

TFRC 协议友好性与平稳性改进算法研究

姜 明¹, 吴春明², 张 旻¹, 蒋 翊²

(1. 杭州电子科技大学软件与智能技术研究所, 浙江杭州 310018;

2. 浙江大学人工智能研究所, 浙江杭州 310027)

摘 要: 本文针对 TFRC(TCP-Friendly Rate Control)流与 TCP 流竞争带宽时的友好性问题, 分析了影响 TFRC 协议 TCP 友好性的因素, 通过对 TFRC 速率计算公式中丢包率的不同幂级项引入权重系数, 增加网络拥塞严重时的发送速率, 减少网络拥塞较轻时的发送速率, 从而降低了网络拥塞程度对 TFRC 流传输速率的影响. 仿真实验表明该方法对 TFRC 协议具有较明显改进作用, 提高了 TFRC 流的传输平稳度和 TCP 友好性, 从而能更有效地适应多媒体流的传输要求.

关键词: TFRC; 友好性; 平稳性

中图分类号: TP393

文献标识码: A

文章编号: 0372-2112 (2009) 08-1723-05

Research of the Algorithm to Improve Fairness and Smoothness of TFRC Protocol

JIANG Ming¹, WU Chun-ming², ZHANG Min¹, JIANG Yi²

(1. Institute of Software and Intelligent Technology, Hangzhou Dianzi University, Hangzhou, Zhejiang 310018, China;

2. AI Institute of Zhejiang University, Hangzhou, Zhejiang 310027, China)

Abstract: This paper analyzes the factors which affect the fairness between TFRC protocol and TCP protocol, and introduces two weighted coefficients to calculate the packet loss event rate which is used to calculate the sending rate in equation of TFRC protocol so as to adjust the sending rate based on network congestion level. Simulation results show that through this method, the fairness and smoothness of TFRC at different congestion levels are effectively improved, thus meeting the requirements of multimedia stream more effectively.

Key words: TFRC; fairness; smoothness

1 引言

近年来计算机网络的爆炸式发展, 特别是大量多媒体实时业务的广泛应用, 对传统的以 TCP 为主的传输层协议提出了新的挑战. 大部分多媒体应用对时延很敏感, 但却具有一定的容错机制, 能容忍少量的报文丢失. 此类应用要求传输协议开销小响应快, 并且不必保证传输完全可靠. 此外, 抖动也是对多媒体应用有较大影响的问题. 而 TCP 拥塞控制使用的 AIMD 策略, 会引起发送速率很大的抖动. 所以目前多媒体实时应用多采用 UDP 协议来传输, 但由于 UDP 协议缺乏拥塞控制机制, 当基于 TCP、UDP 的应用共享网络带宽时, TCP 流因具有拥塞退避机制而导致“饥饿”, 而 UDP 将不公平地获得大量带宽. 此外, 大量使用 UDP 协议会加剧网络拥塞, 影响 Internet 的鲁棒性, 因此随着 UDP 协议的应用在

Internet 所占比例的增大, 其不足之处逐渐暴露.

研究和完善用于各种网络环境并能解决公平性的新一代端到端的拥塞控制协议具有十分重要的理论价值和前景. 而 TCP-Friendly 协议作为其中一类典型协议, 目的在于适应多媒体实时应用的要求, 并能在同样网络和应用环境下获取与 TCP 流相近的吞吐量, 根据协议的机制特点不同^[1], 当前对 TCP 友好协议进行分类的主要依据有:

- (1) 基于窗口的 TCP 友好协议和基于速率的 TCP 友好协议;
- (2) 单播的 TCP 友好协议和组播的 TCP 友好协议;
- (3) 单速率的 TCP 友好协议和多速率的 TCP 友好协议;
- (4) 端到端的 TCP 友好协议和需要路由器支持的 TCP 友好协议.

收稿日期: 2008-08-04; 修回日期: 2009-03-10

基金项目: 国家自然科学基金 (No. 60773182); 国家 863 重大项目 (No. 2008AA01A323); 浙江省 2007 年度科技计划项目 (No. 2007C21031)

TFRC 是其中一种基于公式控制发送速率的端到端单播 TCP 友好协议, IETF 于 2003 年正式颁布了 TFRC 协议的 RFC 文档^[2], 近年来该协议受到国内外研究人员的广泛关注与深入研究. Qi. Li 等人^[3]提出一种使 TFRC 发送速率从慢启动到拥塞避免阶段过渡更趋平稳的方法. 耿福泉等人^[4]提出了一种基于 RTCP 反馈的拥塞控制机制, 改进了 TFRC, 使得 TFRC 流开始的发送速率不至于太低而使得接收端在开始有较大的延迟和抖动. 文献^[5]对 TFRC 改进, 提出了一种基于多路复用的 TCP 友好速率控制算法, 该算法在保持 TFRC 良好的速率平滑性的同时, 增加了对多媒体流最低速率阈值限定特性及多路复用链路环境的考虑.

2 TFRC 协议机制介绍

TFRC 协议主要采用 Padhye 等人^[6]给 TCP Reno 建立的流量模型来调整数据发送速率:

$$B(p) \approx \min \left(\frac{W_{\max}}{RTT}, \frac{1}{RTT \times \sqrt{\frac{2bp}{3}} + T_0 \min(1, 3\sqrt{\frac{3bp}{8}} p(1+32p^2))} \right) \quad (1)$$

其中 $B(p)$ 为平均享有带宽; W_{\max} 为接收端的窗口通告值; RTT 为平均 round-trip time; b 为单个 ACK 回复所确认的包的数量, 通常的 TCP 实现设定为 1 或 2; p 为分组丢失率; T_0 为 TCP 传输超时的时间设定.

TFRC 采用公式(1)来调整发送端的发送速率, 目标是达到与相同的网络竞争环境下 TCP 传输数据流具有基本相同的吞吐量, 即 TCP 友好的, 同时避免 TCP 的抖动性等不利因素. 其主要的控制机制如下:

接收端测量丢失事件率和接收报文的时间戳, 将其反馈给发送端;

发送端利用接收端反馈过来的信息根据公式

$$R_{\text{sample}} = (t_{\text{now}} - t_{\text{recdata}}) - t_{\text{delay}}$$

计算出当前 R_{sample} , R 初始值设为 R_{sample} , 然后每收到一个报文则按照以下公式计算

$$R = q \times R + (1 - q) \times R_{\text{sample}}$$

式中 R 即用来计算发送速率的参数 RTT , 而 q 一般被设定为 0.95.

接收端根据 RTT 以及报文的时间戳, 决定多个丢包是否归为相同的丢失事件, 而丢失事件则决定了怎样调整发送速率:

如果 $p = 0$, 则进入慢启动阶段, 即每个 RTT 对发送速率倍增, 直到出现报文丢失事件. 如果反馈计时器超时则发送速率直接减半, 这种情况虽然少见, 但它的出现往往标志着网络中的严重拥塞, 必须立即减小发送速率;

如果丢失事件率 $p > 0$, 则发送端把丢失事件率、 RTT 和 t_{RTO} (一般直接设为 $4 * RTT$) 代入由公式(1)在通常情况下的公式(2)

$$x = \frac{1}{RTT \times \sqrt{\frac{2bp}{3}} + t_{RTO} \times 3\sqrt{\frac{3bp}{8}} p(1+32p^2)} \quad (2)$$

计算出合适的带宽.

TFRC 的以上机制确保了它在大部分网络情况下具有稳定的传输性能, 但同时也不可避免的存在一些问题. 王光阳等人^[7]指出: 在链路带宽较小时, 无论比例如何, TFRC 与 TCP 协议间的友好性都比较差. 随着比值增大, 协议间友好性得到改善; 在链路带宽较大时, 无论比例如何, 协议间的友好性都较好, 但随着比值增大, 友好性下降.

也有学者从理论角度分析了 TFRC 发送速率在某些情况下不能达到 TCP 友好的原因^[8,9]. 前者提出 TCP-Friendly 这个概念比较笼统, 应该细分为多个子条件, 用于有区别的判断一个协议在与 TCP 协议共存时的表现, 比如更容易衡量的保守性 (conservativeness)、报文丢失事件等. 还指出了用于 TFRC 发送速率计算的 TCP 速率公式(2)的凹凸性给 TFRC 带来的性能问题. 而 I. Rhee 等人^[9]继续分析了报文丢失事件发生率的一些影响因素, 否定了在同样的网络环境中 TFRC 和 TCP 报文丢失率基本相同这个假设, 指出了 TFRC 速率与预期发送速率严重偏离的一些原因, 得出以下结论:

(1) 采用 $4 * RTT$ 的 RTO 估计导致低网络延迟时, TFRC 的 RTO 相比于 TCP 偏小, 而在高网络延迟的环境中偏大;

(2) TFRC 对丢包的响应比较慢, 于是没有在拥塞发生时及时降低发送速率到一定程度, 导致其丢包率比 TCP 高, 而且由于使用了 loss event rate, 进一步加剧发送速率不同, 与 TCP 所见的丢包率差异也会更大. 并提出了影响 TFRC 吞吐量的三个主要因素: TCP-friendly 公式, 丢包事件率, 网络延迟 (包括 RTO), 同时建议:

① TFRC 使用与 TCP 中同样的 RTO 算法, 由于 TFRC 对丢包事件的响应比 TCP 慢, 它更应该在丢包率时采用更具有竞争性的发送速率, 而在低丢包率时保持跟 TCP 同样的发送速率.

② TFRC 在不同的 RTT 下应该具有跟该 RTT 下相同的 TCP 竞争能力, 而实际上由于 RTO 算法的不同 TFRC 在低延迟网络中的 RTO 比 TCP 小得多, 而在高延迟网络中又比它大得多.

由于 TFRC 协议的以上缺陷, 影响了 TFRC 的友好性, 而且在较高网络拥塞情况下吞吐量会严重下降. I. Rhee 等人^[9]为了改进高拥塞网络状态下的 TFRC 吞吐量下降问题, 对公式(2)分母的后半部分乘以一个常数

c,并指出在极低丢包率的情况下,公式(2)分母后半部分几乎可以忽略,而在高丢包率 p 的情况下,后半部分可以发挥作用,给公式(2)加入一个常数 c 后可以使 TFRC 在更广的拥塞度范围内保持较好的 TCP 友好性.

3 TFRC 友好性与平稳性改进

本文采用 TFRC 和 TCP 传输平稳后一段时间的总吞吐量来考察两者的友好性,友好性可以用 F_r 参数来衡量:

$$F_r = \frac{T_{TFRC}}{T_{TCP}}$$

其中 T_{TFRC} 为 TFRC 的吞吐量, T_{TCP} 为 TCP 协议的吞吐量,为简单起见,即为二者在该时间内发送的总字节数. F_r 值一般在 0.7 ~ 1.5 之间可视为协议具有较好的 TCP 友好性^[10].

对协议传输速率平稳性的评价主要采用基于标准差的公式

$$CoV = \sqrt{\frac{\sum_{i=1}^n (X_i - \bar{X})^2}{n}} / \bar{X}$$

此公式得出的 CoV 值是传输速率标准差(Standard Deviation)与均值的比值,是一个无量纲的值,此值反映出传输速率的波动,值越小表示平稳性越好.

本文在 I. Rhee 等人研究基础上提出一种新的改进方法,使之具有更好的 TCP 友好性,对公式(2)进行修改如公式(3):

$$x = \frac{1}{\alpha \times RTT \sqrt{\frac{2bp}{3}} + t_{RTO} \times 3 \sqrt{\frac{3bp}{8}} p (1 + 32 \frac{1}{\beta} p^2)} \quad (3)$$

为公式(2)分母多项式中 p 的低次幂部分加入一个常数 α , 最高次幂项加入一个常数参数 $1/\beta$. $\alpha = 1, \beta = 1$ 的时候即为原公式(2),由于丢包率 p 幂次的不同,调整 α 和 β 的值可以得到如下效果:

(1) $\alpha > 1$ 时可以使发送速率在丢包率 p 较小的情况下低于原有公式,而丢包率 p 较高的情况下与原公式基本相同;

(2) $\beta > 1$ 时则刚好相反,可以使发送速率在丢包率 p 较高的情况下高于原有公式,而丢包率 p 较低的情况下与原公式基本相同.

公式(2)与公式(3)的理论发送速率值随丢包率 p 的变化关系曲线图如图 1:

以 50ms RTT 为例(采用其他 RTT 值并设置 α 和 β 值效果类似,此图略),标有 1 号的改进函数曲线是 I Rhee 等人^[9]提出的一种参数配置,随丢包率 p 而变化的曲线在丢包率约为 2% ~ 3% 以上发送速率总体高于公式(2)的 0 号曲线.调整 β 值为 30 可以使得标有 2 号

曲线在丢包率约 7% ~ 8% 时开始高于公式(2)的 0 号曲线,这意味着如果将此改进后的公式用于 TFRC 的发送速率控制,将提高在丢包率约为 7% ~ 8% 时的 F_r 值,即改善友好度曲线在丢包率 5% ~ 10% 以上开始偏低这种状况.而调整 α 值为 1.2,即得到图中的 3 号曲线,在丢包率 10% 以下的网络环境中发送速率普遍比原公式得到的速率低,所以 α 值可以用来降低良好网络环境中 TFRC 与 TCP 竞争时的 F_r 值.

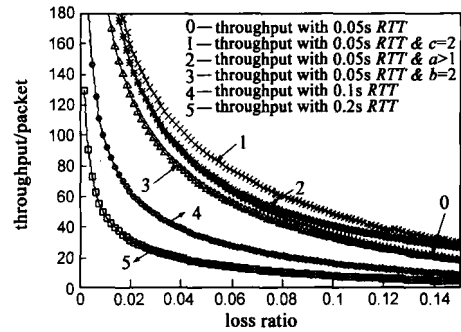


图1 修改TCP吞吐量公式(2)得到的函数曲线

综合调整 α, β 这两个值,将使 TFRC 在不同的网络环境中得到更好的友好性,避免了只加入常数 $1/c$ 提高 TFRC 发送速率,但却会导致在低丢包率网络环境下,与 TCP 流之间的友好性变差的问题.

4 基于 NS 仿真实验的性能分析

为了研究公式(2)的改进对 TFRC 协议的影响,我们使用 NS-2.31 对 TFRC 与 TCP 协议竞争时的情况进行仿真实验^[11],实验采用哑铃型网络拓扑,如图 2 所示,各参数如表 1:

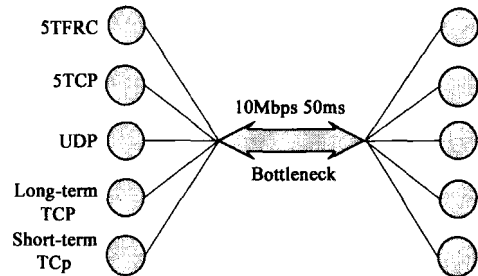


图2 实验采用的基本网络拓扑

表 1 实验基本场景参数

参数名称	参数值
瓶颈带宽	10Mb
瓶颈延迟	50ms
受监控 TFRC 和 TCP 链接带宽	1Mb
受监控 TFRC 和 TCP 链接个数	各 5 个
Long-Term TCP 背景流	5 ~ 250 个
Short-Term TCP 背景流	50 个随机突发
Pareto UDP 背景流	200Kb 峰值带宽突发
路由队列管理算法	RED
实验时间	1200s

实验中最主要用来改变网络拥塞状况的是 5 ~ 250 个 100Kb 的 long-term TCP 背景数据流, 为避免同步效应, 这些数据流采用早期随机产生, 然后长时间的参与网络资源竞争, 另外增加了 short-term TCP 流和突发的 Pareto UDP 流, 以此带来不同程度和随机突发的网络拥塞[6].

通过一段时间的实验与统计, 我们发现公式(3)中 α 、 β 两个值的取值对 TFRC 的各方面性能改变是比较明显的, 但同时由于 TFRC 现有的传输原理已经决定了它所能达到的性能, 所以这种改进方法对 TFRC 的提高也有一定限制. 首先通过大量实验确定 α 、 β 值的基本范围, α 一般可以在 1.0 ~ 2.0 之间取值, 由于网络环境的不同, TFRC 的 TCP 友好性也不一样, α 主要用来减小低丢包率下 TFRC 的发送速率, 避免 TFRC 的友好性指数 Fr 过高, 而 β 一般可以在 30 ~ 50 左右取值, 其作用主要为了增加高网络拥塞情况下 TFRC 的传输速率, 改进 TFRC 在高拥塞网络中的性能. 以下将从 TCP 友好性、传输平稳性以及丢包率三个方面评价此改进方法的效果.

图 3 是 TFRC 原协议和改进后参数定为 $\alpha = 1.2, \beta = 40$ 的 TCP 友好性对比, 不难看出, 以上提出的改进方案是确实有效的, 首先改进后的 TFRC 协议在低于 50 个 TCP 背景流情况下 Fr 值从 1.4 以上降低到 1.3 以下, 最后 200 ~ 250 个 TCP 背景流情况下, 改进后的 TFRC 协议相对提高 Fr 值从 0.7 以下到 0.8 左右, 最后得到整个实验过程中 TFRC 的 TCP 友好性范围在 0.776 ~ 1.271, 而原来的 TFRC 协议在这种网络情况下的 Fr 值在 0.624 ~ 1.455.

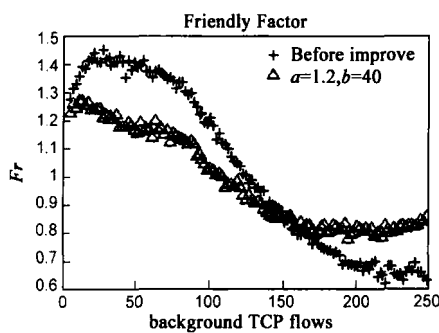


图3 采用 $\alpha=1.2, \beta=40$ 的TFRC协议与原协议的TCP友好性对比曲线

图 4 是 TFRC 使用参数 $\alpha = 1.2, \beta = 40$ 改进后的传输平稳性 CoV 指数与改进前 TFRC 的 CoV 指数对比. 与改进前相比, 改进方案将原协议在 150 个 long-term TCP 背景流以上的高网络拥塞情况下的 CoV 值基本降低到 0.15 以下, 具有明显的改善作用.

由于 α 和 β 值对最后 TFRC 的传输速率是一种综合的影响, α 值提高会降低 Fr 值, 而 β 值提高会提高 Fr

值, 只是显著提高和降低的区段不同, 但并不是单纯的线性提高和降低的作用, 对其表现较接近的几组取值也例举如表 2:

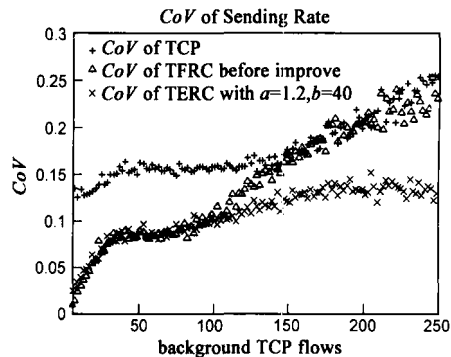


图4 采用 $\alpha=1.2, \beta=40$ 的TFRC协议与原协议的 CoV 对比曲线

表 2 不同的 α, β 值得到的 TFRC 友好度和平稳度

α	β	Fr 值下限	Fr 值上限	CoV 最大值
1.2	30	0.775	1.254	0.146
1.2	40	0.776	1.271	0.152
1.2	50	0.776	1.289	0.154
1.3	30	0.761	1.180	0.141
1.3	40	0.763	1.189	0.146
1.3	50	0.764	1.197	0.138
1.4	30	0.708	1.158	0.142
1.4	40	0.733	1.170	0.139
1.4	50	0.769	1.182	0.138

表 2 是改进时参数 α 和 β 分别取不同数值时的性能对比, 基本反映了参数调整为合适值时可以缩小不同网络拥塞情况下 TFRC 的 TCP 友好度变化范围, 使其较原协议更加平稳, 同时在拥塞程度较高的情况下 TFRC 的发送速率平稳性也有所改善.

5 结论与展望

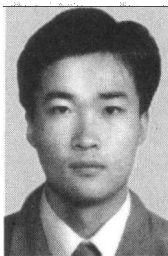
本文研究了不同网络拥塞程度下 TCP 流量公式(2)的分母多项式各项参数的影响力, 通过对 TFRC 速率计算公式中报文丢失事件率 p 的不同幂级项引入权重调整参数, 较大的改善了 TFRC 在不同网络拥塞程度下的 TCP 友好性与传输平稳性, 且使友好性波动范围较原协议更小, 在不同的网络拥塞情况下维持较稳定的协议特性.

由于 TFRC 这类协议是基于 TCP Reno 吞吐量数学模型的, 随着 TCP 协议的不断发展, TCP 新版本的传输性能越来越好, 网络竞争力也越来越高, 这时 TFRC 的弊端便显现出来, 它只能跟 TCP Reno 保持友好性, 而竞争力弱于其他更高的 TCP 版本, 这给大规模推广和可扩展性带来了很大的阻碍. 另一方面, 由于网络环境的各不相同, TCP 协议在高速和低速网络中均有不同的改进版本, 如果要适应这类网络, TFRC 也需要不断完善与改进.

参考文献:

- [1] Qian Wang, Keping Long, Shiduan Cheng, Runtong Zhang. TCP-friendly congestion control schemes in the Internet [A]. IEEE Info-Tech and Info-net International Conference [C]. Beijing: People's Post & Telecommunications Publishing House, 2001. 211 - 216.
- [2] M Handley, S Floyd, J Padhye, J Widmer. TCP Friendly Rate Control (TFRC): Protocol Specification [S]. RFC 3448, January 2003.
- [3] Q Li, D Chen. Analysis and improvement of TFRC congestion control mechanism [A]. IEEE Wireless Communications, Networking and Mobile Computing [C]. Wuhan: IEEE Communication Society, 2005. 2. 1149 - 1153.
- [4] 耿福泉, 方敏, 等. 基于 RTCP 反馈的 TCP 友好的实时流媒体拥塞控制机制 [J]. 东北大学学报, 2006, 27(11): 1204 - 1207.
Geng Fu-quan, Fang Min, et al. A TCP-friendly real-time streaming media congestion control based on RTCP feedback [J]. Journal of Northeastern University, 2006, 27(11): 1204 - 1207. (in Chinese)
- [5] 王东, 陈明, 张大方. 一种基于多路复用的多媒体流 TCP 友好拥塞控制机制 [J]. 电子学报, 2006, 34(3): 567 - 572.
Wang Dong, Chen Ming, Zhang Da-fang. A multiplexing-based TCP-friendly congestion control mechanism for multimedia Streaming applications [J]. Acta Electronica Sinica, 2006, 34(3): 567 - 572. (in Chinese)
- [6] J Padhye, V Firoiu, D Towsley, J Kurose. Modeling TCP throughput: A simple model and its empirical validation [J]. Acm Sigcomm Computer Communication Review, 1998, 28(4): 303 - 314.
- [7] 王光阳, 徐昌彪, 陈前斌. TFRC 与 TCP 流数之比对协议间公平性影响的研究 [J]. 计算机应用研究, 2005, 22(11): 16 - 18.
- Wang Guang-yang, Xu Chang-biao, et al. Research on fairness between FFRC and TCP in terms of ratio of flows [J]. Application Research of Computers, 2005, 22(11): 16 - 18. (in Chinese)
- [8] D Bansal, H Balakrishnan, S Floyd, S Shenker. Dynamic behavior of slowly-responsive congestion control algorithms [J]. Acm Sigcomm Computer Communication Review, 2001, 31(4): 263 - 274.
- [9] I Rhee, L Xu. Limitations of equation-based congestion control [J]. IEEE/Acm Transactions on Networking, 2007, 15(4): 852 - 865.
- [10] S Hassan, M Kara. Simulation-based performance comparison of TCP-friendly congestion control protocols [A]. Proceedings of the 16th Annual UK Performance Engineering Workshop [C]. Durham: Academic Press, 2000. 199 - 210.
- [11] S McCanne, S Floyd. ns-LBNL Network Simulator [CP]. <http://www.nrg.ee.lbl.gov/ns/>, 1996.

作者简介:



姜 明 男, 1982 年生于江苏如皋, 博士, 杭州电子科技大学软件与智能技术研究所副教授. 研究方向为网络服务质量、网络虚拟化和覆盖网.

吴春明 男, 1967 年生于浙江萧山, 博士, 浙江大学计算机学院教授, 博士生导师. 主要从事网络服务质量、可重构网络技术、网络虚拟化和覆盖网、人工智能等领域方向的科学研究工作. E-mail: wuchunming@cs.zju.edu.cn