

文章编号: 1001-0920(2012)05-0761-07

基于无线传感器网络汇聚传输的两阶段实时调度算法

张晓玲^{1,2}, 梁 炜¹, 于海斌¹

(中国科学院沈阳自动化研究所 工业信息学重点实验室, 沈阳 110016)

摘要: 多对一的汇聚传输是无线传感器网络,特别是工业无线传感器网络中的一个典型的数据传输模式. 针对具有星形和网状两层拓扑结构的工业无线传感器网络汇聚传输的实时性要求,分析簇-线型路由下完成汇聚传输所需时隙数和信道数的下限值,并基于理论值提出一种基于紧凑搜索的两阶段实时调度算法. 仿真结果表明,存在最优的两阶段实时调度算法,且相比集中式调度算法,时间开销降低了6~8倍,报文开销降低了近2倍.

关键词: 无线传感器网络; 汇聚传输; 实时调度; 两阶段算法

中图分类号: TP393

文献标识码: A

Two-stage and real-time scheduling algorithm for convergecast in wireless sensor networks

ZHANG Xiao-ling^{1,2}, LIANG Wei¹, YU Hai-bin¹

(Key Laboratory of Industrial Informatics, Shenyang Institute of Automation, Chinese Academy of Sciences, Shenyang 110016, China. Correspondent: ZHANG Xiao-ling, E-mail: zhangxiaoling@sia.cn)

Abstract: The many-to-one convergecast is a kind of typical transmissions for wireless sensor networks, specially for wireless industrial sensor networks. This paper establishes the lower bounds on the number of timeslots and channels used for real-time convergecast in hybrid star and mesh wireless industrial sensor networks. And a two-stage and real-time scheduling algorithm based on compact search is proposed subsequently. Simulation results show that the two-stage and real-time scheduling algorithm based on compact search is optimal and the time overhead and packet overhead can be reduced by 6 to 8 times and nearly 2 times, respectively.

Key words: wireless sensor networks; convergecast; real-time scheduling; two-stage algorithm

1 引言

无线传感器网络技术与特定应用背景相关的,针对不同的应用领域,其实现技术各异^[1]. 在无线传感器网络的多种应用场景中,各传感器节点需要在短时间内将采集到的数据传输至汇聚节点,例如:网络状态检测应用中节点对自身剩余能量的扫描;火灾现场探测应用中节点对采集温度的汇报;过程工业中对温度、压力、流量等参数的汇报;环境检测应用中对潮流潮位的监控和水文监测等^[2]. 但由于网络中往往只有一个汇聚节点(一般由网关担当),使在上述场景中网络的数据传输呈现“多对一”的传输模式. 这种“多对一”的传输模式称为“汇聚传输”.

汇聚传输由于多对一的数据收集方式导致的冲突尤为严重. 当大量的数据涌向汇聚节点时,距离汇

聚节点较近的几个传感器节点容易成为通信瓶颈并导致拥塞. 现有基于竞争的通信算法,如载波侦听多访问(CSMA)等,其引入的退避重传机制和错误阻塞现象无法保证数据传输的实时性和可靠性^[2],而基于时分多路访问(TDMA)的传输调度算法因其可有效地避免冲突而成为汇聚传输的首选. 目前,大部分基于TDMA的汇聚传输调度算法主要从提高网络吞吐量和能量效率的角度出发,对数据传输的实时性考虑相对较少. 近年来,部分学者开展了对于汇聚传输所需时隙数和信道数下限值的研究,并据此设计了优化算法. Gandham等人^[3]从理论上证明了对于节点数量为 N 的网络,通过传输调度最多只需要 $3N$ 个时隙就能完成一次全网数据的汇聚传输;孙利民等人^[2]提出了一种可在 $1.6N$ 和 $1.8N$ 个时隙数中完成

收稿日期: 2010-10-25; 修回日期: 2011-05-24.

基金项目: 国家自然科学基金项目(61174026, 61100159, 61172145); 国家863计划项目(2011AA040103); 国家973计划前期研究专项课题(2010CB334705).

作者简介: 张晓玲(1982-),女,博士,从事工业无线传感器网络标准、实时可靠的传输调度算法的研究; 于海斌(1964-),男,研究员,博士生导师,从事无线通信、分布控制系统等研究.

一次全网数据汇聚传输的调度算法; Soldti^[4]面向无线 HART 网络, 分析了节点在单报文缓存和多报文缓存情况下完成一次全网数据汇聚传输所需的时隙数和信道数的下限值. 然而, 上述研究均采用集中式的方法求解汇聚传输的调度策略^[4-5], 需要汇聚节点掌握网络全局信息, 在网络规模较大时计算复杂度很高. 一些针对汇聚传输的分布式传输调度策略^[2-3]由各个节点自主决定传输时隙的分配, 但每个节点需要掌握网络拓扑中的分支总数、各分支长度、节点和各分支相互间的传输干扰关系等信息, 在计算复杂度上已经接近于集中式调度.

WIA-PA 标准^[6]是中国工业无线联盟针对过程自动化的迫切需求而率先制定的系统结构和通信规范, 定义网络拓扑为星形和网状混合的两层结构. 在实际应用中, 需要针对两层拓扑设计相应的两阶段调度算法. 目前的研究比较初步, 如文献 [7] 等, 缺乏完整的理论体系和算法支撑. 本文针对上述问题, 根据理论分析和两阶段实时调度算法设计了两个角度, 并对基于簇-线型路由的汇聚传输进行了系统研究, 致力于寻找保证汇聚传输实时性的低复杂度调度算法.

2 系统模型

2.1 网络模型

WIA-PA 标准定义网络拓扑为星形和网状混合的拓扑结构. 其中: 网关设备和路由设备构成网状网络, 路由设备及现场设备或手持设备构成星形网络(又称簇). 网络中仅有现场设备产生数据且以一定的数据更新率更新数据. 现场设备将采集到的数据转交给路由设备, 由路由设备负责转发, 在一定程度上降低了现场设备的能耗. 采用星形和网状混合的拓扑结构, 可使网络灵活且易于管理.

采用图论的方法将网络构建为一个无向图 $G = (V, E)$. 其中: E 为网络中所有链路的集合; $V = \{GW, R, F\}$ 表示网络中的所有设备. 这里: GW 表示网关设备, R 表示路由设备的集合, F 表示现场设备的集合. 出于硬件和节能的考虑, 假设现场设备仅具有单报文缓存能力. 因为路由设备需要转发簇内所有现场设备的报文, 所以假设具有多报文缓存能力. 假设网络中的各设备通过唯一的标识符 ID 进行标识, 且通信范围相等. 网关设备在执行调度算法之前已知各设备的位置及其两跳范围内的邻居设备. 此假设可通过如下方式实现: 网络中的设备汇报各自的邻居信息; 为了避免冲突, 等待汇报的时间与设备 ID 号成正比. 假设全网时间同步, 且设备在每个时隙可完成一次数据交互(数据+ACK). 在同一个时隙内, 并行传输的设备要求使用不同的信道. 网络中的每个设备均安装半双工模式的收发器, 因此在设计调度算法时, 需要满足设备不能同时收发数据的要求.

2.2 路由模型

在汇聚传输开始前, 每个现场设备最多产生一个数据, 等待通过路由设备转发给网关设备. 数据传输过程沿着路由拓扑 $T = (V, E^*)$ 传向网关设备 GW , 其中 $E^* \subseteq E$. 本文的研究建立在如图 1 所示的簇-线型路由拓扑.

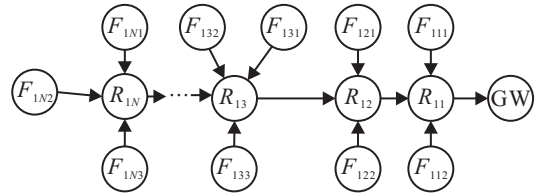


图 1 簇-线型路由

不失一般性, 网关设备 GW 位于最右侧. 网络中的 N 个路由设备 $R_{11} \sim R_{1N}$ 以线型方式连接, 且从右向左排列. 假设以 R_{1i} 为簇首的簇中, f_{1i} 表示现场设备(即簇成员)的数量; F_{1ij} 表示第 j 个现场设备, 且 C_{1ij} 表示 F_{1ij} 的数据更新率. 其中, $i \in [1, N]$, 且 $j \in [0, f_{1i}]$.

2.3 超帧模型

对于 WIA-PA 星形和网状混合的两层网络拓扑结构, 汇聚传输的通信组织方式基于 IEEE STD 802.15.4-2006 超帧, 并对其进行了扩展. 扩展的超帧将 IEEE STD 802.15.4-2006 超帧非活动期的时隙用于簇内通信、簇间通信以及休眠, 如图 2 所示. 本文研究的重点在于分析完成簇内汇聚传输和簇间汇聚传输所需时隙数和信道数的下限值, 并设计完成簇内汇聚传输和簇间汇聚传输的两阶段实时调度算法.

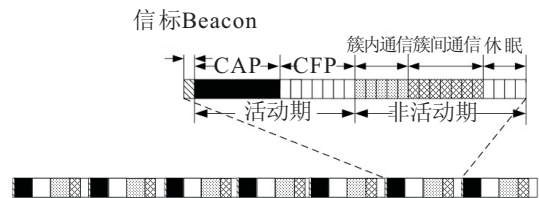


图 2 超帧结构

3 两阶段实时调度算法

首先研究单个簇内现场设备的数据更新率部分相同情况下, 完成簇内和簇间汇聚传输所需时隙数和信道数的下限值. 其次, 以理论分析的下限值为依据, 设计完成簇内和簇间汇聚传输的两阶段实时调度算法. 如果要求单个簇内现场设备的数据更新率部分相同, 则对于以 R_{1i} ($i \in [1, N]$) 为簇首的任意一个簇, 需要满足

$$C_{1ij} = C_{1ik}, \exists j, k \in [1, f_{1i}], \text{ 且 } j \neq k. \quad (1)$$

3.1 簇内汇聚传输时隙数和信道数下限值分析

定理 1 簇内汇聚传输所需时隙数下限值. 以 R_{1i} ($i \in [1, N]$) 为簇首的簇中, 假设簇内数据更新

率为 $C_{1i1}, C_{1i2}, \dots, C_{1iQ_i}$ ($Q_i \leq f_{1i}$ 为簇内不同数据更新率的种类和) 的现场设备的数量分别为 $N_{1i1}, N_{1i2}, \dots, N_{1iQ_i}$, 且满足 $C_{1i1} \leq C_{1i2} \leq \dots \leq C_{1iQ_i}$ 以及 $\sum_{j=1}^{Q_i} N_{1ij} = f_{1i}$. 则该簇完成簇内汇聚传输所需时隙

数的下限值为 $\sum_{j=1}^{Q_i} y_{1ij}, i \in [1, N], j \in [1, Q_i]$. 其中

$$y_{1i1} = \left\lceil \frac{N_{1i1} \times C_{1i1}}{C_{1i1}} \right\rceil = \lceil N_{1i1} \rceil. \quad (2)$$

$$y_{1i2} = \left\lceil \frac{N_{1i2} \times C_{1i1}}{C_{1i2}} \right\rceil. \quad (3)$$

$$y_{1ik} = \begin{cases} 0, & C_{1i1}(N_{1i(k-1)}C_{1ik} + N_{1ik}C_{1i(k-1)}) / \\ & (C_{1i(k-1)}C_{1ik}) < y_{1i(k-1)}; \\ \left[\left(N_{1ik} - \left(y_{1i(k-1)} \frac{C_{1i(k-1)}}{C_{1i1}} - \right. \right. \right. \\ \left. \left. \left. N_{1i(k-1)} \frac{C_{1ik}}{C_{1i(k-1)}} \right) / \left(C_{1ik} / C_{1i1} \right) \right], & \text{else.} \end{cases} \quad (4)$$

其中 $k = 3, 4, \dots, Q_i$.

证明 时隙分配原则为: 在时隙数不增加的前提下, 搜索未被占用的时隙; 如果不存在未被占用时隙, 则时隙数增1. 以 C_{1iQ_i}/C_{1i1} 个超帧周期作为分配周期.

最小数据更新率 C_{1i1} 所对应的 N_{1i1} 个现场设备需要的时隙数为

$$y_{1i1} = \left\lceil \frac{N_{1i1} \times C_{1i1}}{C_{1i1}} \right\rceil = \lceil N_{1i1} \rceil.$$

数据更新率为 C_{1i2} 所对应的 N_{1i2} 个现场设备, 需要从前 C_{1i2}/C_{1i1} 个超帧周期簇内通信段的第2个时隙开始搜索未被占用的时隙. 需要增加的时隙数为

$$y_{1i2} = \left\lceil \frac{N_{1i2} \times C_{1i1}}{C_{1i2}} \right\rceil.$$

数据更新率为 C_{1ik} ($k = 3, 4, \dots, Q_i$) 所对应的 N_{1ik} 个现场设备需要从前 C_{1ik}/C_{1i1} 个超帧周期簇内通信段的第2个时隙开始搜索未被占用的时隙. 前 C_{1ik}/C_{1i1} 个超帧周期簇内通信段的前 $\sum_{j=1}^{k-1} y_{1ij}$ 号时隙中空闲时隙的数量为

$$\left(y_{1i(k-1)} \times \frac{C_{1i(k-1)}}{C_{1i1}} - N_{1i(k-1)} \right) \times \frac{C_{1ik}}{C_{1i(k-1)}}.$$

需要增加的时隙数为

$$y_{1ik} =$$

$$\begin{cases} 0, & \frac{C_{1i1}}{C_{1i(k-1)}C_{1ik}}(N_{1i(k-1)}C_{1ik} + \\ & N_{1ik}C_{1i(k-1)}) < y_{1i(k-1)}; \\ \left[\left[\left(N_{1ik} - \left(y_{1i(k-1)} \times \frac{C_{1i(k-1)}}{C_{1i1}} - \right. \right. \right. \\ \left. \left. \left. N_{1i(k-1)} \right) \times \frac{C_{1ik}}{C_{1i(k-1)}} \right] \times C_{1i1} \right] / C_{1ik}, & \text{else.} \end{cases}$$

完成簇内汇聚传输所需时隙数为各部分时隙增

加数的累加, 即 $\sum_{j=1}^{Q_i} y_{1ij}$. \square

因为簇内通信段为每个簇成员分配一个时隙, 所以不存在簇内数据传输冲突的情况, 仅需考虑与其他簇之间避免干扰即可. 所需信道数取决于实际的要求和具体算法.

3.2 簇间汇聚传输时隙数和信道数下限值分析

完成簇间汇聚传输所需的时隙数, 取决于距离网关设备 GW 最近的一个路由设备转发其上所有数据所需的时隙数.

定理 2 簇间汇聚传输所需时隙数下限值. 对于具有簇-线型路由的网络, 如果单个簇内现场设备的数据更新率部分相同, 则完成簇间汇聚传输所需时隙数的下限值为

$$L_s = 2 \sum_{i=2}^N \sum_{j=1}^{Q_i} y_{1ij} + \sum_{j=1}^{Q_1} y_{11j}.$$

其中: N 为网络中路由设备的数量; Q_i 为以 R_{1i} ($i \in [1, N]$) 为簇首的簇内不同数据更新率的种类和.

证明 利用 L_s 表示完成簇间汇聚传输所需时隙数的下限值, g_{1i} 表示一个超帧周期内 R_{1i} ($i \in [1, N]$) 上需要转发的报文数量.

R_{11} 负责完成两项任务: 1) 转发(接收和发送)来自其他路由设备的数据; 2) 将自身簇内产生的数据发送给网关设备. 由定理1可知, 每个簇完成簇内通信所需时隙数的下限值为 $\sum_{j=1}^{Q_i} y_{1ij}$, 且满足式(2)~(4).

因此, 每个超帧周期路由设备上的报文数至少为 $\sum_{j=1}^{Q_i} y_{1ij}$, 即满足 $g_{1i} \geq \sum_{j=1}^{Q_i} y_{1ij}$. R_{11} 转发 R_{12} 至 R_{1N}

的数据至少需要 $2 \sum_{i=2}^N g_{1i}$ 个时隙; R_{11} 将自身簇内的数据发送给 GW 至少需要 g_{11} 个时隙. 因此, 完成簇间通信所需的最少时隙数为两部分时隙数的累加, 即

$$L_s = 2 \sum_{i=2}^N g_{1i} + g_{11} = 2 \sum_{i=2}^N \sum_{j=1}^{Q_i} y_{1ij} + \sum_{j=1}^{Q_1} y_{11j}. \quad \square$$

定理 3 簇间汇聚传输所需信道数下限值. 对于具有簇-线型路由的网络, 给定可以在

$$L_s = 2 \sum_{i=2}^N \sum_{j=1}^{Q_i} y_{1ij} + \sum_{j=1}^{Q_1} y_{11j}$$

个时隙内完成的调度 S . 如果单个簇内的现场设备的数据更新率各不相同, 则完成簇间汇聚传输所需信道数的下限值为

$$\left\lceil \frac{L_s + 1}{2} - \sqrt{\left(\frac{L_s + 1}{2} \right)^2 - \sum_{i=1}^N \left[i \sum_{j=1}^{Q_i} y_{1ij} \right]} \right\rceil.$$

其中: N 为网络中路由设备的数量, Q_i 为以 R_{1i} ($i \in$

$[1, N]$) 为簇首的簇内不同数据更新率的种类和。

证明 利用 C_s 表示网络中可利用的最大信道数。PT_{max}(t) 表示在时隙 $t \in [1, L_s]$ 内可并行传输的最大设备数。为了保证汇聚传输可在 L_s 个时隙内完成, 需要满足下列条件:

1) 在第 L_s 个时隙, 网络中仅剩 R_{11} 上的最后一个数据, 且 R_{11} 将其发送给 GW。因此, 在第 L_s 个时隙仅允许一个设备传输。

2) 在第 $L_s - 1$ 个时隙, 可能有以下两种传输情况: ① R_{12} 将其上的最后一个数据发送给 R_{11} , 再由 R_{11} 在第 L_s 个时隙将该数据发送给 GW; ② 在第 $L_s - 1$ 个时隙, 网络中仅剩 R_{11} 上的两个数据, 利用第 $L_s - 1$ 个时隙将其上的一个数据发送给 GW, 再在第 L_s 个时隙将最后一个数据发送给 GW。因此, 在第 $L_s - 1$ 个时隙仅允许一个设备传输。

3) 以此类推, 在时隙 t 最多允许 $\left\lfloor \frac{L_s - t + 1}{2} \right\rfloor$ 个设备并行传输。

综合考虑上述条件以及网络中的最大信道数限制, 在时隙 t 允许并行传输的设备数为

$$\text{PT}_{\max}(t) = \begin{cases} C_s, & t \in [1, L_s - 2(C_s - 1)]; \\ \left\lfloor \frac{L_s - t + 1}{2} \right\rfloor, & \\ t \in [L_s - 2(C_s - 1) + 1, L_s]. \end{cases} \quad (5)$$

由式(5)的结论可得, 利用 L_s 个时隙和 C_s 个信道进行传输的最大传输次数为

$$\begin{aligned} \sum_{t=1}^{L_s} \text{PT}_{\max}(t) &= \\ & \sum_{t=1}^{L_s - 2(C_s - 1)} C_s + \sum_{t=L_s - 2(C_s - 1) + 1}^{L_s} \left\lfloor \frac{L_s - t + 1}{2} \right\rfloor = \\ & -C_s^2 + C_s(L_s + 1). \end{aligned} \quad (6)$$

分析具有簇-线型路由的网络的结构可知, 完成簇间通信所需的传输次数至少为 $\sum_{i=1}^N if_{1i}$ 。因此有

$$-C_s^2 + C_s(L_s + 1) \geq \sum_{i=1}^N if_{1i}. \quad (7)$$

求解不等式(7), 可得如下关系式:

$$C_s \geq \left\lceil \frac{L_s + 1}{2} - \sqrt{\left(\frac{L_s + 1}{2}\right)^2 - \sum_{i=1}^N \left[i \sum_{j=1}^{Q_i} y_{1ij} \right]} \right\rceil,$$

其中

$$L_s = 2 \sum_{i=2}^N \sum_{j=1}^{Q_i} y_{1ij} + \sum_{j=1}^{Q_1} y_{11j}. \quad \square$$

3.3 两阶段实时调度算法设计

基于3.1节和3.2节对簇内汇聚传输和簇间汇聚传输所需时隙数和信道数的理论分析及其证明思路, 本节提出一种基于紧凑搜索的两阶段实时调度算

法(TRSP)。该算法的主要思想为: 在时隙分配前, 判断是否需要增加时隙, 充分利用不导致时隙数增加的空闲时隙, 即要求紧凑搜索。TRSP算法由GW和路由设备各自执行。其中: GW负责为路由设备之间以及路由设备和GW之间的通信分配时隙和信道; 路由设备负责为簇内现场设备到路由设备的通信分配时隙和信道。

TRSP算法的执行包括以下3个步骤:

Step 1 簇间通信段时隙数和信道数的分配, 遵循以下4个规则:

规则 1 簇间通信段, 时隙数和信道数受定理2和定理3的限制;

规则 2 簇间通信段, 时隙 t 内分配的设备数不多于式(5)中的PT_{max}(t);

规则 3 簇间通信段, 从距离GW最远的路由设备开始, 直至网络中的所有数据发送或转发至网关设备为止;

规则 4 簇间通信段, 利用前向搜索和后向搜索结合的方法为路由设备到网关的整条路径分配时隙和信道。

前向搜索过程为: 假设 R_{1i} 利用时隙 t 发送数据, 如果 $R_{1(i-2)}$ ($i \in [3, N]$) 上有报文需要发送或者转发, 则 $R_{1(i-2)}$ 在时隙 t 利用 R_{1i} 所用信道的下一个信道发送数据; 否则, 依次考察 $R_{1(i-3)}, \dots, R_{11}$, 判断是否有报文发送或转发。

后向搜索过程为: 如果在后向搜索过程中, $R_{1(i-2)}, \dots, R_{11}$ 均没有报文需要发送或转发, 则反向考察 $R_{1(i+2)}, \dots, R_{1N}$, 判断是否有报文需要发送或者转发。

TRSP算法Step 1的执行过程如下:

算法 1 TRSP簇间通信段时隙和信道分配算法。

输入: 路由 $T = (V, E^*)$,

输出: 簇间调度结果 $S[t][\text{ch}]$ 。

begin

根据定理1和定理3计算 L_s 和 C_s ; //规则1

$Tx(R_{1i}) = 0; N_0(R_{1i}) = f_{1i};$

$\forall R_{1i} \in V - \{\text{GW}\}; N_0(\text{GW}) = 0;$

for($t = 1; t \leq L_s; t++$) do

ch = 1; 根据式(5)计算PT_{max}(t); //规则2

first_router=SearchFirstRouter(N);

//搜索报文数不等于0的首个设备

if (first_router == 1) then // 如果开始设备是 R_{11}

$S[t][\text{ch}] = R_{11}; N_t(R_{11}) = N_{t-1}(R_{11}) - 1;$

$N_t(\text{GW}) = N_{t-1}(\text{GW}) + 1;$

```

if (first_router == N) then //如果开始设备是  $R_{1N}$ 
   $S[t][ch] = R_{1N}; k = N;$ 
  for ( $i = k - 2; i > 0; i --$ ) do
    for ( $ch = 2; ch \leq PT_{\max}(t); ch ++$ ) do
      if ( $N_{t-1}(R_{1i}) > 0 \&\& S[t] \neq R_{1i}$ ) then //规则3
         $S[t][ch] = R_{1i};$ 
        for ( $j = i; j > 0; j ++$ ) do
           $S[t][ch --] = R_{1j}; ch ++;$ 
           $N_t(R_{1i}) = N_{t-1}(R_{1i}) - 1;$  //更新报文数
           $N_t(R_{1(i-1)}) = N_{t-1}(R_{1(i-1)}) + 1;$ 
           $k = k - 2;$ 
        else
          break;
      if ( $1 < \text{first\_router} < N$ ) then
        //如果开始设备是  $R_{1N}$ 
         $S[t][ch --] = R_{1\text{first\_router}}; k = \text{first\_router};$ 
        for ( $ch = 2; ch \leq PT_{\max}(t); ch ++$ ) do
          //规则4 —— 向前搜索
          for ( $i = k - 2; i > 0; i --$ ) do
            if ( $N_{t-1}(R_{1i}) > 0 \&\& S[t] \neq R_{1i}$ ) then
               $S[t][ch] = R_{1i};$ 
              for ( $j = i; j > 0; j ++$ ) do
                 $S[t][ch --] = R_{1j}; ch ++;$ 
                 $N_t(R_{1i}) = N_{t-1}(R_{1i}) - 1;$ 
                 $N_t(R_{1(i-1)}) = N_{t-1}(R_{1(i-1)}) + 1; k = k - 2;$ 
              //规则4 —— 后向搜索
            else if ( $N_{t-1}(R_{1i}) == 0 \parallel S[t] == R_{1i}$ ) then
              for ( $k = \text{first\_router} + 2; k < N + 1; k ++$ ) do
                if ( $N_{t-1}(R_{1k}) > 0 \&\& S[t] \neq R_{1k}$ ) then
                   $S[t][ch --] = R_{1k};$ 
                  for ( $j = k; j > 0; j ++$ ) do
                     $S[t][ch --] = R_{1j}; ch ++;$ 
                     $N_t(R_{1k}) = N_{t-1}(R_{1k}) - 1;$ 
                     $N_t(R_{1(k-1)}) = N_{t-1}(R_{1(k-1)}) + 1; k = k + 2;$ 
                  else
                    break;
                end
            end
          //Step2 路由设备抽取属于自身的子调度.
          各路由设备收到簇间通信段的调度结果后, 抽取相应的子调度. 子调度的抽取算法采用文献[5]的方法. GW将Step1中计算得到的簇间通信段的调度结果存储在二维数组  $S[t][ch]$  中. 其中:  $S[t][ch]$  的第1维存储时隙分配结果, 第2维存储信道分配结果. 在每个时隙, 设备对应3种工作状态: 传输(T), 接收(R)和

```

休眠(S). 路由设备将抽取得到的子调度存储在二维数组 $S_{-s}[t][2]$ 中. 其中: $S_{-s}[t][1]$ 用于存储设备在时隙 t 的工作状态, $S_{-s}[t][2]$ 存储设备在时隙 t 的信道偏移量.

对于时隙 t , 路由设备 R_{1i} 抽取子调度的原则如下:

- 1) 如果 $S[t][ch] = R_{1i}$, 则 R_{1i} 工作在传输状态, 且信道偏移量为 ch ;
- 2) 如果 $S[t][ch] = R_{1(i+1)}$, 则 R_{1i} 工作在接收状态, 且信道偏移量为 ch ;
- 3) 如果上述两条均不满足, 则 R_{1i} 工作在休眠状态.

TRSP算法Step2的执行过程如下:

算法2 TRSP子调度的抽取.

输入: $S[t][ch]$,

输出: 子调度 $S_{-s}[t][2]$.

```

begin
  for ( $t = 1; t \leq L; t ++$ ) do
    for ( $ch = 1; ch \leq C; ch ++$ ) do
      if ( $S[t][ch] = R_{1i}$ ) then //作为发送方
         $S_{-s}[t][1] < -T; S_{-s}[t][2] < -ch;$ 
      if ( $S[t][ch] = R_{1(i+1)}$ ) then //作为接收方
         $S_{-s}[t][1] < -R; S_{-s}[t][2] < -ch;$ 
      else //休眠
         $S_{-s}[t][1] < -S;$ 
    end

```

Step3 簇内通信段时隙和信道的分配.

TRSP算法Step3实现的功能是: 路由设备将自身簇内通信段的时隙分配给簇内的现场设备. 该阶段的分配原理为: 簇内的现场设备按照标识符从小到大的顺序排列, 依次选择未被占用的时隙. 簇内通信段的所有时隙对应相同的信道. 信道的选择以避免簇与簇之间的冲突为原则, 可由网关设备采用着色算法分配给每个簇.

TRSP算法Step3的执行过程如下:

算法3 TRSP簇内通信段时隙和信道分配.

输入: f_{1i} , 标识符数组 $ID[f_{1i}], C_{1ij}, N_{1ij}$,

输出: 簇内调度结果 $S_{-tra}[t][ch]$.

begin

Step1: 初始化.

$ch = CH;$

从小到大排列 $ID[f_{1i}]$ 中的元素;

for ($j = 1; j \leq L_{sa}; j ++$) do

for ($k = 0; k < C_{1iQ_{1i}}/C_{1i}; k ++$) do

//将 C_{1iQ_i}/C_{1il} 个超帧周期簇内通信段的 L_{sa}

//个时隙初始化为空闲状态

$t[j + kC_{1i1}\overline{S_f}] = 0$.

Step 2: 簇内调度.

for($l = 0; l < f(1i); l++$) do

for($j = 1; j \leq L(sa); j++$) do

for($k = 0; k < C_{1iQ_{ii}}/C_{1il}; k++$) do

idle_slot_number=1; //起始时隙号

for($m = 1; m \leq C_{1il}/C_{1i1}; m++$) do

if($t[\text{idle_slot_number} + mC_{1i1}\overline{S_f}] = 0$) then

for($n = 0; n < C_{1iQ_{ii}}/C_{1il}; n++$) do

$S_{\text{tra}}[\text{idle_slot_number} + mC_{1i1}\overline{S_f} + nC_{1il}/C_{1i1}][\text{ch}] = \text{ID}[l];$

$t[\text{idle_slot_number} + mC_{1i1}\overline{S_f}] = 1;$

else if($t[\text{idle_slot_number} + mC_{1i1}\overline{S_f}] \neq 0$)

then

idle_slot_number++;

end

TRSP算法 Step1 的时间复杂度为 $O(N^2)$; Step2 的时间复杂度为 $O(NC_s)$; Step3 的时间复杂度为 $O((\max_{i \in [1, N]} f_{1i})^2)$. 一般情况下, 信道数量 $C_s < N$ 且 $\max_{i \in [1, N]} f_{1i} \leq N$. 因此, TRSP算法的时间复杂度为 $O(N^2)$.

4 仿真分析

本节利用 OPNET10 网络仿真软件模拟簇-线型路由拓扑下采用 TRSP 算法后网络的性能. 将网络部署在 $100 \times 100 \text{ m}^2$ 的区域内. 设备的物理层和数据链路层采用 IEEE STD 802.15.4-2006 标准. 时隙长度规定为 31.25 ms.

图3为不同数据更新率情况下完成汇聚传输所需时间的最优值和仿真值, 以及与现有典型算法^[5]的比较. 单个簇内现场设备数据更新率完全相同和全部不同的情况为部分相同情况的特例. 结果表明, 存在一个最优算法使得簇-线型路由拓扑的网络完成汇聚传输的时间最短.

图3同时说明, 网络结构、网络规模和现场设备的数据更新率决定完成汇聚传输的时间. 星形和网状混合拓扑的网络相比网状网络, 完成汇聚传输的时间较短; 规模较大的网络完成汇聚传输所需时间较长; 现场设备的数据更新率种类较多的网络完成汇聚传输的时间较短, 特别是更新率各不相同情况下对应的汇聚传输时间最短. 无论节点的数据更新率如何分配, 在10个节点的网络中, 两阶段实时调度算法完成汇聚传输所需的最长时间仅为0.37s;

15个节点的网络中, 完成汇聚传输所需的最长时间不到0.66s. 而无线HART的集中式传输调度算法^[5]对应于10个节点和15个节点, 完成汇聚传输所需时间分别为0.594s和0.906s. 相比两阶段传输调度算法, 集中式算法的延时较大.

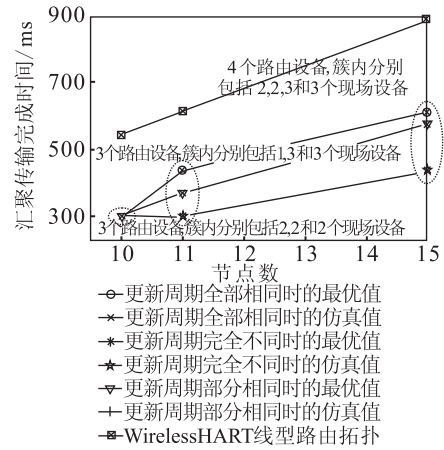
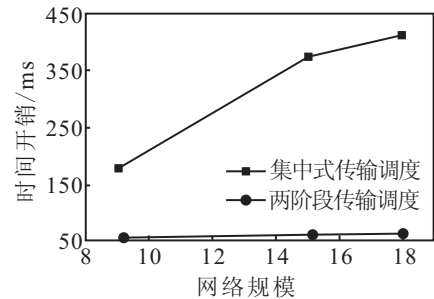
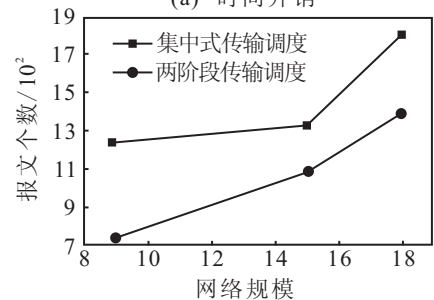


图3 汇聚传输所需时间



(a) 时间开销



(b) 报文开销

图4 算法的时间开销和报文开销

图4比较了基于簇-线型路由的集中式汇聚传输调度算法和两阶段实时汇聚传输调度算法的开销, 包括时间开销和控制报文开销. 采用两阶段实时汇聚传输调度算法可使完成汇聚传输的时间开销与集中式方法相比降低了6至8倍, 报文开销降低了近2倍, 显著地节省了网络资源, 从而证实了两阶段实时汇聚传输调度算法轻型快速的特点.

5 结论

本文分析了簇-线型路由下完成汇聚传输调度所需时隙数和信道数的下限值, 以及网络结构、网络规模和数据更新率对于时隙数和信道数下限值的影响.

并基于理论分析的下限值, 针对无线传感器网络实时性的要求, 设计了簇-线型路由下基于紧凑搜索的两阶段实时调度算法. 该算法通过最小化调度所需时隙数和信道数, 即最大化并行传输的设备和最小化汇聚传输时间, 缓解了无线资源受限的问题, 从而实现了充分且实时地利用无线通信资源的目标. 最后, 仿真结果验证了面向具有簇-线型路由的星形和网状两层拓扑网络的两阶段实时调度算法的有效性, 并说明了该算法在实时性和复杂度方面的优势.

参考文献(References)

- [1] 龚本治. 工业无线网络实时路由协议研究与实现[R]. 武汉: 武汉理工大学, 2009.
(Gong B Z. Research and realization of real-time routing in industrial wireless sensor networks[R]. Wuhan: Wuhan University of Technology, 2009.)
- [2] 柯欣, 孙利民, 吴志美. 基于无线传感器网络汇聚传输实时性的分布式调度算法[J]. 通信学报, 2007, 28(4): 44-50.
(Ke X, Sun L M, Wu Z M. Distributed scheduling for real-time convergecast in wireless sensor networks[J]. J of Communications, 2007, 28(7): 44-50.)
- [3] Gandham S, Zhang Y, Huang Q F. Distributed minimal time convergecast scheduling in wireless sensor networks[C]. Proc of the 26th Int Conf on Distributed Computing Systems(ICDCS). Lisboa: IEEE Press, 2006: 50-57.
- [4] Soldti P. On cross-layer design and resource scheduling in wireless networks[D]. Sweden: KTH School of Electrical Engineering, 2009.
- [5] Huanli P S, Ramamritham K. Scheduling messages with deadlines in multi-hop real-time sensor networks[C]. Proc of IEEE Int Symposium on Real Time and Embedded Technology and Applications. San Francisco: IEEE Press, 2005: 415-425.
- [6] Industrial communication network - Fieldbus specifications - WIA-PA communication network and communication profile[S]. <http://www.iec.ch/>.
- [7] Zhang X L, Liang W, Zheng M, et al. Distributed and dynamic TDMA channel scheduling algorithm for WIA-PA[C]. Proc of the IEEE Int Conf on Intelligent Computing and Intelligent Systems(ICIS). Shanghai: IEEE Press, 2009: 462-466.
- [7] 易平涛, 郭亚军, 张丹宁. 密度加权平均中间算子及其在多属性决策中的应用[J]. 控制与决策, 2007, 22(5): 515-524.
(Yi P T, Guo Y J, Zhang D N. Density weighted averaging middle operator and application in multi-attribute decision making[J]. Control and Decision, 2007, 22(5): 515-524.)
- [8] 易平涛, 郭亚军. 广义实型密度加权平均中间算子及其应用[J]. 系统工程学报, 2010, 25(2): 190-195.
(Yi P T, Guo Y J. Generalized real density weighted averaging middle operator and its application[J]. J of Systems Engineering, 2010, 25(2): 190-195.)
- [9] 易平涛, 高立群, 郭亚军. 基于多源密度信息集结算子的组合评价方法[J]. 系统工程与电子技术, 2009, 31(12): 2882-2887.
(Yi P T, Gao L Q, Guo Y J. Combined evaluation based on multi-sources information density aggregation operator[J]. Systems Engineering and Electronics, 2009, 31(12): 2882-2887.)
- [10] Herrera F, Herrera-Viedma E, Martinez L. A fusion approach for managing multi-granularity linguistic terms sets in decision making[J]. Fuzzy Sets and Systems, 2000, 114(1): 43-58.
- [11] Herrera F, Martinez L. A 2-tuple fuzzy linguistic representation model for computing with words[J]. IEEE Trans on Fuzzy Systems, 2000, 8(6): 746-752.
- [12] 王欣荣, 樊治平. 基于二元语义信息处理的一种语言群决策方法[J]. 管理科学学报, 2003, 6(5): 1-5.
(Wang X R, Fan Z P. Method for group decision making based on two-tuple linguistic information processing[J]. J of Management Sciences in China, 2003, 6(5): 1-5.)

(上接第760页)