

# 支持异构数据通信的 IEEE 802. 15. 4 实时带宽分配算法

胡显俊 陈建新 周生强 李一帆

(南京邮电大学“无线宽带通信与传感网络技术”教育部重点实验室 南京 210003)

**摘要** IEEE 802. 15. 4 协议的 GTS 分配机制能够支持实时和延迟约束的应用,目前已经应用于医疗健康、工业控制和楼宇自动化等物联网领域。但是,在高速实时异构数据传输应用中仍然存在局限性,它不能解决多于 7 个设备需要实时服务、时延约束小于超帧长度以及不同周期异构数据传输等问题。为了摆脱这些限制,提出一种新的支持异构数据通信的 IEEE 802. 15. 4 实时带宽分配算法,其根据不同周期任务的数据传输信息,调整部分传输任务的发送时间。性能分析显示,该算法可以严格遵循时延约束条件,满足异构数据通信,提高带宽利用率,提升整个网络的性能。

**关键词** 无线传感网络, IEEE 802. 15. 4, GTS 分配, 实时通信, NPGAS

**中图分类号** TP393 **文献标识码** A **DOI** 10. 11896/j. issn. 1002-137X. 2016. 6. 022

## IEEE 802. 15. 4 Real-time Bandwidth Allocation Algorithm Supporting Heterogeneous Data Communication

HU Xian-jun CHEN Jian-xin ZHOU Sheng-qiang LI Yi-fan

(Key Lab. of Broadband Wireless Communication & Sensor Network Technology, Ministry of Education, Nanjing University of Posts & Telecommunications, Nanjing 210003, China)

**Abstract** GTS allocation mechanism on IEEE 802. 15. 4 protocol can support real time and delay constrained applications. It has been used in the Internet of Things field such as medical health, industrial control, building automation and etc. But in the applications of high-speed real-time heterogeneous data transmission, there are still some limitations. It does not support real-time services required by more than seven devices, delay constraints less than the superframe length and different periodic heterogeneous data transmission applications. To overcome these limitations, this paper proposed a new IEEE 802. 15. 4 real-time bandwidth allocation algorithm supporting heterogeneous data communication, and adjusted the transmission time of some of the transfer tasks according to the data transmission information of different periodic tasks. Performance analysis result shows that this algorithm is in strict compliance with delay constraints, and can meet the needs of heterogeneous data communication, improve the performance of the bandwidth utilization, and then improve the performance of the overall network.

**Keywords** Wireless sensor network, IEEE 802. 15. 4, GTS allocation, Real-time communication, NPGAS

## 1 引言

过去几年,无线通信技术飞速发展,伴随着物联网应用的推广,出现了越来越多的无线传感网络应用。这类应用普遍采用低功耗无线通信技术,以较高频率采集数据,从而对低功耗实时通信技术提出新的挑战。IEEE 802. 15. 4 协议是一种标准的低速率、低功耗和低成本的短距离无线通信协议,不仅可以应用于医疗健康(如人体健康生理参数实时监测),也可以应用于工业控制、环境监测以及基础设施如桥梁等质量监测<sup>[12]</sup>。

IEEE 802. 15. 4 标准因其介质访问控制子层(Media Ac-

cess Control, MAC)的灵活性,采用保障时隙(Guaranteed Time Slot, GTS)介质访问控制机制来提供实时保障传输,使其适合于无线传感器网络中的实时通信传输。但是,协议在低功耗、高速率、实时传输应用中仍存在不足和局限性,因而如何分配 GTS 是 IEEE 802. 15. 4 支持满足延时约束的应用的关键,该协议的 GTS 实时通信调度算法是当前研究的热点。

Koubaa 等使用网络微积分的方法研究了 GTS 分配的带宽利用率和时延问题<sup>[1,3,4]</sup>,提出一种 i-GAME 的 GTS 分配算法。算法根据网络中各节点的数据传输量和时延要求等信息实现网络中不同节点对同一信标周期中相同 GTS 的共享,

到稿日期:2015-05-05 返修日期:2015-07-24 本文受中国国家自然科学基金(61201165),泛在无线通信和无线传感器网络教育部重点实验室开放课题项目(NYKL201306),江苏省高校自然科学基金(13KJB510026),同济大学嵌入式系统与服务计算教育部重点实验室项目(ESSCKF201305)资助。

胡显俊(1990—),男,硕士生,主要研究方向为无线传感器网络,E-mail:1013010415@njupt.edu.cn;陈建新(1973—),男,博士,副教授,主要研究方向为无线可穿戴计算;周生强(1991—),男,硕士生,主要研究方向为无线传感器网络;李一帆(1992—),男,硕士生,主要研究方向为室内定位。

即动态地把 GTS 时隙分配给不同的节点,显著地提高了带宽利用率。Cheng 等提出一种称为 16-mts 的 GTS 时隙分配算法<sup>[5]</sup>,将标准协议的保障时隙划分为 16 个等长的微小时隙,降低一个单位时隙的长度,让不同的设备在相同的周期内共享带宽,将 CFP 中带宽资源浪费最小化,从而提高了 GTS 时隙的带宽利用率。文献<sup>[6]</sup>提出一种 EDF (Earliest Deadline First)算法,该算法将 GTS 分配给众多设备节点中期限或延迟最早的,很好地解决了时延问题,但是会增加有延迟约束的事件突发性传输的负载。

许多关于 GTS 分配算法的研究考虑的时延约束通常都要大于一个帧长,而对于时延约束要求比较严格的应用,如体域网应用里的高速运动、人体实时生理参数的监测等,数据采集速率高达几十赫兹,上述所提出的 GTS 分配算法已经无法应用。基于此,Chen 等提出了一种 EGSA 算法<sup>[11]</sup>,通过最大化 CFP 长度将 GTS 进一步划分为更短的微小时隙来支持多节点、周期性、有延时约束的应用,带宽利用率和延时约束都显著提高。但是,EGSA 算法也有其局限性,即主要解决多节点、同周期、高速实时任务传输,并不适合不同周期、异构数据的高速实时通信。在 EGSA 算法的基础上,本文提出一种新的支持异构数据通信的 IEEE 802.15.4 实时带宽分配算法,主要解决不同周期下高速实时通信数据的传输,有效地提高了异构数据高速实时传输任务的网络性能,满足体域网或物联网中异类传感器高速实时数据采集和发送。

## 2 问题描述

### 2.1 IEEE 802.15.4 协议简介

IEEE 802.15.4 标准主要定义了物理层(PHY)和介质访问控制子层(MAC)。MAC 层主要支持两种模式:信标模式和非信标模式。在信标模式中设备通过超帧结构传输数据,超帧结构如图 1 所示,主要分为活跃和不活跃两个部分。在不活跃期间,节点进入休眠状态以节省能量。活跃期间划分为 16 个相同大小的时隙,主要由 3 个部分组成:信标阶段、竞争访问时段(Contention Access Period, CAP)和非竞争访问时段(Contention Free Period, CFP)。其中,信标阶段主要用于协调器周期性地与设备进行同步,CAP 时段用于非实时应用需求,CFP 时段主要用于有延时约束的实时业务需求<sup>[12]</sup>。

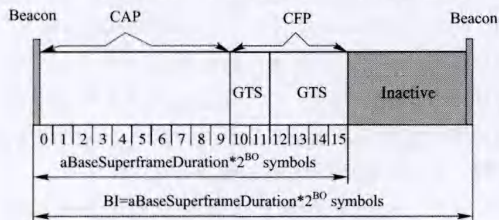


图 1 IEEE 802.15.4 超帧结构

在标准协议的超帧结构中,超帧长度(SD)和信标间隔(BI)分别为:

$$SD = aBaseSuperFrameDuration \times 2^{SO} \quad (1)$$

$$BI = aBaseSuperFrameDuration \times 2^{BO} \quad (2)$$

其中,SO 表示超帧序列值,BO 表示信标序列值,SO 决定了超帧的长度,BO 决定了信标间隔的长度,并且  $0 \leq SO \leq BO \leq$

14。IEEE 802.15.4 标准定义最小超帧长度  $aBaseSuperFrameDuration$  为 960 符号,在 2.4GHz 频率下,每个符号 4bit。因此,当  $0 \leq SO = BO \leq 14$  时,超帧的长度等于其活跃期间的长度,超帧的持续时间在 15.36ms 和 251.66s 之间。当 SO 等于 BO 时,超帧的活跃期间划分为 16 个相同大小的时隙,所以超帧的每个时隙的长度(单位:bit)表示为:

$$L_{TS} = \frac{960 \times 4 \times 2^{SO}}{16} = 240 \times 2^{SO} \quad (3)$$

### 2.2 问题定义

考虑在 IEEE 802.15.4 信标使能的无线传感器网络中 1 个协调器节点和若干个设备节点组成的星型拓扑结构网络。每个设备节点采用不同的周期向协调器节点发送数据,每个设备节点的非周期性任务表示为  $(N_i, T_i, L_i, D_i)$ ,其中  $N_i$  表示第  $i$  个节点 ID,  $T_i$  表示节点传输数据的发送周期,  $L_i$  表示传输数据的长度,  $D_i$  表示数据的发送期限。每个设备节点数据传输任务记为一个事务,表示为  $transaction(addr, startT, endT)$ ,其中  $addr$  表示执行事务的节点的地址,  $startT$  表示事务开始时间,  $endT$  表示事务的截止时间。根据上述表示方法可以计算出每个节点的事务列表<sup>[2]</sup>。于是,文中需要解决的问题为:设计一种带宽分配和调度算法,能够满足非周期多节点异构数据高速实时通信,在满足延时约束的基础上,使得系统具有较高的带宽利用率。

## 3 非周期实时带宽分配算法

### 3.1 算法的基本思想

在自动化系统等无线传感网络应用中,传感器节点用于控制或实时监测等,其数据发送和传输都具有相同周期。EGSA 算法解决了同设备、同周期高速实时数据传送的调度。然而,有些应用中,由于每个节点的状态不同,产生数据的周期也不相同,超帧需要匹配最短的周期,使得每个节点都可以在规定时间内发送其数据,并且分配给每个节点超帧的 GTS 并不总是适用,导致带宽利用率较低,整个网络的性能较差。

基于此,在 EGSA 算法的基础上,提出一种新的支持异构数据通信的 IEEE 802.15.4 实时带宽分配算法(Non-periodic GTS Allocation Scheme, NPGAS),主要解决非周期异构数据的实时传输。当不同的节点采用不同的周期采集数据并且发送数据时,由于这些参数都是已知的,采用 EDF 动态时隙调度算法需要消耗较多的系统资源和能量,并且带宽利用率较低<sup>[6,14]</sup>。而 NPGAS 算法是一种预分配的静态离线调度算法,根据不同节点的传输任务参数,在数据发送之前分配完成,无须每次调度都进行动态的分配。算法根据不同周期任务的数据传输信息,通过调整部分传输任务的发送时间和周期,充分利用数据发送周期中空闲部分导致的带宽浪费,从而提高带宽利用率,提升整个网络的性能。

### 3.2 算法步骤与分析

#### 3.2.1 CFP 长度的计算

在本系统中,假设数据的发送周期为 2 个部分:数据部分和空闲部分,其中数据部分包括数据包的长度、帧间隔的长度等。图 2 显示了不同节点的发送周期,有  $n$  类不同周期的数据需要发送,其中  $L_1$  表示的是节点 1 的数据部分长度,  $K_1$

表示的是其空闲部分长度,  $T_1$  表示的是节点 1 发送的周期, 其他节点同理。IEEE 802.15.4 为标准的 GTS 分配机制, 每个 GTS 可以由若干个时隙组成, 最多只能分配 7 个 GTS, 每个设备节点至少占用一个 GTS, 所以当  $SO$  值增大时, 每个时隙的长度增大, 浪费的空间逐渐增多, 带宽利用率也随之降低。

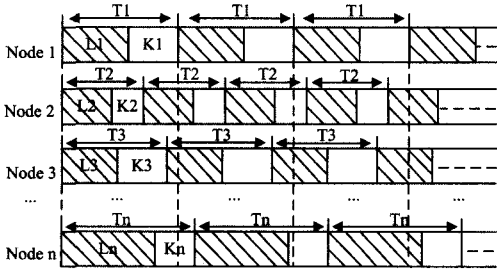


图 2 不同节点的发送周期

在超帧的不活跃期间, PAN 网络中的设备之间不通信, 为了节省能量, 设备进入睡眠模式, 数据不能传送。而不活跃期间又是由  $SO$  和  $BO$  决定的, 所以为了支持高速实时数据的传输, 令  $SO=BO$ , 使整个超帧的长度等于活跃期间的长度。

其次, CAP 时段主要用于非实时应用需求。在本系统中, 每个节点传送数据都需要满足一定的时延要求, 所以在超帧结构中, 让 CAP 的长度尽可能小, 不仅可以避免许多节点在过长的 CAP 期间因为无法传送数据而超时, 而且, 通过最大化 CFP 的长度, 能够提供更多的带宽用于实时通信。IEEE 802.15.4 协议规定 CAP 最小长度  $aMinCAPLength$  为 440 符号(在 2.4GHz 频率为 220 字节), 信标帧的最小长度为 13 个字节, 加上物理层开销总共为 25 个字节, 最终定义最小的 CAP 长度  $L_{minCAP}$  为 245 个字节<sup>[11]</sup>。但是, CAP 的末端必须与 TS 边界对齐, 把超帧起始的信标帧与紧随后的 CAP 记作  $\overline{CAP}$ , 则  $\overline{CAP}$  的最小长度为:

$$n_{CAP} = \left\lceil \frac{L_{minCAP}}{L_{TS}} \right\rceil = \left\lceil \frac{245}{30 \times 2^{SO}} \right\rceil \quad (4)$$

$$L_{CAP} = n_{CAP} \cdot L_{TS} \quad (5)$$

其中,  $n_{CAP}$  表示 CAP 占用的时隙个数,  $L_{TS}$  表示标准时隙长度, 由式(3)可以得出,  $L_{CAP}$  表示 CAP 的长度。则 CFP 的长度为:

$$L_{CFP} = (timeSlotsNumber - n_{CAP}) \cdot L_{TS} \quad (6)$$

其中,  $timeSlotsNumber$  表示标准时隙的总个数。

### 3.2.2 $SO$ 值的选取

根据每个设备节点传输任务的参数  $(N_i, T_i, L_i, D_i)$ , 可以获得能够满足传输应用延时约束的  $SO_{min}$  和  $SO_{max}$  值。首先, 延时大小可以表示为:

$$d_g = \frac{L_{CAP} + T_1}{R} \quad (7)$$

其中,  $L_{CAP}$  表示的是 CAP 的长度,  $T_1$  表示 3.2.3 节中按照空闲比率排序后的首个数据的发送周期,  $R$  表示的是比特的速率。由于  $0 \leq SO \leq 14$ , 将  $SO$  值的取值代入式(7)中, 可以得出满足等式的  $SO$  值大小。

然而, 数据传输任务需要的带宽可能不满足 CFP 的带宽, 因此还需要满足下面的条件:

$$L_{CAP} + T_1 \leq SD \times 4 - L_{minCAP} \quad (8)$$

式(8)保证了传输的消息流至少有一个传输能够满足 CFP 的带宽。注意, 式(8)中数字 4 表示的是符号和位之间的转换。

### 3.2.3 空闲比率的计算

根据每个节点不同发送周期的数据长度和空闲长度, 计算每个节点的空闲比率:

$$\lambda = K/L \quad (9)$$

其中,  $K$  为空闲部分长度,  $L$  为数据部分长度。假设当前有 2 个节点传输数据。则根据图 2 中发送周期的空闲比率可以得出  $\lambda_1 = K_1/L_1$ ,  $\lambda_2 = K_2/L_2$ 。比较空闲比率大小, 从大到小排序。

### 3.2.4 算法的分配过程

NPGAS 算法解决了不同周期情况下多节点高速实时通信数据的传输, 算法的具体流程可总结如下。

Step 1 根据 3.2.3 节计算的空闲比率  $\lambda_1, \lambda_2$ , 若  $\lambda_1 > \lambda_2$ , 则用节点 2 的数据部分填充节点 1 的空闲部分。如果  $K_1/L_2 > 1$ , 计算  $K_1$  能够容纳节点 2 的周期数目:

$$m = \begin{cases} \left\lfloor \frac{K_1}{L_2 + K_2} \right\rfloor + 1, & K_1 \bmod (L_2 + K_2) > L_2 \\ \left\lfloor \frac{K_1}{L_2 + K_2} \right\rfloor, & K_1 \bmod (L_2 + K_2) < L_2 \end{cases} \quad (10)$$

根据式(10)可知, 节点 1 的空闲部分可由  $m$  个节点 2 的周期填充, 满足分配。如果  $K_1/L_2 < 1$ , 表示节点 1 的空闲部分小于节点 2 的数据长度, 不满足此分配, 则计算  $K_2/L_1$ , 重复上述步骤。当  $K_1/L_2 < 1, K_2/L_1 < 1$  时, 表示上述分配未满足, 因此考虑节点自身的空闲部分大小, 分别计算  $K_1/L_1$  和  $K_2/L_2$ , 重复上述步骤。若分配能满足, 将另外一个节点放入下一次分配中; 若两个节点都无法满足分配, 则跳转到 Step 2。

Step 2 为了充分利用空闲的带宽资源, 考虑将数据段进行分割传输。计算  $L_2/K_1$ , 若结果小于或等于 2, 则满足算法的分配。由于数据段的分割会产生相应的分割开销  $\gamma$ , 若  $L_2/K_1$  的结果大于 2, 表示节点 2 的分割次数多于 2 次, 影响后续节点的带宽分配, 并且开销  $\gamma$  过大, 因此不满足算法的分配。继续计算  $K_2/L_1$ , 重复该步骤的过程。若该步骤的两种分配皆无法满足, 则跳转到 Step 3。

Step 3 首先计算  $K_1/L_1$ , 若结果小于或等于 2, 则表示可以分割。接着判断  $L_1 - K_1$  是否小于  $K_2$ , 若小于, 表示  $L_1$  切割完的数据部分可以被节点 2 的空闲部分填充, 即  $L_1$  切割完剩余的部分由节点 2 的空闲部分填充; 否则节点 2 的空闲部分过小, 不考虑。若  $L_1/K_1$  计算的结果大于 2, 表示产生的分割开销  $\gamma$  过大, 不满足分配。继续计算  $L_2/K_2$ , 重复该步骤的过程。若无法满足分配, 跳转到 Step 4。

Step 4 当上述分配都无法满足时, 说明 2 个周期的空闲部分相对于数据部分都十分小, 无需进行分配。如果空闲部分的绝对长度比较大, 可以变动较大的发送周期为数据部分的长度, 减少空闲长度导致的带宽浪费。当所有的分配完成后, 计算整个 CFP 长度, 统计分配的情况, 在满足  $d_g < D$  的情况下将 CFP 中分配完剩余的带宽移至 CAP 部分, 其中  $d_g$  表示延时约束大小, 可重新定义为:

$$d_g = \frac{L_{CAP} + t_{\Delta} + T_1}{R} \quad (11)$$

其中,  $t_{\Delta}$  表示的是 CFP 中不能用于传输任何数据剩余的长度大小。

上述步骤显示了不同周期的任务传输时算法的具体分配过程, 算法通过修改部分传输任务的发送时间和周期, 充分利用节点发送周期中因空闲部分长度导致的带宽资源浪费, 提高了带宽利用率。算法的分配流程不需要每次调度时进行计算, 只需在整个传输任务起始阶段进行调度分配, 降低了系统的性能开销。

### 3.3 算法实现

NPGAS 算法只对 IEEE 802.15.4 标准做了细小的改变, 在 MAC 协议中的超帧部分做了微小改动, 不再分配标准的 GTS 时隙, 所以需要对标帧结构中的 GTS 信息子域做相应的修改。

标准的 GTS 信息子域格式如表 1 所列, 在第一个字节的 GTS 描述域结构中, 第 0-2 位表示 GTS 描述符计数, 第 7 位表示 GTS 允许位, 其余位均为保留。为了实现 NPGAS 算法, 首先在信标帧中使用一个标识位指示是否使用 NPGAS 算法。标准的 GTS 信息子域中, 第 3-6 位是保留位。于是, 使用第 3 位作为 NPGAS 算法的标识位, 如表 2 所列, 当该位为 0 时, 使用 IEEE 802.15.4 标准 GTS 分配算法, 当该位为 1 时, 使用 NPGAS 分配算法。

表 1 标准的 GTS 信息子域

位: 0-2	3-6	7
GTS 描述符计数	保留	GTS 允许位

表 2 NPGAS 信息域的 GTS 信息子域

位: 0-2	3	4-6	7
GTS 描述符计数	NPGAS 标识位	保留	GTS 允许位

当使用 NPGAS 调度算法时, 信标帧中需要附带分配信息, 算法将该信息添加在信标载荷中, 如表 3 所列。根据算法分配的结果, 起始 4 个字节表示分配的组数, 后面紧跟节点地址列表, 每个地址占 2 个字节, 第  $i$  个地址就表示第  $i$  个分配给该地址的节点。

表 3 NPGAS 信息域的分配信息

字节: 4	2	2	...	2	2
分配组数	地址	地址	...	地址	地址

MAC 协议的帧头、超帧描述域、GTS 描述域、帧校验共占 12 字节, 分配组数占 4 字节, 地址列表占 2k 字节。算法首先根据式(6)计算得到最长的 CFP 时段, 再利用各个节点的发送数据的信息, 调用 NPGAS 算法计算每个分配的节点地址, 填入地址列表中, 将分配的组数填入计数域。于是信标帧构造完成, 协调器节点将其广播出去, 设备节点在信标帧中获取自己被分配到的序号列表, 计算出数据发送的起始时间, 到达该时间时便可以发送数据。

## 4 性能分析

本节分析 NPGAS 算法的实时调度性能, 并与其他算法进行比较。首先, 当所有节点传输数据的周期相同时, NPGAS 算法与 EGSA 算法的策略相同, 即通过最大化 CFP 长度将 GTS 进一步划分为更短的微小时隙, 让相同周期的节点组成块进行传输, 从而能够支持多节点周期性应用。但是,

EGSA 算法考虑的场景仅仅是 NPGAS 算法的一个特例, 即所有节点采用相同的周期, 当节点发送数据的周期不同时, EGSA 算法无法进行调度。接着, 考虑一个应用实例来分析 NPGAS 算法的实时调度性能, 并与 IEEE 802.15.4 标准的分配算法和 16-mTS 算法进行比较。

本节考虑的应用场景为若干可穿戴无线传感器节点非周期性地发送数据, 根据周期的不同, 选择系统中两个简单的节点, 其发送周期如图 3 所示, 分别为 (50, 90) 和 (25, 15), 即节点 1 的周期为 140, 数据长度为 50, 空闲长度为 90; 节点 2 同理。当 SO 值为 4 时, 根据式(4)可以计算出 CAP 最小长度占用一个时隙, CFP 的长度为 230.4ms。带宽利用率如式(12)所示:

$$U_{CFP} = \frac{t_{tx}}{t_{CFP}} \quad (12)$$

其中,  $t_{tx}$  表示算法在 CFP 期间传输所有任务的时间。

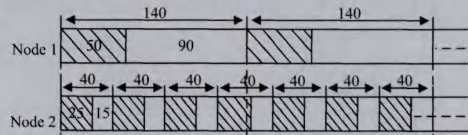


图 3 不同节点的发送周期

采用标准的 GTS 分配算法, 超帧的活跃期间划分为 16 个时隙, 每个 GTS 可以由若干个时隙组成, 每个节点设备至少占有一个 GTS, 最多只能支持 7 个设备传输数据。具体分配结果如图 4 所示, 当传输的速率小, 使用的带宽小于分配到的 GTS 带宽时, GTS 的带宽使用率十分低, 导致资源的浪费。

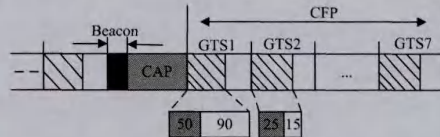


图 4 IEEE 802.15.4 标准 GTS 算法分配结果

16-mTS 算法通过降低一个单位时隙的长度, 将 CFP 的长度划分为 16 个微小的时隙来减少带宽资源的浪费, 并且能够保证帧格式的完整性, 具体的分配结果如图 5 所示。当 SO 值小时, 算法通过将 CFP 分成 16 个微小时隙可以有效减少带宽资源的浪费。但是, 随着 SO 值的增大, CFP 带宽增大, 算法无法支持更多的节点加入, 带宽资源浪费逐渐增加, 带宽利用率较低。

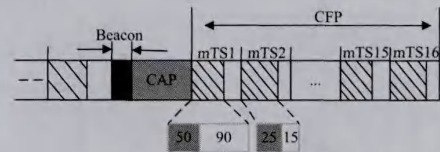


图 5 16-mTS 算法分配结果

采用 NPGAS 算法的具体分配结果如图 6 所示。具体的分配过程由 3.2.4 节得出, 其中  $d_g$  表示的是延时约束的大小, 由式(5)可以得出,  $\beta_k$  表示分配的周期。NPGAS 算法通过调整一个传输任务的发送时间, 减少了因发送周期中空闲部分导致的带宽浪费, 带宽的利用率较高。

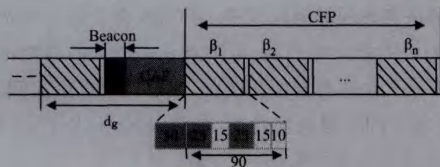


图6 NPGAS算法分配结果

在 Windows 平台下,使用 C++ 编程,对 3 种算法进行仿真,分别得到 3 种算法  $SO$  取不同值时的带宽利用率,如图 7 所示。IEEE 802.15.4 标准分配机制中 CFP 中的 GTS 分配数量最多为 7 个,而 16-mTS 算法将 CFP 长度划分为 16 个 mTS,这两种算法都是随着  $SO$  值增大,每个时隙的长度增大,带宽资源的浪费逐渐增多,带宽利用率也随之降低。而采用 NPGAS 算法,通过最大化 CFP 的长度,有效利用数据发送周期中空闲部分导致的带宽浪费,时延约束得到了有效的保证,并且带宽利用率较其他两种算法也得到了显著的提高。

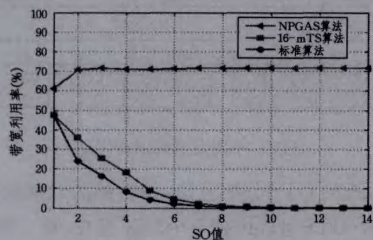


图7 不同算法带宽利用率比较

当  $SO$  值较小时,时隙长度较小,由于 CAP 需要与时隙边界对齐,取最小长度的 CAP 仍占用较多时隙,剩下可以调度的时隙数较少,因此 NPGAS 算法的带宽利用率较低。当  $SO$  值增大时,由于有效利用了部分空闲带宽资源,算法的带宽利用率较高。而 16-mTS 算法将 CFP 长度划分为 16 个 mTS,让不同的设备在相同周期内共享带宽,与标准算法相比,带宽利用率更高。但是,随着  $SO$  值增大,时隙长度增加,由于标准的算法和 16-mTS 算法可以提供的非周期传送的次数有限,带宽浪费增加,带宽利用率随着  $SO$  值的增大逐渐减小。

NPGAS 算法的延时约束可以通过式(11)计算得出,主要取决于 CAP 的长度和算法分配首个周期长度。其他两种算法的延时约束可以通过式(3)和式(7)得出。图 8 显示了当  $SO$  值变化时,NPGAS 算法与其他两种比较算法的延时性能对比。请注意,图中 Y 轴上使用的是对数尺度。

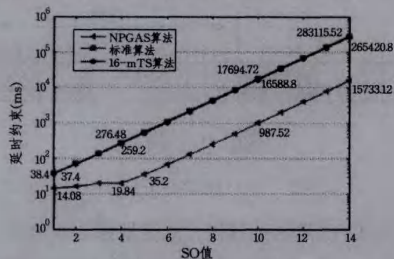


图8 不同算法延时约束比较

当  $SO$  等于 4 时,可以比较 3 种算法的延时性能,NPGAS 算法的延时约束是 19.84ms,而标准算法和 16-mTS 算法的

延时分别是 276.48ms 和 259.2ms。从图 8 中可以发现,当  $SO$  值小于 4 时,NPGAS 算法的延时约束变化较小。NPGAS 算法的延时约束主要取决于最小的 CAP 大小。而标准算法和 16-mTS 算法的延迟约束随着  $SO$  值增加而成倍增加,这主要由于两种算法的延时约束和 TS 的大小相关。当  $SO$  等于 1 时,标准算法和 16-mTS 算法提供延时分别为 38.4ms 和 37.4ms,不能满足许多应用严格的延时约束要求,而 NPGAS 算法可以提供一个较低的延时约束为 14.08ms。同时从图 8 中还可以发现,随着  $SO$  值的变化,NPGAS 算法的延时大小一直远远小于其他两种算法。

**结束语** 本文提出了一种支持异构数据通信的 IEEE 802.15.4 实时带宽分配算法,主要解决多节点高速率非周期数据传输的问题。算法通过最大化 CFP 的长度,让更多的带宽用于高速实时数据传输,并充分利用了不同节点异构数据传输过程中空闲部分导致的带宽浪费。仿真结果表明,与标准的 IEEE 802.15.4 协议和 16-mTS 算法相比,NPGAS 算法满足了应用的时延约束条件,显著地提高了带宽利用率。并且,算法与标准 GTS 分配算法完全兼容,改动较小,实现方便。未来的工作将进一步着眼于分析不同时刻、不同节点数、非周期的高速实时应用和延时约束保障之间的关系,为高速实时数据传输应用奠定理论基础。

## 参考文献

- [1] Koubaa A, Alves M, Tovar E. Gts allocation analysis in IEEE 802.15.4 for real-time wireless sensor networks[C]//Proc of 14th International Workshop on Parallel and Distributed Real-Time Systems, Island of Rhodes, Greece; IEEE, 2006; 25-29
- [2] Bu Xiang, Zhang Yun, Chen Jian-xin. A Rapid and Real-time Data Transmission Algorithm for IEEE 802.15.4[J]. Computer Technology and Development, 2013, 3(4): 14-18 (in Chinese)
- [3] 卜翔,章韵,陈建新. IEEE 802.15.4 支持高速实时数据传输算法[J]. 计算机技术与发展, 2013, 03(4): 14-18
- [4] Koubaa A, Alves M, Tovar E. i-GAME: an implicit GTS allocation mechanism in IEEE 802.15.4 for time-sensitive wireless sensor networks[C]//Proc of the 18th Euromicro Conference on Real-Time Systems, Dresden, Germany; IEEE, 2006; 183-192
- [5] Koubaa A, Alves M, Tovar E. Energy and delay trade-off of the gts allocation mechanism in IEEE 802.15.4 for wireless sensor networks; Research articles[J]. Communication Systems, 2007, 20(7): 791-808
- [6] Liang C, Bourgeois A G, Xin Z. A new GTS allocation scheme for IEEE 802.15.4 networks with improved bandwidth utilization[C]//International Symposium on Communications and Information Technologies, 2007 (ISCIT 07). IEEE, 2007; 1143-1148
- [7] Buttazzo G C. Rate Monotonic vs. EDF; Judgment Day [J]. Real-Time Systems, 2005, 29; 5-26
- [8] Lee D H, Roh H T, et al. Performance Analysis of the IEEE 802.15.4 MAC Protocol[C]//2013 International Conference on ICT Convergence (ICTC). 2013; 398-401
- [9] Chen Chen, X Wei, et al. A Delay Sensitive Time Slots Schedu-

ling in Vehicular Sensor Networks[C]//2013 5th International Conference on Intelligent Networking and Collaborative Systems (INCoS). IEEE, 2013; 448-455

- [9] Rasyid M U H A, Lee B H, Sudarsono A. PEGAS; Partitioned GTS Allocation Scheme for IEEE 802. 15. 4 Networks[C]//2013 International Conference on Computer, Control, Informatics and Its Applications (IC3INA). Nov. 2013; 29-32
- [10] Xia Feng, Hao Ruo-nan, Cao Yang, et al. ART-GAS; an adaptive and real-time GTS allocation scheme for IEEE 802. 15. 4[C]//Proc of AINTEC '11. New York: ACM, 2011; 96-103
- [11] Chen Jian-xin, Ferreira L, Tovar E. An Explicit GTS allocation algorithm for IEEE 802. 15. 4[C]//Proc of Emerging Technologies & Factory Automation (ETFA), 2011 IEEE 16th Conference. Toulouse; IEEE, 2011; 1-8
- [12] IEEE 802. 15. 4 standard : Wireless Medium Access Control (MAC) and Physical Layer (PHY) Specifications for Low-Rate

Wireless Personal Area Networks (WPANs) [R]. October 2003; 1-133

- [13] Lee D H, Roh H T, et al. Performance Analysis of the IEEE 802. 15. 4 MAC Protocol[C]//2013 International Conference on ICT Convergence (ICTC). 2013; 398-401
- [14] Shen Zhuo-wei. Schedulability Analysis for Non-Preemptive EDF Scheduling Algorithm[J]. Computer Engineering and Applications, 2006, 42(9): 10-12 (in Chinese)  
沈卓炜. 不可抢占式 EDF 调度算法的可调度性分析[J]. 计算机工程与应用, 2006, 42(9): 10-12
- [15] Qiao Guan-hua, Mao Jian-lin, Guo Ning, et al. Research and Improved Design in IEEE 802. 15. 4 MAC Protocol for Service Distinguishing[J]. Computer Science, 2014, 41(10): 149-153 (in Chinese)  
乔冠华, 毛剑琳, 郭宁, 等. 基于业务区分的 IEEE 802. 15. 4 协议分析及改进[J]. 计算机科学, 2014, 41(10): 149-153

(上接第 71 页)

着 RegionServer 结点数量的增加呈间隔性线性增长, 证明了系统具有良好的可拓展性。因此在租户以及传感器数量增加的情况下, 可以通过增加 HBase 的 RegionServer 结点的数量来提高系统的吞吐量。

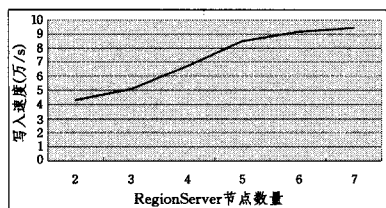


图 8 不同 RegionServer 节点下的写入速度

**结束语** 本文针对实时数据库在处理大规模历史数据方面的缺陷, 鉴于 HBase 在大数据处理方面的优越性, 提出了基于融合数据库的海量传感器信息存储架构。该架构将传感器数据集中式存储, 采用多租户机制, 将原来分散的传感器数据集中式存储, 通过将传感器元数据与数据分离, 维持了实时数据库原有查询、数据组织结构的特点, 并对 HBase 写入进行了优化。经过实验验证, 该系统具有良好的可拓展性和读写性能, 有效避免了写入热点。

同时, 该架构有一定的不足之处, 进一步的改进方向包括: 1) 如何在保证高速并发写入的情况下, 使系统有较好的查询性能; 2) 如何对存储的海量数据信息进行进一步挖掘。

### 参 考 文 献

- [1] Ding Zhi-ming, Gao Xu. A Database Cluster System Framework for Managing Massive Sensor Sampling Data in the Internet of Things[J]. Chinese Journal of Computers, 2012, 35(6): 1175-1191 (in Chinese)  
丁治明, 高需. 面向物联网海量传感器采样数据管理的数据库集群系统框架[J]. 计算机学报, 2012, 35(6): 1175-1191
- [2] Chen Qing-kui, Zhou Li-zhen. HBase-based storage system for large-scale data in wireless sensor network[J]. Journal of Com-

puter Application, 2012, 32(7): 1920-1923, 1977 (in Chinese)  
陈庆奎, 周利珍. 基于 HBase 的大规模无线传感网络数据存储系统[J]. 计算机应用, 2012, 32(7): 1920-1923, 1977

- [3] Lu Ting, Fang Jun, Qiao Yan-ke. HBase-based Real-time Storage System for Traffic Stream Data[J]. Journal of Computer Application, 2015, 35(1): 103-107, 135 (in Chinese)  
陆婷, 房俊, 乔彦克. 基于 HBase 的交通流数据实时存储系统[J]. 计算机应用, 2015, 35(1): 103-107, 135
- [4] Lu Hui-ming, Zhou Zhao, Liao Chang-bin. Historical Data Processing Based On Real-time Database System [J]. Electric Power Automation Equipment, 2012, 29(3): 127-131 (in Chinese)  
陆会明, 周钊, 廖常斌. 基于实时数据库系统的历史数据处理[J]. 电力自动化设备, 2012, 29(3): 127-131
- [5] George L. HBase: The Definitive Guide [M]. 2. Inc, USA: O'Reilly Media, 2013; 339-350
- [6] Ku W Y, Chou T Y, Chung L K. The CloudBased Sensor Data Warehouse[C]//International Symposium on Grids and Clouds and the Open Grid Forum. Taipei, Taiwan, 2011; 21-24
- [7] Carstou D, Cernian A, Olteanu A. Hadoop Hbase-0. 20. 2 performance evaluation[C]//2010 4th International Conference on New Trends in Information Science and Service Science (NISS). IEEE, 2010; 84-87
- [8] The Apache Software Foundation[OL]. <http://hadoop.apache.org>
- [9] Rabl T, Gómez-Villamor S, Sadoghi M, et al. Solving Big Data Challenges for Enterprise Application Performance Management [J]. Proceedings of the VLDB Endowment, 2012, 5(12): 1724-1735
- [10] Kallman R, Kimura H, Natkins J, et al. H-store: a high-performance, distributed main memory transaction processing system [J]. PVLDB, 2008, 1(2): 1496-1499
- [11] Lakshman A, Malik P. Cassandra: a decentralized structured storage system[J]. SIGOPS Operating Systems Review, 2010, 44(2): 35-40
- [12] <http://planetcassandra.org/nosql-performance-benchmarks>