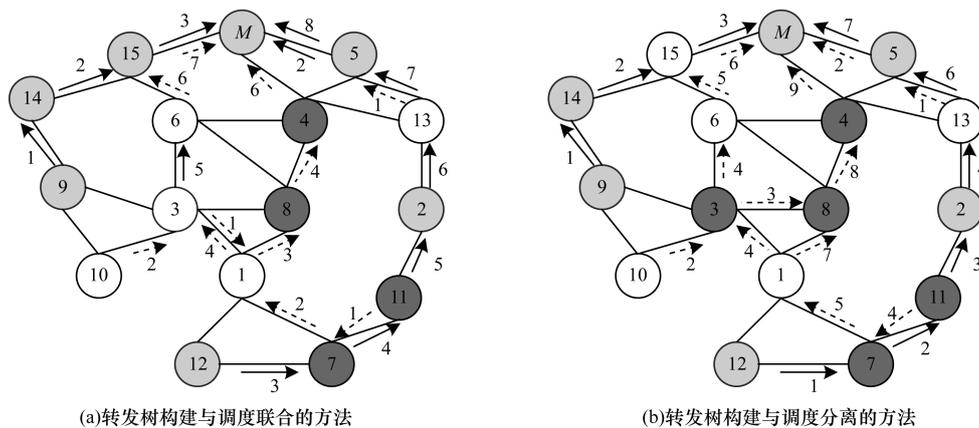


图 1 FTCS 问题示例

2 问题描述

本文将 FTCS 联合问题看成一种优化问题, 目标是确定一组转发树, 并在转发树调度后于最短调度周期内将测量数据发往汇点, 同时在传输延时和采集能效间实现平衡。该问题从数学角度可以表述为混合整数线性规划问题。

为了便于描述, 表 1 给出了文中用到的标记符号。其中, x'_{ij} 是个表示树 t 节点 i 和 j 间是否存在链

路的二元变量。 λ_s 是个表示时隙 s 是否具有至少 1 条活跃链路的二元变量, 有:

$$\lambda_s = \begin{cases} 1, & \text{时隙 } s \text{ 内至少 1 条链路处于活跃状态} \\ 0, & \text{其他} \end{cases} \quad (3)$$

本文目的是构建转发树, 以便在测量数据采集过程所需要的链路数量(数据采集能耗)和转发树调度所需时隙数量(数据采集延时)之间实现平衡, 即:

$$\text{Minimize } \sum_{t \in T} \sum_{(i,j) \in E} x'_{ij} + \sum_{s \in S} \lambda_s \quad (4)$$

表 1 问题描述中的符号及含义

符号	含义
V	网络中的节点集合
E	网络中的边集合
n	节点总数量
m	投影总数量
$ I_t $	集合 I_t 中兴趣节点的总数量
P	节点的功耗
S	一轮数据采集的时隙数量
G_{ij}	发射节点 i 到接收节点 j 的信道增益
β	SINR 最小阈值
η	背景噪声
f_{ij}^t	树 t 链路 (i, j) 的流量
x_{ij}^t	链路 (i, j) 是否属于树 t
a_{ij}^t, s	树 t 中的链路 (i, j) 在时隙 S 期间是否处于活跃状态
λ_s	时隙 S 期间是否有至少一条链路处于活跃状态

在式(4)中,第 1 项求和对应于所构建树中的链路总量,第 2 项求和表示调度长度。下面给出约束条件。

1) 流量守恒约束。此约束要求所有入向流量加上节点生成的流量等于所有出向流量。设 f_{ij}^t 表示树 t 内节点 i 和 j 间的流量。对各个树 t 及属于树(投影)的兴趣节点集合(向量集 I_t),下式约束要求从兴趣节点到汇点存在 1 个流量。

$$\sum_{j:(i,j) \in E} f_{ij}^t - \sum_{j:(j,i) \in E} f_{ji}^t = \begin{cases} -|I_t|, & i \text{ 为汇点} \\ 1, & \forall i \in I_t \\ 0, & \text{其他} \end{cases}$$

$$\forall t \in T \quad (5)$$

2) 汇聚约束。母节点对来自子节点的数据进行汇聚或压缩,然后在 1 个报文内把压缩数据向上传输,具有正流量的边表示为树的链路。因此, $f_{ij}^t = 0 \Leftrightarrow x_{ij}^t = 0$ 且 $f_{ij}^t > 0 \Leftrightarrow x_{ij}^t = 1$ 。通过以下 2 个不等式可以保证树中流量和链路间的连通性:

$$\begin{cases} f_{ij}^t - x_{ij}^t \geq 0 \\ x_{ij}^t - m^{-1} f_{ij}^t \geq 0 \end{cases} \quad \forall (i, j) \in E, t \in T \quad (6)$$

3) 外向链路约束。此约束要求每个节点在每个树中最多只有一次外向传输,以避免出现环路。

$$\sum_{j:(i,j) \in E} x_{ij}^t \leq 1 \quad \forall i \in V, t \in T \quad (7)$$

4) 半双工约束。此约束可保证节点在同一时隙内不能同时传输和接收数据。

$$\sum_{t \in T} a_{ij}^t, s + \sum_{t \in T} a_{jk}^t, s \leq 1 \quad \forall (i, j) \in E, (j, k) \in E, s \in S \quad (8)$$

5) 传输约束。此约束可保证发送节点在同一时隙内不得向多个接收节点传输数据。

$$\sum_{t \in T} \sum_{j:(i,j) \in E} a_{ij}^t, s \leq 1 \quad \forall i \in V, s \in S \quad (9)$$

6) 接收约束。此约束可保证接收节点在同一时隙内不得接收多个发送节点的数据。

$$\sum_{t \in T} \sum_{j:(i,j) \in E} a_{ij}^t, s \leq 1 \quad \forall j \in V, s \in S \quad (10)$$

7) 链路调度约束。此约束要求树中链路在同一时隙内只能调度一次。

$$\sum_{s \in S} a_{ij}^t, s = x_{ij}^t \quad \forall (i, j) \in E, t \in T \quad (11)$$

8) 传输次序约束。此约束可保证树中节点只有接收到子节点的所有报文后才会向外传输数据。也就是说,如果通往其子节点的所有链路在时隙 s 前(即时隙 $1 \sim s-1$)调度完毕,则树 t 在时隙 s 内的链路 (i, k) 可被调度。即:如果 $\sum_{\bar{s}=1}^{s-1} \sum_{j:(j,i) \in E} a_{ji}^t, \bar{s} \geq \sum_{j:(j,i) \in E} x_{ji}^t$, 则 $a_{ik}^t, s = 1$ 。按照 LP 格式,上述条件可重写为:

$$\sum_{\bar{s}=1}^{s-1} \sum_{j:(j,i) \in E} a_{ji}^t, \bar{s} + B(1 - a_{ik}^t, s) \geq \sum_{j:(j,i) \in E} x_{ji}^t \quad \forall (i, k) \in E, s \in S, t \in T \quad (12)$$

其中, B 表示较大常数,其数值大于 m 个树任意组合后的链路总数。因此,如果 $a_{ik}^t, s = 1$, 则上述约束成立。

9) SINR 约束。此约束可保证每条活跃链路的 SINR 大于阈值 β 。

$$P \cdot G_{ij} + B_{ij}^t, s(1 - a_{ij}^t, s) \geq \beta(\eta + \sum_{T \in T(k,h) \in E; k \neq i} P \cdot G_{kj} \cdot a_{kh}^t, s) \quad \forall (i, j) \in E, s \in S, t \in T \quad (13)$$

其中, B_{ij}^t, s 为常数且满足下式:

$$B_{ij}^t, s \geq \eta + \sum_{t \in T(k,h) \in E; k \neq i} P \cdot G_{kj} - \alpha \cdot a_{kh}^t, s$$

在式(13)中,如果树 t 的链路 (i, j) 在时隙 s 内处于活跃状态(即 $a_{ij}^t, s = 1$), 则该式可转化为式(1)。

10) 确定调度期间被占用的时隙。此约束可确定时隙 S 期间是否具有至少一条活跃链路。

$$\lambda_s \geq a_{ij}^t, s \quad \forall s \in S, t \in T, (i, j) \in E \quad (14)$$

3 本文算法

为解决 FTCS 问题计算复杂性较高的问题,本文将其分为 2 个子问题:转发树的构建和链路调度,然后给出分布式求解算法。

3.1 转发树的分布式构建

每个转发树将把经过压缩的测量数据从网络传输到汇点,笔者希望构建出的转发树将数据传输到汇点时,能效最高、延时最低。转发树的构建步骤具体如下:

步骤 1 通过传播检测报文。

汇点首先将检测报文发往相邻节点。每个节点在接收到报文时,将其广播出去,以便与汇点相距较远的其他节点也可以接收到检测报文。此步骤与基于宽度优先搜索策略(Breadth-first Search, BFS)的网络遍历做法相类似。因此,每个节点 v 将学习通

往汇点的最短路径 P_{vs} 以及路径的跳数。另外,节点 v 将检测出相邻节点集合 $N(v)$ 。在检验存储在其内存内的矩阵 Φ 时,节点 v 需要确定节点 $u \in N(v)$ ($\forall u$) 是否属于树 t 的兴趣节点集合 I_t 。此步骤的时间复杂度为 $O(n)$ 。

步骤2 通过路由传输检测报文。

每个树 t 的每个节点 v (如果为兴趣节点) 在运行完算法1后,确定通往汇点的上行路径上哪个节点为母节点。对每个兴趣节点,通过分配一个属性来确定其母兴趣节点 π_v^t ,同时分配一个决策标记 $flag_v^t$ 来表示 v 的母兴趣节点是否固定。

算法1 节点 v 的路由检测

1. if $root \in N(v)$ then
2. $\pi_v^t = s$;
3. 设置并广播 $flag_v^t = 1$;
4. else if $b \in N(v)$ 且 $b \in I_t$ 且 $flag_b^t = 1$ then
5. $\pi_v^t = b$;
6. 设置并广播 $flag_v^t = 1$;
7. else if $b \in N(v)$ 且 $b \in I_t$ 且 $h_b^t < h_v^t$ then
8. $\pi_v^t = b$;
9. $flag_v^t = 0$;
10. else if $b \in N(v)$ 且 $b \in I_t$ 且 $h_b^t = h_v^t$ 且 b 的连续母节点到达跳数较少的节点或者 $flag = 1$ 的节点或非母节点同时不到达 v then
11. $\pi_v^t = b$;
12. 设置 $flag_v^t = 0$;
13. else
14. 以 $h_v^t - 1$ 为半径从 v 开始执行 BFS 策略;
15. if 在该半径内找到兴趣节点 then
16. 通过最短路径将 v 与最近兴趣节点相连;
17. 设置 $flag_v^t = 0$;
18. else
19. 利用最短路径将节点 v 与根节点相连

算法1中的第1行~第3行表明,作为根节点相邻节点的每个兴趣节点选择根节点作为其母节点,同时设置其决策标记 $flag_v^t = 1$ 并加以传播。此时,如果兴趣节点 v (算法第4行~第6行)不是根节点的相邻节点,但其某个相邻节点 b 为兴趣节点且 $flag_b^t = 1$,则 v 选择 b 作为其母兴趣节点,并做出决策 ($flag_v^t = 1$)。如果 v 的所有相邻兴趣节点均未设置决策标记 ($flag_b^t = 0$),则 v 将选择通往汇点的路径跳数最小的相邻兴趣节点作为其母兴趣节点 (算法1第7行~第9行)。此时,如果找到通往汇点的路径跳数相同的相邻兴趣节点 (算法1第10行~第12行),则 v 选择 b 。选择标准是 b 的连续母节点通过较少跳数即可到达兴趣节点,或决策标记 $flag = 1$,或 b 为非母节点且不可到达 v (以避免出现环路)。如果上述条件均未满足,则 v 通过 BFS 策略搜索半径为 $h_v^t - 1$ 的邻域,以找到通往汇点路径跳数较少的兴趣节点 b ; 否则,寻找决策标记

为 $flag_b^t = 1$ 的兴趣节点 (算法1第14行~第17行)。节点 v 不选择 $\pi_v^t = v$ 的兴趣节点 b ,以避免出现环路。最后,如果仍没找到兴趣节点,则 v 通过最短路径将自己与汇点直接相连 (这条路径可在检测阶段获得)。如果节点 v 从相邻节点接收到通知报文,表明网络发生变化 (如决策标记变化,或由于网络移动性或信道受损导致链路或节点发生故障,并触发路由维护),则节点 v 将重复路由检测步骤。此步骤在最好情况下 (节点选择一个相邻节点) 的时间复杂度为 $O(1)$,在最坏情况下 (节点没有相邻的兴趣节点) 的时间复杂度为 $O(\gamma)$,其中 γ 表示节点 v 周围半径 $h_v^t - 1$ 内的节点数量。

通过图2来阐述本文算法树构建的改进思想。其中,箭头表示数据汇聚树上的链路。每个箭头的活跃时隙在旁边显示。

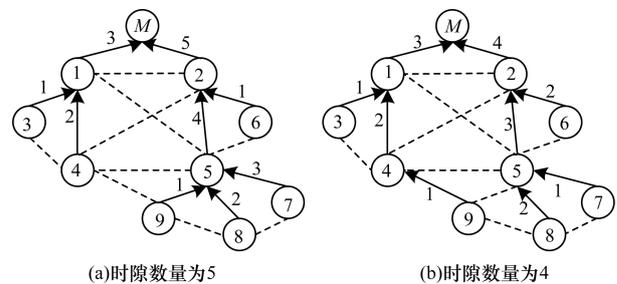


图2 平衡树中节点度的示例

本文对树的选择进行改进,以期减少转发树的链路数量,提高能效,同时降低其调度周期,以缩短延时时。例如,节点9可选择节点4或节点5作为其母节点。虽然无论哪个节点,转发树的链路数量相同,但是对于调度子问题,这两种选择所对应的转发树,在数据采集延时方面有所区别 (通过节点5需要5个时隙,通过节点4只需要4个时隙)。很显然,如果树中母节点的节点度较高,支持多个入向传输,则需要对这些传输进行依次调度,因此,应该避免这种现象。然而,树较高会导致调度周期较长,因为通往汇点路径上的母节点在转发自己的测量数据前,需要等待所有下行测量数据采集完毕。受这些现象的启发,对本文树的构建过程进行改进,以提高转发树的效率,具体内容见步骤3。

步骤3 寻找更为高效的路由,并在调度子问题中加以利用。

如算法2所示,每个节点 v 需要确定其是否属于集合 I_t 中的兴趣节点。如果是,则节点 v 运行算法2,寻找效率更高的路由或像上文讨论那样寻找可以降低调度长度的母节点。

算法2 节点 v 的转发树改进方法

1. 删除从节点 v 到首个兴趣节点之间的连续链路,对于在树 t 中具有1个以上子节点的节点,从节点 v 到这种节点之间的连续链路同样删除。用 b 表示检测到的节点。

2. $R \leftarrow$ 删除链路总量;
3. $\text{Best}_{\text{candidate}} \leftarrow b$;
4. 以 R 为半径,从 v 开始运行 BFS 算法;
5. if 在该半径内从非连通主树 t 上找到除 b 之外的其他节点 then
 6. 候选节点 \leftarrow 将距离最近候选节点加到列表中;
 7. $\text{Best}_{\text{weight}}$ 无穷大;
 8. for 候选列表中的每个节点 g do
 9. $H_g \leftarrow$ 从 g 到汇点间的跳数
 10. $D_g \leftarrow$ 节点 g 的度数;
 11. if $(D_g + H_g) < \text{Best}_{\text{weight}}$ then
 12. $\text{Best}_{\text{weight}} = D_g + H_g$;
 13. $\text{Best}_{\text{candidate}} \leftarrow g$;
 14. 利用最短路径将节点 v 与 $\text{Best}_{\text{candidate}}$ 相连。

如果某个节点为兴趣节点或具有 1 个以上子节点,则将删除从节点 v 到该节点之间的所有连续链路,如图 3(a) 所示。对节点 5,用箭头和 \times 表示从节点 5 到节点 2 的路径被删除(请注意,节点 2 有 2 个子节点)。设 b 表示具有不止一个子节点的节点, R 表示被删除的链路总量。在此步骤中,兴趣节点及其子节点从主树 t 中脱离(如图 3(a) 中已经从树脱离的节点 5)。然后,为检测出将 v 与主树 t 相连的替代路径, v 按照宽度优先搜索(Breadth-first Search, BFS)策略搜索周围半径为 R 的邻域,以便找到树 t 中可以改进调度长度并减少传输次数的节点(如图 3(b) 已被删除的节点 5 通过节点 4 可再次与树相连。因此,传输总次数从 5 降为 4)。为了找到最合适的路径,算法将树 t 半径 R 内找到的最近邻候选节点加到候选(Candidates)列表中。另外,对 Candidates 列表中的每个节点 g ,获取其节点度 D_g 和与汇点之间的跳数 H_g 。由于这些信息已经在步骤 2 中确定,因此可从各个节点处获得上述信息。如上文所示,使节点度和子树高度最小的候选节点将被选为新的母节点(算法 2 中的第 8 行~第 13 行)。

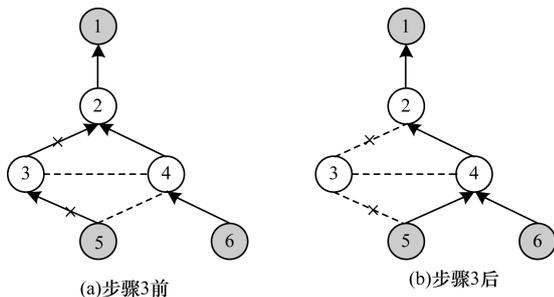


图 3 删除树中连续链路的示例

设 δ 表示半径 R 内的节点数量,利用 BFS 算法对 R 内所有节点进行遍历的复杂度为 $O(\delta)$ 。确定 Candidates 列表内所有节点的最优候选节点需要时间 $O(\rho)$,其中 ρ 表示 Candidates 列表的规模。算法 2 的复杂度为 $O(\delta + \rho)$,其中 ρ 是 δ 的子集, δ 大于 ρ 。因此,算法的时间复杂度为 $O(\delta)$ 。

3.2 分布式链路调度算法

在转发树构建完毕后,本文采用 TDMA 的思想进行链路调度,假设时间被分割为长度相等的时隙。每个时隙又分为调度周期和传输周期。在调度周期内构建调度方案,确定可同时调度的链路配置(配置的定义如上文所示)。在传输周期内,被选配置中的链路在传输周期内传输各自报文,一条链路一个报文,报文中包含各自经过压缩的测量数据。本节介绍一种分布式调度算法,目的是实现每条链路本地传输调度的同时,避免违反传输次序并遵守干扰约束,以保证数据传输的效果。为此,本文为每条链路定义一个以链路接收节点为中心的干扰邻域。确定落入链路干扰邻域内并处于活跃状态的发射节点所造成的累积干扰,并对干扰进行控制。另外,发射节点位于链路 l 干扰邻域内的所有链路可以与 l 的发射节点交换数据,进而进行调度。

对长度为 d_l 的每条链路 l (比如发射节点/子节点 i 和接收节点/母节点 j 间的数据传输),利用文献[14]中给出的干扰定位方法,构建以链路 l 接收节点为中心、半径为 $K_l \times d_l$ 的干扰邻域。对每条链路构建邻域后,邻域外的干扰对接收信号几乎没有影响。为保证链路 l 的数据传输效果,链路 l 接收节点可容忍的最大干扰可表示为:

$$I_l \max \triangleq \frac{P d_l - \alpha}{\beta} \quad (15)$$

其中, P 表示传输功率; α 表示功率损耗指数; β 表示为了保证误码率而提前确定的 SINR 阈值。文献[14]表明,已知常数 ε 且 $0 < \varepsilon < 1$,为了保证链路 l 的可用性,位于链路 S 干扰邻域外的活跃链路的发射节点所产生的干扰上限不应超过 $\varepsilon I_l \max$,在干扰邻域内的数据传输所产生的总体干扰效应不应超过 $(1 - \varepsilon) I_l \max$ 。干扰邻域的半径($K_l \times d_l$)取决于 ε 的值。 ε 越小,干扰邻域越大,调度开销也越大。可利用 ε 的值来控制调度开销。另外,每条链路的接收节点可根据无线干扰检测理论(Radio Interference Detection, RID)^[15],估计干扰邻域内每条链路发射节点产生的干扰功率。

下面介绍分布式调度算法。设 $\Delta_l(K_l)$ 表示链路 l 发射节点位于干扰邻域内时所有链路 k 构成的集合, SC 表示并行调度的链路集合。每个时隙开始时, SC 为空。从较高层面讲,通往叶节点的链路或子节点没有数据需要传输的链路进入就绪状态(它们无需等待任何下行数据)。这些链路的发射节点向其干扰邻域 $\Delta_l(K_l)$ 内的所有节点广播其优先级信息。根据 2 个指标来确定各个节点的优先级:母节点的度和通往根节点的跳数(在树构建阶段确定)。通过综合上述 2 个因素便可定量确定节点的优先级。母节点度较高且通往汇点跳数较大的节点,其优先级较大。发送节点的 ID 可打破这种关联(ID 较大的节点具有较高优

优先级)。链路 l 的优先级信息传播给 l 干扰邻域内的所有链路(发射节点)。此时,对于优先级在其干扰邻域内的所有链路(就绪状态)中最高并处于就绪状态的每条链路 l ,如果它的累积干扰 I_l 未超过 $(1 - \varepsilon)I_l \max$,且其接收节点没有为其他任何链路调度过,则可并行运行算法 3,使自己加入当前调度 SC 中。重复上述过程,直到没有更多处于就绪状态的链路加入当前调度 SC 为止。对于下个时隙,如果新链路的前任链路在前一时隙已被调度过,则将这些新链路加入就绪状态。相应地,重复上述步骤,直到所有链路都被调度为止。

算法 3 链路 l 的分布式调度算法

1. 链路 l 的发射节点向 $\Delta_l(K_l)$ 中的所有链路广播 SchReq;
2. 在将链路 l 添加到 S 后,链路 $k \in S \cap \Delta_l(K_l)$ 的接收节点计算干扰值 $I_{k \text{ tem}}$;
3. if 对链路 k 的任意接收节点有 $I_{k \text{ tem}} > (1 - \varepsilon)I_{k \text{ max}}$ then
4. 链路 k 向链路 l 发送一个 NotAcc 报文;
5. if 链路 l 接收到至少一个 NotAcc 报文 then
6. 链路 l 不把自己添加到调度 SC 中;
7. 链路 l 广播 RemSch 报文;
8. 接收到 RemSch 报文时,所有链路 k 将链路 l 从当前调度 SC 中删除;
9. else if 链路 l 未接收到 NotAcc 报文 then
10. 将链路 l 添加到当前调度 SC 中;
11. 链路 l 广播 AccSch 报文;
12. 接收到 AccSch 报文时,所有链路 k 通过将链路 l 添加到 SC 中来对 SC 进行更新。

4 实验评估

本文利用 OMNet++ 模拟器来评估所提算法的性能。传感器基于 IEEE 802.15.4 协议进行通信。无线频段为 2.4 GHz,数据传输率为 250 kb/s。评估指标为不同网络规模、拓扑结构和投影数量(压缩数据采集时)条件下完成数据采集所需要的时隙数量、传输次数和能耗。假设传感器网络具有 n 个节点,均匀分布在 700×700 单位距离的区域上,且生成的图形为连通图。同时假设所有节点采用正规化发射功率 $P = 1$ 。此外,假设路径损失指数 $\alpha = 3$,成功传输时的 SINR 阈值 $\beta = 2$ 。利用 CPLEX 来求解本文优化模型,利用 Java 来部署本文分布式算法。

以图 1(a)中包含 15 个节点的网络为例,考察基于 ILP 模型的 FTCS 联合问题的解,并与本文分布式解进行比较,如图 4 所示。显然,2 种方法构建出来的转发树具有相同数量的链路(因此收集传感数据的传输次数也相同)。但是 2 种方法的链路调度性能存在差异。首先,ILP 解可获得最优链路调度(调度长度为 8 个时隙),但其解为集中式解。本文分布式解随着 ε 值的不同而不同, ε 值越少表明干扰邻域越大,需要进行传输调度的区域越大,因此获得的解也更优。然而,此时调度开销也较大。当 $\varepsilon = 0.3$ 时,调度长度为 11 个时隙,与最优解相差 27% 左右。然而,ILP 集中式方法的计算成本远高于分布式方法。

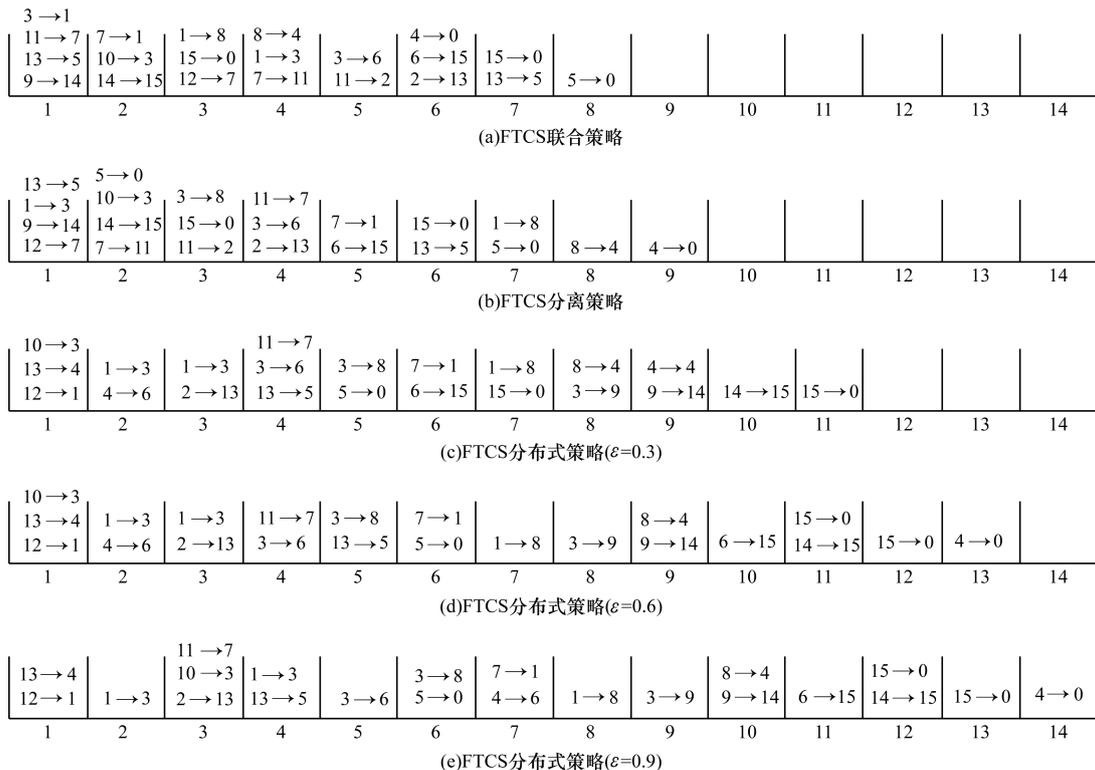


图 4 链路调度问题的解

比较 FTCS 分布式算法 (D-FTCS) 与其他 2 种方法的性能,这两种方法是将转发树构建与调度分开对待的方法。第 1 种方法为集中式算法,可获得 2 个子问题的最优解 (OTC-OLS);第 2 种算法只获得调度子问题的集中式最优解 (DTC-OLS)。具体结果如表 2 所示。

表 2 FTCS 算法性能 ($m = 20\% n$)

节点数量	D-FTCS	DTC-OLS	OTC-OLS
10	6	6	6
15	14	11	10
20	14	12	11
25	19	13	15
30	21	16	17
35	22	16	17
40	27	19	20

调度长度可反映网络的数据采集延时。由表 2 可以看出,可获得链路调度子问题最优解的 OTC-OLS 和 DTC-OLS,可使调度长度最小化,因此测量数据的采集速度更快。同时还可以看出,DTC-OLS 对大规模网络的调度长度较短,这是因为分布式方法对转发树的构建进行了改进。然而,OTC-OLS 方法通过构建转发树使包含的边数量最小,但是没有对转发树的构建进行改进。另一方面,D-FTCS 的性能较优,在最坏情况下与其他方法的性能差异不超过 26%。从计算复杂性角度来看,本文分布式算法的求解时间不到 2 s (网络包含 40 个节点),而 OTC-OLS 方法对包含 40 个节点 (107 条边) 的网络,求解时间需要 2.5 d。也就是说,转发树的构建只需几秒钟,链路调度需要 2.5 d。对小规模网络 (比如包含 20 个节点和 41 条边的网络),分布式方法的求解时间不到 1 s,OTC-OLS 方法需要数分钟 (转发树的构建需要数秒钟)。上述结果表明,干扰约束条件下链路调度问题的复杂性较高,难以按照集中式策略求解。

下面考虑一个包含 200 个节点的网络,并通过改变 FTCS 问题中的投影数量 m 来比较 D-FTCS 和 LLHC-MWF^[8] 的性能,实验结果如图 5 所示。该图表明,当投影数量较小时,D-FTCS 在传输次数和调度长度方面的性能均显著优于 LLHC-MWF。例如,当 FTCS 中的 $m = 5\% n$ 和 $m = 10\% n$ 时,用于网络数据采集的转发树的构建数量较少。由于采用压缩采集思想,采集数据时需要的传输次数显著降低,因此在较短时间内便可完成这些数据传输的调度。调度

长度基本小于等于 LLHC-MWF 的一半。然而,当投影数量增加时,构建的转发树增多,需要的传输次数也增多。相应地,调度长度和传输次数开始上升。同时,对于不同规模的网络,投影数量较少时 D-FTCS 的性能仍远优于 LLHC-MWF,如图 6 所示。例如,当 $m = 20\% n$ 时,FTCS 相对于 LLHC-MWF 的性能增益达到 29% ~ 44%。

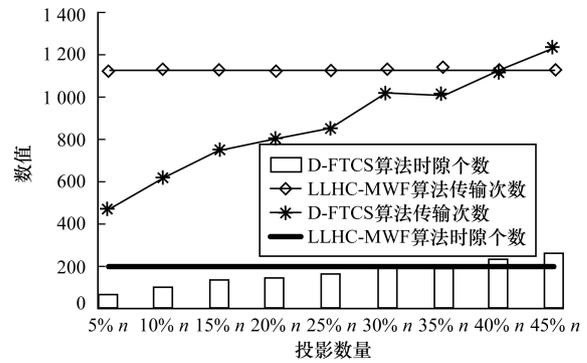


图 5 不同投影下 D-FTCS 和 LLHC-MWF 算法性能比较

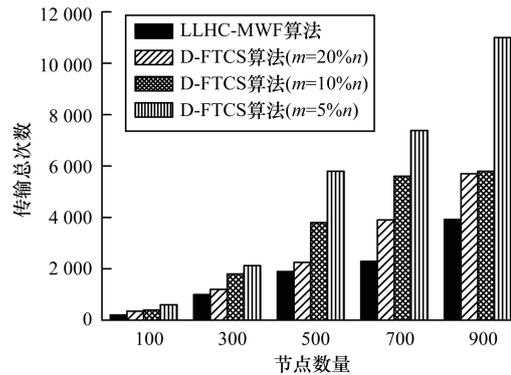


图 6 D-FTCS 和 LLHC-MWF 算法传输总次数比较

为体现本文方案的优越性,将 D-FTCS 和 LLHC-MWF 在同一实验环境下进行了能耗比较,实验结果如图 7 所示。可以看出,随着节点数量的增加,2 种方案的能耗都在增加,2 种方案在稀疏网络中的能耗都要低于在密集网络中的能耗。但总体而言,D-FTCS 的能耗要远低于 LLHC-MWF。仔细分析其原因可知,这主要是因为:1) LLHC-MWF 采用次采样的思路来进行节点上的压缩数据收集,相比于 D-FTCS 而言,LLHC-MWF 需要传输的数据量更多,通信开销更大。2) D-FTCS 以压缩感知理论为基础,通过构建转发树来收集节点上的测量数据,并对转发树的链路在干扰模型下进行了优化调度,避免了数据传输冲突,充分发挥了空间复用的优势,因此,取得了比 LLHC-MWF 更好的能耗表现。

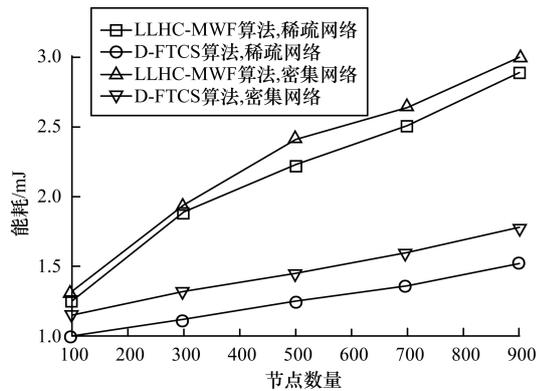


图7 D-FTCS和LLHC-MWF算法能耗比较

5 结束语

本文研究无线传感器网络在物理干扰模型下的压缩数据采集和调度,从数学角度阐述转发树构建和链路调度联合问题,目标是提高数据采集的能效,同时实现采集延时最小化。首先分析问题的复杂性,然后给出分布式求解算法,最后通过仿真实验验证所提算法的有效性。下一步将在多种应用场景下研究数据汇集可靠性与汇集树构建及调度算法之间的关系,以进一步提高数据汇集的质量。同时,基于压缩感知理论分析数据汇集树的构建过程对于延长网络生命周期的影响,设计基于压缩感知的数据汇集算法,提高网络生命周期。

参考文献

- [1] 戴志峰,李元香. 多层 Agent 数据融合的无线传感器决策网络及其多样性研究[J]. 计算机工程,2015,41(3): 198-203.
- [2] 陈慧杰,韩江洪,刘磊. 无线传感器网络中自然能收集分簇路由算法研究[J]. 计算机工程,2016,42(3): 143-147.
- [3] 林春立,崔杰. WSN 中自适应多路径安全路由协议[J]. 计算机工程,2016,42(6): 144-150.
- [4] LIU Xiaoyang, ZHU Yanming, KONG Linghe, et al. CDC; Compressive Data Collection for Wireless Sensor Networks [J]. IEEE Transactions on Parallel and Distributed Systems, 2015, 26(8): 2188-2197.
- [5] ZHENG Haifeng, YANG Feng, TIAN Xiaohua, et al. Data Gathering with Compressive Sensing in Wireless Sensor Networks: A Random Walk Based Approach [J]. IEEE Transactions on Parallel and Distributed Systems, 2015, 26(1): 35-44.
- [6] EBRAHIMI D, ASSI C. Optimal and Efficient Algorithms for Projection-based Compressive Data Gathering [J]. IEEE Communications Letters, 2013, 17(8): 1572-1575.
- [7] EBRAHIMI D, ASSI C. A Distributed Method for Compressive Data Gathering in Wireless Sensor Networks [J]. IEEE Communications Letters, 2014, 18(4): 624-627.
- [8] YAO Yanjun, CAO Qing, VASILAKOS A V. EDAL: An Energy-efficient, Delay-aware, and Lifetime-balancing Data Collection Protocol for Heterogeneous Wireless Sensor Networks [J]. IEEE/ACM Transactions on Networking, 2015, 23(3): 810-823.
- [9] MILOSEVIC B, CAIONE C, FARELLA E, et al. Sub-sampling Framework Comparison for Low-power Data Gathering: A Comparative Analysis [J]. Sensors, 2015, 15(3): 5058-5080.
- [10] GONG Dawei, YANG Yuanyuan. Low-latency SINR-based Data Gathering in Wireless Sensor Networks [J]. IEEE Transactions on Wireless Communications, 2014, 13(6): 3207-3221.
- [11] ZHENG Yuanqing, LI Mo. P-MTI: Physical-layer Missing Tag Identification via Compressive Sensing [J]. IEEE/ACM Transactions on Networking, 2015, 23(4): 1356-1366.
- [12] ANDREWS M, DINITZ M. Maximizing Capacity in Arbitrary Wireless Networks in the SINR Model: Complexity and Game Theory [C]//Proceedings of IEEE INFOCOM '09. Washington D. C., USA: IEEE Press, 2009: 1332-1340.
- [13] DAMS J, HOEFER M, KESSELHEIM T. Scheduling in Wireless Networks with Rayleigh-fading Interference [J]. IEEE Transactions on Mobile Computing, 2015, 14(7): 1503-1514.
- [14] LE L B, MODIANO E, JOO C, et al. Longest-queue-first Scheduling Under SINR Interference Model [C]//Proceedings of the 11th ACM International Symposium on Mobile Ad Hoc Networking and Computing. New York, USA: ACM Press, 2010: 41-50.
- [15] CHIWewe T M, MBUYA C F, HANCKE G P. Using Cognitive Radio for Interference-resistant Industrial Wireless Sensor Networks: An Overview [J]. IEEE Transactions on Industrial Informatics, 2015, 11(6): 1466-1481.
- [14] ZHANG Wenshu, DUAN Dongliang, YANG Liuqing. Relay Selection from a Battery Energy Efficiency Perspective [J]. IEEE Transactions on Communications, 2011, 59(6): 1525-1529.
- [15] DUAN Dongliang, QU Fengzhong, YANG Liuqing. Modulation Selection from a Battery Power Efficiency Perspective [J]. IEEE Transactions on Communications, 2010, 58(7): 1907-1911.
- [16] VICARIO J L, BEL A, MORELL A. Outage Probability Versus Fairness Trade-off in Opportunistic Relay Selection with Outdated CSI [J]. EURASIP Journal on Wireless Communications and Networking, 2009(1).

编辑 金胡考

编辑 金胡考