支持 QoS 服务的 SCTP 中伪共享问题的研究

李晓夏,杜中军,贾 钊
LI Xiao-xia,DU Zhong-jun,JIA Zhao

四川大学 计算机学院,成都 610065
Department of Computer,Sichuan University,Chengdu 610065,China
E-mail:strackor@163.com

LI Xiao-xia,DU Zhong-jun,JIA Zhao.Research about false-sharing of SCTP based on QoS.Computer Engineering and Applications,2009,45(15):130-131.

Abstract: Multi-stream is an important characteristic of SCTP,which provides an aggregation mechanism for SCTP association to accommodate heterogeneous objects,which originate from the same application but may require different type of QoS from network.The current SCTP specification is not aware of QoS provided by networks,and an extension of SCTP provided to support QoS.However,another problem——false-sharing comes out.This paper provides a method which solves the false-sharing problem,and is validated by simulate experiments.

Key words: Stream Control Transmission Protocol(SCTP);Quality of Service(QoS);sub-flows;congestion control;false-sharing

摘 要: 多流是流控制传输协议(SCTP)的一个重要特点,利用这个特点可以在同一个关联中传输不同优先级的数据包。然而当前的 SCTP 协议规范并没有提供对 QoS 服务的支持,提出对 SCTP 的一种改进,但是由此引出伪共享问题。主要研究了伪共享对于 SCTP 性能的危害,并通过对 SCTP 拥塞系统的改进解决了伪共享问题,最后通过仿真实验进行了证实。

关键词: 流控制传输协议;服务质量;子组流;拥塞控制;伪共享
DOI: 10.3778/j.issn.1002-8331.2009.15.037 **文章编号:** 1002-8331(2009)15-0130-02 **文献标识码:** A **中图分类号:** TP393

1 引言

流控制传输协议(Stream Control Transmission Protocol,SCTP)是继 TCP 和 UDP 之后由 IETF 开发的第三个通用传输协议,是一种面向连接的和面向消息的可靠传输协议的。它的设计主要是为了克服 TCP 和 UDP 的不足,因此被加入了众多的新特性,如:多宿、多流、信息边界保护、队头阻塞的消除等等。

“流”在 TCP 中指一系列的字节,而在 SCTP 中指的是一个在 STCP 关联(关联:由 TCP 的连接引申出,但比其含义更广,每个关联都由两个端口号和两个 IP 地址列表组成) 内部单独进行排序的消息序列。关联支持流的数目是不固定的,由源端与目的端商定。

QoS 服务是当前研究较多的一个方向, 可根据 SCTP 流的特性支持 QoS。可以来自同一应用程序的不同类型的对象可以根据网络中 QoS 要求的不同被分配给不同的流, 利用同一个关联中的不同流并发的传送(如,网页中的 JPG、TXT、VIDEO 就可以放在同一关联中的不同流传输时,TXT 文件就可要求比 JPG 等优先级更高的 QoS)。

然而目前的 SCTP 规范中并没有对支持 QoS 有任何的定义。为此文献[1]提出了子流组(sub-flows)的概念,即将相同优先级的流聚合成一个子流组,该子流组中的每一个流在网络中受到的待遇(如丢包率、延迟等)是相同的。这样就可以解决了 SCTP 对 QoS 的支持问题。然而这又引申出另一个问题,伪共享(false-sharing)。

2 伪共享的发生与解决

由于 SCTP 的拥塞控制算法是基于 RFC2581 的,在关联中使用类似于 TCP 的拥塞控制算法, 即关联通过收集各个流的拥塞信息, 不加区别的对各个流进行拥塞控制。在各个流的 QoS 优先级相同的情况下, 这种做法显然不会引起任何问题。但是在优先级不同的情况下,就会引起伪共享。

文献[2]指出,伪共享的发生基于以下事实,即共享拥塞状态信息的多个流并不共享相同瓶颈。假设基于 QoS 服务的 SCTP 将优先级相同的包通过同一子组流传输, 那么各流中的包根据其 Differentiated Services Codepoints(DSCPs)的区别,在网络中受到的待遇不尽相同,各流的拥塞参数值也不尽相同。在此,对于拥塞控制信息的共享就引起了伪共享的发生。

伪共享的发生将会极大的影响 SCTP 的整体性能。在各流的包丢失率各不相同的情况下,传输层将会根据共享的拥塞控制信息调整各流的传输性能,高优先级的流将会因为与低优先级的流共享拥塞控制信息而在性能上受到削弱。同样,在 RTTs 不同的情况下,各流也会因为不必要的快速重传和超时而使性能受到影响。

作者简介: 李晓夏(1983-),男,硕士研究生,研究方向:计算机网络;杜中军(1965-),男,副教授,硕士生导师,研究方向:计算机网络;贾钊(1979-),男,讲师,研究方向:网络安全。

收稿日期: 2008-04-01 **修回日期:** 2008-06-23

为此可借鉴文献[3]中并行 TCP(parallel TCP)的思想,将关联中各个子组流之间的关系看为并行的 TCP 连接之间的关系,如同并行 TCP 中每个 TCP 都拥有自己的拥塞控制系统一样,各子组流拥有自己的拥塞控制系统,即将拥塞控制系统从关联级下放到子组流级。尽管各子组流在网络中的拥塞参数不尽相同,但是由于其拥塞控制系统是相互独立的,各个子组流之间不会互相影响,高性能的子组流不会在因为平衡负载的需要而受到惩罚,从而就避免了伪共享的发生。

3　SCTP 传输性能分析

本章从理论上对 SCTP 的传输性能进行讨论,并在一个 DS 网络环境下设计出 SCTP 的三种情况分别分析各自的性能,仿真实验将在第 4 章实现。

由文献[4]可知,在流的传输状态稳定后,由包在传输队列的丢失率 p 可推出 SCTP 关联的吞吐量:

$$\frac{B(p)}{M}=\frac{E[N]}{E[A]}=\frac{\frac{6\eta}{p}+6-2\eta+2\sqrt{(3-\eta)^2+\frac{24\eta}{p}}}{(9-\eta+\sqrt{(3-\eta)^2+\frac{24\eta}{p}})\times RTT}\approx \frac{\frac{6\eta}{p}+2\sqrt{(3-\eta)^2+\frac{24\eta}{p}}}{(\sqrt{(3-\eta)^2+\frac{24\eta}{p}})\times RTT} \tag{1}$$

其中 M 为 MTU 的大小,$\eta=K/M$,M 为数据包的大小,RTT 为往返时延的平均值。$E[A]$表示上一次超时事件与下一次超时事件发生之间的时间间隔,$E[N]$表示在此时间间隔内发送的平均数据量。如果考虑到数据包的超时重传,上述公式可以改为:

$$\frac{B(p)}{M}=\frac{E[N]+Q^*E[R]}{E[A]+Q^*E[T^{TO}]} \tag{2}$$

其中 Q 表示丢包事件属于超时事件的概率,$E[R]$表示因超时事件而重传的数据量,$E[T^{TO}]$表示重传数据所花费时间的平均值。

在此使用一个简单的 DS 环境下分别讨论三种情况,并假设三种情况下关联中流的数目以及数据包的大小都相同。

情况 1　不支持 QoS 的源 SCTP。

情况 2　支持 QoS 的 SCTP,由位于关联级的拥塞控制对各子组流的拥塞控制进行统一管理。

情况 3　支持 QoS 的 SCTP,拥塞控制放在子组流级,由各个子组流自行管理。

对于情况 1 来说,将 p 直接带入公式(1)或(2)即可得出关联的吞吐量。

在情况 2 下,各流对于拥塞窗口的分配采用 WRR(Weighted Round-Robin)算法,λ_i 表示流 SFi 分配的拥塞窗口占全部拥塞窗口的百分比,则包的平均丢失率为:

$$\bar{P}Case2=\sum_{i=1}^{n}\lambda p_i$$

由文献[5]可知,在 p 极小的情况下,可近似的使用 $Pcase2$ 作为情况 2 的关联的包的丢失概率。

对于情况 3 来说,由于其拥塞控制放在子组流级上,各个子组流间的关系近似于并行 TCP,则将各个子组流的吞吐量相加即可近似得出关联的吞吐量。

4　仿真实验

使用 NS-2 网络模拟器对上述情景进行模拟。图 1 为仿真网络设计,该网络使用两类流量 BE(Best Effort)和 AF(Assured Forward)。网络对不同的流提供不同的数据包的丢失概率,端点 0、1、2 分别模拟情况 1、2、3,端点 3 模拟网络背景流。端点 0 建立一个 SCTP 关联,该关联仅有一个 BE 流。端点 1 建立一个拥有一个 BE 流和一个 AF 流的关联,由关联级的拥塞控制参数对两流的拥塞进行统一管理。情况 3 使用一个 BE 流和一个 AF 流,由单独的拥塞控制分别管理两流,相当于在端点 2 建立了两个 SCTP 关联。端点 4 由 10 个 BE 流和 4 个 AF 流作为网络背景流。

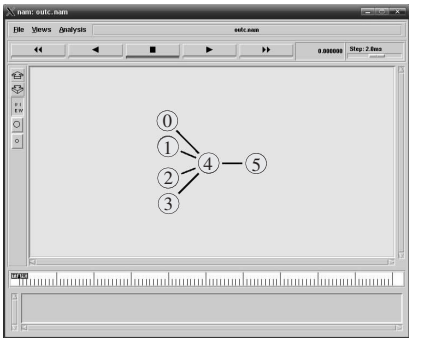

图 1　仿真实验网络拓扑

端点 4 起网络核心路由的作用,端点 5 作为各流的终端,并假设端点 5 的输入缓存无限大,即不会因为端点 5 的输入缓存不足而造成包的丢失。端点 0、1、2、3 到端点 4 的链路大小和延迟 40 Mb 和 10 ms,端点 4 到 5 的链路大小和延迟为 10 Mb 和 20 ms(即在此制造链路瓶颈)。同时,所有的 SCTP 关联不使用延迟确认(Delayed-SACK),当接收端成功接收数据包后立即发送确认,已确保仿真的精确性。

表 1 给出了端点对于不同通信流的拥塞窗口的分配策略,其中 Q 表示输出队列大小,a 为分配给 AF 流的窗口大小所占输出队列大小的百分比。实验中 a 取 10%~90%,递增量为 5%。

表 1　拥塞窗口分配参数

传输队列	Min/max thresh	Max drop rate
BE	$0.2/0.8\times Q\times(1-a)$	0.05
AF	$0.2/0.8\times Q\times a$	0.05

图 2 为从实验中得出的数据值、观察图表,可得出两个结论。

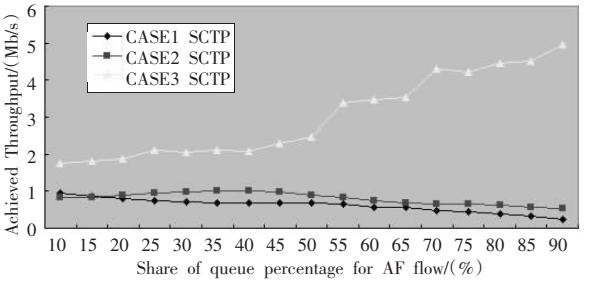

图 2　SCTP 在不同丢失率下的吞吐量

(1)情况 2 的关联有一个 BE 流和一个 AF 流组成,但是其吞吐量确与情景 1 差不多,由此可看出,伪共享极大的影响了 AF 流的吞吐量,关联级的拥塞控制对于高性能的 AF 流进行了惩罚,使其保持与低性能流之间的性能平衡。忽视了 QoS 对于流之间的性能区别的要求。

(2)尽管情况 2 和情况 3 的关联都有一个 BE 流和一个

(下转 157 页)