

一种改进 HSTCP 公平性的拥塞控制算法^①

沈 燕, 王绪宛

(南阳理工学院 软件学院, 南阳 473000)

摘 要: 标准传输控制协议在高速、长距离等应用条件下存在网络带宽不能充分利用的问题, 高速传输协议的提出可以解决此问题, 但在数据包往返时延上存在较大不公平性. 在分析传输公平性基础上, 通过离散事件网络模拟器进行公平性模拟验证, 提出一种通过添加公平因子改进高速传输协议公平性的算法, 经仿真验证, 该算法减轻了因 RTT 不同造成的不公平性.

关键词: 高速传输控制协议; 拥塞控制; 高速网络; RTT 不公平性

An Algorithm of Congestion Control on Improving Fairness of HSTCP

SHEN Yan, WANG Xu-Wan

(Software Institute, Nanyang Institute of Technology, Nanyang 473000, China)

Abstract: Standard TCP can not make full use of network bandwidth in the condition of high-speed and long distance. High-speed TCP has been proposed to solve this problem, but has severe RTT unfairness. Though analyzing and simulating, an algorithm based on congestion control is put forward to improve the fairness by adding fair factor. Simulation experiments show that the modified algorithm can reduce the unfairness of different RTT effectively.

Key words: high-speed tcp; congestion control; high-speed networks; unfairness of RTT

TCP 是目前 internet 上使用最广泛的传输协议, 实践证明在低带宽环境下, 能满足对吞吐量的要求, 并能保证公平性^[1]. 随着高性能计算、网络备份和大型数据库同步等大数据量访问的需要, 出现了带宽大于 1Gbps 的高速网络, 由于标准 TCP 协议自身的问题, 吞吐量不能达到实际需求^[2-4].

为克服标准 TCP 在高速网络环境下的局限性, S.Floyd 提出了一种实验性的 High-Speed TCP(简称 HSTCP)算法^[1]. HSTCP 在标准 TCP 拥塞避免算法基础上, 通过改变参数, 实现了根据拥塞窗口大小快速增长和慢速减小, 最终保持较大的拥塞窗口以充分利用网络带宽, 提高吞吐量^[5], 不过实验表明, HSTCP 在长时延的 HSTCP 流和短时延的 HSTCP 流相互竞争时存在严重的数据包往返时延 RTT 不公平性^[6-9], 也有学者通过增加公平性因子方法, 调整 RTT 的公平性, 但通用性不强^[8,10,11].

基于此, 论文在分析 HSTCP 的 RTT 不公平性基础上, 进行仿真实验, 提出 HSTCP 改进算法, 该算法

通过添加更为通用的公平因子, 降低对拥塞窗口增加速率的影响, 使不同 RTT 流的拥塞窗口增量趋于相等, 减轻了原算法中由于 RTT 不同造成的不公平性.

1 HSTCP 拥塞控制原理及算法

1.1 HSTCP 拥塞控制原理

2003 年 Floyd 针对 TCP 算法在高速网络环境下的不足, 提出了一种基于包丢失的 AIMD(additive increase multiplicative decrease)拥塞窗口调整算法 HSTCP. 该算法与标准 TCP 相比, 修改了最大慢启动门限和随拥塞窗口 W_{curr} 变化的拥塞反馈系数——加法因子 $a(w)$ 和乘法因子 $b(w)$, 并设窗口阈值为 W_{low} .

在慢启动阶段, 该算法使用与标准 TCP 相同的慢启动算法, 在拥塞避免阶段采用 AIMD 规律改变拥塞窗口大小: 当 $W_{curr} \leq W_{low}$ 时, HSTCP 参数与标准 TCP 参数相同, 当 $W_{curr} > W_{low}$ 时, 比 TCP 较快速度增加拥塞窗口, 较慢递减拥塞窗口, 比标准 TCP 维持一

^① 收稿时间:2013-03-06;收到修改稿时间:2013-04-07

个较大的拥塞窗口, 提高了网络吞吐量^[7,11].

1.2 HSTCP 拥塞控制算法

在 HSTCP 算法中引入了 4 个参数: W_{low} , W_{high} , P_{low} 和 P_{high} , 分别表示拥塞窗口和丢包率的最小、最大值. 为了保持和标准 TCP 的兼容性, 规定当拥塞窗口 W_{ccurr} 小于 W_{low} 时, 使用 and 传统 TCP 相同的响应函数, 当拥塞窗口 W_{ccurr} 大于 W_{low} 时, 使用 HighSpeed TCP 响应函数, 并且响应函数由传统 TCP 的响应函数式 $w = \sqrt{1.5/p}$ 改为:

$$w = 0.12 / P^{0.835} \quad (1)$$

当发送端接收到 1 个 ACK 后拥塞窗口增长函数变为:

$$w = w + \frac{a(w)}{w} \quad (2)$$

拥塞发生后拥塞窗口的调整函数为:

$$w = w - b(w)w \quad (3)$$

对传统 TCP, $a(w)=1$, $b(w)=0.5$, 与 w 无关. 当 $W_{curr} \leq W_{low}$ 时 HSTCP 等价于传统 TCP; 而当 $W_{curr} > W_{low}$ 时, $a(w)$ 和 $b(w)$ 是 w 的函数:

$$a(w) = \frac{2w^2 \times p(w) \times b(w)}{2 - b(w)} \quad (4)$$

$$b(w) = \frac{\log w - \log W_{low}}{\log W_{high} - \log W_{low}} (b(W_{high}) - 0.5) \quad (5)$$

其中, $p(w)$ 为在拥塞窗口为 W_{curr} 时的丢包率, 值为 0.078, 并设 $b(W_{high})=0.1$, 使用推荐值, 可以算出 $w^{1.2}$

不同 w 对应的 $a(w)$ 和 $b(w)$. 在实际使用中, $a(w)$ 和 $b(w)$ 为静态表, 如表 1 所示, 从中可看出, $a(w)$ 的值随着 w 的增加而增大, $b(w)$ 的值相应减小, 因此 HSTCP 能够维持一个大的拥塞窗口, 提高了高速网络的利用率.

表 1 HSTCP 中 $a(w)$ 和 $b(w)$ 的取值

w	a(w)	b(w)	w	a(w)	b(w)
38	1	0.50
118	2	0.44	84035	71	0.10
221	3	0.41
347	4	0.38			

2 HSTCP 公平性

HSTCP 在高速网络环境中能够比标准的 TCP 更好地利用网络带宽, 但在 RTT 公平性方面却存在严重的问题. 根据 HSTCP 算法, 加法因子 $a(w)$ 的值随 w 的增加而增大, 乘法因子 $b(w)$ 随窗口增加而减小, 当网

络拥塞时, 拥塞窗口不会迅速减小, 而没有拥塞发生时, 拥塞窗口却迅速增大, 从而导致 HSTCP 维持一个较大的拥塞窗口, 尽管提高了网络利用率, 但却不可避免地出现了带宽竞争现象, 尤其在 RTT 流对网络资源争用情况下, 由于带宽反比于 RTT, 使得 RTT 较小的流获得较大的带宽, 影响到其他 RTT 较大流对带宽的占用, 产生了不公平现象^[6].

为了验证 HSTCP 的不公平性, 选用目前广泛使用的离散事件网络模拟器第二版 (Network Simulator version 2, NS2) 进行模拟仿真, 仿真场景对应的网络拓扑结构如图 1 所示, 其中, 节点 0、1、2 为发送端, 5、6、7 为接收端, 节点 3、4 间的瓶颈链路带宽设为 500MBbps, 其他链路带宽为 1Gbps, 延时均为 30ms, 其他参数取缺省值, 第一条数据流为 0-3-4-5, 第二条为 1-3-4-6, 第三条为 2-3-4-7, 路由队列管理采用队尾丢弃算法.

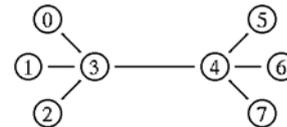


图 1 网络拓扑图

为了与标准 TCP 公平性进行比较, 分别对 TCP 和 HSTCP 协议进行仿真, 利用式 6 得出每条数据流不同时刻的流量.

$$F = \frac{(\sum_{i=1}^n x_i)^2}{n \sum_{i=1}^n x_i^2} \quad (6)$$

通过计算得到 TCP 和 HSTCP 协议的公平性在不同时刻的量化指标. 式 6 是衡量 TCP 协议公平性好坏的标准, 其值位于 (0,1) 之间, F 值越接近 1, 其公平性越好. TCP 和 HSTCP 协议在不同时刻的公平性指标如图 2 所示, 在高速网络中, TCP Reno 的公平性索引值达到 0.94, 是最好的, 而 HSTCP 的公平性索引值仅为 0.51 左右, 和标准 TCP 相比, HSTCP 存在严重的不公平性.

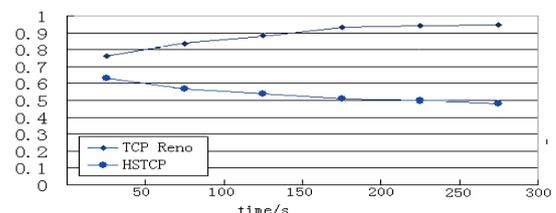


图 2 TCP 和 HSTCP 的公平性

3 HSTCP 公平性改进算法: AR-HSTCP

3.1 AR-HSTCP 算法介绍

AR-HSTCP 算法在拥塞避免阶段, 每收到一个

ACK, 窗口的变化由式 7 进行更新.

$$w(t+1) = w(t) + a(w) \times \frac{\Delta t}{RTT} \quad (7)$$

从式(7)可以看出, 每经过 Δt HSTCP 流的拥塞窗口将增加 $a(w) \times \frac{\Delta t}{RTT}$. AR-HSTCP 算法的基本思想是根据 RTT 的不同, 通过增加一个公平因子, 使不同 RTT 的流在相同的时间内窗口的增加趋势程度相同.

为了简化问题的复杂度, 只讨论窗口在拥塞避免阶段的变化情况, 假设网络中存在两条 HSTCP 流 1 和 2, RTT 分别为 RTT_1 和 RTT_2 , 在初始时刻窗口的大小均为 w , 并利用式 8 计算各个流的 RTT 加权平均数.

$$avg = \frac{RTT_1 + RTT_2}{2} \quad (8)$$

取公平性因子: $c_i = \frac{RTT_i}{avg}$, (其中, $i=1,2$).

在 HSTCP 拥塞避免阶段把式 7 转换为式(9):

$$w(t+1) = w(t) + c \times a(w) \times \frac{\Delta t}{RTT} \quad (9)$$

经过 Δt 后, 拥塞窗口增大为 $c \times a(w) \times \frac{\Delta t}{RTT}$, 即流 1 的拥塞窗口为:

$$\Delta w_1 = c_1 \times a(w_1) \times \frac{\Delta t}{RTT_1} \quad (10)$$

流 2 的拥塞窗口为:

$$\Delta w_2 = c_2 \times a(w_2) \times \frac{\Delta t}{RTT_2} \quad (11)$$

其中, $c_1 = \frac{RTT_1}{avg} = \frac{2 \times RTT_1}{RTT_1 + RTT_2}$, $c_2 = \frac{RTT_2}{avg} = \frac{2 \times RTT_2}{RTT_1 + RTT_2}$,

将 c_1, c_2 二者分别代入式 10 和 11 中可得 $\Delta w_1 = \Delta w_2$.

通过该方式, 不同 RTT 流在拥塞避免阶段获得相同的拥塞窗口增加, 避免了不同流的 RTT 对窗口变化的影响. 若 RTT_1 为 50ms, RTT_2 为 100ms, 在引入公平因子前, 流 1 在 $t+50ms$ 后窗口大小为 $w + a(w)$, 在 $t+100ms$ 后, 窗口大小为 $w + a(w) + a(w + a(w))$, 其值约等于 $w + 2a(w)$, 而流 2 在 $t+100ms$ 后窗口大小为 $w + a(w)$, 窗口大小存在差异, 而采用改进算法之后, 不管 RTT_1 和 RTT_2 的大小如何, 都使窗口在相同的时间内增加同样的大小.

为实现上述算法, 借助 TCP 的相关机制对 HSTCP 进行修改使其可以计算流 RTT_i , 然后在下层用 IP 包头尚未定义的空间传递 RTT_i 的估算值, 再在路由器中存放不同流的 RTT 信息, 路由器通过这些信息计算各

流公平因子 c_i , 最后按照式 9 增加窗口. 公平因子在路由器中的具体计算流程描述如图 3 所示.

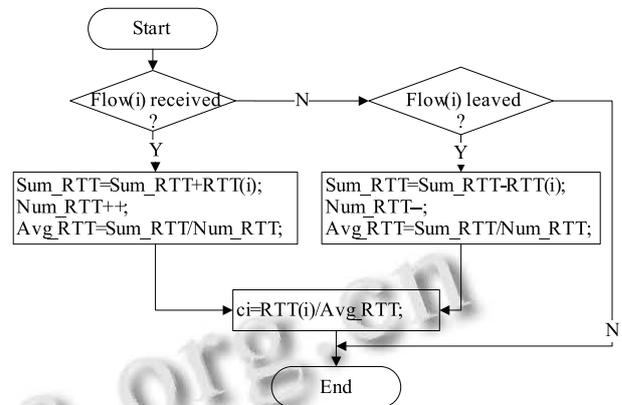


图 3 公平因子的计算

在初始窗口相同的条件下, 根据发送速率公式 $v(t)=w(t)/RTT$, 理论上改进算法的 RTT 不公平性与 RTT 成反比.

3.2 仿真及其结果分析

为了验证算法有效性, 笔者在个人电脑上采用 NS2 进行仿真模拟, 其硬件条件为 Inter Core2 T5750 处理器, 2GB 内存, ATI 的 HD 3470 显卡, Windows XP Sp3 操作系统.

仿真场景网络拓扑结构如图 1 所示, 瓶颈链路带宽设为 1Gbps, 延时 20ms, 其他链路带宽均为 2.5Gbps, 延时为 5-100ms, 所有链路队列类型均为 DropTail, 路由队列缓冲大小设置为 8000 个包大小; 包的大小设置为 3500Bytes, 其他值取缺省值.

流 0、1、2 的 RTT 分别为 80ms, 150ms, 230ms, 仿真过程中观察各个流的拥塞窗口的变化, 得出改进算法前后各流拥塞窗口的变化如图 4 和图 5 所示.

图 4 和 5 中, 上中下三条曲线对应的 RTT 值分别为 0.08s、0.15s、0.23s, 通过图 4 可以看出: RTT 为 80ms 的流 0 的拥塞窗口随着时间变大, 最后到达 8300 个 MSS 以上, 而其他两个长 RTT 流(1, 2)的窗口越来越小, 分别为 800 和 400MSS 左右, 由此计算三个流的发送速率分别为 103750MSS/s、5333MSS/s 和 1739MSS/s.

图 4 中, 流 0 的发送速率分别是流 1 和 2 的 19.5 和 59.7 倍左右, 造成流 0 在竞争瓶颈链路带宽时处于统治地位, 掠夺了几乎所有的带宽, 使得流 1 和 2 对带宽的占有率非常低, 从而产生了不公平现象. 采用 AR-HSTCP 算法后, 三个 RTT 不同流的发送窗口逐渐接近, 拥塞窗口分别为 4500、3500、3100MSS 左右, 经计算, 发送速

率分别为 56250MSS/s、23333 MSS/s、13478 MSS/s。

图 5 中, 流 0 的发送速率分别是流 1 和 2 的 2.4 和 4.2 倍左右, 三个流的发送速率差距比较接近, 对瓶颈链路的带宽分配也较均匀, 有效改进了公平性。

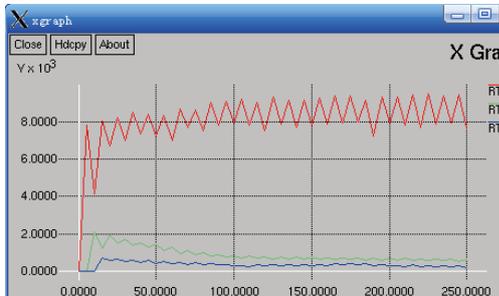


图 4 改进算法前窗口随时间的变化情况

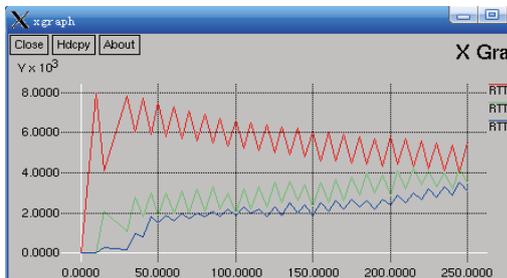


图 5 改进算法后窗口随时间的变化情况

为了直接观察三个流的带宽分配情况, 经过计算得到三个流在算法改进前后对瓶颈链路带宽的占有情况, 如表 2 和表 3 所示, 改进算法前流 1 和流 2 对瓶颈链路带宽的占有率为 10.6% 和 6.1% 左右, 采用改进方案后, 长 RTT 的流 1 和流 2 的占有率提高至 37.9% 和 20% 左右, 而且随着时间的进行, 带宽的公平分配将会得到更好的体现。

表 2 改进算法前各流对瓶颈带宽的利用率

时间(s)	各流占总流量的比率(%)		
	流 0	流 1	流 2
50-100	81.47	12.35	6.18
100-150	82.98	10.87	6.15
150-200	83.26	10.62	6.12
200-250	83.27	10.63	6.10

表 3 改进算法后各流对瓶颈带宽的利用率

时间(s)	各流占总流量的比率(%)		
	流 0	流 1	流 2
50-100	62.83	24.16	13.10
100-150	51.75	33.68	14.57

150-200	44.91	37.94	17.15
200-250	40.07	38.14	21.79

4 结论

针对高速网络中 HSTCP 严重的 RTT 不公平性, 采用 AR-HSTCP 算法, 对 RTT 不公平性进行了一定程度的改进。算法根据 RTT 的不同, 通过添加公平因子, 降低不同 RTT 对拥塞窗口增加速率的影响, 使不同流的拥塞窗口增量趋于相等, 从而公平地共享瓶颈链路的带宽, 最后在 NS2 中进行了仿真对比。理论分析和模拟实验表明, AR-HSTCP 在保证 HSTCP 原有优点的基础上, 有效地提高了高速流之间的公平性, 且实施简单, 可以方便地用在高带宽、长距离和低复合的网络中, 随着网络带宽的不断增长, 该算法具有广阔的应用前景。

参考文献

- 刘文远, 信春雷, 龙承念, 关新平. 一种基于 HSTCP 改进的公平性算法. 小型微型计算机系统, 2008, 29(1): 31-34.
- Jin C, Wei DX, Low SH. FAST TCP: motivation, architecture, algorithm, performance. Proc. of IEEE INFOCOM 2004.
- Kelly T. Scalable TCP: Improving performance in high-speed wide area networks. ACM ICGCOMM Computer communication Review, 2003, 33(2): 83-91.
- 信春雷. 基于高速 TCP 的网络拥塞性能分析研究[博士学位论文]. 秦皇岛: 燕山大学, 2007.
- 王思思, 朱金鹏, 王建峰. 一种改善 RTT 公平性的 TCP 慢启动算法. 空军工程大学学报(自然科学版), 2009, 10(3): 86-89.
- 王崇, 张大方, 曾彬. 高速网络中 TCP 拥塞控制算法的研究和改进. 计算机技术与自动化, 2007, 26(3): 30-34.
- Xu LS, Harfoush K, Rhee I. Binary Increase Congestion Control (BIC) for Fast Long-Distance Networks. Proc. of the IEEE INFOCOM, Hong Kong, 2004, 2514-2524.
- Zhang KP, Li ZZ. Study on TCP congestion control of supporting transfer fairness in the internet. Mini-Micro Systems, 2004, 25(9): 1597-1601.
- 龙承念, 杨会龙, 李欣, 关新平. EHSTCP: 改进的高速 TCP 算法. 计算机学报, 2008, 31(3): 440-449.
- 杨小刚, 单宝松, 陈钺, 吕卫峰. 基于二项式的高速网络拥塞控制协议. 江苏大学学报(自然科学版), 2009, 30(6): 618-621.
- 茹新宇, 刘渊. 一种新的 TCP 慢启动策略. 计算机工程, 2008, 34(5): 116-119.