

无线传感器网络中基于协同压缩方法的 MAC 协议*

胡玉鹏¹, 林亚平^{1,2+}, 蒋红艳², 李小龙³, 周四望¹

¹(湖南大学 软件学院,湖南 长沙 410082)

²(湖南大学 计算机与通信学院,湖南 长沙 410082)

³(桂林电子科技大学 数学与计算科学学院,广西 桂林 541004)

MAC Protocol for Wireless Sensor Networks via Collaborative Compression

HU Yu-Peng¹, LIN Ya-Ping^{1,2+}, JIANG Hong-Yan², LI Xiao-Long³, ZHOU Si-Wang¹

¹(Software School, Hu'nan University, Changsha 410082, China)

²(College of Computer and Communication, Hu'nan University, Changsha 410082, China)

³(School of Mathematics & Computational Science, Guilin University of Electronic Technology, Guilin 541004, China)

+ Corresponding author: E-mail: yplin@hnu.cn, http://ss.hnu.cn/lnml/

Hu YP, Lin YP, Jiang HY, Li XL, Zhou SW. MAC protocol for wireless sensor networks via collaborative compression. Journal of Software, 2009,20(9):2483-2494. http://www.jos.org.cn/1000-9825/3316.htm

Abstract: Based on the broadcast nature of wireless signal, this paper proposes an approach that exploits the overhearing data omitted in conventional MAC protocols to eliminate the spatial correlation. Particularly, sensor nodes can prevent the transmission of redundant data in link layer by using the data they overhear to compress their own sensory data collaboratively before the transmission. Firstly, this paper formulates the problem that sensor nodes collaborate their compression to optimize their lifetime, and establish linear programming model for it. This paper also proposes a lower complexity ($O(N^2)$) heuristic node selection algorithm while achieving a nearly optimal performance. Based on that, an efficient Collaborative ComPression-based MAC (CCP-MAC) protocol is designed to implement the above node selection algorithm in a distributed way. As a result, the corresponding node can receive the data it overhears from the selected sensor node subset to compress redundant data before the transmission. Experimental results show that by exploiting the data nodes overhear CCP-MAC can collaborate the nodes to compress sensory data, thereby conserve energy significantly to prolong the network lifetime.

Key words: wireless sensor networks; spatial correlation; overhearing; node selection algorithm; CCP-MAC protocol

摘要: 基于无线信号的广播本质,利用传统 MAC 协议所忽略的串音(overhearing)数据,提出一种方法以在媒介访问控制层去除数据的空间相关性.根据串音所接收到的数据,事件监测节点间协同地对自身的数据进行压缩后再发送,从而在链路层减少冗余信息的传输.首先针对节点间的协同数据压缩问题进行量化,建立线性规划模型;进而提出一种近似最优的、更低时间复杂度($O(N^2)$)的启发式节点筛选算法.在此基础上,设计一种能量有效的、基于协

* Supported by the National High-Tech Research and Development Plan of China under Grant No.2006AA01Z227 (国家高技术研究发展计划(863)); the Hu'nan Provincial Natural Science Foundation of China under Grant No.06JJ20049 (湖南省自然科学基金)

Received 2007-12-24; Accepted 2008-03-14

同压缩方法的 MAC 协议(CCP-MAC),可分布式地控制节点实现该节点筛选算法,相应节点可从筛选出的被压缩节点子集中接收串音数据,融合冗余数据以后再进行发送.实验结果表明,CCP-MAC 利用串音数据协调节点进行数据压缩,可在很大程度上节约能量,延长网络的生命周期.

关键词: 无线传感器网络;空间相关性;串音;节点筛选算法;CCP-MAC 协议

中图分类号: TP393 **文献标识码:** A

无线传感器网络(wireless sensor networks,简称 WSN)在军事和民用领域潜在的广泛应用前景,如战场监视、环境和交通监测、灾难救助等,吸引了众多研究者进行研究^[1-4].由于单个节点的资源和能力有限,WSN 中节点往往密集分布,使得节点的数据间存在较大的空间相关性.而链路层协议是节点通信的基础,因此在链路层研究如何有效减少冗余数据的传输,由此减少进入网内的数据量,对于节省能量、延长 WSN 的网络生命周期具有十分重要的意义.

目前,MAC 协议已经吸引了研究人员进行了广泛的研究,提出了多种 MAC 协议,包括 S-MAC^[5],T-MAC^[6],B-MAC^[7],P-MAC^[8]和 SCP-MAC^[9]等.但这些协议主要沿用传统无线网络或 Ad-hoc 网络 MAC 协议的设计方法,着重讨论能耗与延时的折衷,而忽略了 WSN 中传感数据的内在特性.Vuran 等人将数据相关性与 MAC 协议设计结合起来考虑,基于所建立的空间相关性模型,设计一种选择代表性节点来传回数据的 MAC 协议 CC-MAC^[10].然而,与 CC-MAC 及传统 MAC 协议不同,本文提出的方法是利用以往 MAC 协议忽略的串音数据,结合协同的数据压缩方法达到在链路层去除冗余数据的目的.

基于无线信号的广播本质,WSN 中的多数 MAC 协议^[5,6,9]通常需要采用串音避免机制来节省能耗,即节点在接收到发给其他节点的 MAC 报头或 RTS/CTS 包时,通过检验目标地址判断是否为需要接收的数据.串音避免机制原本是为了在一定程度上降低节点的无效接收能耗,但实际上已经不可避免地浪费了相当的侦听和接收能量.尤其是对于采用唤醒前导机制的 MAC 协议如 B-MAC^[7],串音会浪费更多的能量,无论目标地址是否为本身,一旦监测到信号,节点均需要持续侦听信道,接收邻居节点数据.

因此,本文将利用节点接收的串音数据,根据串音所接收到的数据,节点之间协同地对自身数据进行压缩后再发送,从而在链路层减少冗余信息的传输.首先,本文针对节点间的协同数据压缩问题进行量化,建立网络生存周期向量,以获取最大的网络生命周期.然后将网络生存周期向量的最优解(lifetime vector optimization,简称 LVO)问题归结为一组线性规划问题.在此基础上,提出一种更低时间复杂度($O(N^2)$)的启发式节点筛选算法,近似最优地解决 LVO 问题.最后,针对事件监测应用,设计了一种基于协同压缩方法的 MAC 协议 CCP-MAC(collaborative ComPression-based MAC),可分布式地控制节点实现该节点筛选算法,相应的,节点从筛选出被压缩节点子集接收串音数据,融合冗余数据后再发送结果给 sink.实验结果表明,CCP-MAC 利用串音数据协调节点进行数据压缩,可以在很大程度上节约能量,延长网络生命周期.

本文第 1 节、第 2 节对一般协同压缩问题进行量化分析.第 3 节对 LVO 问题进行线性规划建模.第 4 节阐述启发式节点筛选算法.第 5 节描述基于协同压缩方法的 CCP-MAC 协议.第 6 节通过实验对 CCP-MAC 协议的性能进行评价.第 7 节总结全文,并讨论下一步的研究计划.

1 问题描述

WSN 通常可应用于事件监测,代表性的传感器节点如 MICAz 系列,节点有足够的存储空间用来存放必要的协议参数,突发事件均会被多个传感器节点探测到.由于节点分布密集,监测事件的节点间存在一定的空间相关性.节点在网络初始化阶段通过传感板蜂鸣器向邻居节点广播声音样本的方法估算出与邻居节点的距离 D .将单个事件监测节点看作随机信息源,若节点事先预置了空间相关性模型(本文假设仅与距离 D 相关)^[11],然后运用熵编码,如 Slepian-Wolf 编码^[12],节点可以在不通信的前提下估算出与邻居节点的条件熵及相等的压缩编码率.故在本文中若节点 B 的数据量(byte)为 L_B ,则基于与邻居节点 A 的距离 D ,可以估算出基于节点 A 的信息进行压缩后的数据量 $L_{B|A}$ 的值,及压缩率($L_{B|A}/L_B$),这是节点能进行下文协同数据压缩的前提条件.

发送和接收能耗分别为 $E_{tx}(L)=P_{tx}L t_{Byte}$, $E_{rx}(L)=P_{rx}L t_{Byte}$, 其中, t_{Byte} 为节点发送/接收单字节的时间, L 为数据长度(byte), P_{tx} 为节点发送功率, P_{rx} 为接收/侦听功率. 节点用于压缩 L 字节的能耗为 $E_{cp}(L)=P_{cp}L t_{cp}$, 其中, P_{cp} 为 CPU 压缩数据时的功率, 通常比射频模块的功率小许多. t_{cp} 为处理单字节的时间, 通常, $t_{cp} \ll t_{Byte}$.

首先考虑最简单的网络场景, 假设网络中只有 2 个相邻的节点 A 和 B , 当某事件发生时, A, B 同时监测到信息, 若让 A 先发送数据往 sink, 邻居节点 B 可以选择是否侦听 A 的数据, 对自身数据进行压缩后再转发给 sink. 若 B 选择不进行压缩, 直接发送数据 L_B 往 sink, 其能耗为 $E_B=P_{tx}L_B t_{Byte}$; 若 B 选择先侦听并接收 A 的数据 L_A , 基于压缩算法^[13,14]对自身数据进行压缩后得到 $L_{B|A}$, 然后再发送 $L_{B|A}$ 给 sink, 其总能耗为 $E_{B|A}=P_{rx}L_A t_{Byte}+P_{cp}L_B t_{cp}+P_{tx}L_{B|A} t_{Byte}$. 为了便于定量分析 B 进行压缩的前提条件, 假设 $L_B=L_A=L$, $L_{B|A}=L^*$, 且 $L^* \leq L$. 易知, 当 $E_{B|A} < E_B$ 时, 即公式(1)成立时, B 选择压缩后再发送数据将会更节省能量.

$$\frac{L^*}{L} < 1 - \frac{1}{P_{tx}} \times \left(P_{rx} + P_{cp} \times \frac{t_{cp}}{t_{Byte}} \right) \quad (1)$$

由公式(1)可知, 由于 $(t_{cp}/t_{Byte}) \ll 1$, 一般情况下, 节点发送功率大于接收和压缩功率, 公式(1)右边略小于 1, 故左边的压缩率 (L^*/L) 很可能会小于某个门限值, 从而节点会选择进行压缩后发送数据. 接下来需要分析的问题是如何设计合理的协同压缩方法, 可以最大化节点的生命周期, 直到其中某个节点 A 或 B 失效(本文指能量耗尽). 有两种基本的协同压缩方案可以选择:

(1) 方案 P_1 . 让节点 A 先发送数据给 sink, 节点 B 根据估算出的压缩率进行判断, 若能节省能耗则选择基于 A 的数据压缩后再发送; 否则直接发送数据给 sink;

(2) 方案 P_2 . 同理, 让节点 B 先发送数据给 sink, 节点 A 根据压缩率进行判断是否直接发送数据给 sink.

P_1 中, A, B 所消耗的能量分别为 $E_A^1 = E_A = P_{tx}L_A t_{Byte}$, $E_B^1 = \min\{E_B, E_{B|A}\}$; P_2 中, A, B 所消耗的能量分别为 $E_A^2 = \min\{E_A, E_{A|B}\}$, $E_B^2 = E_B = P_{tx}L_B t_{Byte}$.

若 P_1, P_2 在某节点失效前分别使用了 r_1, r_2 轮(即数据采集的次数), 设 e_A, e_B 分别为节点 A, B 的初始能量, 因此, 如何最大化节点的生命周期直到某个节点失效的问题可以形式化地表示为

目标函数:

$$\arg \max_{r_1, r_2} \{r_1 + r_2\}.$$

约束函数(s.t.):

$$\begin{aligned} r_1 &\geq 0, r_2 \geq 0, \\ E_A^1 \times r_1 + E_A^2 \times r_2 &\leq e_A, \\ E_B^1 \times r_1 + E_B^2 \times r_2 &\leq e_B. \end{aligned}$$

实际应用中, 节点的生命周期远大于每一次数据收集的时间, 故 r_1, r_2 可以作为实数看待, 则以上问题仍可看作标准的线性规划问题. 下节将针对更一般的网络场景进行量化分析, 建立网络生命周期向量.

2 网络生命周期向量

WSN 中通常某个事件发生时有多节点监测到, 节点间存在空间相关性, 如图 1 所示. 节点通过竞争无线信道, 将各自的数据发回 sink, 且每次监测到事件的节点可能不同, 节点数目也可能不相同. 首先给出以下参数说明: $M=\{1, 2, \dots, N\}$ 为监测到某事件的所有节点集合, N 为监测事件的平均节点个数, $M_n=\{i \in M | i \text{ 为 } n \text{ 的邻居}, n \in M\}$ 为 M 中所有 n 的邻居节点, $e_n(n \in M)$ 为节点 n 的初始能量, $E_n(U)(n \in M, U \subset M, n \notin U)$ 为节点 n 基于节点子集 U 进行串音数据接收、压缩、发送数据的能耗, $E_n(\emptyset)$ 表示节点 n 不进行压缩时直接发送数据的能耗.

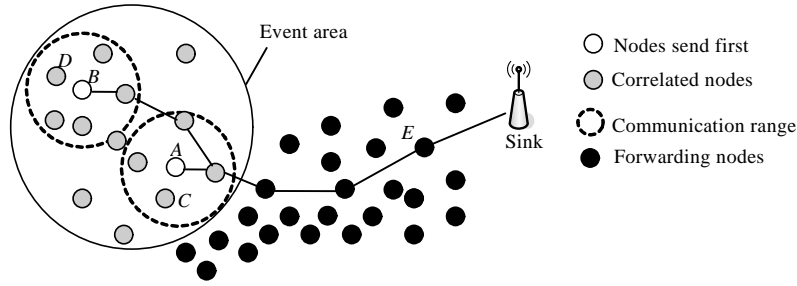


Fig.1 Spatial correlation in WSN

图1 无线传感器网络的空间相关性

2.1 协同压缩方案与策略

由第 1.2 节中的基本协同压缩方案可知,要使得节点在每一轮的数据采集过程中进行协同的数据压缩,所设计的协同压缩方案最后需要达到两个作用:

- (1) 控制哪些节点先发,哪些节点后发;
- (2) 若发送顺序已经确定时,节点需要知道基于哪些节点进行数据压缩.

在给出协同压缩方案的定义之前,本文规定必须遵循此约束条件:即在每轮的数据采集过程中,任何监测到事件的节点只能基于集合 U 中节点的传感数据进行压缩,而 U 中的所有节点均不进行数据压缩.因此,拓展到一般网络场景,协同压缩方案定义如下.

定义 1. 协同压缩方案(collaborative compression policy,简称 CCP)为一个映射函数 $P: v \rightarrow V$,其中, $v \subseteq M$, v 为 M 的某个子集, V 为 v 的所有子集(包括空集)的集合.对于 $\forall n \in v$,节点 n 均能基于集合 $P(n)$ 中节点的数据进行压缩.基于上文的约束条件可知 $P(i) = \emptyset, i \in P(n), \forall i, n, n$ 和 $P(n)$ 中的节点分别称为压缩节点和被压缩节点.

由定义 1 可知,当 P 确定时,首先,节点的发送顺序即可确定,让不压缩的节点集合 $\cup P(n) (\forall n \in v)$ 中的节点先发,需要压缩的节点集合中的节点后发.然后,当节点 n 可以接收到多个邻居节点的串音数据时,可以基于 $P(n)$ 中节点的串音数据进行压缩,并发送数据.且当只有局部节点参与通信时(如部分节点睡眠,失效), P 仍然是可行的.

定义 2. 协同压缩策略(collaborative compression strategy,简称 CCS)是由一套协同压缩方案所组成的方案——轮数集合 $S = \{(P_1, r_1), \dots, (P_m, r_m)\}$,其中, m 为正整数, (P_i, r_i) 表示协同压缩方案 P_i 在某段时间内在节点子集 $v \subseteq M$ 的节点上执行了 r_i 次.若 e_n^{rem} 表示节点 n 在执行 S 前的剩余能量,则 S 必须要满足公式 $\sum_{i=1}^m E_n(P_i(n)) \times r_i \leq e_n^{rem}, \forall n \in v$ 以保证其可行性,即 v 中的所有节点均不会消耗多于自身的能量.

2.2 网络生命周期向量

节点执行协同压缩任务过程中会由于能量耗尽而相继失效,设 N 个监测节点分 $K (1 \leq K \leq N)$ 个阶段陆续失效,为便于说明,本文假设 $K=N$,阶段 $k (1 \leq k \leq K)$ 表示从第 $k-1$ 个节点失效开始直到 k 个节点失效结束.

定义 3. 网络生命周期向量 $LV = (LV_1, \dots, LV_K)$ 是由 K 个元素组成的一个向量,其中, LV_k 为该向量的第 k 个元素,表示第 k 个阶段结束时(即第 k 个节点失效)的时间开销(从网络工作开始计时).且该向量是 K 个实际可行的协同压缩策略 S 执行后的结果,即每个协同压缩策略 S 控制一个阶段.

对于不执行协同压缩策略时的网络生命周期向量 \widetilde{LV} 满足 $\widetilde{LV}_k \leq LV_k, 1 \leq k \leq K, LV$ 为其他任何执行协同压缩策略时的生命周期向量.因每次发生事件的地点及参与任务的节点集合不同,若仅关注小部分节点的生命周期,即 LV 中的某些元素 LV_k 的值,并不能达到最大化网络生命周期的目标,最后,少量节点并不一定能够连通以协同工作.因此,需要将 LV 中的各个元素依次进行最大化,让尽可能多的节点能同时工作,即通过保证第 $1, 2, \dots, (k-1)$ 个元素最大化的条件下,再保证第 k 个元素条件性的最大化,使得节点几乎同时失效.

定义 4. 最优网络生命周期向量 \overline{LV} 满足, $\forall LV, LV \neq \overline{LV}$, 则 $\exists k, 1 \leq k \leq K$, 使得:

- (1) $\overline{LV}_i \geq LV_i, \forall i \in \{1, \dots, k\}$;
 (2) 且至少存在一个严格的不等式,即至少满足 $\overline{LV}_k > LV_k$.

定义 5. 在定义 4 的基础上,LVO(lifetime vector optimization)问题定义为:设计 K 个可行的协同压缩策略 S 分别控制 K 个阶段,使得最后得到的生命周期向量 $LV = \overline{LV}$.

本文使用下面的迭代控制算法所产生的协同压缩策略 S 来解决 LVO 问题.证明略.

算法. 解决 LVO 问题的迭代控制算法.

Step 1. 监测事件 N 个节点拥有最初的能量,首先找到第 1 个协同压缩策略 S ,控制第 1 个阶段, $LV_1 = \overline{LV}_1$.

Step k . $2 \leq k \leq K$,由于前 $k-1$ 个步骤已经找到 $\overline{LV}_1, \overline{LV}_2, \dots, \overline{LV}_{k-1}$,那么在第 k 步需要找到 k 个可行的协同压缩策略 S ,使得第 i 个阶段结束时的时间开销为 $\overline{LV}_i, \forall 1 \leq i < k$,且使得第 k 个阶段结束时的时间最大化,即为 \overline{LV}_k .

3 线性规划建模

本节将通过 LVO 问题进行线性规划建模,求解出可以获得最大生命周期向量 \overline{LV} 的协同压缩方案 P 和协同压缩策略 S .本节以迭代控制算法的第 1、第 2 阶段为例建模,然后可以推广到后续的 $K-2$ 个阶段.每个阶段均要找出一个协同压缩策略 S ,能使对应于每个 P_i 的 r_i 的和为最大值,从而最大化网络生命周期.

考虑第 1 阶段,要求解 P ,首先要确定让哪些节点成为先发送数据的被压缩节点,设在使用某个协同压缩方案 P 时所有不进行压缩的节点的集合为 U_1 ,则 U_1 必须满足 $\forall n \in (M - U_1), U_1 \cap M_n \neq \emptyset$.也就是意味着集合 $(M - U_1)$ 中的每个节点至少要能够接收到 U_1 中某一个节点的数据,并设 \mathcal{U}_1 为所有 U_1 的集合:

$$\mathcal{U}_1 = \{U_1 \subseteq M \mid \forall n \in (M - U_1), U_1 \cap M_n \neq \emptyset\}.$$

一旦确定了不进行压缩的节点集合 U_1 ,则所有其他节点可以从 U_1 中选择最节约能耗的串音数据进行压缩,从而确定每一轮的协同压缩方案 P .因此,可以将协同压缩方案 P 的范围缩小并确定所要求的 P 如公式(2)所示.其中,上式中的“1”均表示第 1 阶段.

$$P(n) = \Psi_1^n(U_1) = \begin{cases} \emptyset, & \forall n \in U_1 \\ \arg \min_{W \subseteq (U_1 \cap M_n)} \{E_n(W)\}, & \forall n \in (M - U_1) \end{cases} \quad (2)$$

设 $r_1^{U_1}$ 表示第 1 阶段基于集合 U_1 进行协同数据压缩的总轮数,如前面提到的,由于节点的生命周期远大于每次数据采集的时间,故 $r_1^{U_1}$ 可作为实数来研究.第 1 阶段的最大生命周期问题可以归结为以下线性规划问题:

目标函数:

$$\arg \max_{r_1^{U_1}, \forall U_1 \in \mathcal{U}_1} \sum_{\forall U_1 \in \mathcal{U}_1} r_1^{U_1} \quad (3)$$

约束函数(s.t.):

$$\begin{cases} r_1^{U_1} \geq 0, & \forall U_1 \in \mathcal{U}_1 \\ \sum_{\forall U_1 \in \mathcal{U}_1} (r_1^{U_1} \times E_n(\Psi_1^n(U_1))) \leq e_n, & \forall n \in M \end{cases} \quad (4)$$

因此,解决以上线性规划问题则可获得一个协同压缩策略 S ,从而最大化第 1 阶段的网络生命周期为 \overline{LV}_1 (实际上,在 \overline{LV}_1 时刻失效的节点可能不止一个,这可以通过检查节点剩余的能量得知具体的失效节点个数).然后考虑第 2 阶段,需要找到两个协同压缩策略 S ,使得第 1 个阶段的网络生命周期为 \overline{LV}_1 ,且第 2 个阶段的生命周期最大化.假设 \overline{LV}_1 时第 1 阶段只有一个节点 n_1^* 失效,类似地,可以构造 $\Psi_2^n(U_2), \forall n \in (M - \{n_1^*\})$ 以及 U_2, \mathcal{U}_2 ,故第 2 阶段可以归结为以下线性规划问题:

目标函数:

$$\arg \max_{r_1^{U_1}, r_2^{U_2}, \forall U_1 \in \mathcal{U}_1, \forall U_2 \in \mathcal{U}_2} \sum_{\forall U_2 \in \mathcal{U}_2} r_2^{U_2} \quad (5)$$

约束函数(s.t.):

$$\begin{cases} r_1^{U_1} \geq 0, \forall U_1 \in \mathcal{U}_1, r_2^{U_2} \geq 0, \forall U_2 \in \mathcal{U}_2 \\ \sum_{\forall U_1 \in \mathcal{U}_1} r_1^{U_1} = \overline{LV}_1 \\ \sum_{\forall U_1 \in \mathcal{U}_1} (r_1^{U_1} \times E_{n_1^*}(\Psi_1^{n_1^*}(U_1))) \leq e_{n_1^*} \\ \sum_{\forall U_1 \in \mathcal{U}_1} (r_1^{U_1} \times E_n(\Psi_1^n(U_1))) + \sum_{\forall U_2 \in \mathcal{U}_2} (r_2^{U_2} \times E_n(\Psi_2^n(U_2))) \leq e_n, \forall n \in (M - \{n_1^*\}) \end{cases} \quad (6)$$

若第 1 阶段多于一个节点失效,则要在公式(6)中对每个可能的 n^* 进行说明.

基于公式(5)、公式(6)可以推广到后续 $K-2$ 个阶段.解决该线性规划问题的复杂度取决于集合 U_i 中元素的个数,第 1 步复杂度最小,仅取决于 U_1 ,而 U_1 由监测事件节点个数和网络拓扑等因素决定.最坏情况下,若监测事件节点均互为邻居,则可能包含所有 M 的非空子集. U_1 元素个数为 (2^N-1) ,因此,解决第 1 步的时间复杂度为 $O(2^N)$.下面介绍复杂度相对更低且近似最优的启发式节点筛选算法,用于筛选出被压缩节点子集,解决 LVO 问题.

4 启发式节点筛选算法

本节提出低时间复杂度、近似最优的启发式节点筛选算法,用逐步贪婪的方法实现以上协同压缩方案和策略.设第 1 阶段的协同压缩策略 $S = \{(P_1, r_1), \dots, (P_m, r_m)\}$ 且 $r_i = R (i=1, \dots, m, 1 \leq R)$,每个协同压缩方案使用 R 轮.设时间间隔 $t, t=1, 2, \dots$,表示从 $(t-1) \times R + 1$ 轮开始直到 $t \times R$ 结束时所经历的时间段.设 $e_n^t (e_n^t \geq 0, \forall n \in M)$ 表示节点 n 在时间间隔 t 开始时的剩余能量.算法使用的 R 值越小,越能及时反映节点的能量状态,故算法的精确度越高.

该启发式节点筛选算法构造第 1 阶段协同压缩策略 S 的伪代码如图 2 所示,其中,第 3~19 行为构造每个时间段 t 的协同压缩方案 P_t 的代码.本算法的主要思想是:循环构造各阶段的协同压缩方案 P_t ,该 P_t 使得在 t 结束后,所有 N 个节点中剩余能量最小的节点的能量最大化,即在时间段 t 寻找的最佳方案 P_t^* ,满足:

$$P_t^* = \arg \max_P \{ \min_{n \in M} \{ e_n^{t+1} | e_n^{t+1} = e_n^t - R \times E_n(P_t(n)) \} \}.$$

为了构造 P_t^* ,首先从所有节点均不进行压缩开始,接下来逐步贪婪每个时间段的最佳方案,在第 7 行筛选出剩余最小能量的节点 n^* ;然后在第 8 行筛选出能让 n^* 节约能耗的被压缩邻居节点子集 W .若 $W = \emptyset$,则节点不宜压缩;否则,让 n^* 基于这些节点进行压缩.若有多个选择的对象 l ,筛选满足第 12 行条件的最佳节点 l^* 进行压缩 (1.(.)表示括号内条件成立时为 1,否则为 0).最后,在第 13 行中,节点要么属于子集 U ,不进行数据压缩,即被其他节点用来进行数据压缩或/且自身不宜进行数据压缩(如第 10 行所示);要么属于子集 V ,即基于 U 的节点进行协同数据压缩.基于所构造的协同压缩方案 P_t ,该节点筛选算法构造的协同压缩策略如图 2 中第 20~25 行代码所示.图 2 算法的输入为所有节点的初始能量值 (e_1, \dots, e_N) ,输出则为一系列的协同压缩方案 CCP 和第 1 阶段的协同压缩策略 CCS.最后,对于后续的阶段 2, 3, ..., K 可以采用类似的方法得到.相对第 1 阶段,由于节点数量逐渐减少,后续阶段的时间复杂度更低.

由于后续 $K-1$ 个阶段节点相继失效,故 loop² 内的代码最多从 N 个节点中筛选出能量最小的节点,其复杂度为 $O(N)$.同样地,loop² 内的代码最多被执行 N 次,故该启发式节点筛选算法用于获得协同压缩方案的第 3~19 行代码的时间复杂度最坏为 $O(N^2)$.而算法中的 loop¹ 执行次数上限为 $e_n / E_{lowest} E_{lowest}$ 为每次数据采集时节点的最低能耗,仅与节点的生命周期有关,与节点的规模 N 无关.因此,其时间复杂度仍然为 $O(N^2)$.

5 基于协同压缩的无线传感器网络 MAC 协议(CCP-MAC)

本节设计一种基于协同压缩方法的空间相关性 MAC 协议 CCP-MAC,以分布式地协调各事件监测节点进行协同数据压缩和发送.CCP-MAC 是针对空间相关性的 MAC 协议,除了引入新的信道竞争机制来代替以往 MAC 协议的信道竞争机制,CCP-MAC 完全可以和一些底层的 MAC(如 S-MAC, B-MAC)协议结合使用.

```

Heuristic_NodeSelection_Algorithm_for_CCS( $e_1, \dots, e_N$ )
1.  $S \leftarrow []; t \leftarrow 0; e_n^{rem} \leftarrow e_n, \forall n \in M$ 
2. loop1
3.    $t \leftarrow t + 1; e_n^t \leftarrow e_n^{rem}, \forall n \in M$ 
4.    $P_t(n) \leftarrow \emptyset, \forall n \in M$ 
5.    $U \leftarrow \emptyset; V \leftarrow \emptyset$ 
6.   loop2
7.      $n^* \leftarrow \arg \min_{n \in M - (U \cup V)} \{e_n^t - R \times E_n(P_t(n))\}$ 
8.      $W \leftarrow \{n \in (M - V) \mid E_n(n) < E_n(\emptyset)\}$ 
9.     if  $W = \emptyset$ 
10.       $U \leftarrow U \cup \{n^*\}$ 
11.     else
12.       $l^* \leftarrow \arg \max_{l \in W} \{\min\{e_n^t - R \times E_n(\{l\}), (e_n^t - R \times E_n(\emptyset)) / 1. (l \notin U)\}\}$ 
13.       $P_t(n^*) \leftarrow \{l^*\}; U \leftarrow U \cup \{l^*\}; V \leftarrow V \cup \{n^*\}$ 
14.     endif
15.     if  $U \cup V = M$ 
16.       break loop2
17.     endif
18.   end loop2
19.   return  $P_t$ 
20.  $S \leftarrow [S, P_t]; e_n^{rem} \leftarrow e_n^{rem} - R \times E_n(P_t(n)), \forall n \in M$ 
21. if  $\min_{n \in M} \{e_n^{rem}\} \leq 0$ 
22.   break loop1
23. endif
24. end loop1
25. return  $S$ 

```

Fig.2 Heuristic node selection algorithm for CCS of phase 1

图 2 构造第 1 阶段 CCS 的启发式节点筛选算法

5.1 数据包结构

CCP-MAC 基于 TinyOS1.X 所定义的数据包如图 4 所示. CCP-MAC 需要在数据包中定义额外的字段 $cTag$, 用于表明数据包的压缩标识. 若 CCP-MAC 使用 RTS/CTS 包, 如图 4(a)所示, 其中有单独的 $cTag$ 字段, 当节点接收到 RTS 包时, 将回复包含相同 $cTag$ 值的 CTS 包, 这样周围的节点均可以知道数据包的类型. 若采用基于低功耗侦听技术的唤醒前导机制而不使用 RTS/CTS/DATA/ACK 机制, 则在 MAC 层数据包报头定义 $cTag$ 字段, 如图 4(b)所示, 节点通过 MAC 报头即可以判断数据包的类型, 进行相应的数据压缩或直接丢弃.

<pre> // control packet —— RTS, CTS, ACK typedef struct { PhyHeader phyHdr; char type; // indicates RTS/CTS/ACK uint8_t cTag; // 0,1 or 2 uint16_t toAddr; uint16_t fromAddr; uint16_t duration; int16_t crc; // for CRC check } __attribute__((packed)) MACCtrlPkt; </pre>	<pre> // MAC header typedef struct { PhyHeader phyHdr; char type; uint8_t cTag; uint16_t toAddr; uint16_t fromAddr; uint16_t duration; uint8_t fragNo; } __attribute__((packed)) MACHeader; </pre>	<pre> // MAC packet typedef struct { MACHeader hdr; int8_t data[MAX_MAC_PAYLOAD]; int16_t crc; } MACPkt; </pre>
(a) Control packet structure (a) 控制包结构	(b) MAC header structure (b) MAC 报头结构	(c) MAC packet structure (c) MAC 层数据包结构

Fig.4 Packet structures defined in TinyOS1.X for RTS/CTS/ACK, MAC header, and MAC packet

图 4 针对 TinyOS1.X 定义的 RTS/CTS/ACK 数据包、MAC 报头及 MAC 层数据包

5.2 CCP-MAC 协议

CCP-MAC 的核心思想是: 让不宜压缩或剩余能量多的节点先发送数据, 能量少的节点基于先发节点的数据进行协同压缩后再发送. 因此, 每个节点 n 首先根据与邻居节点间的距离, 估算出相应的压缩率, 并基于公式(1)筛选出能用来进行协同数据压缩的节点集合 W , 若 $W = \emptyset$, 即表示自身不宜进行压缩, 需要最优先发送数据; 否则, 节点计算在不压缩时发送数据后的剩余能量 $e_n^{rem} = e_n^{rem} - R \times E_n(\emptyset)$, 并让剩余能量多的节点先发. CCP-MAC 中, 节点需要存储一个邻居信息表, 包含节点 ID、距离和剩余能量, R 表示为更新信息表的周期, 如每 R 个事件更新

一次,从而了解哪些邻居节点已失效、目前邻居节点的距离及剩余能量(由于电池的消耗将导致节点电压下降,故剩余能量可由节点工作电压值表示).

首先,CCP-MAC 采用一种基于能量优先级的、非均匀概率分布的竞争窗口机制决定发送顺序,节点在不同的时槽选择不同的发送概率,让不宜压缩或剩余能量多的节点选择前面时槽的概率较高,而能量少的节点选择偏后时槽的概率较高.该竞争窗口机制采用线性的窗口增长机制 $CW_i = (i+1) \times CW_0$,节点在竞争窗口 $[1, CW_i]$ 内选择发送时槽 c .采用线性窗口机制是由于前两个竞争阶段的窗口大小与 802.11MAC 相同,且研究表明,发生 3 次以上连续冲突的概率几乎可以忽略^[15],故 CCP-MAC 设定的退避级数 i 最大值为 3,从而有利于减少竞争延时.节点在时槽 c 选择发送数据的概率为 $\Pr(c) = \beta \left(\frac{1}{1+e^{-\alpha(c+1)}} - \frac{1}{1+e^{-\alpha c}} \right)$, $\alpha = \frac{e_n^{rem}}{e_n}$, $\forall n \in M, c \in [1, CW_i]$. 其中, $\alpha(0 < \alpha \leq 1)$ 为剩余能量与初始能量的比值, β 为平衡系数.图 5 描述了 α 对不同时槽的发送概率的影响. Y 轴为发送概率, X 轴为时槽号.由图 5(a)可知, α 固定时,越前面的时槽发送数据的概率越大,越后面的时槽发送概率相对较小.但随着 α 的增大,越前面的时槽发送概率会增大,而越后面的时槽发送概率有所缩小,从而可以很好地控制能量大的节点优先选择前面的时槽发送数据,能量少的节点优先选择偏后的时槽发送数据.图 5(b)描述 $\alpha=0.3$ 、取不同 β 值时概率曲线变化情况. β 越大时曲线变化越明显,后面时槽的发送概率也越高.CCP-MAC 对于不宜压缩的节点,设 $\alpha=1$,故能够以最优先发送数据.

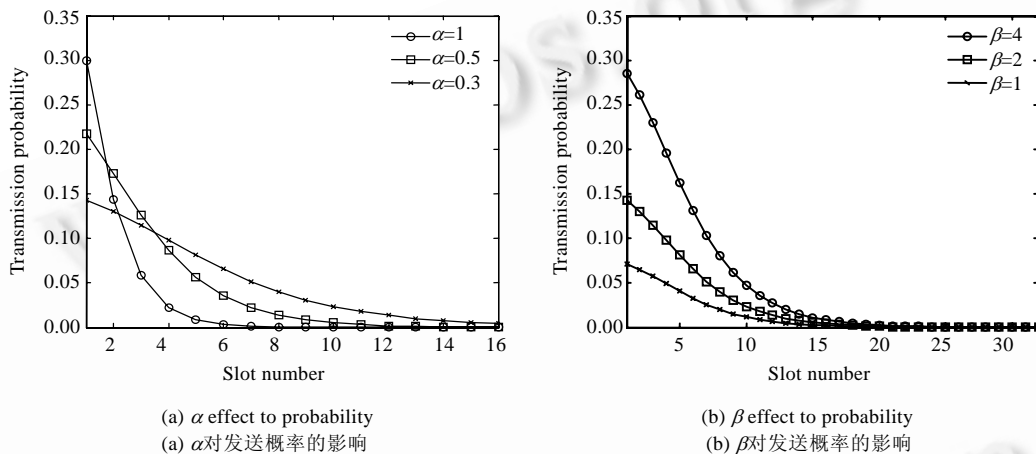


Fig.5 Transmission probability for each slot

图 5 节点在各时槽的发送概率

然后,节点在选择自己的发送时槽时,若其不属于不宜压缩的节点,则根据图 2 的第 12 行计算自己的最优压缩节点 l^* (节点最初不知道集合 W 中哪些节点属于 U 及剩余能量,故先假设节点均不属于 U ,在接下来几轮数据发送过程中,通过侦听邻居节点的数据包的 $cTag$ 是否为 0,从而记录哪些邻居节点属于 U ;而邻居节点的剩余能量可以通过周期性的更新邻居节点信息表后得到,并不影响协议性能),紧接着执行以下协议流程.

- (1) 节点在时槽等待过程中,若侦听到某数据包的目标地址不是本节点,则通过数据包或 MAC 报头 $cTag$ 来进行相应地处理:
 - 1.1) $cTag=0$,表示数据包是由不宜压缩或高能量节点发出,若该数据包源节点为 l^* ,则接收语音数据并和自身数据进行压缩打包(可由多个节点同时接收),并设 $cTag=1$,等下一次信道空闲时发送;若数据包源节点不为 l^* ,则直接转入睡眠状态,以节省侦听能量;
 - 1.2) $cTag=1$,表示数据包已经压缩过,不宜再进行压缩,节点直接转入睡眠;
 - 1.3) $cTag=2$,表示中转数据包,与本地数据相关性小,故不进行压缩,节点也转入睡眠.
- (2) 若节点在时槽等待过程中,数据包的目标地址为本节点,则节点接收该数据:

- 2.1) 若 $cTag < 2$, 则首先在本地进行数据压缩后, 将 $cTag$ 设定为 2, 等待下一次发送;
 - 2.2) 若 $cTag = 2$, 表明为中转数据, 可能与本地数据相关性较小, 故不再进行压缩, 等待下一次信道空闲再进行发送, 且设定 $\alpha = 1$, 从而可以最优先发送数据, 减少延时.
- (3) 若节点在选定时槽到来时还未侦听到数据, 表明自己属于 U , 将数据包 $cTag$ 设置为 0, 并立即发送数据.

CCP-MAC 应用场景如图 1 所示, 事件监测区域的节点 A, B 能量较多, 优先竞争到信道发送数据, 周围邻居节点则根据数据包源地址 $fromAddr$ 判断该节点是否可用来进行数据压缩, 如节点 C, D 在等待发送时槽过程中分别发现 A, B 的串音数据可用来进行压缩, 则接收数据并进行压缩, 设 $cTag = 1$, 等待下次信道空闲再发送. 而中转节点 E , 接收到数据包时由于发现 $cTag = 2$, 表明为中转数据, 则会设定 $\alpha = 1$, 从而可以最优先发送数据.

6 性能评估

6.1 网络参数

本节基于 OMNET++ 模拟平台开发模拟程序, 对筛选算法和 CCP-MAC 在较大规模 WSN 中的性能进行评估, 参考 MICAz 节点, 使用以下网络场景和参数:

- (1) 800 个节点随机分布在 $600 \times 600 \text{m}^2$ 的区域, 在某些子区域以一定频率模拟突发事件, 监测节点每次发送 5 个 40Byte 的数据包往 sink, 监测节点数目 $N \in [20, 100]$;
- (2) 将 CCP-MAC 与 S-MAC, B-MAC 结合, 测试节点的生命周期. 占空比为 5%, $P_{tx} = 60 \text{mW}$, $P_{rx} = 45 \text{mW}$, $P_{cp} = 20 \text{mW}$, 发送速率为 250Kbps, 通信距离为 60m;
- (3) 邻居节点信息表更新周期 $R \in [500, 10000]$ (事件数), 压缩率 $L^*/L \in [0.1, 0.9]$.

6.2 实验分析与比较

实验首先验证在理想状况下(节点有序无冲突的发送数据)线性规划模型的性能, 并比较基于多个被压缩节点与基于单个被压缩节点压缩数据时, 节点的最大生命周期增益. 当监测节点不进行压缩时, 每次事件需要传输 200Byte 的数据, 假设基于某节点进行数据压缩后, 数据量为 L^* , 为便于分析, 假设基于另外 m 个节点进行数据压缩后变为 $(L^* m^z)$, 即压缩率均为 z . 若采用线性规划模型来求解 CCS, 图 6 描述了第一阶段结束时节点的最大生命周期的增益, LV_1 表示第 1 阶段. 其中, $z = 0.45, 0.85, 1, L^*/L \in [0.1, 0.9]$, $z = 1$ 表示只基于一个被压缩节点压缩数据. $z = 0.45$ 时, 节点生命周期增益要稍好; 当 $z = 0.85$ 时, 相对于 $z = 1$, 基本没有增益. 实际应用中, $z = 0.45$ 的概率不大, 可见基于多个压缩节点进行压缩有些得不偿失, 这是因为相对于压缩节省的发送能耗, 基于多个节点进行数据压缩需要增加更多的接收能耗. 该结果证实了启发式筛选算法基于一个最优节点压缩数据的合理性. 压缩率为 0.1 时, 增益最多为 120% 左右, 这是因为并非所有节点均能进行压缩, 某些节点要作为被压缩节点发送数据.

图 7 描述 $z = 0.85$ 时的节点生命周期增益. 实验比较了第 1、第 5、第 10 阶段 (LV_1, LV_5, LV_{10}) 的增益情况, LV_1 的增益最大, LV_{10} 最小, 本文的设计目标主要通过延长最早失效节点的生命周期来保证网络的协同工作. 启发式算法的性能几乎与最优 CCS 一致, 且在 LV_5, LV_{10} 中性能要优于最优 CCS, 这是因为最优 CCS 基于多个节点进行数据压缩, 为了 LV_1 有最大值, 必定要多个节点来发送数据, 帮助第 1 个失效的节点达到最好压缩效果, 造成后续的节点生命周期变短. 实验将 CCP-MAC 与 B-MAC, S-MAC 相结合, 使用 CCP-MAC 时节点生命周期明显延长. 采用 CCP-MAC 后的 B-MAC, S-MAC 协议的增益要低于最优 CCS 的性能, 这是因为冲突造成一定的丢包率, 抵消了部分增益; 采用 CCP-MAC 后的 B-MAC 的增益比 S-MAC 要多, 这是由于 B-MAC 浪费的串音能耗更多.

图 8 描述了邻居信息表更新频率 R 取不同值时的 CCP-MAC 性能. 节点总能量可支持约 100 000 个事件, 图 6、图 7 的实验均是基于 $R = 500$ 的情况.

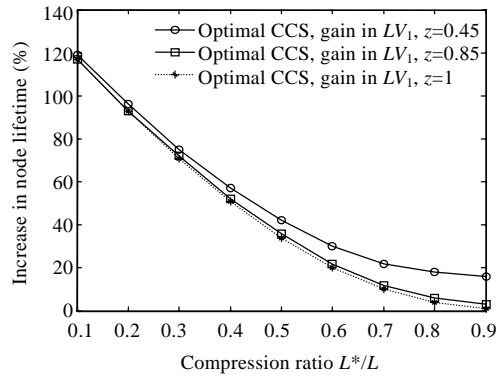


Fig.6 Gain in lifetime if optimal CCS is applied
图 6 最优 CCS 的生命周期增益

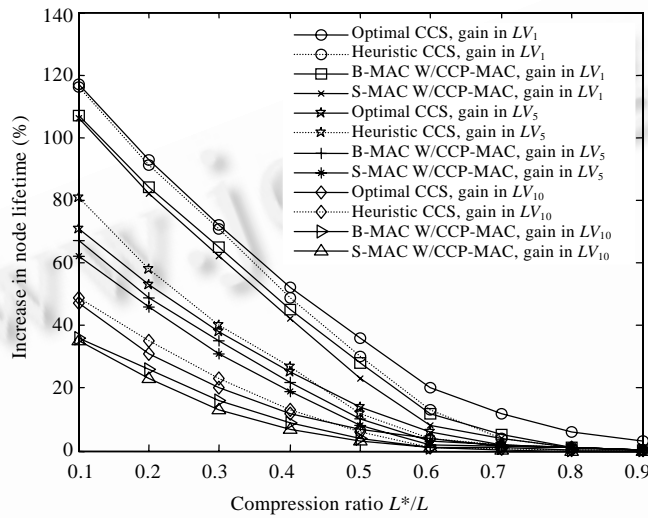


Fig.7 Gain in lifetime when z=0.85
图 7 z=0.85 时的生命周期增益

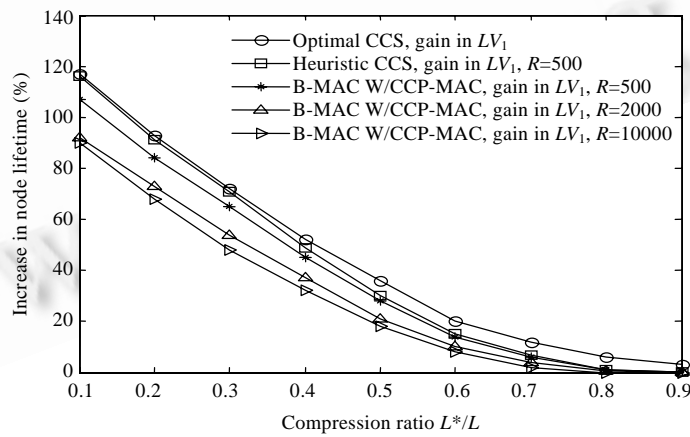


Fig.8 Gain in lifetime vs varying R
图 8 不同 R 值时的生命周期增益

从图 8 可见,启发式算法在保证低复杂度的前提下,其性能仍然保持较好的精确度,与最优 CCS 基本相当. R 越大时,CCP-MAC 协议的增益越小,这是由于节点不能及时了解邻居节点的剩余能量和目前可以通信的邻居节点所致.但尽管如此,当 $R=10\ 000$ 时,相对于不使用 CCP-MAC 协议,其增益仍然相当可观.因此如上文中所述,邻居信息表的更新频率可以很低,其更新能耗基本不影响协议的整体性能.

图 9 显示了当监测事件节点数量 N 取不同值时的增益情况.图 6~图 8 实验中的事件监测节点数均为 $N=20$,图 9 显示随着监测节点数的增加,节点生命周期 LV_1 也随之增加.这是因为 N 增大时,每个节点可以进行选择的邻居节点数目也相应增加,其选择余地增大,从而提高了 CCP-MAC 的压缩性能.且当 N 增加到一定程度时,增益的增长率变缓,如图 9 所示, $N=100$ 比 $N=80$ 时的增益上升幅度有所减缓.

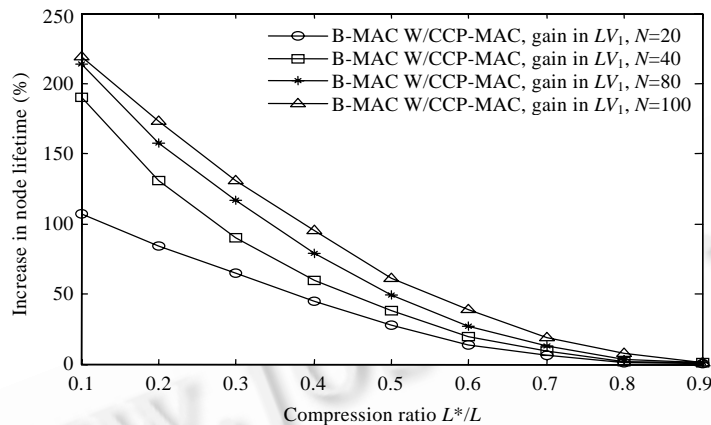


Fig.9 Gain in lifetime vs varying N

图 9 不同 N 值时的生命周期增益

7 结束语

本文基于无线信号的广播本质,利用传统 MAC 协议所忽略的串音数据,提出一种协同数据压缩方法在媒介访问控制层去除数据的空间相关性.本文不仅对一般协同数据压缩问题进行量化分析,建立了线性规划模型,还提出一种近似最优的、更低时间复杂度($O(N^2)$)的启发式节点筛选算法.在此基础上,设计了一种能量有效的、基于协同压缩的 MAC 协议(CCP-MAC),可以分布式地控制节点实现该筛选算法.实验结果表明,该节点筛选算法与 CCP-MAC 协议可以在很大程度上延长节点的生命周期.

References:

- [1] Akyildiz IF, Su W, Sankarasubramaniam Y, Cayirci E. A survey on sensor networks. IEEE Communication Magazine, 2002,40(8): 102-114.
- [2] Ren FY, Huang HN, Lin C. Wireless sensor networks. Journal of Software, 2003,14(7):1282-1291 (in Chinese with English abstract). <http://www.jos.org.cn/1000-9825/14/1282.htm>
- [3] Li JZ, Li JB, Shi SF. Concepts, issues and advance of sensor networks and data management of sensor networks. Journal of Software, 2003,14(10):1717-1727 (in Chinese with English abstract). <http://www.jos.org.cn/1000-9825/14/1717.htm>
- [4] Cui L, Ju HL, Miao Y, Li TP, Liu W, Zhao Z. Overview of wireless sensor networks. Journal of Computer Research and Development, 2005,42(1):163-174 (in Chinese with English abstract).
- [5] Ye W, Heidemann J, Estrin D. Medium access control with coordinated adaptive sleeping for wireless sensor networks. IEEE/ACM Trans. on Networking, 2004,12(3):493-506.
- [6] Dam TV, Langendoen K. An adaptive energy-efficient MAC protocol for wireless sensor networks. In: Akyildiz IF, ed. Proc. of the ACM SenSys 2003. New York: ACM, 2003. 65-72.

- [7] Polastre J, Hill J, Culler D. Versatile low power media access for wireless sensor networks. In: Stankovic JA, ed. Proc. of the ACM SenSys 2004. New York: ACM, 2004. 95–107.
- [8] Zheng T, Radhakrishnan S, Sarangan V. Pmac: An adaptive energy-efficient MAC protocol for wireless sensor networks. In: Bader DA, ed. Proc. of the IEEE IPDPS 2005. Denver: IEEE CS, 2005. 65–72.
- [9] Ye W, Silva F, Heidemann J. Ultra-Low duty cycle MAC with scheduled channel polling. In: Campbell A, ed. Proc. of the ACM SenSys 2006. New York: ACM, 2006. 321–334.
- [10] Vuran MC, Akyildiz IF. Spatial correlation-based collaborative medium access control in wireless sensor networks. IEEE/ACM Trans. on Networking, 2006,14(2):316–329.
- [11] Pattem S, Krishnamachari B. The impact of spatial correlation on routing with compression in wireless sensor networks. In: Ramchandran K, ed. Proc. of the IPSN 2004. New York: ACM, 2004. 28–35.
- [12] Slepian D, Wolf J. Noiseless coding of correlated information sources. IEEE Trans. on Information Theory, 1973,19(4):471–480.
- [13] Pradhan SS, Kusuma J, Ramchandran K. Distributed compression in a dense microsensor network. IEEE Signal Processing Magazine, 2002,19(2):51–60.
- [14] Zhou SW, Lin YP, Zhang JM, Ouyang JC, Lu XG. Compressing spatial and temporal correlated data in wireless sensor networks based on ring topology. Journal of Software, 2007,18(3):679–690 (in Chinese with English abstract). <http://www.jos.org.cn/1000-9825/18/679.htm>
- [15] Bottigligo M, Casetti C, Chiasserini CF, Meo M. Short-Term fairness for TCP flows in 802.11b WLANs. In: Li VOK, ed. Proc. of the IEEE INFOCOM 2004. Piscataway: IEEE Press, 2004. 1383–1392.

附中文参考文献:

- [2] 任丰原,黄海宁,林闯.无线传感器网络.软件学报,2003,14(7):1282–1291. <http://www.jos.org.cn/1000-9825/14/1282.htm>
- [3] 李建中,李金宝,石胜飞.传感器网络及其数据管理的概念、问题与进展.软件学报,2003,14(10):1717–1727. <http://www.jos.org.cn/1000-9825/14/1717.htm>
- [4] 崔莉,鞠海玲,苗勇,李天璞,刘巍,赵泽.无线传感器网络研究进展.计算机研究与发展,2005,42(1):163–174.
- [14] 周四望,林亚平,张建明,欧阳竞成,卢新国.传感器网络中基于环模型的小波数据压缩算法.软件学报,2007,18(3):679–690. <http://www.jos.org.cn/1000-9825/18/679.htm>



胡玉鹏(1981—),男,湖南祁东人,博士,CCF会员,主要研究领域为无线传感器网络中的MAC协议,数据认证,网络数字媒体.



李小龙(1981—),男,博士,主要研究领域为无线传感器网络中的覆盖控制,定位技术.



林亚平(1955—),男,博士,教授,博士生导师,CCF高级会员,主要研究领域为通信网络,机器学习.



周四望(1971—),男,副教授,主要研究领域为无线传感器网络中的数据压缩,信号处理,小波分析.



蒋红艳(1966—),女,博士生,主要研究领域为计算机通信网络.