

主动队列管理机制研究^{*}

Researches on Active Queue Management

李方敏^{1,2} 叶澄清² 李仁发¹

(湘潭工学院计算机系 湘潭 411201)¹(浙江大学计算机系 杭州 310027)²

Abstract Currently, there are two methods suggested to handle congestion. The first approach uses packet or link scheduling on multiple logical or physical queues to explicitly reserve and allocate output bandwidth to each class of traffic. The second approach, called Active Queue management, uses advanced packet queuing disciplines on an outbound queue of a router to actively avoid congestion with the help of cooperative traffic sources. On the basis of analyzing the two approaches, in order to efficiently support flow-controlled continuous multimedia flows, we compare and analyze RED, FRED, CBT, D-CBT. Finally, we point out the shortcomings and the next research topic.

Keywords Congestion control, Link scheduling, Active queue management, Multimedia

1. 引言

随着 Internet 的日益发展, 它已不再是一个单纯的数据通信网络, 诸如 IP 电话、视频会议、VoD 等应用已开始在 Internet 上运行, 而这些多媒体应用导致业务容量和活跃流数目戏剧性地增长, 也使得路由器的拥塞控制和公平性机制显得越来越重要。多媒体应用有自己的特点, 对延迟界限和延迟变化相当敏感, 路由器机制如何有效地同时支持多媒体数据流、TCP 流以及其它业务流成为目前业界一个研究热点。图 1 给出了目前主要的路由器队列机制。

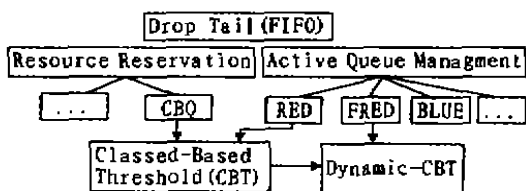


图 1 路由器队列机制

目前处理拥塞控制一般有两种方法: 第一种方法是在多个物理或逻辑输出队列上, 使用分组或链路调度方法^[1], 为每个业务类 (traffic class) 明确预留或分配输出带宽, 每个类可以是单个流或相似流的聚合, 公平队列算法以及基于类的队列算法 (CBQ) 都是采用这

种思想。结合接入控制, 这种机制不仅能解决拥塞问题, 而且能对多媒体业务提供潜在的性能保证。但这种方法明确的资源预留机制改变了 Internet 传统的基于尽力传送的特性以及对公平性的传统定义, 因此需要对网络管理和记帐机制作相应改变。第二种方法是主动队列管理机制, 不同于传统的基于 FIFO 的裁尾队列方式, 它采用高级的分组队列机制, 与业务源协作一起主动地处理拥塞。例如, TCP 将分组丢失作为网络拥塞指示, 它的 AIMD (加法增加、乘法减少) 拥塞控制机制减少发送方注入网络的数据。

下面我们将分析比较四种不同的主动队列机制: RED^[2]、FRED^[3]、CBT^[4]、D-CBT^[5]。

2. RED 算法

最早的主动队列机制是 RED (Random Early Detection), 它监视输出队列, 根据平均队列长度和最大、最小阈值的比较来随机选择包丢失, 或采用 ECN (明确的拥塞通知) 机制, 当发送方检测到包丢失或 ECN, 则减慢发送速度。在 RED 算法中, 用三个参数配置: red_min、red_max、p_max, 其工作方式如图 1 所示, x 轴表示平均队列长度, y 轴表示丢包概率, red_min、red_max 将 x 轴分成三个区间 [0, red_min]、[red_min, red_max]、[red_max, ∞], 分别对应正常操作、拥塞避免、拥塞控制三个阶段, 在正常操作阶段, 平均队列长

^{*} 本文受国家自然科学基金资助 (编号: 69974031)。李方敏 博士, 副教授, 研究方向为网络服务质量、新型网络协议、多媒体通信技术。叶澄清 教授, 博导, 研究领域为高性能体系结构、计算机网络、多媒体应用等。李仁发 教授, 研究方向为分布式计算、虚拟技术、智能控制。

度小于 red_min , 丢包概率为 0; 在拥塞避免阶段, 平均队列长度介于两个阈值之间, 其丢包概率是 p_{max} 的函数, 队列管理通过丢包或采用明确的拥塞通知 (ECN) 来告诉发送方减慢发送速率, 从而减少平均队列长度; 当平均队列长度超过 red_max 时, 丢包概率或 ECN 标记概率为 1。

RED 不能有效地保护 TCP 流, 没有拥塞控制或采用比 TCP 更贪婪的非 TCP 流将比 TCP 流攫取更多的网络带宽, 最坏情况下, 没有拥塞控制的非 TCP 流将独占所有的输出带宽。这种不公平性的主要原因是在拥塞发生时非 TCP 流不降低发送速率或降低的程度比 TCP 少, 而 RED 对所有的流都采用同样的丢包比率, 导致采用 AIMD 拥塞控制的 TCP 流不能公平地获取带宽。

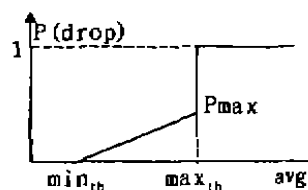


图 2 RED 算法

3. FRED 算法

为解决公平性问题, 文[3]提出 FRED (Fair Random Early Drop) 记录每个活跃流的状态, 将每个流相互隔离, 它惩罚没有拥塞控制或采用比 TCP 更贪婪的非 TCP 流, 而对不超过其共享带宽的流能保证传送, 不会丢包。但 FRED 存在两个问题, 首先, 记录每个流的状态开销大, 并且其可伸缩性差; 其次, 对 TCP 友好的流, 它消耗的带宽超过其公平共享带宽, 对该流只采用 RED 算法丢包, 而对标记为非 TCP 友好的流, 一旦该流消耗的带宽超过其公平共享带宽, 则丢弃所有进来的包, 这样, 有流控制的交互式多媒体流 (基于 UDP) 很有可能标记为非 TCP 友好的, 导致它比 TCP 流经历更多的丢包而不能获得它的公平共享带宽。

4. CBT 算法

为解决 FRED 算法存在的问题, 文[4]提出一种新的主动队列管理机制 CBT (Class-Based), 它在 RED 机制的基础上增加了基于类的静态带宽预留机制, 使用 RED 在单个队列上实现了 CBQ (Class-Based Queue) 明确的资源预留机制, 没有采用 CBQ 的包调度机制, 一方面使得 UDP 流不再遭受严格基于每个流的惩罚, 另一方面又可以保护 TCP 流公平共享带宽, 其工作原理如图 3 所示。CBT 的基本思想是使用

基于类的静态阈值 (用包或分组的数目表示) 来决定在拥塞发生时, 每个 UDP 类可在输出队列中占用的队列长度 (用包衡量), 而对 TCP 流则没有阈值测试, 直接应用 RED 管理。CBT 定义了三类流, 标记的 UDP 流 (交互多媒体流), 其它 UDP 流 (未标记) 和 TCP 流, 它为两类 UDP 流指定预先决定的静态阈值, 并且记录每类加权平均队列长度 (用在队列中的包数目表示), 一旦类的加权平均队列长度超过类的静态阈值, 则丢弃该类进来的包。每当有新 UDP 包到达时, 更新其平均队列长度, 并进行相应的阈值测试, 这样, 可以将 TCP 流从 UDP 流中隔离出来, 同时, 又可以保护多媒体 UDP 流不受其它 UDP 流的影响。

CBT 在保持 RED 的拥塞避免机制的同时, 又在单个队列上实现了 CBQ 的基于类的隔离机制, 因此在增加较小处理负荷的同时取得了接近 FRED 的公平性, 但是, Internet 业务改变非常快, 这种静态资源预留机制并不能完全反映快速变化的业务状态。

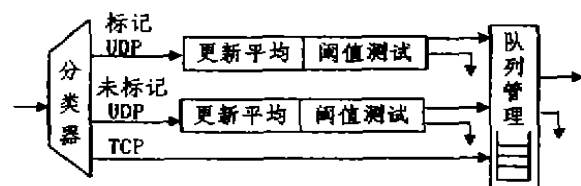


图 3 CBT 算法示意图

5. D-CBT 算法

5.1 D-CBT 概述

D-CBT (Dynamic Class-Based Threshold) 加强了不同流类之间的公平性, 提高了 UDP 流类的队列资源使用效率, 图 4 给出了其设计框图。D-CBT 主要在两个方面不同于 CBT: 第一, 动态更新每类的公平阈值; 第二, D-CBT 动态监视 RED 的拥塞状态, 只有当 red_avg (RED 平均队列长度) 大于 red_min 时, 阈值测试才进行。这样, 通过动态更新每个类的活跃流数目在总的活跃流中所占的比例, 可动态更新每个类的阈值, 结合上述第二点, 允许 UDP 类在 CBT 中更有效地使用队列资源。实际上, D-CBT 可看成基于类的类 FRED 机制。

类似于 CBT, D-CBT 将所有的流分成三类: TCP 流、标记 UDP 流、未标记的 UDP 流, 但在 D-CBT 中它将有流控制的多媒体流从没有流控制的多媒体中区分出来, 它只标记有流控制的多媒体流, 其它的都归类为未标记流, 而在 CBT 中标记所有的多媒体流。其目的是保护有流控制的多媒体流, 鼓励多媒体应用采用类似 TCP 的拥塞避免机制。实际上, D-CBT 根据不同

流的拥塞响应特性,将流分成了三类:TCP 友好的、有流控制但不是 TCP 友好的多媒体流、没有流控制的多媒体流,这样,更便于不同的类采用不同的管理方法。

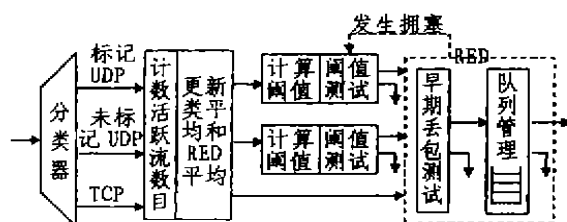


图 4 D-CBT 算法示意图

5.2 UDP 活跃流计数模块

我们比较图 3 和图 4, D-CBT 在 CBT 中增加了活跃流计数模块, 它的目的是能更动态、公平地反映不同类之间的公平共享。活跃流计数模块跟踪每类的状态信息, 它的思路来源于 FRED^[3], 文[3]对活跃流的定义是在输出队列中至少有一个包或分组, 它预先定义一个检查间隔, 在这个间隔里面一个流至少有一个包进入了输出队列。但 D-CBT 对此作了一些改变, 紧跟在分类器后的活跃流计数模块保存每个类的排序链表, 该链表包括每个流的描述符、最后接收包的时间以及对应类的流计数器。

每当有新包经分类器分类后, 活跃流计数模块更新相应的数据结构, 它插入或更新流信息和当前的本地时间, 如果插入新的流信息, 则对应类的活跃流计数器增 1。每隔一定的间隔 (如 300ms), 计数模块扫描每个类的链表, 删除过时的流信息, 减少对应类的流计数器。

D-CBT 不同于基于流计数的队列机制, 它并不需要精确的活跃流计数, 虽然更精确的计数可更公平地在每个类之间分配带宽。因此, 每个类链表的更新可在路由器空闲时进行, 如果路由器是多处理器系统, 可由一个特定处理器作扫描更新工作。在 D-CBT 的当前实现中, 它采用排序链表, 插入、更新、扫描删除的复杂度是 $O(n)$, 如果改用哈希表实现, 则插入、更新复杂度是 $O(1)$, 扫描删除复杂度仍然是 $O(n)$, n 是该类的活跃流数目。

5.3 UDP 阈值计算方法

标记 UDP 流和未标记的 UDP 流其阈值计算方法如下:

$$\text{标记 UDP 阈值} = \frac{\text{标记的活跃流数}}{\text{标记的活跃流总数}} \times \text{red_avg} \quad (1)$$

$$\text{未标记 UDP 阈值} = \frac{\text{未标记的活跃流数}}{\text{未标记的活跃流总数}} \times [\text{red_min} + (\text{red_max} - \text{red_min}) \times 0.1] \quad (2)$$

在计算公平 UDP 阈值时, 如公式 (1) 所示, 标记

UDP 类根据 RED 的平均队列长度计算其在输出缓冲区的平均公平共享; 而未标记的 UDP 类只根据 RED 的最小阈值 (red_min) 加上一个很小的补偿来计算其在输出缓冲区的平均公平共享, 如公式 (2) 所示。从 5.1 节 UDP 流的分类看出, 标记 UDP 类具有流控机制, 能对网络拥塞响应, 主动降低其发送速率, 减少输出队列的平均队列长度, 因此在 RED 的拥塞避免阶段允许标记 UDP 流使用输出缓冲区 (red_avg - red_min), 直到其占用的缓冲区数量达到平均公平共享。而对没有流控机制的 UDP 流, 由于它不对拥塞指示响应, 因此在拥塞避免阶段不允许使用输出队列缓冲区, 故其动态阈值的计算只是 red_min 加上适当补偿 $([\text{red_max} - \text{red_min}] \times 0.1)$ 中所占的比例。

每当有一个包到达时, 根据它的流标识信息, 插入或更新在计数模块中的对应类链表, 更新 RED 平均队列长度、标记 UDP 类和未标记 UDP 的阈值, 然后将包传递给对应的测试模块。如果这时 RED 指示拥塞发生 (即 red_avg > red_min), 则 UDP 包要经过阈值测试后才传递给 RED 管理, 否则, 直接传递给 RED, 工作过程如图 4 所示。

5.4 仿真实验结果及分析

文[5]在图 5 所示的网络拓扑图上比较了 RED、CBT、D-CBT 三种机制在网络带宽分配方面的公平性, 每次实验除路由器队列机制不同外, 其它方面都是一样的。其业务源设置在 0 秒时启动 25 个 FTP-TCP 流和 10 个有流控机制的 UDP 流, 在第 10 秒时启动 2 个 CBR-UDP 流, 在第 30 秒时再启动 30 个 FTP-TCP 流, RED 参数使用文[2]的推荐设置 [60, 15, 5, 0.002, 0.1], 分别对应队列限制、最大阈值、最小阈值、权重、丢包概率。分组或帧大小设置为 1k 字节, CBR 源速率设置为 5Mbps (无流控机制), 多媒体 UDP 流根据 MPEG-1 的编码特点, 使用五种发送速率 300, 500, 700, 900 和 1100kbps (即五种规格, 编码从 0~4), 其流控机制是每丢失一帧则将规格减少一半, 如果在一个 RTT 内没有丢帧, 则规格加 1。

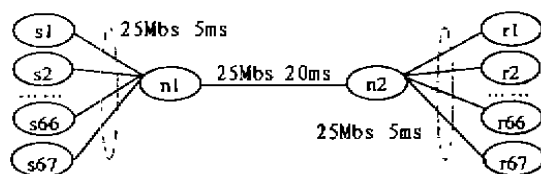


图 5 仿真网络拓扑示意图

表 1 第 3 到第 5 列是文[5]给出的仿真实验结果, 第 6 列是我们根据公式 (3) 得到的公平性值,

$$F = \left(\sum_{i=1}^n b_i \right)^2 / n \left(\sum_{i=1}^n b_i^2 \right) \quad (3)$$

表1 RED、CBT、D-CBT类的
流平均吞吐率和公平性比较

	秒	TCP-Reno	标记UDP (MPEG-1流)	未标记UDP (CBR流)	公平性
RED	0~10	537	851	0	0.951
	10~20	293	720	4681	0.480
	20~30	161	601	4503	0.447
CBT	0~10	587	760	0	0.984
	10~20	615	739	624	0.993
	20~30	305	657	318	0.829
D-CBT	0~10	622	653	0	0.999
	10~20	659	660	637	0.999
	20~30	356	378	313	0.943

比较第6列的公平性值,我们可看到D-CBT能获得更好的公平性。

结束语 本文比较分析了RED、FRED、CBT、D-CBT四种不同的主动队列机制,重点分析了D-CBT, D-CBT将所有的业务分成三类:TCP、有流控的UDP流、没有拥塞控制的UDP流,对它们应用不同的队列策略,从而可限制每个UDP类最多使用它们应该得到的平均队列缓冲区空间。比较而言,D-CBT有如下优点:①D-CBT具有RED的拥塞避免功能;②D-CBT在增加较小负荷的情况下,能取得类似FRED的公平

特性;③D-CBT对每类进行惩罚,不是使用FRED执行严格的每个流的惩罚方法,因此D-CBT允许有流控机制的多媒体流能更公平地使用其输出带宽;④D-CBT的阈值动态更新能使所有流更公平地共享带宽。

我们下一步将改进D-CBT,研究它对区分服务的支持,并且结合端-端拥塞控制机制研究其公平性。

参考文献

- 1 Floyd S, Jacobson V. Link-sharing and Resource Management Models for Packet Networks. IEEE/ACM Trans. on Networking, 1995, 3(4): 7~9
- 2 Floyd S, Jacobson V. Random Early Detection Gateways for Congestion Avoidance. IEEE/ACM Trans on Networking, 1993
- 3 Lin D, Morris R. Dynamics of Random Early Detection. In: Proc. of SIFCOMM'97, Cannes, France, 1997
- 4 Parris M, Jeffay K, Smith F D. Lightweight Active Router-Queue Management for Multimedia Networking. Multimedia Computing and Networking, SPIE Proc. Series, Vol 3020, San Jose, CA, January 1999
- 5 Chung J, Claypool M. Dynamic CBT—Better Performing Active Queue Management for Multimedia Networking [Tech Report WPI-CS-TR-00-10]. Worcester Polytechnic Institute, April 2000

(上接第44页)

据格式,从而达到改变服务级别的目的。例如,对于采用分层编码的视频流而言,当QoS管理者决定降低服务级别时,可以命令过滤器过滤掉与图像细节相关的高层编码数据,只传送视频图像的基本信息,这样由视频服务器端发往点播客户端的数据流量就会减少,整个网络的拥塞程度将会有所改善。

由于在视频点播系统中采用了上述QoS管理方案,使得该系统能够较好地适应各种网络环境,并且在网络通信条件不断变化的情况下,仍然能够提供必要的QoS控制与保障。

结束语 由于目前大多数的网络系统(如Internet)尚不能对QoS要求提供充分的支持,因此采取适当的QoS管理和控制手段以保证点播质量对现有的视频点播系统是非常必要的,本文介绍的QoS管理方案在这方面具有较高的实用价值。随着通信与计算机网络技术的不断发展,未来的Internet网将对QoS提供良好的支持,在这种情况下只需引入QoS管理者与Internet网络设备之间的QoS申请、协商机制,本文介

绍的方案同样会取得良好的效果。

参考文献

- 1 Campbell A, et al. Integrated Quality of Service for Multimedia Communications. Proc. IEEE INFOCOM'93, San Francisco, USA, 1993, 3: 732~739
- 2 Fry M, Seneviratne A. A QoS Managed Video Conferencing Application. [Technical report]. CRC for Distributed Systems Technology, 1994
- 3 Neumair B. Modeling resources for integrated performance management. In: Proc. of the Third Int Symposium on Integrated Network Management, San Francisco CA, April 1993, 109~121
- 4 Pacifici G, Stadler R. An architecture for performance management of multimedia networks. In: Proc. of the Fourth Int Symposium on Integrated Network Management, Santa Barbara CA, May 1995, 174~186
- 5 Lazar A A. An architecture for real-time control of broadband networks. In: Proc. of the IEEE Global Telecommunications Conf. Phoenix AZ, December 1991, 289~295
- 6 Schulzrinne H, et al. RTP: A Transport Protocol for Real-time Applications. IETF, Nov. 1995
- 7 Buss I, et al. Dynamic QoS control of multimedia applications based on RTP. Computer Communications, 1996, 19: 49~58