

一种高信道利用率的分簇传感器网络数据聚合协议

陈磊^{1,2,3}, 赵保华^{1,2,3}

(1. 中国科学技术大学计算机科学与技术系, 230027, 合肥; 2. 网络与交换技术国家重点实验室, 100876, 北京; 3. 中国科学技术大学安徽省计算与通讯软件重点实验室, 230027, 合肥)

摘要: 为了避免分簇无线传感器网络中簇内数据聚合时产生的簇间干扰, 提高簇的信道利用率, 提出了一种基于时分多址的无冲突数据聚合协议(IFDA). IFDA 的帧分配算法采用一种概率调度策略. 通过调度簇头进行广播, 即在每个时隙中进行簇帧号分配, 为相邻簇分配不同的时间帧聚合数据, 与已有的协议的帧分配算法相比可降低广播的次数. 同时, IFDA 通过信道扩充(CE)算法来提高各个簇的信道利用率. CE 算法中采用了一种基于时钟退避的调度策略, 可均衡簇头间的信道利用率. 仿真结果表明, IFDA 协议中簇的平均信道利用率较已有的协议提高了约 90%.

关键词: 无线传感器网络; 簇间干扰; 时分多址; 数据聚合

中图分类号: TP393 **文献标志码:** A **文章编号:** 0253-987X(2009)02-0038-05

A Highly-Channel-Utilized Data Aggregation Protocol for Clustered Sensor Networks

CHEN Lei^{1,2,3}, ZHAO Baohua^{1,2,3}

(1. Department of Computer Science and Technology, University of Science & Technology of China, Hefei 230027, China; 2. State Key Laboratory of Networking and Switching Technology, Beijing 100876, China; 3. Anhui Province Key Laboratory of Software in Computing and Communication, University of Science & Technology of China, Hefei 230027, China)

Abstract: In order to solve the inter-cluster-interference problem during data aggregation in wireless sensor networks and to enhance the cluster channel-usage-radio, a TDMA (time division multiple access)-based data aggregation protocol (IFDA) is proposed. The frame assignment algorithm of the IFDA uses a probability-based schedule strategy to schedule cluster heads for broadcasting so that different frames are assigned among neighboring clusters, and the number of broadcasts can be reduced compared with the existing protocols. Meanwhile, the IFDA uses a channel expansion (CE) algorithm to enhance the cluster channel-usage-radio, while the CE algorithm uses a clock-back-off schedule strategy to balance the channel usage radio of each cluster. Simulation results show that the average cluster channel-usage-radio of the IFDA is about 90% higher than that of the existing protocols.

Keywords: wireless sensor network; inter cluster interference; time division multiple access; data aggregation

大多数分布式无线传感器网络(WSN)分簇算法最后形成的簇的覆盖区域是有重叠的, 为了解决由此产生的簇间传输干扰问题, LEACH^[1]采用直

接序列扩频(DSSS)技术来避免簇间通信干扰. 文献[2]提出了一种结合了时分多址(TDMA)、频分多址(FDMA)的算法, 即簇内按 TDMA 进行调度, 相

邻簇使用不同的频率通信. DSSS 扩频技术或 FDMA 均增加了硬件的成本,不适用于 WSN 大规模布撒的应用场景. 文献[3]提出了一种自适应的时间片分配算法(SRSA),来解决最小化簇间干扰问题,即根据簇头在收集数据过程中的干扰冲突等反馈信息来调换成员节点调度时的时间片序号. 该算法自适应过程的收敛速度与网络的密度、簇间重叠度密切相关,并不能保证完全消除簇间干扰. 文献[4]提出了一种基于两级 TDMA 调度协议(IFCT)来避免簇间冲突,即将时间分成帧,每一帧包含若干时隙,并为相邻簇分配不同的帧号,再将帧的时隙分配给成员节点,从而避免了簇间干扰. IFCT 由于采用了 TDMA 调度,节点平均能耗小于基于竞争的调度,同时还采用一种基于概率簇覆盖的模型^[5]来确定每个簇中实际应工作的节点数,所以随着节点密度的提高,仍能保证较为恒定的通信延迟.

本文在 IFCT 的基础上,提出了一种基于时分多址的无冲突数据聚合协议(IFDA). 与 IFCT 相同的是,IFDA 也采用了基于簇的两级调度,但是帧分配(FA)算法较之 IFCT 降低了广播次数,又由于利用了信道扩充(CE)算法,所以它避免了 IFCT 中的信道浪费问题,提高了信道利用率.

1 IFDA 协议

1.1 网络模型

假设 N 个节点随机、静态部署在 $M \times M$ 的二维平面上,网络中存在一个 sink 节点,整个网络以统一的簇半径 R_c 分簇. 簇头节点在接收簇内节点感知到的数据之后进行聚合,并将聚合后的数据以簇间多跳的方式传递给 sink 节点. 又设网络具有以下性质:①所有节点同构,并且能够进行数据聚合;②网络是静态部署的,无需人为维护;③节点的发射功率可调节;④所有节点的时钟同步. 前 3 项为一般 WSN 的典型设置(见文献[1,5]),而对于 TDMA 协议而言,节点的时钟同步是必须的,它可以通过全球定位系统(GPS)的时钟同步方法或者其他的时钟同步协议来实现.

1.2 协议描述

IFDA 协议继承了 IFCT 的帧结构^[4],并对 IFCT 的簇间调度阶段进行了修改. 在 IFDA 的簇间调度阶段,将依次运行下面的 FA、CE 算法.

(1)FA 算法. 该算法负责为每个簇分配一个初始的帧号,保证相邻簇之间的帧号不相同,其与文献[6]算法相同的是,假设在建树(tree building)阶段

每个簇头已获得其邻居簇头的列表,则广播一次邻居簇头节点集合,使得所有簇头都获得二跳邻居簇头 ID 的集合.

文献[6]提出的 NCR (Neighborhood-Aware Contention Resolution) 算法是,利用节点 ID 的唯一性和伪随机序列的可预测性,通过伪随机数生成器(RAND)来调度节点,使之在相应的时隙中进行无冲突广播. 定义簇头 i 的一跳邻居簇头的集合为 I_i ,二跳邻居集合为 M_i ,则 NCR 满足定理 1.

定理 1 假设 RAND 将 x 映射到大小为 M 的区间 Ω 上,则对于足够大的 M ,NCR 保证任意节点 i 在任意时隙 t 以 $1/(|M_i|+1)$ 概率被调度.

引理 1 对于任意的正整数 K ,式(1)成立

$$\lim_{n \rightarrow \infty} \frac{1}{n^{K+1}} \sum_{i=1}^n i^K = \frac{1}{K+1} \quad (1)$$

证明 用数学归纳法,显然当 $K=1,2$ 时,式(1)成立. 假设存在某个 k ,当 $K \leq k-1$ 时,式(1)均成立. 若 $K=k$,则有

$$\begin{aligned} 1 &= \lim_{n \rightarrow \infty} \frac{(n+1)^{k+1} - 1}{n^{k+1}} = \lim_{n \rightarrow \infty} \frac{1}{n^{k+1}} \sum_{i=1}^n ((i+1)^{k+1} - i^{k+1}) \\ &= \lim_{n \rightarrow \infty} \frac{1}{n^{k+1}} \sum_{i=1}^n \left((k+1)i^k + \sum_{j=0}^{k-1} \binom{k+1}{j+2} i^j \right) = \\ &= \lim_{n \rightarrow \infty} \frac{(k+1) \sum_{i=1}^n i^k}{n^{k+1}} + \lim_{n \rightarrow \infty} \frac{1}{n^{k+1}} \sum_{j=0}^{k-1} \binom{k+1}{j+2} \sum_{i=1}^n i^j \end{aligned} \quad (2)$$

当 $j \leq k-1$ 时,有

$$\begin{aligned} \lim_{n \rightarrow \infty} \frac{1}{n^{k+1}} \sum_{j=0}^{k-1} n^{j+1} \binom{k+1}{j+2} \left(\frac{1}{n^{j+1}} \sum_{i=1}^n i^j \right) &= \\ \lim_{n \rightarrow \infty} \frac{1}{n^{k+1}} \sum_{j=0}^{k-1} n^{j+1} \binom{k+1}{j+2} \frac{1}{j+1} &= 0 \end{aligned} \quad (3)$$

将式(3)代入式(2),得到 $\lim_{n \rightarrow \infty} \frac{1}{n^{k+1}} (k+1) \sum_{i=1}^n i^k = 1$,因此 $\lim_{n \rightarrow \infty} \frac{1}{n^{k+1}} \sum_{i=1}^n i^k = \frac{1}{k+1}$. 证毕.

对 Ω 进行排序,令 $K = |M_i| + 1$,节点 i 取得 Ω 上第 w 个值并且其他 $K-1$ 的节点取值均小于 w ,其概率为 $((w-1)/M)^{K-1}$,因此有

$$\begin{aligned} P(K, M) &= \sum_{w=1}^M \frac{1}{M} \left(\frac{w-1}{M} \right)^{K-1} = \frac{1}{M^K} \left(\sum_{w=1}^M w^{K-1} - \right. \\ &\quad \left. M^{K-1} \right) = \frac{1}{M^K} \sum_{w=1}^M w^{K-1} - \frac{1}{M} \end{aligned}$$

根据引理 1,当 M 足够大时, $P(K, M) \approx 1/K$. 证毕.

FA算法的描述如图1所示,其中 $p_i^t = \text{true}$ 表示节点 i 在当前时隙 t 被调度, f_i^t 为簇头 i 的所有邻居簇头选择的帧号集合, $\text{frame}\#_i$ 表示当前与 f_i^t 中所有帧号都不相同的最小的帧号, C_i 表示当前尚未广播自身帧号的邻居簇头的个数. 在每个时隙 $t(t=1, 2, 3, \dots)$, 当簇头 i 在当前时隙可以被调度时, 簇头 i 将广播附带自身帧号的帧消息(图1中第3~第5行); 在收到其他簇头的广播时, 簇头 i 更新当前邻居簇头的帧号集合 f_i^t , 如果簇头 i 尚未获得帧号, 则根据当前的 f_i^t 来更新 $\text{frame}\#_i$, 以保证自身选择的帧号与邻居簇头不相同.

```

1. Done=false;  $f_i^t=\emptyset$ ;  $\text{frame}\#_i=1$ ;  $C_i=|I_i|$ ;
2. REPEAT at every slot  $t$ :
3.   IF  $p_i^t=\text{true}$  and Done=false
4.     FRAME. frame#= $\text{frame}\#_i$ ;
5.     BROADCAST(FRAME);
6.     Done=true;
7.   IF received FRAME;
8.      $f_i^t=f_i^t \cup \{\text{FRAME. frame}\#_i\}$ ;
9.      $C_i=C_i-1$ ;
10.    IF Done=false
11.      update  $\text{frame}\#_i$ ;
12. UNTIL  $C_i=0$ .
```

图1 FA算法描述

(2)CE算法. 在FA算法完成之后, 可以通过汇聚和扩散方式使得每个簇头获得网络中最大帧号 n_f . 本文采用文献[4]方法, 即基于仿真模拟的方式确定 $n_f=6$. 由于 IFCT 仅为每个簇头分配一个帧号, 当相邻的簇仅占据帧号 1、2、3 时, 将造成帧号 4、5、6 浪费, 因此我们引入了 CE 算法.

CE算法的描述如图2所示, 其中 S_f 表示所有 n_f 的帧号集合, 簇头 i 所选择的帧号集合为 F_i , ϵ 和 K 是事先设定好的常数, 初始时 $F_i=\{\text{frame}\#_i\}$. CE算法对 F_i 进行扩充, 在算法结束时, 下面2个条件成立: ①若 i, j 相邻, 则 $F_i \cap F_j = \emptyset$; ② $F_i \cup f_i^t = S_f$. 条件①保证了簇间不干扰, 条件②是为了最大化信道利用率.

由于簇头 i 在扩充自身帧号集合的同时可能会减小其邻居簇头可扩充帧号的集合, 因此可以设定每个节点的广播退避时钟 TimeOut_i 在任何时刻都与当前的 $|S_f - F_i \cup f_i^t|$ 成正比, 保证在各时隙具有较小可扩充帧号集合的簇头优先广播. ϵ 过小则无法突出时钟退避(TB)的效果, ϵ 过大将增加算法运行的时间, 所以本文取 $\epsilon=5$. 簇头 i 在任意时隙以 $1/(|M_i|+1)$ 概率被调度, 则节点 i 相邻2次被调度

```

1. Done=false;  $S_f=\{1, 2, \dots, n_f\}$ ;  $F_i=\{\text{frame}\#_i\}$ ;  $C_i=|I_i|$ ;
    $\text{TimeOut}_i=\epsilon|S_f - F_i \cup f_i^t|$ ;
2. REPEAT at every slot  $t$ :
3.   IF  $F_i \cup f_i^t = S_f$ 
4.     Done=true;
5.   ELSE IF  $p_i^t=\text{true}$  AND  $\text{timeOut}_i - t \leq K(|M_i|+1)$ 
6.      $F_i=S_f - f_i^t$ ;
7.     FRAME. frame#set= $F_i$ ;
8.     BROADCAST(FRAME);
9.     Done=true;
10.  ELSE IF received FRAME;
11.     $f_i^t=f_i^t \cup \text{FRAME. frame}\#_i$ ;
12.     $\text{TimeOut}_i=\epsilon|S_f - F_i \cup f_i^t|$ ;
13. UNTIL Done=true.
```

图2 CE算法描述

的时隙差的期望值为

$$E(\Delta t_i) = \sum_{t=1}^{\infty} t \frac{1}{|M_i|+1} \left(1 - \frac{1}{|M_i|+1}\right)^{t-1} = \frac{1}{|M_i|+1}$$

如果当前的时隙 t 有 $p_i^t = \text{true}$, 则通过以下规则可判断簇头 i 是否超时, 即: 如果 $\text{TimeOut}_i - t \leq 0$, 认定超时; 否则只要 $\text{TimeOut}_i - t$ 相对于下一次 $p_i^t = \text{true}$ 的平均时隙差 $|M_i|+1$ 为一个较小的值, 则认定 i 超时. 本文设定 $K=0.2$.

1.3 IFDA 分析

本文主要对 IFDA 的 FA、CE 算法的时间复杂度和消息复杂度进行了分析, 并与 IFCT 进行对比.

(1)消息复杂度分析. 由于 NCR 算法需要每个簇头节点知道其二跳邻居列表, 因此就需要每个簇头广播一次其一跳邻居的 ID 集合 D , 该算法的复杂度为 $O(D)$. FA 算法需要每个簇头广播一次消息, CE 算法需要每个簇头最多广播 1 条消息. 因此, IFDA 在簇间调度阶段需要每个簇头进行 $O(1)$ 次广播, 该算法复杂度为 $O(D)$. 由于 IFCT 的 FA 算法运行轮数为 $O(D)$, 并且不能保证每轮相邻簇头广播的保留消息的帧号不相同, 因此簇头最多需进行 $O(D)$ 次广播, 所以算法复杂度为 $O(D)$.

(2)算法时间复杂度分析. 由定理1可知, 在 IFDA 的 FA 算法中, 簇头 i 进行一次广播平均要等待的时隙数为 $|M_i|+1$, 上限为 $D(D-1)+1$, 因此 FA 算法的时间复杂度为 $O(D^2)$. 在 CE 算法中, 簇头 i 在广播前要进行 $|N_f - F_i \cup f_i^t|$ 个时隙的时钟退避, 上限为 $O(n_f)$. 由于在 FA 算法中簇头选择未被邻居选择的最小帧号, 因此 $n_f \leq D$, 故 CE 算法中每个簇头的时钟退避上限为 $O(D)$, 则该阶段的时间复杂度为 $\max(O(D^2), O(D))$, 即为 $O(D^2)$. 因

此,IFDA 在簇间调度阶段的总时间复杂度为 $O(D^2)$. IFCT 协议中的 FA 算法的运行时间复杂度为 $O(Dt)$,其中 t 要保证足够长,以便接收所有邻居簇头发送的数据,即 $t=O(D)$,因此 IFCT 的 FA 算法的时间复杂度同为 $O(D^2)$,这与 IFDA 相同.

2 仿真

2.1 RAND 函数的设计

本文仿真借助了一些已有的伪随机序列生成器来设计 RAND 函数. 计算机上常用的生成伪随机序列的线性求余递推公式为

$$\left. \begin{aligned} X_n &= ((Y_n \times 214\ 013L + 2\ 531\ 011L) \gg 16) \& 0x7fff \\ Y_n &= Y_{n-1} \times 214\ 013L + 2\ 531\ 011L \\ n &= 1, 2, 3, \dots \end{aligned} \right\} (4)$$

其中 Y_0 为 4 B 的种子值.

由于线性同余法具有随机性好、周期长、易于计算机实现和运算速度快等特点,所以本文采用式(4)生成伪随机序列. 假设网络中节点的 D 的取值范围为 $[0, V]$, 则 NCR 算法在时隙 t 为节点计算所得权值为

$$\text{RAND}(D \oplus t) \oplus D = 2^{\lfloor \lg V \rfloor + 1} X_{(t-1)V+D} + D \quad (5)$$

2.2 仿真环境设置

本文在 Visual C++ 平台上实现了仿真程序, 仿真环境假设 100 个节点均匀分布在 $300\text{ m} \times 300\text{ m}$ 的平面区域中, 簇半径 R_c 为 $50 \sim 100\text{ m}$ (每次递增 10 m). 利用 DEEG[2] 进行分簇并构建簇间路由树, V 取 127, 将其中的 1 到 100 作为 ID 赋予所有的节点, Y_0 取 200. 在每一次分簇前, 又设所有节点的能量充足且重新均匀分布. 仿真结果是在每个簇半径下进行 100 次分簇的相对平均值.

2.3 仿真结果

图 3 给出了 $R_c = 60\text{ m}$ 时, 某一次的分簇结果 (总共 16 个簇头节点) 在运行 NCR 算法后簇头连续被调度 30 次平均需要等待的时隙数. 在式(5)中, RAND 函数的映射区间 Ω 的大小 M 为 $0x7fff$, 其倒数约为 $0.000\ 030\ 52$, 可以忽略不计. 由图 3 可以看出, 期望和仿真曲线拟合度较好.

图 4 给出了 IFDA 与 IFCT 簇的信道利用率的比较. 定义簇头 i 的信道利用率为 $|F_i|/n_i$. IFCT 规定 n_i 为 6, 因此 IFCT 中每个簇进行数据聚合的信道利用率为 $1/6$. IFDA 中的 CE 算法对各簇头的帧号集合进行了扩充, 所以簇的平均信道利用率比

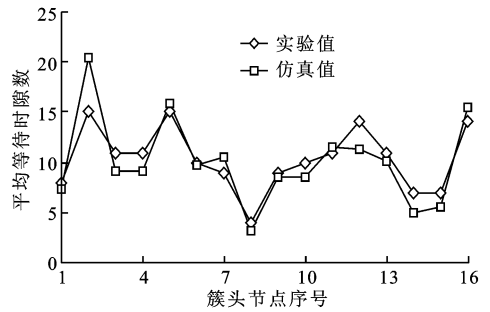


图3 簇头节点获得广播权平均等待的时隙数

IFCT 约提高了 90%, 网络中最高的簇信道利用率比 IFCT 约提高了 274%.

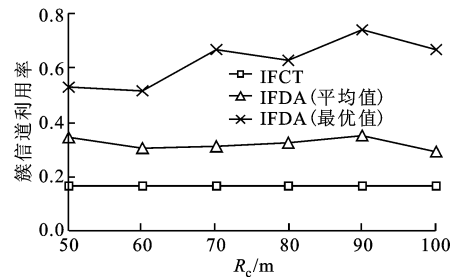
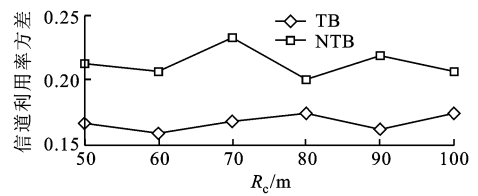
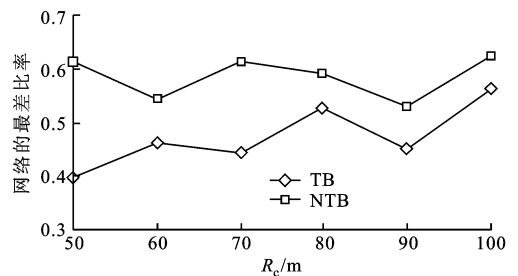


图4 IFCT与IFDA的信道利用率比较

图 5 给出了 CE 算法中运用 TB 进行广播所带来的优化效果. 由定理 1 可知, 具有较小二跳邻居数的簇头将优先被调度, 但该簇头一般也具有较小的一跳邻居簇头数. 初始时, 具有较少邻居数的簇头将有可能具有较大的可扩充帧号集合, 因此对于不采用时钟退避 (NTB) 的情形, CE 算法将使得具有较



(a) 簇头信道利用率方差



(b) 网络的最差比率

图5 CE算法采用时钟退避产生的优化效果

大可扩充帧号集合的簇头提前广播,导致簇头信道利用率不均衡.图 5a 给出了相同分簇的结果在 TB 与 NTB 的情形下分别运行 CE 算法后簇信道利用率的方差比较.定义帧号个数为 1 的簇头占所有簇头数的比率为网络的最差比率,图 5b 显示了在 TB、NTB 情形下网络的最差比率.可以看出,在 NTB 的情形下,网络有更多的簇的帧号集合无法得到扩充.

3 结 论

本文提出了一种单一信道下基于分簇路由的 WSN 无冲突数据聚合协议.与 IFCT 相同,IFDA 通过基于两级 TDMA 调度来避免数据聚合时的簇间干扰. IFDA 还使用了基于 TDMA 的 NCR 算法,即在每个时隙中进行簇帧号分配,较之 IFCT 降低了广播的次数,同时通过一种基于时钟退避策略均衡了各簇头的信道利用率,仿真实验证实了该时钟退避策略的有效性.实验表明,较之 IFCT,IFDA 具有更高的信道利用率.

参考文献:

[1] HEINZELMAN W, CHANDRAKASAN A, BALAKRISHNAN H. An application-specific protocol architecture for wireless microsensor networks [J]. IEEE Transactions on Wireless Communications, 2002, 1(4): 660-670.

[2] WAHARTE S, BOUTABA R. Performance comparison of distributed frequency assignment algorithms for wireless sensor networks [C]//Proceedings of the IFIP International Conference on Network Control. Los Alamitos, CA, USA: IEEE Computer Society, 2004: 217-228.

[3] WU Tao, BISWAS S. Minimizing inter-cluster interference by self-reorganizing MAC allocation in sensor networks [J]. Wireless Networks, 2007, 13(5): 691-703.

[4] GONG Haigang, LIU Ming, WANG Xiaomin, et al. An interference free cluster-based TDMA protocol for wireless sensor networks [C]//Wireless Algorithms, Systems, and Applications. Berlin, Germany: Springer, 2006: 217-267.

[5] 刘明, 龚海刚, 毛莺池, 等. 高效节能的传感器网络数据收集和聚合协议 [J]. 软件学报, 2005, 16(12): 2106-2116.

LIU Ming, GONG Haigang, MAO Yinchu, et al. A distributed energy-efficient data gathering and aggregation protocol for wireless sensor networks [J]. Journal of Software, 2005, 16(12): 2106-2116.

[6] BAO Lichun, GARCIA-LUNA-ACEVES J. A new approach to channel access scheduling for ad hoc networks [C]//Proceedings of the 7th Annual International Conference on Mobile Computing and Networking. New York, USA: ACM, 2001: 210-221.

(编辑 苗凌)