一种基于优先级的加权公平队列调度算法

闵 捷, 周红琼, 王晓东*
（宁波大学 信息科学与工程学院, 浙江 宁波 315211）

摘要: 根据在无线传感器网络上传输实时业务的要求, 深入研究了加权公平队列调度算法. 在分析其无法保证实时业务的QoS的原因后, 提出一种基于优先级的加权公平队列调度算法P-WFQ. 这种算法利用滑动窗口将 WFQ 和基于优先级的机制结合起来, 通过有效的优先级划分策略, 在带宽受限的情况下, 保证高优先级数据的QoS要求. 仿真结果表明: P-WFQ在实时业务的时延性能上比 WFQ 有明显的改善, 而且能够将实时业务的最大时延控制在允许范围之内.

关键词: 无线传感器网络; 调度算法; 加权公平队列; 优先级
中图分类号: TP393 **文献标识码:** A **文章编号:** 1001-5132（2012）02-0042-05

为了满足日益增长的实时业务的 QoS (Quality of Service)需求, 无线传感器网络(Wireless Sensor Network, WSN)所采取的调度机制需要支持多种的业务特性[1], 同时兼顾传感器网络带宽资源有限的特点. 因此, 队列调度机制成为目前无线传感器网络的研究热点之一, 其中最著名的是加权公平队列(Weighted Fair Queueing, WFQ) [2]. WFQ 支持通用处理机共享模型(Generalized Processor Sharing, GPS) [3], GPS 系统定义了虚拟时间和虚拟结束时间, WFQ 调度算法借鉴了 GPS 的个概念, 其总是优先选择虚拟结束时间最小的数据包进行发送. WFQ 是一类公平调度算法, 所有的业务根据其平均速率分享带宽, 在无线传感器网络资源受限的情况下, 如何保障实时业务的端到到时延是要解决的一个问题.

许多研究在 WFQ 的基础上进行改进. 自同步公平排队(Self Clocked Fair Queuing, SCFQ)是 WFQ 一个简单的变体[4]. SCFQ 取数据包获取服务的时间作为虚拟起始时间. 这极大地减少了追踪虚拟时间所需的计算量. 但是与 WFQ 相比, SCFQ 的端到端时延增加了一个部分 $\sum_{i=1}^{N} M_i / \gamma^h$. 而开始时间公平排队(Start-time Fair Queueing, SFQ)是另一个公平排队的算法[5].

与 SCFQ 相比, SFQ 优先选择具有最小虚拟发送时间的数据包发送, 而不是最小虚拟结束时间. SFQ 的端到端时延比 SCFQ 略小. 最坏公平加权公平排队(Worst-case Fair Weighted Fair Queueing, WF2Q)同时使用 GPS 模型中的数据包开始时间和结束时间[6], 在系统中虚拟开始时间比实际时间小, 这样就避免了实际时间超过参考系统中的时间. WF2Q 中的端到端时延非常接近 WFQ 中的端到端时延, 而目前 WFQ 的改进算法都无法很好地解决实时业务的端到到时延问题.

笔者在此提出一种基于优先级的加权公平队列调度算法 P-WFQ, 其能保证将业务最大时延控制在允许的范围之内. P-WFQ 通过使用滑动窗口, 将 WFQ 和基于优先级的机制结合起来. 首先, 它定义了一个滑动窗口, 虚拟结束时间落在滑动窗口内的数据包被认为是符合条件的待发送数据包. 然后, 通过控制在任何给定时刻内的待发送数据包的个数, 来保证每类多媒体业务在该时刻可分配的带宽. 最后, 通过在待发送数据包内部使用基于优先级的调度机制, 来保证高优先级数据包的时延.

收稿日期: 2011−11−03. 宁波大学学报（理工版）网址: http://3xb.nbu.edu.cn
基金项目: 国家自然科学基金（61071198）; 浙江省大学生创新创业孵化项目（2010R405064, 2010R405066）.
第一作者: 闵 捷（1985−）, 女, 上海人, 在读硕士研究生, 主要研究方向: 无线传感器网络. E-mail: 184735917@qq.com
*通讯作者: 王晓东（1970−）, 男, 浙江上虞人, 副教授, 主要研究方向: 软件工程. E-mail: wangxiaodong@nbu.edu.cn

?1994-2015 China Academic Journal Electronic Publishing House. All rights reserved. http://www.cnki.net

1　P-WFQ 队列调度算法

1.1　滑动窗口机制

传统基于优先级的机制是按优先级顺序发包, 即当高于优先级 k 的队列中还有数据没有发送完毕之前, 优先级 k 队列中的数据包只能等待发送. 因此, 一个超载的高优先级队列将会降低优先级队列的信道利用率. 如果此时直接将基于优先级的机制用在 WFQ 中, 将不能保证各个队列独立的信道利用率.

因此, 我们采用的办法是改变虚拟时间的颗粒度, 这样拥有“相似”结束时间的数据包可以根据其优先级被调度. 为了定义“相似”, 我们在虚拟时间轴上引入滑动窗口. 一个滑动窗口定义了一个虚拟时间间隔, 所有虚拟结束时间落在这个窗口的数据包被认为有“相似”的虚拟结束时间. 当节点要发送数据时, 就从当前滑动窗口中选择优先级最高的数据包发送. 窗口由当前时间开始, 沿时间轴滑动, 窗口大小由调度器定义. 如果当前的窗口内没有合适的候选包, 调度器就依旧选择拥有最小虚拟结束时间的数据包发送.

图 1 详细描述了 P-WFQ 的滑动窗口机制. 假设队列 2 的优先级比队列 1 高. $F_{i,k}$ 是队列 i 的第 k 个包的虚拟结束时间. 如果使用的是 WFQ, 那么包的发送顺序为: $(P_1^1, P_1^2, P_1^3, P_2^1, P_1^4, \cdots)$, 如果使用的是 P-WFQ, 那么落入窗口内的包为 P_1^1, P_1^2, P_1^3, P_2^1. 节点从具有发送资格的候选数据包里, 选择最高优先级的包先发送, 因为队列 2 的优先级高于队列 1, 所以先发送的包是 P_2^1. 因此, 包的发送顺序为: $(P_2^1, P_1^1, P_1^2, P_1^3, P_1^4, \cdots)$.

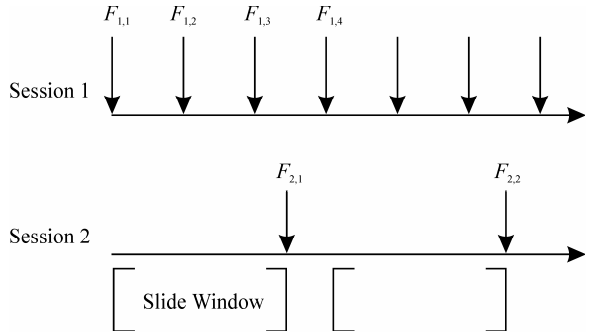

图 1　P-WFQ 中的滑动窗口机制

1.2　计算最大时延

为便于描述, 我们假设每个数据包的大小为 1 bit. 根据文献[2-3]的分析, 在 P-WFQ 中, 队列 i 的最大时延情况发生在以下几种状态:

(1) 所有队列都处于贪婪状态;

(2) 落入当前竞争窗口内的所有队列的优先级都比队列 i 高;

(3) 所有队列都满足 $g_i = \rho_i$, 即令牌产生的固定速率等于所要求的数据服务率.

假设 1 个包含 B bit 的新的数据包 β, 在时刻 t 到达队列 i. 根据漏桶模型, 最差的情况是数据包 β 之前还有 σ_i bit 的数据在队列中等待发送. 因此数据包 β 最大时延由三部分组成: 进入滑动窗口所需的时间、在滑动窗口中比队列 i 优先级高的队列中所有数据发送完的时间以及最大长度数据包发送完毕需要的时间.

首先, 计算数据包 β 进入滑动窗口所需的时间. 数据包 β 的虚拟结束时间为 $t + \sigma_i / g_i$, 为了使其落在滑动窗口内, 必须等到滑动窗口移动到 $\sigma_i / g_i - W$. 我们用 D_i^B 代表这部分时延:

$$D_i^B \leqslant \max\left(\frac{\sigma_i}{g_i} - W, 0\right). \tag{1}$$

然后计算滑动窗口内比该数据包优先级高的数据全部发送成功所需的时间. 队列 i 由比它优先级高的队列 j 引起的, 时延的计算方法如下所示.

设 HP_i 为优先级比队列 i 高的队列的集合. 定义 D_i^H 为滑动窗口内由 HP_i 导致的时延. $g_j W$ 为在滑动窗口内队列 j 待发送的比特数, 但是因为受到漏桶模型的限制, $g_j W$ 不能超过 σ_i. 设 Δt_1 为队列 i 的初始化时延, 假设滑动窗口内的数据以比特率 r 发送到链路上. 队列 i 中的数据以比特率 g_i 进入滑动窗口, 即队列 i 相对于滑动窗口的移动速度为 g_i. 因此, 我们用迭代的方法计算预期时延 Δt_1, $\Delta t_2, \cdots, \Delta t_n$, 其中, Δt_2 是发送间隔 Δt_1 内到达的比特流所导致的时延, Δt_3 是发送间隔 Δt_2 内到达的比特流所导致的时延, 以此类推.

相应算式如下:

$$D_i^H = \Delta t_1 + \Delta t_2 + \cdots + \Delta t_k = \frac{\Delta t_1}{1-q} = \frac{1}{1 - \sum_{j \in HP_i} g_j / r}\left(\sum_{j \in HP_i} \frac{\min(\sigma_j, g_j W)}{r} + \frac{\min(\sigma_i, g_i W)}{r}\right) =$$

?1994-2015 China Academic Journal Electronic Publishing House. All rights reserved.　http://www.cnki.net

$$\frac{\min(\sigma_i, g_i W) + \sum_{j \in HP_i} \min(\sigma_j, g_j W)}{r - \sum_{j \in HP_i} g_j}. \quad (2)$$

最后计算将最大长度数据包发送完毕需要的时间. 与 GPS 相比, WFQ 的区别是设置了最大包长度 $L_{\max}$. 因此, P-WFQ 和 GPS 的最大时延的差值为 $L_{\max}/r$.

因此, 在 P-WFQ 机制中, 队列 i 的最大时延为:

$$D_i^* = \max\left(\frac{\sigma_i}{g_i} - W, 0\right) + \frac{\min(\sigma_i, g_i W) + \sum_{j \in HP_i} \min(\sigma_j, g_j W)}{r - \sum_{j \in HP_i} g_j} + \frac{L_{\max}}{r}. \quad (3)$$

1.3 设置队列优先级

实时网络应用中, 我们假设所有队列的 D_i, g_i, σ_i 已知. 在 WFQ 机制中, 由于 D_i^* 和 g_i 是一一对应的, 所以可以通过调节 g_i 使得 D_i^* 接近 D_i, 但是这样会导致信道资源分配不均. 在 P-WFQ 机制中, 我们通过设置合适的优先级和滑动窗口大小来使得 D_i^* 接近 D_i.

首先, 我们选择一种简单的方法来设置队列的优先级, 即通过队列的 WFQ 时延边界和队列所要求的时延边界 D_i 的差值来设定队列优先级. 差值越大的, 优先级越高.

优先级设置方法包含三步:

(1) 通过(3)式, 计算队列 i 的 WFQ 时延边界:

$$D_i^{*(wfq)} = \frac{\sigma_i}{g_i} + \frac{L_{\max}}{r}; \quad (4)$$

(2) 计算 $D_i^{*(wfq)}$ 和 D_i 之间的差值为:

$$\Delta_i = D_i^{*(wfq)} - D_i; \quad (5)$$

(3) 将 N 个队列的 Δ_i 降序排列, Δ_i 最大的, 优先级最高.

假设队列 i 的 D_i、g_i、σ_i 已知, 而且优先级已经设置好, 我们可以通过下面不等式来求解滑动窗口的大小 W:

$$D_i^B + D_i^H + \frac{L_{\max}}{r} \leqslant D_i, 1 \leqslant i \leqslant N. \quad (6)$$

根据已经设置好的优先级 $P_1, P_2, \cdots, P_N$, 我们可以选择 $\sigma_1/g_1, \sigma_2/g_2, \cdots, \sigma_N/g_N$, 并且分别定义它们为 $S_1, S_2, \cdots, S_N$. 我们定义 N+1 个时间段为 $(0, S_1], (S_1, S_2], \cdots, (S_N, +\infty)$. 从时间段 $(0, S_1]$ 开始, 我们可以一个一个地利用(6)式来计算最大、最小值, 目的是寻找一个能同时满足 N 个不等式的 W 值; 当一个合适的 W 值找到时, 这种比较就停止. 因此寻找窗口大小的复杂度为 $O(N^2)$.

2 仿真结果及分析

我们将利用 NS-2 网络仿真平台[7]来仿真 WFQ 与 P-WFQ, 并对仿真结果进行比较. 仿真拓扑如图 2 所示. 所有的数据包都假设拥有同样的 1 KB 大小.

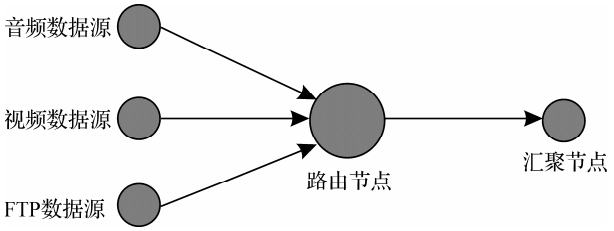

图 2 仿真拓扑图

在仿真中, 我们设置有 3 种数据. 第一类数据是带宽为 64 Kbps 的语音数据, 语音数据的队列时延必须小于 100 ms; 第二类数据是带宽为 2 Mbps 的视频数据, 视频数据的队列时延必须小于 33 ms; 第三类数据是 FTP 上传输的普通数据, 此类数据的带宽为 7.936 Mbps, 队列时延必须小于 200 ms.

2.1 计算优先级与窗口大小

仿真参数见表 1.

根据表 1 的仿真参数, 在开始仿真之前, 我们要确定 2 个重要参数: 数据优先级和滑动窗口大小.

表 1 系统仿真参数

	D_i /ms	$D_i^{*(wfq)}$ /ms	σ_i /byte	源速率/Mbps	ρ_i /Mbps	g_i /Mbps	ϕ_i /%
音频	100	128.82	1 024	0.32	0.064	0.064	0.64
视频	33	17.20	4 096	10	2	2	20
FTP 数据	200	66.07	65 536	40	7.936	7.936	79.36

?1994-2015 China Academic Journal Electronic Publishing House. All rights reserved. http://www.cnki.net

优先级大小根据(5)式的差值计算法可得, 其结果见表 2.

表 2　优先级参数表

	差值/ms	优先级
音频	28.82	3
视频	-15.80	2
FTP 数据	-133.94	1

2.2　WFQ 与 P-WFQ 仿真结果及分析

为比较 P-WFQ 在不同滑动窗口下的性能, 我们分别选择 3 种大小的窗口进行测试, 分别为 50 ms、30 ms 和 10 ms.

图 3、图 4 和图 5 分别列出了在 3 种滑动窗口下所测量到的时延值. 在图 3 中, 滑动窗口的大小为 50 ms, 我们可以看到音频数据的时延仅为 1 个数据包传送的时间. 在图 4 中, 当滑动窗口大小为 30 ms 时, 三类数据时延要求均能得到满足. 在图 5 中, 当窗口大小为 10 ms 时, 音频数据的时延要求无法得到满足. 这些实验结果可以说明, 通过设置合适的滑动窗口大小, 可以有效地控制多媒体数据的时延上限.

表 3 将 P-WFQ、WFQ 和理论值最大时延进行了对比. 从表 3 中可以看出, WFQ 对时延的保障十分宽松, 特别是音频数据的时延性能保障, 音频的仿真时延值明显高于理论值. P-WFQ 对时延性能的保障也明显优于 WFQ, 可以通过设置合适的滑动窗口大小, 使得最大时延值满足多媒体业务的需求.

3　结论

提出了一种基于优先级的加权公平队列调度算法 P-WFQ. 通过在数据包传输的虚拟时间内引入滑动窗口的概念, P-WFQ 在 WFQ 调度算法中融入了静态优先级区分策略. 利用滑动窗口调节虚拟时间间隔, P-WFQ 将由优先级驱动的机制和由权值驱动的机制结合起来. 与 WFQ 相比, P-WFQ 可以使多媒体数据的最大时延独立于它所占的信道带宽. 因此, P-WFQ 是一种有效的资源分配的调

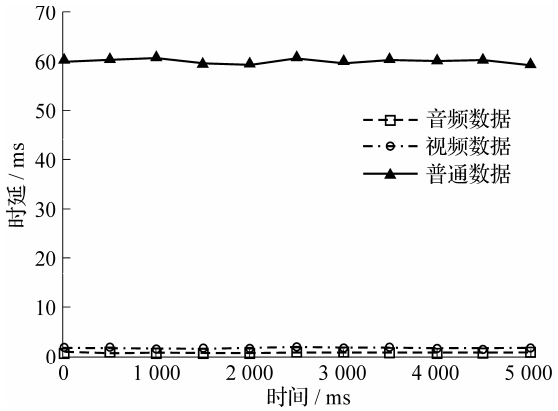

图 3　窗口大小为 50 ms 时的最大时延

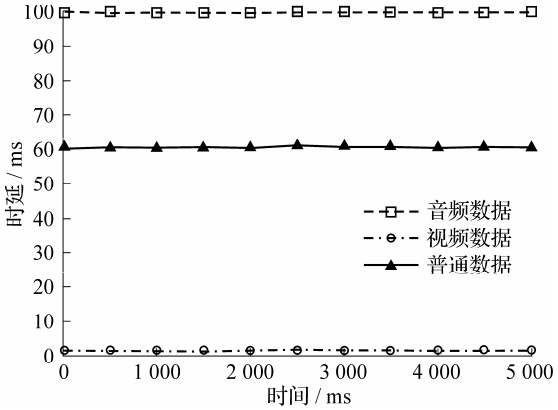

图 4　窗口大小为 30 ms 时的最大时延

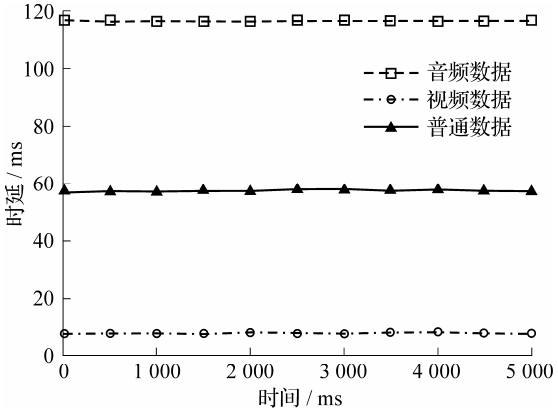

图 5　窗口大小为 10 ms 时的最大时延

表 3　P-WFQ、WFQ 和理论值最大时延对比　　ms

	P-WFQ(50ms)	P-WFQ(30ms)	P-WFQ(10ms)	WFQ	理论值
音频	0.82	98.94	118.78	127.12	100
视频	1.638	1.638	7.373	15.56	33
普通数据	59.65	58.85	56.84	54.92	200

?1994-2015 China Academic Journal Electronic Publishing House. All rights reserved.　http://www.cnki.net

度算法. 仿真实验证明, P-WFQ 在保障带宽方面的性能与 WFQ 十分接近, 在保障最大时延方面的性能远远优于 WFQ.

参考文献:

[1] 任丰源, 黄海宁, 林闯. 无线传感器网络[J]. 软件学报, 2003, 14(7):1282-1291.

[2] Demers A, Keshav S, Shenker S. Analysis and simulation of a fair queuing algorithm[J]. International Research and Experience, 1990, 1(1):3-26.

[3] Parekh A, Gallager R. A generalized processor sharing approach to flow control: The single node case[J]. ACM/IEEE Trans On networking, 1993, 1(3):344-357.

[4] Golestani S. A self-clocked fair queueing scheme for broadband applications[C]//Proc IEEE INFOCOM'94, 1994:636-646.

[5] Goyal P, Vin H, Haichen Cheng. Start-time fair queuing: a scheduling algorithm for integrated services packet switching networks[C]//Proc IEEE/ACM SIGCOMM'96, 1996:690-704.

[6] Bennett J, Hui Zhang. WF2Q: Worst-case fair weighted fair queueing[C]//Proc IEEE INFOCOM'96, 1994:120-128.

[7] 徐雷鸣, 庞博, 赵耀. NS 与网络模拟[M]. 北京: 人民邮电出版社, 2003.

A Priority-based Weighted Fair Queue Scheduling Algorithm

MIN Jie, ZHOU Hong-qiong, WANG Xiao-dong*

(Faculty of Information Science and Engineering, Ningbo University, Ningbo 315211, China)

Abstract: Based on the requirements of the transmission in real-time service in wireless sensor networks, the WFQ algorithm is studied in this paper. Having analyzed the reason why WFQ cannot ensure the quality of real-time service, it proposes P-WFQ, which is a priority-based Weighted Fair Queuing algorithm. With sliding windows, the P-WFQ combines both the WFQ and the priority-based mechanism. Under the condition of limited bandwidth in the link, using the proposed method to assign priority, the quality of high-priority data is ensured. Simulation shows that the P-WFQ can solve the delay problem in real-time service, and confine the worst-case within the bound as required by the real-time service.

Key words: wireless sensor network; queueing algorithm; weighted fair queueing; priority

（责任编辑　章践立）

?1994-2015 China Academic Journal Electronic Publishing House. All rights reserved. http://www.cnki.net