

文章编号: 1000-5013(2008)02 0208-05

网上视频传输拥塞控制策略的实现

陈 霞, 蔡灿辉

(华侨大学 信息科学与工程学院, 福建 泉州 362021)

摘要: 探讨在没有服务质量(QoS)保证的 IP 网络中, 根据反向信道反馈的网络状态信息, 自适应调整输出码率, 实现基于信源的实时视频传输控制策略. 提出一种智能速率整形算法并实现在策略控制模块中, 使传输缓冲区的输出速率与网络带宽相匹配. 采用基于 Lagrangian 代价函数的最优前向自适应编码方法, 根据网络带宽和缓冲区状态确定编码参数. 仿真实验表明, 这种算法非常高效, 通常 10 次以内就能达到收敛, 而且它能保证找到的都是质量较好的点, 既合理利用网络资源又不会导致网络拥塞.

关键词: 服务质量; 拥塞控制; 速率控制; 速率整形; 自适应编码

中图分类号: TN 915.04; TN 919.8 **文献标识码:** A

当前基于局域网的实时视频服务, 其服务质量已达到了一个相当高的水平. 这主要是因为局域网内, 带宽相对充裕且变化幅度较小, 传输延迟可忽略, 因而设计高效的拥塞控制策略以保证视频服务有较高的质量并不难. 但当人们希望跨出局域网的范围, 享受整个互联网的实时视频服务时, IP 网络无法预测的带宽, 以及大范围传输的延迟不可忽略地成了必须面对的问题. 因此, 如何实现高效的拥塞控制, 成为互联网上实现有效实时视频传输的关键. 本文探讨在没有 QoS 保证的 IP 网络中, 如何根据反向信道反馈的网络状态信息, 实现传输缓冲区的智能速率整形和自适应调整视频编码器的输出码率, 以实现既能合理利用网络资源又不会促使网络拥塞的, 基于信源的实时视频传输控制策略.

1 拥塞控制流程

通常基于信源决策的网上视频传输拥塞控制, 是由策略控制模块根据信道与缓冲区的情况确定控制策略, 通过控制自适应编码器的编码率和传输缓冲区, 向网络发送数据的速率来实现的^[1]. 基于信源的网上视频传输拥塞控制, 机制在信源端必须包含策略控制模块、自适应视频编码器、传输缓冲区和传输协议栈, 而在接收端必须有接收状态反馈机制. 其基本框架如图 1 所示. 如上所述, 其基本控制流程包括如下 4 个方面. (1) 策略控制模块. 根据实时传输 (RTP) 协议^[2] 反馈的控制信息, 估算网络有效带宽, 并结合传输缓冲区的占有量状态, 确定传输缓冲区向网络发送数据的速率 R_{ref} 和自适应编码器产生码流的速率 R_e . (2) 自适应编码器^[3]. 根据限制条件 R_e 选择编码的最优量化矢量 (即确定最优比特匹配方案), 然后编码输出. (3) 传输缓冲区. 以 R_{ref} 的速率向网络发送数据, 并在策略控制模块中的速率整形功能调节下智能丢帧. (4) 策略控制模块. 获得新的控制信息, 转至 (1).

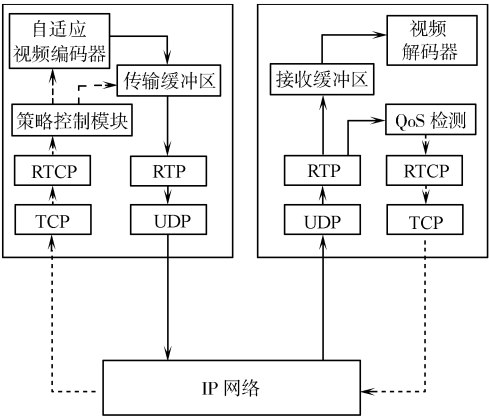


图 1 传输与拥塞控制系统架构
Fig. 1 A framework of transport and congestion control system

收稿日期: 2007-09-25

作者简介: 陈 霞 (1976-), 女, 讲师, 主要从事图形图像的研究. E-mail: xiaoyu@hqu.edu.cn.

基金项目: 国家自然科学基金资助项目 (60472106); 福建省自然科学基金资助项目 (A0410018); 华侨大学科研基金资助项目 (04HZR17)

© 1994-2010 China Academic Journal Electronic Publishing House. All rights reserved. http://www.cnki.net

2 拥塞控制策略与实现方案

2.1 有效带宽的估计

在互联网中,用户的实际有效带宽是不断变化的,而实时视频服务必须建立在对当前有效带宽较准确了解的基础上.所以,首先必须掌握视频服务器与客户端之间的实际有效带宽.为了使控制策略具有传输控制协议(TCP)友好特性,采用类“TCP”公式^[4,5]估计有效带宽,有

$$R_{\text{ref}} = 1.22 \times [U_{T, \max} / t_{RT} \times \sqrt{\rho}], \quad (1)$$

式(1)中, R_{ref} 为某时间段内网络的有效带宽; $U_{T, \max}$ 为最大传输单元,它与具体网络传输环境有关.当网络环境不可知时,默认 $U_{T, \max}$ 值为 576 b, 本文假设为以太网环境,即 $U_{T, \max} = 1\,500\text{ b}$. 往返时间 t_{RT} 为从服务器到客户端往返一次所花费的时间; 分组丢失率 ρ 为短期丢失率, 如一个传输间隔内的丢失率.

从式(1)可知,估算带宽只需知道 3 个参数,它们都可以从构建在实时传输控制协议(RTCP)的报文中获得.这种方法不仅计算简单方便,而且可以与网络中占主流的 TCP 流量比较公平地竞争网络资源,不会对 TCP 的传输产生太大的影响,优势非常明显.

2.2 速率控制

由于互联网是一种变速率传输网络,为了防止网络出现拥塞,信源必须根据网络有效带宽调整自己的传输速率.然而,视频编码自适应控制系统实质上是一个大延迟控制系统^[6],从编码速率设定到数据经互联网传送到用户接收端需要一段延迟时间.那么,这段延迟的长短,以及为考察这段时间内网络带宽所发生变化产生什么影响,需要先考查网上视频传输系统的延迟情况,如图 2 所示.图 2 中, S 为输入帧, S_n 为解码端的重建帧, δ_e 为视频编码器的编码延迟, δ_{be} 为编码器输出在传输缓冲区中的等待传输的延迟时间, δ_{ek} 为帧图像经过信道传输的延迟, δ_{bd} 为图像流到达解码端缓冲区中等待解码的延迟, δ_d 为视频解码器的解码延迟.设第 n 帧视频 S^n 在 T_i^n 时刻进入编码器,在 T_o^n 到达解码端显示.那么,第 n 帧端到端延迟 ΔT^n 可表示为

$$\Delta T^n = T_o^n - T_i^n, \quad \delta_e + \delta_{be} + \delta_{ek} + \delta_{bd} + \delta_d. \quad (2)$$

考虑到一般视频解码延迟 δ_d 较小,为了符号表达简单起见,把解码延迟 δ_d 并入等待解码延迟 δ_{bd} . 即

$$\Delta T^n = \delta_e + \delta_{be} + \delta_{ek} + \delta_{bd}. \quad (3)$$

从实时视频应用的角度出发,实时系统对总延迟的要求较交互系统宽松,而对帧间延迟则有非常严格的限制^[7-8].也就是说,一旦确定了第 1 帧的播放时间,其后任意帧的播放时间也随之确定.假设第 1 帧的延迟为 ΔT^1 , 因为传输的是第 1 帧,在传输缓冲区中无需等待其他数据传输,即 $\delta_{be}^1 = 0$; 而 δ_{bd}^1 是第 1 帧数据在解码缓冲区中等待解码播放的时间.由传输第 1 帧时的网络状况、对延迟的容忍度和解码端缓冲区容量决定,一旦解码就不会因其他因素 δ_{bd} 的改变而改变.为了区别起见用 ΔT_d 表示, 则有 $\Delta T^1 = \delta_e^1 + \delta_{bd}^1 + \Delta T_d$. 从上述分析可知,一旦确定了第 1 帧的延迟 ΔT^1 , 也就限定了其后所有帧的平均延迟时间和最迟到达时间.进一步假设每帧的编码时间 δ_e 基本相等,有 $\delta_e^i + \delta_{ek}^i + \delta_{bd}^i \approx \Delta T^1 - \delta_e$. 上式表明,为了实时显示视频信号,任何帧的编码数据在系统中经历的各种延迟总和必须保持恒定,且直接与第 1 帧延迟限制相关.其中, δ_{ek} 随传输带宽变化而变化,编码传输延迟 δ_{bd} 和解码传输延迟 δ_{bd} 则是为了平滑编码输出、吸纳网络变化的影响、保障延迟限制而引入的.

确切地说,为了让一个实时视频传输系统能正常工作,最重要的一点是要能保证视频在开始播放后有持续稳定的质量.从接收端解码播放的角度考虑,也就是要求所需帧能按时到达,即必须避免接收缓冲区下溢^[9].为简化分析,先不考虑信道延迟和编码延迟,即假设 $\delta_{ek}^i = 0$, $\delta_e = 0$. 这样,不难看出要避免接收缓冲区下溢,只需保证第 i 帧编码结束后,传输缓冲区存放的数据量不超过有效缓冲量 $B_{\text{eff}}(i)$ 即可.

这里,有效缓冲量 $B_{\text{eff}}(i) = \sum_{j=i+1}^{i+\Delta N_d} C_j$, 即从第 i 帧编码结束后到接收端开始解码第 i 帧这段时间内网络所能传送的数据量.其中, i 表示第 i 个帧间隔, C_j 为第 j 个帧间隔的网络传输比特数, ΔN_d 为从第 i 帧

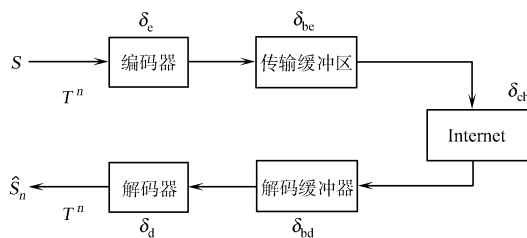


图 2 视频传输系统的延迟情况

Fig. 2 The delay status of the video transport system

编码结束到接收端开始解码第 i 帧所经历的帧间隔数. 由此可知, 第 i 帧的编码率应为

$$R_i = B_{\text{eff}}(i) - [B_{\text{eff}}(i - 1) - C_j] = C_{i + \Delta N_d} \tag{6}$$

由于 $C_{i + \Delta N_d}$ 是将来的传输速率, 不可预知, 因此 $B_{\text{eff}}(i)$ 和 R_i 也都不可预知. 一个合理的解决方法是用当前的速率 C_i 来近似表示 $C_{i + \Delta N_d}$, 即 $R_i = C_i$. 这时, 引入的失配误差 $\Delta R_i = C_{i + \Delta N_d} - C_i$, 这种输入、输出缓冲区的失配不可避免地导致网络拥塞, 使重建视频信号的质量大大下降. 当网络拥塞不是特别严重时, 一种行之有效的拥塞控制策略是, 采样自适应传输速率控制机制, 即根据传输时的网络速率而不是编码速率进行传输. 当 $C_{i + \Delta N_d} < C_i$ 时, 只传输第 i 帧编码的前 $C_{i + \Delta N_d}$ 比特而丢弃剩余的编码数据. 这时, 要求编码器具有内嵌编码功能, 即

$$B_{\text{eff}}(i) = \sum_{j=i+1}^{i+\Delta N_d} C_j = \sum_{j=i+1-\Delta N_d}^i R_j + \sum_{j=i+1-\Delta N_d}^i \Delta R_j \tag{7}$$

2.3 速率整形

考虑到视频编码自适应系统实质上是一个大延迟控制系统, 传输缓冲区的输入、输出失配不可避免. 本文提出一种智能速率整形算法. 实现在策略控制模块中, 使传输缓冲区的输出速率与网络带宽相匹配. 它主要包括两个部分. (1) 基于时间差(网络认知差)的速率整形. 不是根据数据进入缓冲区的速率向网络发送数据, 而是依据策略控制模块最新估算的网络有效带宽发送. (2) 智能丢帧. 一旦缓冲区实际占有量大于最新计算的 $B_{\text{eff}}(i)$ 时, 根据先丢 B 帧, 再 P 帧, 最后 I 帧的原则丢帧. 基于时间差的速率整形, 主要是针对传输缓冲区输入速率和传输速率不匹配的问题, 即传输缓冲区的输入速率必须以最新网络有效带宽的计算值为依据进行发送, 如图 3 所示. 图 3 表明, 缓冲区的吞吐量不匹配, 即 $R_{\text{ref}}(t) \neq R_c(t)$; 或是当缓冲区占有量超过临界值时, 速率整形模块会发挥作用. 它既保障对网络资源的高效利用, 又能较好地处理缓冲区上溢带来的质量严重下降问题. 其中, $R_{\text{ref}}(t)$ 为最近一次计算带宽是在时间 t 的估算值.

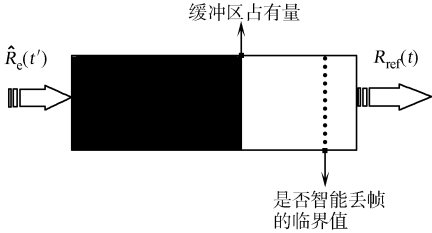


图 3 传输缓冲区状态示意图
Fig. 3 The sketch map of the transport buffer's state

根据对网络带宽最新估算(每接收到一个 RTCP RR 报文计算一次)向网络发送数据, 当因吞吐不匹配导致传输缓冲区有上溢危险时, 有选择性地丢弃某些帧. 这样实现基于时间差的速率整形包括两个部分, 即及时调整发送速率, 以及当缓冲区存在溢出危险时, 如何制定智能的丢帧策略. 对于发送速率的调整频率, 由接收 RTCP 包的频率确定, 每接收一次就相应地调整一次带宽估算值. 对于智能丢帧策略的制定则相对复杂, 具体算法有如下两部分. (1) 开始状态. 记传输缓冲区容量的初始值为 B , 缓冲区占有量为 B_{occ} , 缓冲区最大容量为 B_{max} (这主要是出于发送方的资源有限实时限制的考虑). (2) 确定是否需要丢帧. 根据对失真影响, 从小到大的顺序丢帧. 具体有如下 6 个步骤. (1) 一旦缓冲区占有量 B_{occ} 大于等于达到当前缓冲区容量 B 的 90% 时, 转至步骤(3); 否则, 转至步骤(2). 该步骤用于判断是否有上溢危险. (2) 若 $B_{\text{occ}} \geq B_{\text{eff}}(i)$, 转至步骤(3); 否则转至步骤(1). 该步骤判断是否违反实时限制. (3) 丢弃当前画面组(GOP)的一个 B 帧 $B_{2i}(i = 1, 2, \dots, 5)$, 转至步骤(1); 若丢至 B_{i0} 仍然处于上溢危险区, 转至步骤(4). (4) 丢弃当前 GOP 的一个 B 帧 $B_{2i-1}(i = 1, 2, \dots, 5)$, 转至步骤(2); 若丢至 B_9 仍然处于上溢危险区, 转至步骤(5). (5) 丢弃当前 GOP 的一个 P 帧 $P_i(i = 4, 3, 2, 1)$, 转至步骤(2); 若已经丢弃了 P_1 , 则转至步骤(6). (6) 丢弃 I 帧, 转至步骤(1); 若丢掉 I 帧后还需继续丢帧, 则开始处理下一个 GOP.

3 自适应编码

在实时视频应用中, 自适应调整的准则就是在速率和延迟受限条件下的最小失真. 本文采用基于 Lagrangian 代价函数的最优前向自适应编码方法, 根据网络带宽和缓冲区状态确定编码参数.

算法 1 对给定 λ 设计快速搜索最优 Lagrangian 量化矢量的算法. (1) 对 I 帧所有可选量化步长都用 Lagrangian 函数 $J(Q, \lambda) = D(Q) + \lambda R(Q)$ (Q 为量化步长) 计算成本, 只保留成本最小的点. (2) 保留下来的所有量化选择和 P_1 帧的所有量化选择计算 $J(I, P_1)$, 除去次优的组合和可能会导致传输缓冲区超过 $B_{\text{eff}}(i)$ 组合. (3) 对保留下来的 I, P_1 组合, 选择最小的 $J(B_1) + J(B_2)$, 并去除次优的 $I-B_1-B_2$.

P_1 组合和可能会导致传输缓冲区超过 $B_{\text{eff}}(i)$ 组合. (4) 对所有保留下来的选择组合, 对 $(P_1-B_3-B_4-P_2)$, $(P_2-B_5-B_6-P_3)$ 和 $(P_3-B_7-B_8-P_4)$ 帧集合重复步骤 (2)~ (4).

算法 2 根据网络带宽确定的限制条件, 采用折半搜索法寻找最佳 λ 设初始条件 $\lambda=0, \lambda=\infty$.
(1) 选择 $\lambda \leq \lambda$, 计算是否满足不等式 $\sum_i R_i^*(\lambda_u) \leq R_{\text{ref}} \leq \sum_i R_i^*(\lambda)$. 如果上述不等式中有 1 个相等则停止计算, 表示已经得到一个确切答案, 否则继续. (2) 计算 $\lambda_{\text{next}} \leftarrow \left\lfloor \frac{\sum_i [D_i^*(\lambda) - D_i^*(\lambda_u)]}{\sum_i [R_i^*(\lambda) - R_i^*(\lambda_u)]} \right\rfloor + \varepsilon$, 其中, ε 是一个任意小的正数, 用于保证当 λ_{next} 是一个奇异点时, 可以选取一个较低速率的值. (3) 为选定的 λ_{next} 运行阶段 I 的最优算法, 然后计算. 如果 $\{\sum_i R_i^*(\lambda_{\text{next}}) = \sum_i R_i^*(\lambda)\}$, 那么, 计算停止且 $\lambda^* = \lambda$; 否则, 如果 $(\sum_i R_i^*(\lambda_{\text{next}}) > R_{\text{budget}})$, 就设 $\lambda \leftarrow \lambda_{\text{next}}$, 并转至步骤 (2). 结合算法 1 和算法 2, 能在限制条件下找到一种最佳的各帧量化编码选择.

4 实验结果与分析

为了模拟在 Internet 上的传输, 在网络仿真实验中采用 Cloud 网络仿真软件. 该软件可在实验室条件下仿真端到端广域网, 把标志广域网(WAN)的参数(带宽、延迟和丢包率等)引入到局域网(LAN)环境中. 假设编码器采用独立编码方式每秒固定产生 650 Kb 数据, 发送缓冲区的容量为 3 000 Kb, 接收缓冲区没有容量上限规定. (a) 设 $\Delta N_d = 3$ s, 即接收缓冲区在接收到来自发送端的前 3 s 数据后, 开始解码播放. 采用向安装有 Cloud 软件的服务器指定各时刻传输延迟和丢包率的方法, 来模拟网络传输速率的变化情况, 如图 4, 5 所示. 因为信道带宽相对于源端的实际发送速率不足, 所以在源端的传输缓

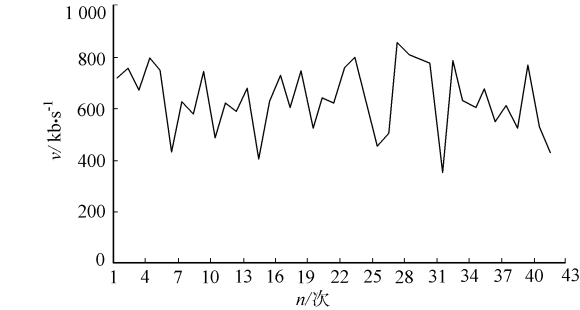


图 4 网络带宽估算情况
Fig. 4 Network bandwidth estimate

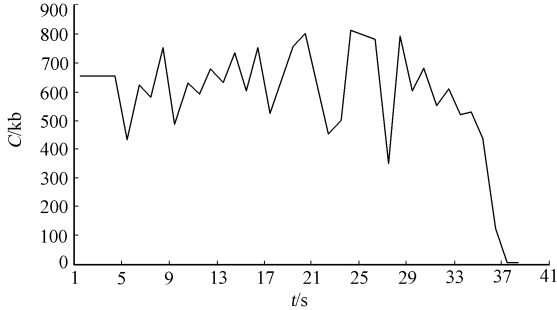
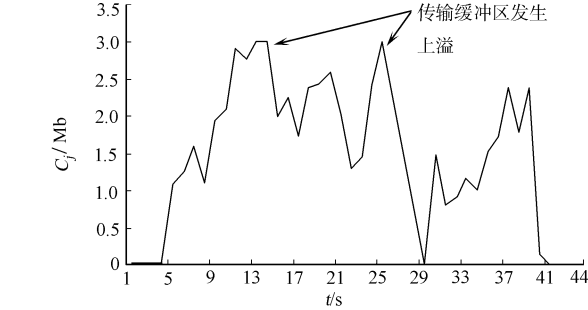
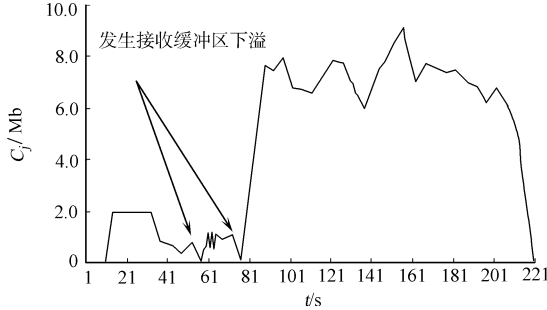


图 5 源端发送速率变化情况
Fig. 5 The source sending rate variety

冲区必然会产生上溢. 本实验依据上文提出的速率整形算法处理该问题, 在尽可能减少失真的情况下, 避免无谓发送或导致网络拥塞的状况发生, 如图 6(a) 所示. 因为网络带宽不足, 而 3 s 的缓冲仍不足以吸纳网络有效带宽的变化, 从而造成应该解码的数据无法及时到达, 即发生接收缓冲区下溢. 接收缓冲区变化情况, 如图 6(b) 所示. 在 $\Delta N_d = 3$ s 且网络状况如图 4 的情况下, 帧丢失率为 20%. (b) 设 $\Delta N_d = 5$ s, 传输缓冲区变化情况不变, 但接收缓冲区情况发生了变化. 当 ΔN_d 从 3 s 增大到 5 s 后, 接收缓冲区



(a) 源端的传输缓冲区



(b) 接收缓冲区

图 6 缓冲区占有量变化情况

Fig. 6 Variety of the buffer occupancy factor

可以较好地吸纳网络带宽的波动,从而因缓冲区下溢导致的数据丢失问题得到了避免.

由于接收缓冲区下溢情况得到缓解,丢失帧数占总帧数的比例下降为 10%,而且这种丢失也是由于传输缓冲区容量不足造成的.如果进一步扩大其容量,那么,丢失率就可以得到更大程度的下降.

5 结束语

本文从 IP 网络的实际情况出发,研究基于信源的拥塞控制策略,获得较理想的效果.本文重点在于整体框架的设计,传输缓冲区和接收缓冲区的大小对控制策略效果的影响,以及根据网络反馈调整发送速率的速率整形算法上.在分析实验数据时,没有考虑自适应编码在本拥塞控制策略中的作用.如果将本系统与自适应视频编码相结合,充分发挥速率整形模块对作用不同帧的选择丢弃功能,就能更好地进行拥塞控制,达到更佳的视频传输质量.

参考文献:

- [1] FLOYD S, ACIRI. Congestion control principles[EB/OL]. [2000-09]. <http://rfc.net/rfc2914.html>
- [2] SCHULZRINNE H, CASNER S, FREDERICK R, et al. RTP: A transport protocol for real time applications [EB/OL]. [2003-07]. <http://rfc.net/rfc3550.html>
- [3] BRADEN R, ZHANG L, BERSON S, et al. Resource reservation protocol (RSVP)[EB/OL]. [1997-09]. <http://rfc.net/rfc2205.html>
- [4] MAHDAVI J, FLOYD S. TCP friendly unicast rate based flow control[EB/OL]. [1997-01]. <http://www.psc.edu/networking/papers/tcpfriendly.html>
- [5] FLOYD S, FALL K. Promoting the use of end to end congestion control in the internet[J]. IEEE ACM Trans Networking. 1999, 7(4): 458-472.
- [6] REIBMAN A R, HASKELL B G. Constraints on variable bit rate video for ATM networks[J]. IEEE Trans on CAS for Video Tech, 1992, 2(4): 361-372.
- [7] DALGIC I, TOBAGI F. Performance evaluation of atm networks carrying constant and variable bit rate video traffic[J]. IEEE J Selected Areas in communications, 1997, 15: 1115-1131.
- [8] LAKHSMAN T V, ORTEGA A, REIBMAN A R. VBR video: Trade offs and potentials[J]. Proceedings of the IEEE, 1998(86): 952-973.
- [9] KRUNZ M, TRIPATHI S. Bandwidth allocation strategies for transporting variable bit-rate video traffic[J]. IEEE Communication Magazine, 1999, 37(1): 40-46.

A Source-based Congestion Control Policy on IP Network for Real-Time Video Transport

CHEN Xia, CAI Canhui

(College of Information Science and Engineering, Huaqiao University, Quanzhou 362021, China)

Abstract: This paper discusses under IP network environment without quality of service (QoS) guarantee, how to adapt output rate according to the network states by backward channel, in order to realize a source-based real-time video transport control policy. This thesis presents a intelligent rateshaping algorithm nested in the policy control module, to insure that transport buffer's output rate matches network bandwidth. This adopts the optimization forward self adaptive encoding based on Lagrangian cost function, to confirm coding parameter by network bandwidth and buffer states. The simulation experiments indicate this algorithm is highly efficient, because it gets convergence within 10 times and can find better quality point, consequently not only utilizes network resource reasonably, but also results in no network congestion.

Keywords: quality of service; congestion control; rate control; rate shaping; adaptive video encoding

(责任编辑: 黄仲一 英文审校: 吴逢铁)