

一种用于 IEEE 802.16 无线城域网 TDD 模式中的带宽调度方案

王洪熙^① 陈剑峰^② 焦文华^② 刘勇^①

^①(北京邮电大学 通信网络综合技术研究所 北京 100876)

^②(朗讯科技 贝尔实验室中国基础科学研究院 北京 100080)

摘要 该文提出了一种用于 IEEE 802.16 宽带无线接入系统 TDD 模式下的公平而有效的带宽分配调度体系。与该领域中传统的固定带宽分配方式相比,该文提出的调度体系结构综合考虑了上下行链路不同业务带宽需求并进行动态带宽分配。该文提出一种新的亏空公平优先级队列(DFPQ)算法来调度不同优先级的业务流,可以为系统提供更好的公平性。仿真结果显示该文提出的调度体系结构能够很好地满足所有类型业务的服务质量(QoS)需求,并提供较好的公平性。

关键词 IEEE 802.16, 宽带无线接入, 带宽分配, 调度算法, 服务质量

中图分类号: TN915.142

文献标识码: A

文章编号: 1009-5896(2006)05-0789-06

A Bandwidth Scheduling Architecture for IEEE 802.16 Wireless MAN in TDD Mode

Wang Hong-xi^① Chen Jian-feng^② Jiao Wen-hua^② Liu Yong^①

^①(School of Telecommunication & Network technology, BUPT, Beijing 100876, China)

^②(Bell Labs Research China, Beijing 100080, China)

Abstract A fair and efficient bandwidth allocation scheduling architecture for IEEE 802.16 Broadband Wireless Access (BWA) systems in TDD mode is proposed. In this area, comparing with the traditional fixed bandwidth allocation, the proposed scheduling architecture allocates uplink and downlink bandwidth simultaneously and dynamically. A Deficit Fair Priority Queue (DFPQ) algorithm is proposed to schedule different service flow, which provides more fairness to the system. The simulation results show the proposed architecture can meet the QoS requirement and fairness for all types of traffic.

Key words IEEE 802.16, Broadband Wireless Access(BWA), Bandwidth allocation, Scheduling, QoS

1 引言

2002年4月, IEEE发布了针对宽带无线接入(Broadband Wireless Access, BWA)系统的IEEE 802.16标准^[1-2], 定义了城域网(Metropolitan Area Network, MAN)范围内10-66GHz的频段上提供高速无线接入服务的标准。IEEE 802.16标准定义了一种点到多点的拓扑结构, 由一个控制基站BS来连接多个用户站SS, 各个SS可以通过BS互相通信并与各种公共网络相连。

在IEEE 802.16中, 定义了频分双工(Frequency-Division Duplexing, FDD)和时分双工(Time-Division Duplexing, TDD)两种工作模式。在两种模式中, TDD模式的优点在于在每一帧中的下行链路(从BS到SS)和上行链路(从SS到BS)共享同一个频率的信道。每个通信帧由一个下行链路子帧和一个上行链路子帧组成, 两个子帧的长度比例可以根据当前的上

下行业务量的具体情况由BS动态地划分。因此在本文中只考虑TDD模式。

IEEE 802.16标准定义了一种面向连接的MAC层协议, 为BS和SS之间的各种业务提供保证服务质量(Quality of Service, QoS)的机制, 包括连接建立、带宽请求和带宽分配等等。在MAC层中, 按照不同业务对QoS的需求定义了4种类型的服务流: (1)主动授予服务(Unsolicited Grant Service, UGS)流, 用于传输固定比特率的业务; (2)实时轮询服务(real-time Polling Service, rtPS)流, 用于传输周期性的产生可变尺寸数据包的实时业务; (3)非实时轮询服务(non-real-time Polling Service, nrtPS)流, 用于传输非周期的产生可变尺寸数据的非实时业务; (4)尽力而为(Best Effort, BE)服务流。每类服务流都有相应的一些关键服务参数, 如最小保留业务速率(Minimum Reserved Traffic Rate), 最大可支持的业务速率(Maximum Sustained traffic rate), 业务优先级(Traffic Priority), 最大时延(Maximum Latency)等。所有的这些参数通过动态服务流管理消息来传递。通过一系列被称为动态服

务添加(Dynamic Service Addition, DSA)、动态服务更改(Dynamic Service Change, DSC)和动态服务删除(Dynamic Service Deletion, DSD)的 MAC 层消息序列, 服务流可以被创建、更改或者删除。BS 将为每个连接记录相应的服务参数作为调度的依据。但是, 标准中并没有对带宽调度和管理的具体和具体实现予以说明。

文献[3,4]中提出了在 BWA 系统中 TDD 模式下的几种带宽分配方案。但是, 两篇文章的作者都仅仅考虑了上行链路子帧的调度, 并且默认为上行链路子帧和下行链路子帧的长度是按照一定的比例固定不变的。例如, 在文献[4]的仿真中, 上行链路和下行链路的带宽就是按各自占 50% 来划分的。然而在现实的传输中, 这种情况并不会经常发生。另外, 以上的方案都是采用完全的优先级队列(Priority Queue, PQ)调度算法来调度各个类型的业务, 当高优先级业务的业务量很大时, 低优先级的业务将被饿死。

本文提出一种带宽分配体系结构, 能够为 IEEE 802.16 标准中定义的不同类型的应用提供 QoS 支持。通过对上行和下行链路带宽的动态调度, 带宽能够得到更有效的利用。此外, 与完全的优先级队列(PQ)算法相比, 本文提出的亏空公平优先级队列(DFPQ)算法, 可以在公平性方面有较大的提高。

本文按照以下顺序组织: 第 2 节介绍整个分层带宽调度体系结构, 第 3 节详细地描述 DFPQ 调度算法, 第 4 节给出了按照该调度体系结构所进行的仿真结果, 第 5 节结束语。

2 分层带宽调度体系结构概述

IEEE 802.16 的 MAC 层为每个业务都建立一个连接和相应类型的服务流(UGS, rtPS, nrtPS, 或者 BE)。BS 为每个连接分配一个唯一的连接标识符(Connection ID, CID)用于该连接在上行链路或下行链路进行数据传输的标识。如图 1 所示, 当一个新的服务流创建或者一个已有的服务流更改参数时, 该服务流将发送请求消息给 BS。接着由 BS 上的接入控制模块决定是否批准该请求; 如果批准该请求, 则由接入控制模块通知调度模块根据请求消息中的参数值来更改当前的调度参数。请求被批准的前提是确保已有的业务的服务质量不会下降, 同时该新请求所要求的服务质量能够被满足。

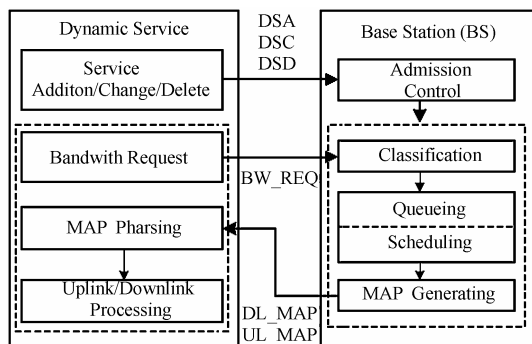


图1 BS 和 SS 的模块设置
Fig.1 Module diagram of BS and SS

本文中采用一种简单的接入控制方法, 即通过估算当前系统可用的剩余带宽来判断是否批准请求。具体的方法是, 当接入控制模块收到服务流的创建、更改或者删除请求时, 将统计所有已有的连接的最小保留业务速率, 估计出系统的剩余可用带宽(C_a)。剩余可用带宽的计算公式为

$$C_a = C_{total} - \sum_{i=1}^I \sum_{j=1}^{J_i} r_{min}(i, j) \quad (1)$$

其中 C_{total} 是无线链路的总容量, I 是服务类型种类的数量, J_i 是第 i 种服务类型的服务流的数量, $r_{min}(i, j)$ 是第 i 种业务流的第 j 个连接的最小保留业务速率。

如果接入控制模块所收到的 DSA 或 DSC 请求中提出的最小保留业务速率参数值小于 C_a , 则该请求被批准, 否则该请求被拒绝。另外, 如果在请求消息中最小保留业务速率参数值为零, 则该请求总是被批准, 但该连接的 QoS 将不会被保证。

本文提出一种用于 IEEE 802.16 TDD 模式下, 可以支持所有类型的服务流的分层带宽分配调度体系。如图 2 所示, 在该体系结构中, 按照传输方向(上行链路或下行链路)和服务类型(rtPS, nrtPS 和 BE)作为划分优先级的依据定义了 6 个队列。由于 UGS 服务是被分配固定的带宽(或者是固定的时间宽度), 无需进行调度, 所以在每次调度之前直接切掉 UGS 服务所占用的带宽。

来自各个连接的带宽请求根据 CID 和服务流类型进行分类, 并按照所属的服务流类型放到相应的队列中进行排队。调度模块将按照调度算法从相应队列中取出带宽请求, 进行带宽分配, 然后根据带宽分配结果创建 UL-MAP 消息和 DL-MAP 消息。

第 1 层调度采用 DFPQ 算法对 6 个队列的进行调度, 确定当前被调度的队列。有关 DFPQ 算法的具体描述将在第 3 节中给出。

第 2 层调度中, 采用 EDF(Earliest Deadline First)算法^[5]来为 rtPS 队列中的请求进行排队, 采用加权公平排队(Weighted Fair Queue, WFQ)算法^[6]来为 nrtPS 队列中的请求进行排队, 采用 RR(Round Robin)算法^[7]来为 BE 队列中的请求进行排队。

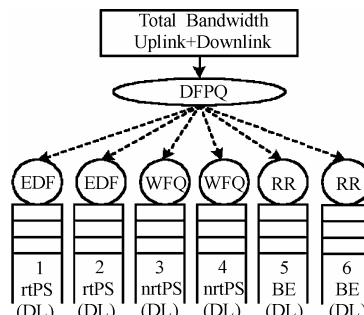


图2 带宽分配分层调度结构
Fig. Hierarchical structure of band width allocation

3 亏空公平优先级队列(Deficit Fair Priority Queue, DFPQ)算法

本文提出一种基于优先级队列的亏空公平优先级队列算法,用于调度 IEEE 802.16 支持的各种 QoS 业务。各个业务的优先级按照下面两个规则来规定:

- (1)服务类型的优先级排列: rtPS > nrtPS > BE;
- (2)传输方向的优先级排列: 下行链路 > 上行链路。

设置下行链路服务的优先级比上行链路服务高的原因是,作为一个中央集中调度的体系结构,BS 需要尽可能快速的中继数据包以避免 BS 的缓存溢出,以及保证时延需要。

基于以上两个规则,设置了6个类型服务的优先级权值,在表1中给出。权值越小的类型,优先级越高。

表1 各种业务类型的优先级

Table1 Priority of each service class

下行	上行	下行	上行	下行	上行
-rtPS	-rtPS	-nrtPS	-nrtPS	-BE	-BE
1	2	3	4	5	6

在文献[4]中,作者采用完全的优先级队列(PQ),即总是服务完高优先级的队列再服务低优先级的队列。需要注意,完全的优先级服务的一个缺点是,高优先级的服务可能会占用所有的带宽,导致低优先级的服务被饿死。而本文中提出的 DFPQ 算法则克服了这个问题,为所有类型的服务提供了更好的公平性。

在 BS 中维护有一个队列活动状态清单。DFPQ 只对处于活动状态清单中的队列进行调度。如果一个队列非空,它将留在活动状态清单中;否则,从活动状态清单中除去。在每一轮调度中,高优先级的队列将总是被先服务。

本文提出的DFPQ算法参考了DWRR(Deficit Weighted Round Robin)算法^[8]。在DFPQ中,每个队列定义一个亏空计数器(DeficitCounter),DeficitCounter的值根据相应队列的量子

值(quantum)来增加。调度程序轮询每个非空队列,并根据 DeficitCounter 的值来决定是否服务该队列头部的数据包。在本文中,队列中排列的是各个连接的带宽请求。调度程序依次访问活动状态清单中的队列(这些队列都肯定非空),确定每个队列头部第1个带宽请求所要求的带宽数量。在每个队列首次被服务时,DeficitCounter要加上该队列的quantum值。如果当前队列的DeficitCounter大于零,则位于当前被访问队列头部的带宽请求将总是被批准,同时DeficitCounter更新为原DeficitCounter 减去被批准的请求所要求的数据值,然后访问该队列中的下一个带宽请求;否则,如果DeficitCounter的值小于等于零,或者当前队列已经为空,则调度程序按照优先级排列接着访问下一个队列。当一个队列为空时,该队列的DeficitCounter值被清零,同时该队列从活动状态清单中清除。

每个队列的quantum值确定方法如下。设Quantum[i]为第i个队列的quantum值,则

$$Quantum[i] = \sum_{j=1}^{J_i} r_{max}(i, j) \tag{2}$$

其中 J_i 是第i个服务类型的服务流的数量, $r_{max}(i, j)$ 是第i种服务流的第j个连接的最大支持的业务速率。如果某个服务流没有相应的 r_{max} 参数,则在公式中用该服务流的 r_{min} 参数值来代替。

在带宽足够可用的情况下,当前被访问的队列头部的带宽请求将总是被批准。当剩余带宽不能满足当前被审批的带宽请求时,由于IEEE 802.16支持分段操作,则当前被审批的带宽请求可以通过分段操作被部分满足。

下面以图3为例说明 DFPQ算法。

第1步 更新初始参数 如果一个服务类型的带宽请求队列为空,则将该队列从活动状态清单中清除。如果一个处

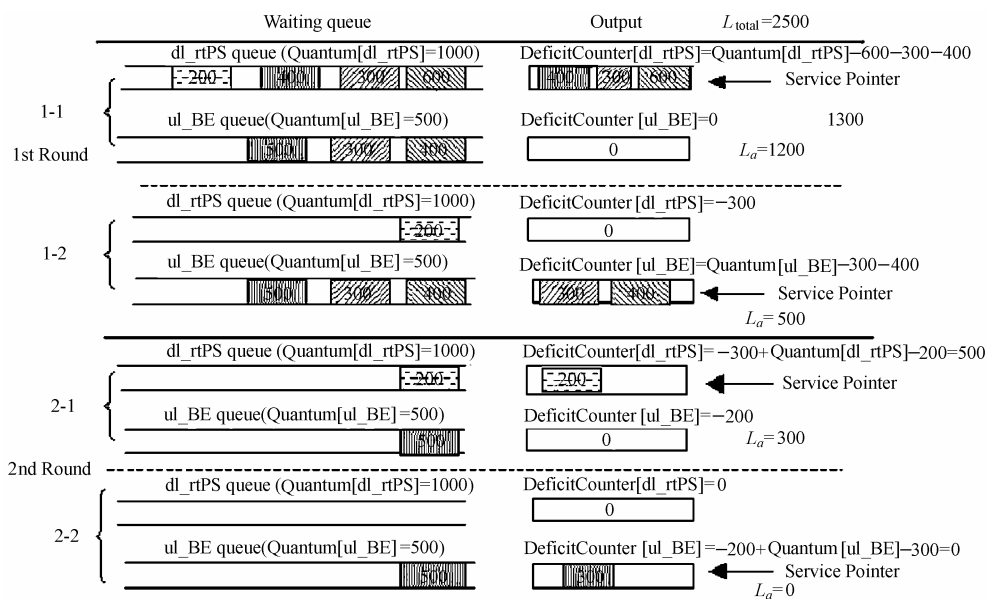


图3 DFPQ调试算法举例

Fig.3 An example of DFPQ scheduling

于空闲状态的队列有带宽请求到达,则把该队列添加到活动状态清单中去。活动状态清单中的队列将按照表2所示的优先级排队。在本例中,只有下行rtPS队列和上行BE队列非空,所以活动状态清单中只有这两个队列。

接着更新每个队列的参数。 L_a 的初始值是: $L_a = L_{total}$,在本例中,该值为1000bit。第*i*个队列的DeficitCounter[*i*]的初始值是:DeficitCounter[*i*]=Quantum[*i*]。每轮开始时,对DeficitCounter进行更新,使DeficitCounter[*i*]=DeficitCounter[*i*]+Quantum[*i*]。其中DeficitCounter_{*i*}指在第*i*轮的DeficitCounter值。

第2步 从最高优先级的队列开始按照优先级依次访问活动状态列表中的各个队列,直到以下条件之一发生:(1)DeficitCounter[*i*] ≤ 0; (2)该类型带宽请求队列为空;(3)已经没有剩余带宽了,即 $L_a \leq 0$;(4)已经到了MAP消息要发送的时间。

在本例中,第1轮开始,首先访问下行rtPS队列。由于Quantum[dl_rtPS]=1000,设置DeficitCounter[dl_rtPS]=1000。在下行rtPS服务队列头部的带宽请求所要求的带宽是600bit,则该带宽请求被批准并出队。同时DeficitCounter[dl_rtPS]减去600bit,得到一个新值400。现在,排在下行rtPS队列头部的带宽请求所要求的带宽为300bit,则该请求也被批准并出队,同时DeficitCounter[dl_rtPS]减去300,剩余值为100。以此类推,下行rtPS队列中的带宽请求将被继续批准,直到DeficitCounter[dl_rtPS]= -300(小于零)。

第3步 如果在第2步中规定的条件(1)或者条件(2)发生,回到第2步继续服务其它较低优先级的队列。如果没有更低优先级的队列了,则回到第1步开始新一轮的调度。

在本例中,当DeficitCounter[dl_rtPS]= -300时,调度算法停止访问下行rtPS队列,接着开始访问上行BE队列。直到DeficitCounter[ul_BE]<0。此时没有更低优先级的队列了,而还有可用带宽,则回到第1步开始第2轮调度。

在第2轮中,仍然是下行rtPS队列拥有最高优先级。首先,更新DeficitCounter[dl_rtPS]的值等于原有值 - 300加上下行rtPS队列quantum值1000,得到新值700。接着排在队列头部的带宽请求(请求的带宽为200)被批准,同时DeficitCounter[dl_rtPS]减去200,剩余值为500。此时条件(2)发生,下行rtPS队列中已没有新的请求,则DeficitCounter[dl_rtPS]置零,同时调度程序开始访问下一个队列。

第4步 如果在第2步中规定的条件(3)或者条件(4)发生,则将MAP消息发送,同时结束当前帧的调度。回到第1步准备下一帧的调度。在本例中,在第2轮调度结束后,满足条件(3)。所以,这一帧的调度结束。要注意的是,在最后一次调度中,上行BE队列中的带宽请求为500bit,而此时调度程序仅能为该请求提供300bit的带宽,因此SS将对数据包执行分段操作,首先发送前300bit,剩余的200bit将在下一帧中被传输。

从上面的例子可以看出,在每一轮调度中,都是优先级

高的队列首先被服务,直到分配给该队列的量值被用完。所以,这是一种基于优先级队列的调度算法。另一方面,由于分配给高优先级业务的量值不一定是充足的(即可能亏空),因此较低优先级的业务也有被服务的机会。与完全的优先级队列相比,本文提出的算法对于较低优先级的业务会更加公平,这也是把这种算法称为亏空公平优先级队列的原因。

4 仿真结果与分析

本文使用MATLAB作为仿真工具来验证提出的调度算法体系结构。在仿真中设定了如下的仿真参数环境:

(1)总的带宽为100Mbps,每一帧的长度 f 为1ms,则每一帧的带宽 L_{total} 为100kbit。

(2)总共设置了6种类型共18个业务源连接,每个连接都有相应的平均速率、最大时延、最大支持速率、最小保留速率等QoS参数,如表2所示。由于在实际情况下,下行的业务量一般要大于上行的业务量,因此仿真中输入的下行链路业务量是60Mbps,上行链路业务量是40Mbps。

(3)假设所有的数据包的到达时间都发生在一帧的开始,每个连接产生的数据流量都服从相应 λ 值速率的泊松(Poisson)分布(其中 λ 取值为该连接的平均速率)。

(4)对于每种服务类型的队列量值(quantum),按照式(2)计算得到。

4.1 动态带宽分配和固定带宽分配策略的比较

在该比较中,吞吐量和业务量速率是要研究的性能指标,两者的计算方式如下:

设系统总共调度的 N 帧数据,则第*i*个连接的吞吐量 Th_i 为

$$Th_i = \sum_{j=1}^N \text{map}_{ij} \quad (3)$$

其中 map_{ij} 为连接*i*在第*j*帧被分配的带宽值。

每个连接的业务量速率 Tr_i 为

$$Tr_i = Th_i / t_d \quad (4)$$

其中 t_d 为调度 N 帧的总时间长度。

图4比较了分别采用动态分配策略(即图2所示的将上行链路和下行链路带宽进行综合分配)和采用固定比例(50Mbps带宽给上行链路,50Mbps带宽给下行链路)情况下的吞吐量性能。

图4(a)显示了上行nrtPS业务的到达曲线和被服务曲线,图4(b)显示了下行nrtPS业务的到达曲线和被服务曲线。从图中可以看出,在上行链路,两种带宽分配方式的服务曲线都能够很好地与业务到达曲线相吻合(实际上,采用固定带宽方式的服务曲线效果更好,因为采用该方式上行链路的带宽充足)。然而,对于固定带宽分配方式来说,尽管上行链路业务被很好地服务,并且还有部分带宽剩余,但是由于下行链路的业务量已经超过了被分配的固定带宽50Mbps,因此下行链

表2 业务源参数
Table 2 Input traffic

CID	类型	平均速率(kbit/ms)	量值(kbit)	最大时延(ms)	最大支持速率(kbit/ms)	最小保留速率(kbit/ms)
1	DL_rtPS	10	36	60	12	8
2		10		40	12	8
3		10		20	12	8
4	UL_rtPS	7	24	70	8.4	5.6
5		7		50	8.4	5.6
6		6		30	7.2	4.8
7	DL_nrtPS	6	21.6	120	7.2	4.8
8		6		80	7.2	4.8
9		6		60	7.2	4.8
10	UL_nrtPS	4	14.4	140	4.8	3.2
11		4		150	4.8	3.2
12		4		60	4.8	3.2
13	DL_BE	2	4.8	240	-	1.6
14		2		200	-	1.6
15		2		160	-	1.6
16	UL_BE	2	3.2	280	-	1.6
17		1		240	-	0.8
18		1		200	-	0.8

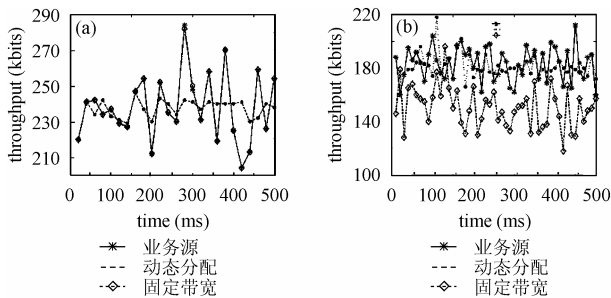


图4 nrtPS业务的吞吐量

- (a)上行链路中的nrtPS业务吞吐量
- (b)下行链路中的nrtPS业务吞吐量

Fig.4Throughput of nrtPS service

- (a)Throughput of nrtPS services in uplink channel
- (b)Throughput of nrtPS services in downlink channel

路的nrtPS和BE业务的服务速率不能满足他们的最小保留业务速率。而本文提出的调度体系结构因为是对上下行链路带宽进行综合分配,在下行链路业务量较大的情况下可以动态地为其分配更多的带宽,因此其在下行链路的性能明显优于固定带宽的方式。

4.2 DFPQ与PQ的公平性比较

一种公平的调度算法应该能够保证所有业务的最小保留业务速率,即使系统中出现违约的业务;同时当业务的流量符合漏桶限制时,业务的最大时延要求也能够被保证。这里的违约业务是指该业务的实际流量超过了其预定的最大

流量。

本文对两种第1层调度算法(本文所提出DFPQ与优先级队列(PQ)算法)进行了仿真比较,仿真中的接入控制按照前文第2节所述规则设置。图5为当其它参数均按照表2规定不变,而nrtPS业务的实际流量从20kbit/ms增加到50kbit/ms(即总的业务量从90kbit/ms增加到120kbit/ms)时的各种业务吞吐量(图5(a))和平均等待时延(图5(b))结果。如图所示,两种调度算法都能很好的满足rtPS业务的需求,但是当总的业务量达到110kbit/ms,超过系统带宽(100kbit/ms)时,PQ调度下的BE业务将由于得不到带宽被饿死;而采用DFPQ调度的nrtPS业务达到其申请的最大支持业务速率就不给其分配带宽了,使BE业务能够保证其最小保留业务速率。在平均等待时延方面,当业务量超过系统带宽后,采用DFPQ的情况下各种类型的业务都能将时延保持在可接受的范围内;而采用PQ调度的结果是, BE业务由于得不到带宽而时延趋于无穷大。

设所有rtPS业务源的流量总和为 S_{rtps} ,所有BE业务源的流量总和为 S_{be} ,则定义rtPS业务与BE业务之间的公平性指标 $FAIR_{r-b}$ 为

$$FAIR_{r-b} = \left| \frac{Th_{rtps}}{S_{rtps}} - \frac{Th_{be}}{S_{be}} \right| \quad (5)$$

$FAIR_{r-b}$ 的值越接近于0,则两者间的公平性越好;反之, $FAIR_{r-b}$ 越接近于1,两者间的公平性越差。

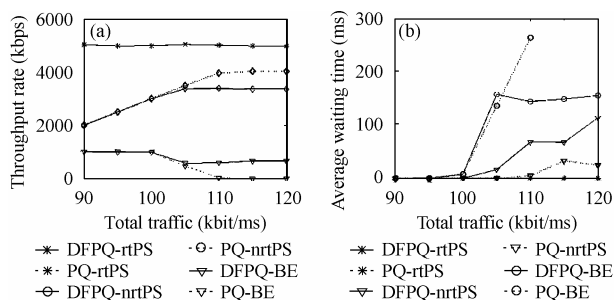


图5 在nrtPS业务增加时的吞吐量和平均等待时延

(a)nrtPS业务增加时的吞吐量变化

(b)nrtPS业务增加时的吞吐量和平均等待时延变化

Fig.5 Throughput rate and average waiting time change when nrtPS services rise

(a) Throughput rate change when nrtPS services rise

(b) Average waiting time change when nrtPS services rise

图6比较了当其它参数均按照表2规定不变, 而rtPS业务的实际流量从40kbit/ms增加到70kbit/ms(即总的业务量从90kbit/ms增加到120kbit/ms)时分别采用两种算法的公平性指标变化。从图中可以看出, 当总的业务量低于100kbit/ms(即没有违约的连接)时, 两种调度算法的公平性相同。当总的业务量超过100kbit/ms(即出现违约的连接)时, DFPQ算法能将公平性指标保持在0.2以下; 相反, 采用PQ算法时, 由于BE业务被饿死, 公平性指标很差。

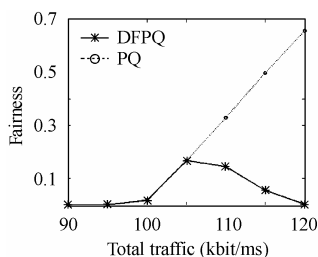


图6 公平性比较

Fig.6 Fairness comparison

5 结束语

本文提出了一种用于IEEE 802.16标准中TDD模式下的二层带宽调度体系结构, 实现了上下行链路带宽的动态分

配。与固定上下行链路带宽分配的方式相比, 该方案提高了在上下行业务量不对称时的吞吐量性能。另外, 与完全优先级队列PQ算法相比, 本文提出的DFPQ算法避免了低优先级业务被抢占资源而饿死的情况, 在系统公平性上有显著的提高。

参考文献

- [1] IEEE 802.16 Standard-Local and Metropolitan Area Networks-Part 16. IEEE Draft P802.16/D3 - 2001.
- [2] IEEE 802.16 Working Group on Broadband Wireless Access. <http://wirelessman.org>.
- [3] Hawa M, Petr D W. Quality of service scheduling in cable and broadband wireless access systems, Tenth IEEE International Workshop on Quality of Service, Miami USA, 2002: 247 - 255.
- [4] Wongthavarawat K, Ganz A. Packet scheduling for QoS support in IEEE 802.16 broadband wireless access systems. *International Journal of Communication Systems*, 2003, 16: 81 - 96.
- [5] Georgiadis L, Guerin R, Parekh A. Optimal multiplexing on a single link: delay and buffer requirements. *Proceedings of IEEE INFOCOM 94, Toronto Canada, 1994*, 2: 524 - 532.
- [6] Demers A, Keshav S, Shenker S. Analysis and simulation of a fair queuing algorithm. *ACM SIGCOMM Computer Communication Review, Symposium Proceedings on Communications Architectures & Protocols, Volume 19 Issue 4*.
- [7] Hahne E L, Gallager R G. Round robin scheduling for fair flow control in data communication networks, *International Conference on Communications, June 1986*: 103 - 107.
- [8] Shreedhar M, Varghese G. Efficient fair queuing using deficit round robin, *Proc. ACM SIGCOMM'95, October 1995*, 25(4): 231 - 242.

王洪熙: 男, 1978年生, 硕士研究生, 研究方向为无线宽带通信。

陈剑峰: 男, 1977年生, 研究员, 研究方向包括下一代 Internet 网。VoIP 和无线城域网。

焦文华: 男, 1975年生, 高级研究员, 研究方向为无线及光网络协议的设计、扩展频谱技术等。

刘勇: 男, 1962年生, 教授, 研究方向为移动通信与多媒体通信。