

文章编号:1001-9081(2006)11-2564-03

大数据量实时视频流单播传输拥塞控制机制研究

宋 健,王 伟

(兰州理工大学 现代网络教育学院,甘肃 兰州 730050)

(nynuping@sohu.com)

摘要:通过对丢包率进行卡尔曼过滤分析预测网络负载状况,结合接收缓冲区大小设置来平滑发送速率,在分析 TCP 友好拥塞控制的基础上,提出了一种基于 RTCP 反馈的 TCP 友好拥塞控制机制。对该机制的 TCP 友好性、视频的回放质量进行了试验和结果分析。

关键词:大数据量;实时;拥塞控制;TCP 友好

中图分类号: TP393.07 **文献标识码:** A

Congestion control mechanism of large quantity real-time video stream unicast

SONG Jian, WANG Wei

(Modern Network Education College, Lanzhou University of Technology, Lanzhou Gansu 730050, China)

Abstract: Based on the analysis of TCP-friendly congestion control mechanism, a TCP-friendly and RTCP feedback-based congestion control mechanism was proposed. The ratio of dropped packet was analyzed by Kalman filter to forecast the status of network load. The sending rate was adjusted by setting up the buffer. The character of TCP-friendly and the quality of playback of this congestion control mechanism was tested and analyzed by network simulation.

Key words: large quantity; real-time; congestion control; TCP-friendly

0 引言

大数据量的视频流实时在网络上传输,对网络设备、网络带宽以及整个网络的负载均衡都带来了极大的挑战。对于多媒体数据的实时传输应用,IETF A/V 工作组制定了实时传输协议(Real-time Transport Protocol, RTP)^[1],提出了一种基于 RTP 协议,利用 RTP 提供的传输控制协议(Real-time Transport Control Protocol, RTCP)进行传输质量反馈,实现多媒体实时传输的解决方案。RTP 协议具有很强的扩展性,采用 UDP 作为传输层协议,有利于时延敏感的实时多媒体应用,但是 UDP 协议本身不提供拥塞控制,而网络上的应用大多采用具有拥塞控制机制的 TCP 协议,大量的 RTP/UDP 流传输,必将产生与传统 TCP 流之间的带宽竞争,缺少拥塞控制的 UDP 流无法适时减少数据发送量,缩小带宽要求,将严重侵占 TCP 流的带宽,严重时会导致网络拥塞崩溃;UDP 本身的尽力传输性还会导致拥塞发生时,将有大量的数据包被抛弃,从而严重影响实时应用的服务质量,同时也不能最大限度提高视频回放的质量。本文在分析 TCP 友好拥塞控制的基础上,提出一种适合大数据量视频流传输、基于 RTCP 反馈的 TCP 友好拥塞控制机制。

1 TCP 友好拥塞控制机制

如果一种通信传输协议与同等条件下的 TCP 流有近似相同的吞吐量,则称这种协议是 TCP 友好的。目前提出的 TCP 友好协议采用的拥塞控制可以分为两类:一种是基于 AIMD^[2](Additive Increase Multiplicative Decrease)的控制机

制;另外一类是以 TFRC(TCP-Friendly Rate Control)^[3]为代表的数学模型控制机制。

基于 AIMD 拥塞退避算法的主要特点是加性增加和乘性减少的速率调节机制,可以用数学公式表示为:

$$\begin{aligned} I: W_{i+R} &\leftarrow W_i + \alpha; \alpha > 0 \\ D: W_{i+\zeta_i} &\leftarrow \beta \times W_i; 0 < \beta < 1 \end{aligned} \quad (1)$$

其中 I 表示在回环时间(Round-Trip Time, RTT)内接收到 ACK 包而引起的窗口增加, α 是增长常量; D 表示遇到拥塞后窗口减小, β 为减小因子; W_i 是 i 时刻窗口的大小, R 是 RTT, α, β 为常数。RAP(Rate Adaptation Protocol)^[4] 协议采用了 AIMD 的速率控制,并对速率改变进行了平滑处理,采用每包确认机制,可以实现与 TCP 流公平分享带宽,但 AIMD 速率调整可能会带来突然、大幅的速率改变,引起传输发生剧烈振荡,这对于实时流媒体应用是很不利的。

基于数学模型的控制机制来源于 Padhye 等提出的 TCP 稳态反应函数,这一反应函数是根据给定丢失事件率和链路回环时间得出平均吞吐量,为了达到 TCP 友好,着重对高带宽流的发送速率进行控制,但速率调整较为平滑,一定程度上有利于实时流媒体传输。但对于带宽要求较大的流媒体数据,采用这种方式在速率调整的同时,大量数据包将会被丢弃以平衡带宽,从而导致回放的视频质量大幅下降。所以在速率调整的同时也必须根据视频帧对回放质量的影响程度有选择地进行码率切换,并且根据网络负载状况对发送速率进行平滑处理。

$$R_{mean} = \frac{S_p}{t_{RTT} \sqrt{\frac{2p}{3}} + 3p(1 + 32p^2) t_{RTO} \sqrt{\frac{3p}{8}}} \quad (2)$$

收稿日期:2006-05-09;修订日期:2006-06-27

作者简介:宋健(1971-),男,江苏徐州人,副教授,主要研究方向:计算机网络通信协议、信息安全;王伟(1977-),男,河南南阳人,硕士研究生,主要研究方向:多媒体通信。

(2)式中 S_p 为数据包大小(单位为 Byte); t_{RTT} 为发送端与接收端之间的回环时间(单位为 s), p 为丢包发生率, t_{RTO} 为 TCP 传输时限(单位为 s)。

2 视频质量的影响帧

对于通过网络传输后的视频数据进行解码,解码的图像质量通常采用峰值信噪比来表示:

$$PSNR = 20 \times \lg \frac{255}{\left\{ \frac{1}{N_1 N_2} \sum_{x=0}^{N_1-1} \sum_{y=0}^{N_2-1} [f(x, y) - f'(x, y)]^2 \right\}^{1/2}} \quad (3)$$

$N_1 \times N_2$ 为图像的大小, f, f' 分别是原始图像和通过网络接收到的数据的解码图像。根据式(3)当网络出现丢包时,视频数据受到影响, $PSNR$ 将随之变小,而且丢包越多, f, f' 差别越大, $PSNR$ 也越小。对于多条标准 MPEG-4 视频流进行传输试验,计算 $PSNR$,可以得出,通常 $PSNR$ 大于 20dB,丢包率小于 1/250 时,视频质量才是可以接受的,并且 P, B, I 帧对视频质量的影响是不同的,视频流中 I 帧比 P, B 帧重要, P 帧又比 B 帧重要^[5]。所以在对视频数据进行封装时,需要对不同的视频帧设置不同的优先级,这可以通过 RTP 扩展报头来实现,在网络传输发生拥塞时,对数据包按优先级进行有选择地丢弃,将会减小回放视频质量的损失。

3 基于 RTCP 反馈的拥塞控制

对于带宽要求高的大数据量视频数据实时传输,必须根据网络状况及时对速率进行调整,并使大量的 RTP/UDP 流与 TCP 流负载均衡,在 TCP 稳态反应函数的基础上,本文提出基于 RTCP 反馈的拥塞控制 (Real-time transport control protocol-based Congestion Control, RCC),合理运用码率切换、采用设置变长接收端缓冲区的方式,对高带宽视频流数据包发送速率进行平滑处理,有效提高回放质量。

3.1 时间参数的测定

3.1.1 回环时间 RTT 的测定

通过在 RTP 扩展报头设置特定的发出时间域,在 RTCP 的报头中增加接收到数据包的时间域,在每次发出数据包时都进行时间域的更新,发送端通过接收端的反馈报告计算出回环时间 RTT。

$$t'_{RTT} = t_{NOW} - t_{LSR} - t_{DLSR} \quad (4)$$

(5)式中 t_{NOW} 为发送端收到 RR (Receiver Report) 的当前时间, t_{LSR} 为接收端接收到来自源端的 SR (Sender Report) 包的时间, t_{DLSR} 为从接收端接到 SR 和发出 RR 之间的延迟。每个 RTCP 报告周期计算出来的 t'_{RTT} 有较大的差异,做如下的平滑处理,得到平滑后的回环时间值:

$$t_{RTT(i+1)} = \delta t_{RTT(i)} + (1 - \delta) t'_{RTT} \quad (5)$$

(6)式中, $0 < \delta \approx 1$,取 $\delta = 0.9$ 。

3.1.2 传输时限的设定

对于大数据量的多媒体实时传输,本文不考虑超时重传的问题,所以设定数据包的传输时限为 $t_{RTO} = 4t'_{RTT}$ 。

3.2 包丢失率的计算

RTP 应用并不直接反馈链路的丢包事件发生率,而只给出 RTCP 报告间隔内丢失包的数目以及丢失包所占的比率。而反馈的包丢失率是网络状况的及时反应,发送端仅利用当前的包丢失比率来改变发送速率将会导致速率的急剧振荡,高带宽的视频流数据将会出现瞬间大量丢包的现象。为了能

得到较好的视频回放质量,必须减小发送速率的振荡,同时能够对下一个时间段的丢包率进行预测,有效利用网络负载。通过统计历史包丢失率,本文运用卡尔曼滤波算法通过迭代过程对 RTCP 间隔内的包丢失率进行预测。取传输过程任意时间段长为 M_{loss} ,设 RTCP 报告间隔 n 时刻时包丢失率为 \hat{p}_n ,这段时间内的包丢失率向量为 $\mathbf{p}_n(n)$:

$$\mathbf{p}_n(n) = [p_n, p_{n-1}, p_{n-2}, \dots, p_{n-M_{loss}+1}]^T \quad (6)$$

取定权值向量 $\mathbf{W}_p(n)$,采用前向校正的误差校正策略,则 $\alpha_n = p_n - \hat{p}_n$,所以:

$$\hat{p}_{n+1} = \mathbf{W}_p^T(n) \mathbf{p}_n(n) \quad (7)$$

$$\mathbf{W}_p(n) = \mathbf{W}_p(n-1) + K_p(n) \alpha_n \quad (8)$$

$$\text{取 } \theta = 0.6, \text{ 卡尔曼增益为 } K_p(n) = \frac{\theta^{-1} \hat{p}_{n-1} \hat{p}_n}{1 + \theta^{-1} \mathbf{p}_n^T(n) \hat{p}_{n-1} \hat{p}_n};$$

$0 < \theta < 2$,取 $\theta = 0.6$, $M_{loss} \approx 1/(1 - \theta)$ 。

3.3 发送端速率调整

根据丢包率的反馈,发送端探测到网络的状况和可用带宽。本文采用在接收端采用设置缓冲区的方式来进行回放视频的平滑处理。接收端缓冲区的大小,至少是一个 RTT 内传输的数据包的大小,接收端缓冲区设置得太大,将严重增加视频的回放时延;接收端缓冲区设置得太小,发送来的数据包将会被丢弃,同样影响视频的回放质量。当网络负载大时,高带宽要求的视频流可以采用码率切换的方式来调整发送速率,发送端有选择地丢弃一定量的视频帧数据,但丢弃数据包的数量也是影响回放质量的重要因素,所以说发送端速率的调整最好能结合网络状况、接收缓冲区的大小以及丢弃数据包的数量综合考虑。本文将这三者有效结合起来,通过预测的丢包率分析网络状态、结合网络状态确定丢包数量、最佳缓冲区大小,最后计算出发送端速率的调整值,进行速率变换,并且在码率切换时采用“保全 I 帧,较少丢弃 P 帧,丢弃 B 帧”的策略。

设接收端视频的平均回放速率为 μ ,接收端期望的发送速率为 δ_n ,发送端的实际发送速率为 λ_n , q_n 为第 n 个 RTCP 间隔 t_n 时接收到的数据包的数目, q_n^v 为第 n 个 RTCP 间隔 t_n 时刻接收端缓冲区的实际大小, \hat{q}_n^v 为第 n 个 RTCP 间隔 t_n 时接收端缓冲区的需求大小, $\bar{\zeta}_n$ 为拥塞发生时丢弃的数据包数目,传输过程的包丢失率为 \hat{p}_n 。

根据公式(2)我们可以计算出 TCP 友好的发送速率为 R_{mean} ,为了达到传输的 TCP 友好,我们设 $\delta = R_{mean}$,当传输过程中带宽充足,那么 $\bar{\zeta}_n = 0$;传输中发生拥塞时,发送端丢弃的数据包的数目 $\bar{\zeta}_n = (\lambda - \delta) t_n$,所以 $\hat{q}_n^v = q_n^v + \bar{\zeta}_{n-1}$,即:缓冲区大小应为当前的长度加上丢弃的数据包数目。可以计算出接收端在第 $n+1$ 个 RTCP 反馈时间间隔 t_{n+1} 时缓冲区应设置为 $\hat{q}_{n+1}^v = q_n^v - \mu t_n + \delta_n (1 - \hat{p}_n) t_n + \bar{\zeta}_n$,综合上面的分析,我们可以计算出发送端的发送速率应为:

$$\lambda_n = \frac{3\mu t_n - q_n^v - \bar{\zeta}_n - \delta_{n-1}}{(1 - \hat{p}_n) t_n} \quad (9)$$

当 $\lambda_n \leq R_{mean}$ 时,表示网络还有多余的链路带宽可以使用,可以按这个速率发送;当 $\lambda_n > R_{mean}$ 时,发送端速率调整为 R_{mean} 。

此后,每当发送端接收到来自接收端的反馈报告,首先根

据(2)式计算出 R_{mean} , 然后对下一个时间间隔应设置缓冲区的大小、丢弃数据包的数量进行分析预测, 最后由(9)式计算出发送端的发送速率, 进行速率的调整。

4 模拟实验与分析

本文使用 NS-2^[6] 作网络模拟, 评估、比较基于 AIMD 的速率控制机制 RAP、基于 TCP 稳态反应函数的 TFRC 以及本文提出的 RCC 的性能。

4.1 模拟实验的网络拓扑及参数配置

端点与路由器之间的瓶颈链路带宽为 10Mb/s, 时延为 10ms, 采用 RED 队列管理(队列长度最小阈值为 5 个包, 最大阈值为 15 个包, 权重 w_q 为 0.02, 临时概率最大值为 0.1), 路由器间的瓶颈带宽为 1.5Mb/s, 时延为 100ms。

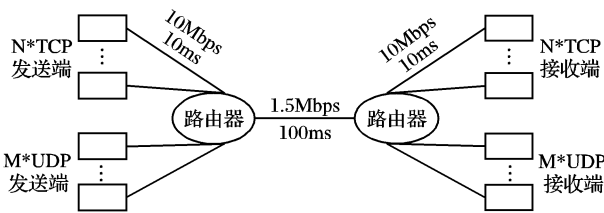


图1 网络模拟拓扑

4.2 实验结果及分析

4.2.1 对 TCP 协议的友好性

采用数据包大小为 10000 字节。我们通过计算协议间的公平指数 F_p^{inter} 作为两种数据流长时公平性的度量。 F_p^{inter} 的定义为:

$$F_p^{inter} = \frac{R_{1/P}}{R_{1/P} + R_p} \quad (10)$$

其中, R_p 为待评估的数据流 P 的归一化平均吞吐量, $R_{1/P}$ 为其他与之竞争带宽的数据流的归一化平均吞吐量, 当 $F_p^{inter} = 0.5$ 时, 我们称它是一种理想的公平。分别对 50% RCC/TFRC/RAP 控制的 RTP/UDP 流与 50% TCP 流进行测试, 结果如图 2 所示, 其中 x 轴表示 TCP 友好流与 TCP 流的总数, y 轴表示协议流的公平指数。可以得出, RTP/UDP 流采用 RCC 拥塞机制在传输过程中, 协议数据流的吞吐量波动平滑, 并且随着协议流总数的变化, 比 TFRC 控制的协议流更趋近于理想的公平状态。

4.2.2 视频回放质量分析

我们通过模拟实际的视频流来进行分析, 设 10min 的时长, 14400 个帧, 28120 个数据包, 其中发送帧率为 30 帧/s, 帧的模式为 IBBPBBPBBPBB; 采用 TFRC 和 RCC 两种拥塞控制, 得出接收端接收到的数据包的百分比, 如图 3 所示, x 轴为网络的归一化负载率, 定义为:

$$W_n^{inter} = \frac{\Delta p_n}{\Delta p_{n-1} + \Delta p_n} \quad (11)$$

其中 Δp_n 为丢包率; y 轴为接收端所收到数据包的百分比, 定义为:

$$A_d^{inter} = \frac{q_n^v}{S_p} \quad (12)$$

其中, S_p 为发送数据包的大小, q_n^v 为接收缓冲区的实际大小。从图 3 可以得出, 整个网络负载较小时, 数据包可以较好地接收, 视频回放质量自然高; 随着网络负载率的变化, TFRC 机制首先对高带宽的数据流进行控制, 丢包率将会增大, 所以接收的数据包也将减少; 而 RCC 控制机制有效进行

速率调整, 从而使接收的数据包的数量的减少比较平滑。

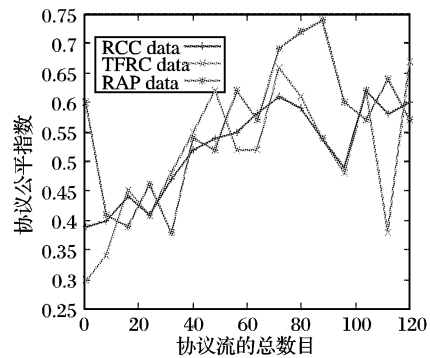


图2 RCC/TFRC/RAP 对 TCP 流公平性比较

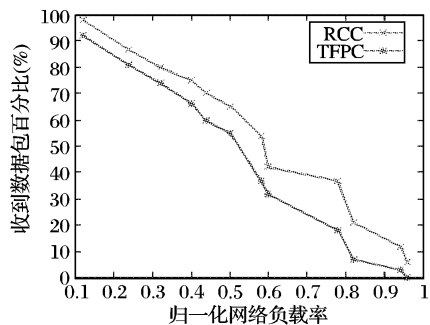


图3 TFRC/RCC 丢包率比较

5 结语

基于 RTP/UDP 的高带宽实时流媒体的广泛应用, 必须要有相应的拥塞控制来提高网络传输的可靠性, 并在合理利用网络带宽的同时, 保证 RTP/UDP 流与 TCP 流之间的负载均衡, 使视频的回放质量达到一定的程度。本文提出的基于 RTCP 反馈的拥塞控制机制, 充分考虑了高带宽视频帧的特点, 结合网络状况分析与变长的缓冲区设置对发送速率进行平滑处理, 弥补了基于数学模型的 TCP 友好拥塞控制机制在大数据量视频流应用中的不足。通过网络模拟实验表明, 该拥塞控制机制具有较好的 TCP 友好性, 并能得到良好的视频回放质量, 由于采用 RTCP 反馈机制, 该控制可以灵活运用至单播或组播的环境。

参考文献:

- [1] SCHULZRINNE H, CASNER S, FREDERICK R, et al. RFC3050, RTP: A Transport Protocol for Real-Time Applications[S/OL]. <http://www.ietf.org/rfc/rfc3050.txt>, 2003-07.
- [2] YANG YR, LAM SS. General AIMD Congestion Control[A]. Proceedings of the ICNP'00, 2000.
- [3] HANDLEY M, PADHYE J. RFC3448, TCP Friendly Rate Control (TFRC) Protocol Specification[S/OL]. <http://www.ietf.org/rfc/rfc3448.txt>, 2003-01.
- [4] REJAIE R, HANDLEY M, ESTRIN D. RAP: An End-to-End Rate-Based Congestion Control Mechanism for Realtime Streams in the Internet[J]. Eighteenth Annual Joint Conference of the IEEE Computer and Communications Societies[C]. 1999, 1337-1345.
- [5] WAH BW, SU X, LIN D. A Survey of Error-concealment Schemes for Real-time Audio and Video Transmission over the Internet[A]. Proceedings of International Symposium on Multimedia Software Engineering[C]. IEEE, 2000, 17-24.
- [6] TOOMEY W, KESHAV S, DUPUY A. The Network Simulator ns-2 [EB/OL]. <http://www.isi.edu/nsnam/ns/index.html>, 2003-12.