

端到端 MPEG-4 FGS 视频 TCP 友好的平滑传输*

尹浩⁺, 林闯, 张谦, 蒋屹新

(清华大学 计算机科学与技术系, 北京 100084)

End-to-End MPEG-4 FGS Video Transmission Based on Smoothing and TCP-Friendly Rate Control

YIN Hao⁺, LIN Chuang, ZHANG Qian, JIANG Yi-Xin

(Department of Computer Science and Technology, Tsinghua University, Beijing 100084, China)

+ Corresponding author: Phn: +86-10-62796495, E-mail: h-yin@mail.tsinghua.edu.cn, <http://www.tsinghua.edu.cn>

Received 2004-02-10; Accepted 2004-10-09

Yin H, Lin C, Zhang Q, Jiang YX. End-to-End MPEG-4 FGS video transmission based on smoothing and TCP-friendly rate control. *Journal of Software*, 2005,16(5):931-939. DOI: 10.1360/jos160931

Abstract: This paper presents the design of an end-to-end adaptive smoothing and TCP-friendly transmission for stored MPEG-4 fine-grained scalable (FGS) videos over the best-effort Internet. The goal is to minimize the playback quality variation when the network conditions are constantly varying. A novel framework for FGS video delivery over a TCP-friendly connection is first presented. In the context of this scheme, and under the assumption of complete knowledge of bandwidth evolution, an offline quality adaptive smoothing algorithm is derived, and an online adaptive smoothing algorithm is also developed based on the predicted available bandwidth to stream FGS video over the TCP-friendly rate control (TFRC) Protocol with the enhanced ARAR model. Through simulation experiments, it has been shown that the online adaptive algorithm performs almost as well as the offline version for a wide-range of the bandwidth scenarios, and a smooth and TCP-friendly video transfer can be accomplished by the proposed scheme.

Key words: MPEG-4; FGS (fine-grained scalable); TFRC (TCP-friendly rate control); ARAR (autoregressive autoregressive) prediction; adaptive smoothing transmission

摘要: 着重研究了 Internet 上 MPEG-4 FGS(fine grained scalable)视频流的自适应平滑传输,其主要目的在于,在网络带宽变化的情况下,提供稳定的视频回放质量.提出了一种新的基于 TFRC(TCP-friendly rate control)的 MPEG-4 FGS 端到端视频流传输系统框架,在此框架的基础上,首先假设完整的可用带宽变化已知,并且提出了一种离线的自适应平滑算法.此后,给出一种基于改进的 ARAR(autoregressive autoregressive)预测技术的在线自

* Supported by the National Natural Science Foundation of China under Grant Nos.90104002, 60372019, 60473086, 60218003 (国家自然科学基金); the National Grand Fundamental Research 973 Program of China under Grant No.2003CB314804 (国家重点基础研究发展规划(973)); the Postdoctoral Science Foundation of China under Grant No.2003034152 (中国博士后科学基金)

作者简介: 尹浩(1974—),男,湖南宁乡人,博士,主要研究领域为计算机网络,视频通信,信息安全;林闯(1948—),男,博士,教授,博士生导师,主要研究领域为系统性能评价,计算机网络,随机 Petri 网,逻辑推理模型;张谦(1973—),男,工程师,主要研究领域为计算机网络,多媒体通信;蒋屹新(1972—),男,博士生,主要研究领域为模型检测,计算机网络及安全.

适应平滑算法.最后,以 NS-2 为实验平台进行了模拟实验.模拟实验表明,提出的离线和在线自适应平滑算法可以充分利用可用网络带宽,并且能够在可用网络带宽持续波动的情况下保证接收方的回放尽可能地平稳,从而达到获得最佳视觉效果的目的.

关键词: MPEG-4;FGS;TFRC;ARAR 预测;自适应平滑传输

中图分类号: TP393 文献标识码: A

近年来,视频点播已成为 Internet 上最具活力和竞争力的服务之一.由于 Internet 上的可用网络带宽在多个时间尺度上均呈现出动态的特性^[1],而且视频传输本身具有严格的时限要求,所以在 Internet 上提供稳定的视频传输服务仍然极具挑战性.尽管通过在客户端建立缓冲区,利用播放时限到达前预取部分数据的方法可以有效地补偿传输资源在较小时间尺度内的波动,但由于缓冲区大小的限制,这种方法不能补偿较大时间尺度内传输资源的变化影响^[2].对于大时间尺度内的波动,通常采用分层编码技术,例如 MPEG-4 FGS(fine grained scalable)编码^[3],根据当前网络传输资源,有选择地传输其中的部分子层或全部子层.在这种情况下,接收方收到的数据越多,回放的质量就越好.但是一般普遍认为,一个图像质量稍差但很稳定的视频流给人的视觉感受要优于一个虽然质量很高但是频繁变化的视频流,所以保持视频流质量的稳定是很重要的^[4].当传输资源不足,必须丢弃一些视频数据时,我们希望在充分利用传输资源的条件下,尽量减少视频流回放时图像质量的波动^[1,5].因此,我们面对的问题是:如何以保持最稳定的回放质量为目标确定相应的传输策略,即在当前的带宽和接收方缓冲区的限制下,确定哪些层的视频数据需要传输,哪些层的视频数据要被丢弃以便保证视频流回放时的质量波动最小,我们将其称为视频流的自适应平滑传输.与此同时,另一种被广泛接受的观点是:在 Internet 上传输视频流时,应该保持与 TCP 传输的公平性^[6-8].按照文献[9]的定义,如果一个流的传输速率不超过在相同网络链路路上的 TCP 连接的传输速率,就被称为是 TCP 友好的传输.近来已相继提出了一些 TCP 友好速率控制算法^[9],由 RFC3448 提出的 TCP 友好速率控制(TCP-friendly rate control,简称 TFRC)协议具有在网络阻塞发生时平滑地调节发送速率的功能.因此在我们的视频传输框架中选择 TFRC 协议,以便保持视频传输过程中的公平性原则.

在本文中,我们在已有工作^[1,10]的基础上,提出了一种基于 TFRC 协议和自适应平滑技术的端到端的 MPEG-4 FGS 视频流传输系统框架,以最大限度地利用可用带宽,并尽量减少视频流回放的质量波动.本文的主要贡献在于:(1) 提出一种新型的基于 TFRC 的 MPEG-4 FGS 视频流传输框架.(2) 对已有的 ARAR (autoregressive autoregressive)算法进行改进,并利用它预测可用网络带宽.(3) 提出一种自适应平滑算法,在可用带宽及接收方缓冲区大小受限情况下提供最为平滑、稳定的回放效果.我们提出的视频流自适应平滑传输在如下方面与已有工作有所不同:文献[1]中的图像质量平滑算法仅适用于 CBR 的分层编码视频流,我们提出的自适应平滑算法适用于以 VBR 形式编码的分层视频流.文献[10]中提出的在线自适应算法采用 MA 预测技术,该方法过于简单,而且不适用于非平稳过程.我们采用改进的 ARAR 预测模型,使预测模型具有更好的自适应能力和更高的预测精度.

1 端到端视频传输系统框架

图 1 所示为本文采用的基于 TFRC 和质量自适应技术的端到端视频传输系统框架图,其中包括:(1) 媒体服务器上的基于 MPEG-4 FGS 编码的视频流.在我们的框架中,视频流分为 3 层,分别为:基层(BL 层)、SNR FGS 层(FGS 层,该层通过降低量化步长而获得较高的 DCT 系数,从而得到较高的图像质量)、时间 FGS 层(TFGS 层,该层通过提供更高的帧率以及运动平滑来获得更高的时间解析度),这 3 层均采用 VBR 编码方式.(2) TCP 友好传输速率控制(TFRC)模块,其功能为:以 TCP 友好的方式在基于 IP 的网络上传输“平滑”后的视频流数据,同时,将每秒钟的平均可用带宽测量值传送给带宽预测模块.(3) 可用带宽预测模块,其功能为:采用改进的 ARAR 预测技术,根据由 TFRC 协议所得到的可用网络带宽观测值,预测后续可用带宽的变化.(4) 图像质量自适应平滑模块,该模块的主要功能是:利用客户端缓冲区和对可用带宽的预测,确定当前时隙中哪些层的 VOP(video object plane)可以传输,哪些层的 VOP 应该丢弃.在接收端,分别为 BL 层、FGS 层、TFGS 层提供各自的数据缓

缓冲区,最后,来自 3 个缓冲区的视频数据经同步后开始播放.

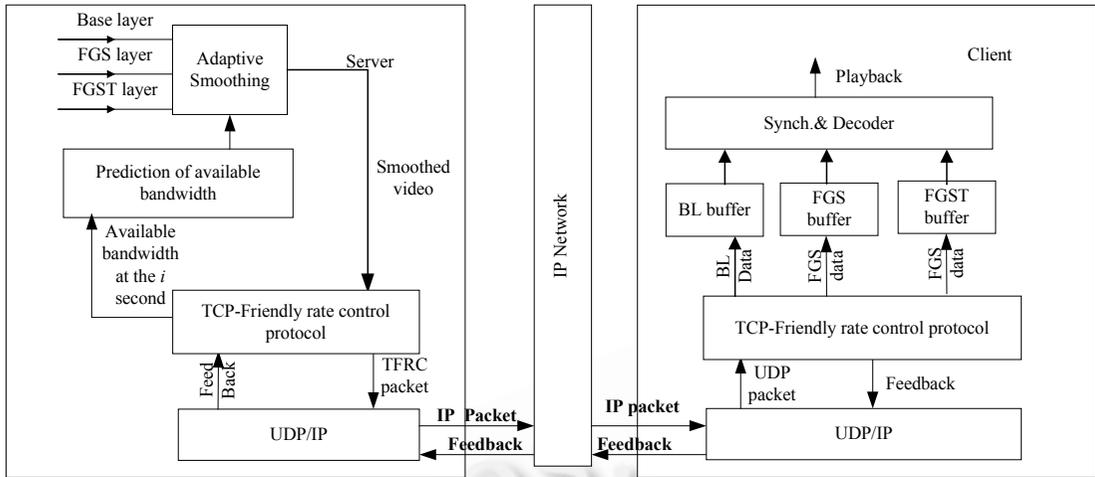


Fig.1 End-to-End video transmission framework based on TFRC and adaptive smoothing

图 1 基于 TFRC 和质量自适应技术的端到端视频传输框架

图 2 所示为以 TCP 友好传输方式在 IP 网络上传输 MPEG-4 FGS 视频流的协议栈.我们的传输系统框架具有如下主要特点:

- (1) 采用分层可变速率编码的视频流.采用分层可变速率的视频流可以根据网络带宽的变化情况动态地选择传输内容,在可用网络带宽不足的情况下可以有选择地丢弃相对不太重要的内容,以便保证较重要的内容能够得以传输.这样一来,保证了在严格的信道条件下得到最佳的视觉效果.
- (2) 回放质量平滑.在网络可用带宽充足的情况下,除了传输当前时隙内的 VOP 之外,还有部分剩余的带宽.这些带宽被最大限度地用来提高视频流回放的质量.在我们的框架中,已经压缩编码的视频流的所有情况都是已知的,所以可以利用动态平滑技术在网络传输资源的情况变坏之前在接收端缓冲区内预取部分数据,以便最大限度地提高收视效果.
- (3) 公平性.视频流以 TCP 友好的方式传输.

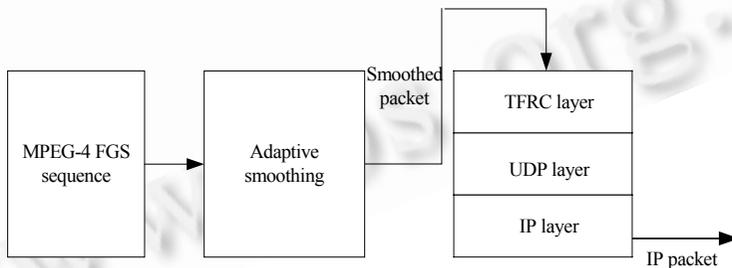


Fig.2 Protocol stack of TCP-friendly MPEG-4 FGS video transmission over IP network

图 2 以 TCP 友好传输方式在 IP 网络上传输 MPEG-4 FGS 视频流的网络协议层次结构

2 TCP 友好传输和可用网络带宽的预测

TFRC 是一种传输速率控制算法,利用 TFRC 不仅可以使得非 TCP 连接具有与 TCP 连接相似的行为特性,而且还可以使得 TFRC 连接与相同链路上的 TCP 连接相比有更加平稳的输出速率.TFRC 连接可以对丢包率等指示的网络拥塞状况的指标作出回应,接收方如果收到了来自发送方的信息,则至少在每个传输延迟时间 (round trip time, 简称 RTT) 向发送方传送反馈信息,反馈给发送端接收端估计到的丢包率与接收速率等信息.发送方根据式(1)^[11]动态地调整其传输速率以回应接收方的反馈.如果发送方在两个 RTT 内未接到来自接收方的任何反馈,则将发送速率减半.

$$b(l) = \frac{S}{t_{RTT} \sqrt{\frac{2bl}{3}} + t_{RTO} \sqrt{\frac{3bl}{8}} l(1+32l^2)} \tag{1}$$

其中 l 是包丢失率, t_{RTT} 是 TCP 连接上的端到端传输延迟, t_{RTO} 是时限值, S 是数据包的大小, $b=1$.

为了在 TCP 友好传输的基础上进行图像质量的自适应传输, TFRC 每秒钟将收集到的网络信息传递给预测模块, 预测模块将该信息作为观测值, 据此预测网络带宽的变化. 预测模块是整个传输框架中最重要的模块, 无论采用何种自适应传输策略, 准确的网络带宽预测都是必须先决条件. 考虑到对预测准确性的要求、预测算法的计算复杂度和可用网络带宽的高度可变性, 我们对时间序列分析中的 ARAR 模型^[12]进行改进, 提出一种基于改进的 ARAR 模型的预测算法.

ARAR 算法的基本思想是对一个具有“长程记忆”特性的时间序列实施自动的“记忆缩短(memory-shortening)”变换, 并使用 Sub-AR 模型拟和变换所得的数据序列. 使用 Sub-AR 模型的优点是可以减少模型参数的个数, 由于每一个参数的估算都会引入一定的误差, 所以过多的模型参数会影响预测的精度. 但是也正是因为采用的是 Sub-AR 模型, 造成了 ARAR 模型的一个不足: 预测模型中缺少 MA 部分. 我们对各种 Box-Jenkins 类的线形时间序列模型的预测能力进行了全面比较, 结果发现, 具有 MA 部分的模型的预测能力优于相同阶数的纯 AR 模型. 为此, 我们改进了 ARAR 模型, 采用 Sub-AR 与 MA 相结合的方式拟和“记忆缩短”后的数据序列. 我们改进后的 ARRA 算法可以描述为:

(1) 记忆缩短变换(LMS): 采用扩展的差分变换对一个长程记忆的数据序列 Y_t 进行记忆缩短变换, 得到序列 X_t .

(2) 通过 Yule-Walker 方法, 以最小化 AIC 值的方式自动获得最佳的 P 个 AR 参数, 计算这 P 个估算参数的标准差, 如果某参数与 1.96 倍的标准差的比值小于 1, 则我们认为这个参数可以被设置为 0, 由此可以确定步长 l_1, l_2 和 l_3 . X_t 的 Sub-AR 模型可以表示为

$$X_t = \phi_1 X_{t-1} + \phi_2 X_{t-2} + \phi_3 X_{t-3} + Z_t, Z_t \sim WN(0, \sigma^2) \tag{2}$$

(3) 按式(3)计算残差 \hat{Z}_t , 以最小二乘法按最小化式(4)的方式估算参数序列 $\beta = (\phi_1, \phi_2, \phi_3, \theta_1, \theta_2, \dots, \theta_q)$:

$$\hat{Z}_t = X_t - \hat{\phi}_1 X_{t-1} - \hat{\phi}_2 X_{t-2} - \dots - \hat{\phi}_p X_{t-p}, t = p+1, \dots, n \tag{3}$$

$$S(\beta) = \sum_{t=m+1+q}^n (X_t - \hat{\phi}_1 X_{t-1} - \hat{\phi}_2 X_{t-2} - \hat{\phi}_3 X_{t-3} - \hat{\phi}_4 X_{t-4} - \theta_1 \hat{Z}_{t-1} - \dots - \theta_q \hat{Z}_{t-q})^2 \tag{4}$$

其中, q 值由最小化 $AIC(\beta)$ 的方式获得, 白噪声序列 Z_t 的方差为

$$\sigma^2 = \frac{S(\hat{\beta})}{n-p-q}$$

下面我们通过一个实例比较 ARAR 模型和改进后的 ARAR 模型的预测能力. 假设某一网络带宽序列如图 3 所示. 我们截取其中的前 150 个网络带宽观测值 Y_t . 首先对该网络带宽序列进行缩短记忆变换, 利用扩展差分变换算法可以得到如图 4 所示的缩短记忆变换后的序列 X_t (已经去除了均值).

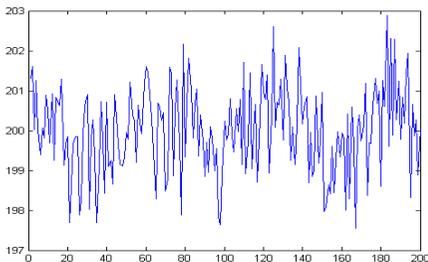


Fig.3 Network bandwidth sequence

图 3 网络带宽示意图

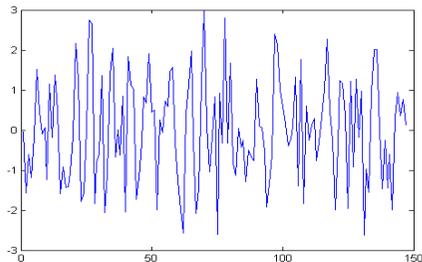


Fig.4 Network bandwidth sequence after LMS

图 4 缩短记忆变换后的网络带宽示意图

我们考察两种模型的一步预测误差, 网络带宽序列的第 151 个观测值为 197.980 6. 见表 1, 以 ARAR 模型得

到的一步预测值为 199.66,预测均方误差为 1.074 36.以改进的 ARAR 模型得到的一步预测值为 198.24,预测均方误差为 0.994 77 可见,改进后的 ARAR 模型的预测能力较之原始 ARAR 模型的预测能力有所提高.

Table 1 Denotations in adaptive smoothing algorithm

表1 自适应平滑算法中的符号标识

Bandwidth $[i]$:	Aavailable bandwidth for layer I
N :	length of the video stream
$X_i[i]$:	Vop size of layer I
B_i :	Client buffer size of layer I
$S_i[i]$:	Select or discard indicator for VOP
Flag:	State indicator select state or discard
discard_first_vop:	The first vop at which a discard state begin
discard_last_vop:	The last vop at which a discard state stop
bandwidth_observation $[i]$:	The past observation of the available bandwidth
num_prediction:	The length of predicted available bandwidth
prefetch_data:	The sum of data that can be fetched in advance

3 自适应平滑算法

自适应平滑算法利用接收方缓冲区,根据可用带宽的预测值制定出相应的传输策略,传输策略由连续丢弃过程和连续传输过程组成,为了减小接收方回放质量的波动,应尽量增长每一个连续传输过程的长度^[1,10].为了达到这个目的,我们利用客户端缓冲区减少带宽波动给视频传输带来的影响,只有在已无可用于预取的额外带宽或客户端缓冲区已满的情况下,才结束连续传输过程并转入连续丢弃过程.根据文献[10]的论述,一个丢弃过程应持续到接收方缓冲区可以被充满时停止,并转入下一个传输过程.为了传输多个层的视频流,我们采用这样一个传输策略:可用带宽首先分配给 BL 层,然后其剩余部分再分配给扩展层 FGS,TFGS 层.这样低层的视频流可以得到最好的传输保障,并且在同一个时隙内,如果下层视频流未被传输,则上层视频流也不会被传输.

首先假设整个传输过程中的带宽变化情况事先已知,我们提出了一种离线自适应平滑算法,具体算法见表 2,其中所用标识见表 1.算法分为初始化、前向扫描和后向扫描阶段.初始化阶段的主要任务是识别出第 1 个丢弃过程结束处的 VOP 的序号 k .这 k 个时隙内的 VOP 将被丢弃,在此期间的可用带宽用于预取后续的 VOP,前向阶段识别出哪些 VOP 需要预取.后向扫描阶段的主要目的是在可用网络带宽和客户端缓冲区容量允许的情况下,尽可能长地向前延伸连续选择过程的长度.下面,我们以图 5 为例说明其具体执行过程.一个 VOP 序列 1~20 如图 5 所示,为了简单起见,我们假设在整个 VOP 序列内的可用网络带宽均为 2,接收方缓冲区的大小设定为 3.图 5 中分别是 VOP 序列图和第 1,2 个选择传输过程的缓冲区占用示意图.

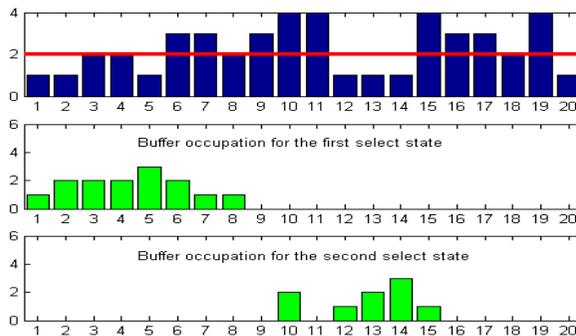


Fig.5 VOP sequence and buffer occupation

图 5 VOP 序列及缓冲区占用示意图

算法从一个丢弃过程开始,首先计算最多持续丢掉多少个时隙的 VOP 可以将缓冲区充满.如图 5 所示,如果连续丢掉第 1、2 个时隙内的 VOP 并将其可用带宽用于后续 VOP 的预取,则共有 4 个单位的带宽可用于预取,已经足以充满接收方的缓冲区,所以连续丢弃这两个时隙内的 VOP 就可以了.第 1 个持续传输过程从第 3 个 VOP 开始.在一个持续传输过程中,首先识别出由于带宽限制而不能被传输的第 1 个 VOP,在我们的例子中为第

6 个 VOP. 由于有剩余可用带宽, 并且接收方缓冲区未充满, 所以第 6 个 VOP 可以通过预取 1 个单位的数据而被传输. 按同样的方法处理, 直到第 9 个时隙的 VOP 后接收方缓冲区已经充满, 所以第 10 个时隙内的 VOP 被放弃, 算法再次进入丢弃过程. 丢弃两个 VOP 后, 第 2 个持续传输过程从第 12 个 VOP 开始, 其后按与上述相同的方法可知该持续传输过程到第 17 个 VOP 停止. 上述处理有一个缺陷, 即第 12, 13, 14 个时隙内有额外的可用带宽未被充分利用, 如果将其也用于后续的 VOP 预取, 则有可能减少前面的连续丢弃过程的长度. 所以, 我们的算法中有一个后向扫描过程, 对这部分额外可用网络带宽加以利用. 在我们的例子中, 第 15, 16 个 VOP 可以通过第 12, 13, 14 个时隙的预取传输, 此时, 第 11 个时隙内的 VOP 也可以被传送而无须丢弃. 与此类似, 对于第 1 个持续传输过程, 通过后向扫描可以发现, 第 1, 2 个 VOP 同样也可传输而无须被丢弃.

在表 2 所述的离线质量自适应算法中, 我们假设整个传输过程中的带宽变化情况已知. 下面, 我们去掉这个假设, 给出一种在线自适应平滑算法. 在在线自适应平滑算法中, 后续可用带宽的变化由改进的 ARAR 预测算法给出, 出于效率和准确性的考虑, 我们使用已有的 40 个可用带宽观测值预测后 5 秒钟的带宽值, 除此之外, 在线自适应平滑算法大致上和离线质量自适应算法相同, 具体算法见表 3. 由于受到预测精度和预测长度的限制, 在线质量自适应算法并不能给出最优的自适应方案, 但是与不采取自适应调整的传输策略相比, 在线自适应平滑算法仍然可以获得较稳定的视觉效果.

Table 2 Offline adaptive smoothing algorithm

表 2 Offline 自适应平滑算法

```

Proc OfflineSmooth(bandwidth[], x[], bi)
  Initialization: discard_first_vop=0; buffered[]=0; k=0; prefetch_data=0;
  While x[k]+prefetch_data<bi do
    Prefetch_data+=x[k]; k++;
  End do
  Discard_last_vop=k;
  For j=k to N-1 do
    //forward scan to detect the last VOP of a select state
    If bandwidth[j]>=Xi[j]
      S[j]=1; flag=1;
    Else if prefetch_data>=Xi[j]-bandwidth[j]
      S[j]=1; prefetch_data=Xi[j]-bandwidth[j]; flag=1;
    Else
      S[j]=0; flag=0;
    End if
    // backward scan to extend to select state as long as possible
    If flag=0
      For n=j to discard_first_vop+1 do
        Buffered[n-1]=buffered[n]-bandwidth[n]+x[n];
      End do
      For m=discard_last_vop to discard_first_vop do
        If bandwidth[m]>=xi[m]+(buffered[m]-buffered[m-1])
          S[m]=1;
        End if
      End do
      Prefetch_data=0; discard_first_vop=j;
      While j<=N-1 and x[j]+prefetch_data<bi do
        Prefetch_data+=x[j]; j++;
      End do
      Discard_last_vop=j;
    End if
  End do
  Updatebandwidth(s[], bandwidth[]);

```

Table 3 Online adaptive smoothing algorithm**表 3** Online 自适应平滑算法

```

Proc OnLineSmooth(bandwidth_observation[],x[],b)
Initialization: discard_first_vop=0; buffered[]=0;k=0;prefetch_data=0;
//forecast the available bandwidth
Bandwidth[]=ARAR_prediction(bandwidth_observation[],num_prediction);
While x[k]+prefetch_data<bt do
  Prefetch_data+=x[k];k++;
End do
Discard_last_vop=k;
For j=k to num_prediction do
//forward scan to detect the last VOP of a select state
  If bandwidth[j]>=Xt[j]
    S[j]=1;flag=1;
  Else if prefetch_data>=Xt[j]-bandwidth[j]
    S[j]=1;prefetch_data=Xt[j]-bandwidth[j];flag=1;
  Else
    S[j]=0;flag=0;
  End if
//backward scan to extend to select state as long as possible
  If flag=0
    For n=j to discard_first_vop+1 do
      Buffered[n-1]=buffered[n]-bandwidth[n]+x[n];
    End do
    For m=discard_last_vop to discard_first_vop do
      If bandwidth[m]>=xt[m]+(buffered[m]-buffered[m-1])
        S[m]=1;
      End if
    End do
    Prefetch_data=0;discard_first_vop=j;
  While j<=N-1 and x[j]+prefetch_data<bt do
    Prefetch_data+=x[j];j++;
  End do
  Discard_last_vop=j;
End if
End do
Updatebandwidth(s[],bandwidth[]);

```

4 模拟结果

为了验证本文提出的框架和 TCP 友好的自适应平滑算法的有效性,我们以 NS-2^[13]为实验平台进行了 MPEG-4 FGS 视频流的端到端传输模拟实验.如图 6 所示为基于 NS-2 的网络拓扑结构.

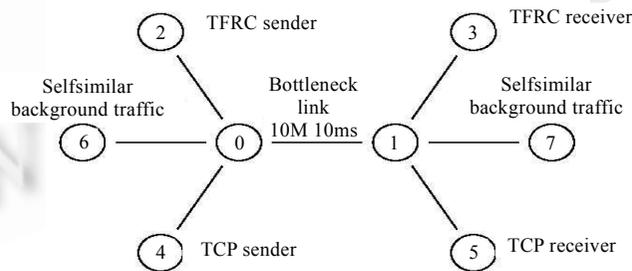
**Fig.6** Network topology of simulation

图 6 模拟实验的网络拓扑结构

在实验中,节点 6 到节点 7 之间加入具有自相似特性的背景流,从节点 4 到节点 5 的背景流为 TCP 流,从节点 2 到节点 3 的流为基于 TFRC 协议的视频流.

在实验中,采用 MPEG-4 FGS 方式压缩编码的视频流“Jurassic Park”的数据文件^[14].其编码参数为每个 GOP 包含 12 个 VOP,每秒钟两个 GOP,整个视频流分为 BL 层、FGS 层、TFGS 三层,每一层都是以 VBR 形式编码,图 7 为视频流“Jurassic Pack”的 BL 层、FGS 层、TFGS 层示意图.在离线和在线算法的实验中,接收方的缓冲区

大小均被设定为 600KB,并按 1:8:16 的比例分别分配给 BL 层、FGS 层、TFGS 层.图 8 给出了 TFRC 的带宽变化和基于改进的 ARAR 算法的预测结果,从图 8 中可以看到其平均稳态吞吐量大致为 3Mbps.如图 8 所示,我们的带宽预测值和真实值很接近,且可以很好地“追踪”网络带宽的变化趋势.

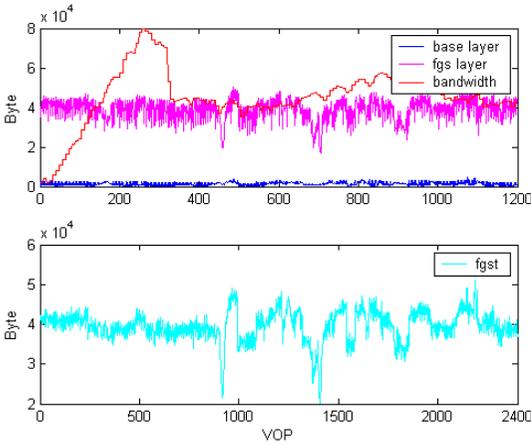


Fig.7 BL,FGS,TFGS layer of “Jurassic Park”

图 7 视频流“Jurassic Park”的 BL,FGS,TFGS 层

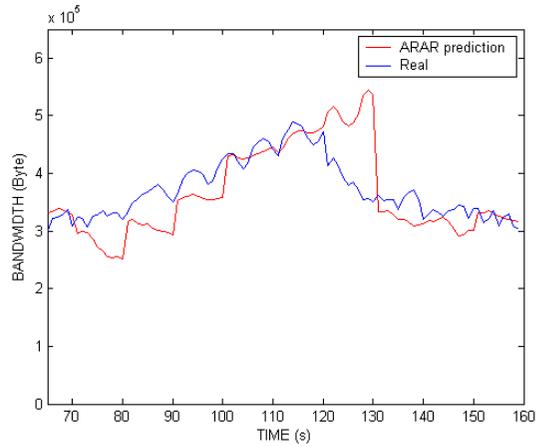
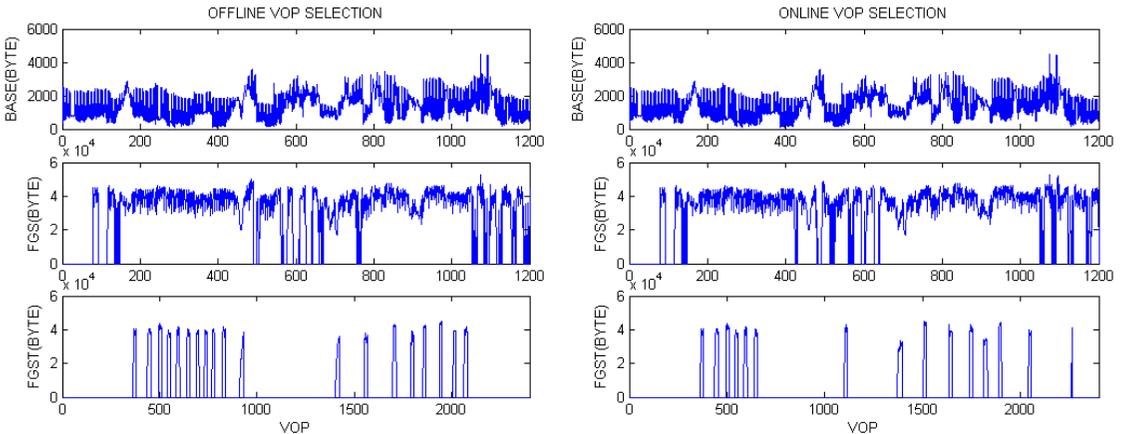


Fig.8 TFRC bandwidth variation and ARAR prediction

图 8 TFRC 带宽变化及其 ARAR 预测

实验分为两个步骤进行,首先通过 NS 模拟获得 TFRC 的可用带宽数据,然后以这些数据为输入进行离线的质量自适应平滑实验.由图 9(a)可以看出,当网络状态持续不理想,TFRC 的可用网络带宽持续走低时,离线自适应平滑算法可以保证回放的相对平稳,从而最大限度地确保获得最佳的视觉效果.实验的第 2 步,结合改进的 ARAR 预测算法,在 NS-2 上进行在线自适应平滑算法实验.为了比较离线和在线算法的效果,我们保持相同的网络设置,图 9(b)为在线情况下的 VOP 传输情况.



(a) Offline
(a) 离线

(b) Online
(b) 在线

Fig.9 VOP selection of adaptive transmission

图 9 自适应传输的 VOP 选择示意图

可以看到,在两种情况下,BL 层均被完整地传输到接收方,其后剩余的可用带宽被依次分配给 FGS 层和 TFGS 层.由于可用带宽不足,所以 FGS 增强层中的一些位平面(bitplane)被丢弃,并且 TFGS 层中大部分位平面也被丢弃.虽然这造成了 FGS 层和 TFGS 层的不连续,但是被选择传输的 VOP 都是尽量保持连续的,所以可以保证播放质量尽可能地“平滑”.

5 结 论

本文着重研究了基于 TFRC/UDP 的 MPEG-4 FGS 视频流的自适应质量平滑传输.我们的目的在于,提出一种低复杂度、高效率的可自适应网络带宽变化的端到端传输方案.为此,我们提出了一个新的基于 TFRC/UDP 的 MPEG-4 FGS 视频流传输系统框架.在该框架的基础上,我们首先假设完整的可用带宽变化已知,并且提出了一种离线的质量自适应算法.此后,我们进一步去除该假设条件,给出一种基于改进的 ARAR 预测技术的在线质量自适应平滑算法.为了验证自适应平滑算法的有效性,我们在 NS-2 模拟平台上进行了离线和在线算法的模拟实验.实验结果表明,我们提出的离线和在线自适应平滑算法可以充分利用可用网络带宽,并且保证接收方的播放尽可能的平稳,从而达到获得最佳视觉效果的目的.

References:

- [1] Nelakuditi S, Harinath RR, Kusmierek E, Zhang ZL. Providing smoother quality layered video stream. In: Proc. of the NOSSDAV. Chapel Hill, 2000. 1–10.
- [2] Salehi JD, Zhang ZL, Kurose J, Towsley D. Supporting stored video: Reducing rate variability and end-to-end resource requirements through optimal smoothing. IEEE/ACM Trans. Networking, 1998,6(4):397–410.
- [3] Li WP. MPEG-4 video standard. IEEE Trans. On Circuits and Systems for Video Technology, 2001,11(3):301–317.
- [4] Zhang J, Joseph H. Applying traffic smoothing techniques for quality of service control in VBR video transmissions. Computer Communications, 1998,21(4):375–389.
- [5] Wu DP, Hou YT, Zhu WW, Zhang YQ, Peher J. Streaming video over the Internet: Approaches and directions. IEEE Trans. on Circuits and Systems for Video Technology, 2001,11(3):282–300.
- [6] Cuetos PD, Ross KW. Adaptive rate control for streaming stored fine-grained scalable video. In: Proc. of the NOSSDEV 2002. Florida, 2002. 3–12.
- [7] Wakamiya N, Miyabayashi M, Murata M, Miyahara H. Dynamic quality adaptation mechanism for TCP-friendly MPEG-4 video transfer. In: Proc. of the 2nd Int'l Workshop on QoS in Multimedia IP Networks. Milano, 2003. 539–550.
- [8] Wang Y, Osermann J, Zhang YQ. Video Processing and Communication. Prentice Hall, 2002. 546–555.
- [9] Widmer J, Denda R, Mauve M. A survey on TCP-friendly congestion control. IEEE Network Magazine, 2001,15(3):28–37.
- [10] Kim T, Ammar MH. Optimal quality adaptation for MPEG-4 fine-grained scalable video. In: Proc. of the IEEE INFOCOM. San Francisco, 2003. 641–645.
- [11] Padhye J, Firoiu V, Towsley D, Kurose J. Modeling TCP throughput: A simple model and its empirical validation. In: Proc. of the SIGCOMM 1998. Vancouver, 1998. 303–314.
- [12] Brockwell PJ, Davis RA. Introduction to Time Series and Forecasting. 2nd ed., New York: Springer-Verlag, 2002. 318–322.
- [13] <http://www.isi.edu/nsnam/ns/>, 2004.
- [14] <http://www.cc.gatech.edu/computing/Telecomm/people/Phd/tkim/>, 2004.