

基于预取的流视频带宽适应性传输算法

谢建国

(广东外语外贸大学信息科学技术学院 广州 510006)
(jgxie@tom.com)

A Prefetching-Based Bandwidth Adaptation Transmission Scheme for the Video Streaming

Xie Jianguo

(School of Informatics, Guangdong University of Foreign Studies, Guangzhou 510006)

Abstract In recent years, the streaming video applications have led to a huge market. But dynamic behavior of the IP network's transmitting bandwidth makes it difficult to provide perceptually good quality of streaming video. The recent technologies of FGS and SVC video coding are proposed to deal with these problems by distributing the data in enhancement layers over a wide range of bit rates. However, data rate of the FGS video coding exhibits significant variability. This variability is meanwhile faced with the problem of the available network bandwidth variability. Therefore, the streaming is faced with the problem of trying to accommodate the mismatch caused by both the available bandwidth variability and the encoded video variability. By using the technologies of prefetching and buffering, this paper focuses on how many encoded video bits of the FGS video should be truncated, and how to truncate them when network available bandwidth isn't enough firstly. Then a scheme of the bandwidth adaptation video transmitting is developed through minimizing to truncate the encoded video bits, and the amount of truncated video bits are analyzed, which can achieve the minimum result theoretically. The scheme can provide a platform of how to compute truncating bit amount for other analogous algorithms. Finally, experimental results show the scheme is efficient and feasible.

Key words video streaming; FGS coding; bandwidth adaptation; buffering; bit rate smoothing

摘要 IP网络带宽的动态特性是流视频要面临的关键问题。可伸缩编码的视频具有带宽适应性,但由于其每一层编码通常是变比特率的,那么在视频传输中,为迎合动态传输带宽,如何裁减这种分层的变比特率视频,并极大化一定的目标性能,需要传输优化。针对上述问题,提出了一个预取或缓冲优化的流视频带宽适应性传输框架,将裁减策略放到一个窗口来考虑,解决了窗口内的最小裁减量问题,其带宽适应性粒度在理论上可以细化到“比特位”,并为其他裁减算法提供综合优化的计算平台。

关键词 流视频; FGS 编码; 带宽适应性; 缓冲; 比特率平滑

中图法分类号 TP393

近年来随着新型的 P2P 流媒体、IPTV 等网络流视频的出现和应用的快速增长,可以预测,网络流媒体将成为未来网络中的主流应用,但 IP 网络传输

带宽的波动性是网络流媒体要面临的关键问题。

流视频(video streaming)在时间上是一个连续的过程,具有时敏性,传输中网络带宽的短期波动可

以通过终端的有限缓冲来消除视频回放的抖动,而长期的带宽波动需要伸缩视频编码和带宽适应性传输来共同消除.

新的信源编码技术在动态网络适应性方面已经开启了卓有成效的研究工作,并形成了第1个标准 MPEG-4 FGS(fine granularity scalability)^[1-2]以及正在形成的 H. 264 扩展版 SVC(scalability video coding)^[3],它们一次编码后能适应网络带宽动态性和终端用户多样性. FGS/SVC 将视频源编码成一个基本层和多个增强层,基本层编码技术类似于现有的 MPEG-2/H. 264 中编码技术,是一个可独立解码的编码流. 增强层是可伸缩编码流(如位平面编码技术),用来增强基本层信息,其伸缩性体现在能根据可用带宽截断传输任意长度的编码数据,被裁减的编码数据只影响视频回放质量. 增强层数据被传输的越多终端视频的解码质量越好.

但 FGS 各编码层采用变比特率编码,这就给裁减算法如何迎合动态带宽、如何保持终端解码质量最大连续平滑性带来了复杂性. 本文研究 FGS 等编码视频在动态网络带宽环境下的最小裁减计算与传输问题,在如何偶合编码率的可变性和带宽的动态性、如何提高带宽利用率等方面,提出了一个精细带宽可适应的流视频传输框架,给出了最小裁减量的计算和分段传输计算方法,并为其他裁减算法提供了一个通用的带宽适应性传输平台.

1 相关研究

对于不可伸缩的编码视频,利用缓冲或预取技术实现带宽适应性传输,在过去的几年有过类似的

研究^[4-5],对于给定的存储视频,利用终端缓冲和起始播放延迟,计算各传输阶段的带宽需求,实现峰值数据需求的预取,从而最大降低峰值带宽需求. 这类算法不考虑视频截断传输,利用缓冲来有限地适应不同传输带宽以及消除带宽的短期波动.

针对 FGS 等可伸缩编码视频,在动态带宽的传输研究中,一类是以缓冲和带宽变化为参数,研究缓冲优化的带宽适应性传输^[6-9];另一类是以视频率失真特性和带宽变化为参数,研究率失真的带宽适应性传输^[10-11]. 在缓冲优化的带宽适应性传输研究中,适应性传输从增强层开始,根据接收端的缓冲状况逐层计算带宽需求后,选择适合当前网络带宽的编码层数进行截断传输. 在这类研究中,其带宽适应性粒度体现在选“编码层”上,首先通过一定的质量步长来适应网络波动,未体现 FGS 可任意截断传输的技术特点;二是算法的分层单独缓冲存在缓冲空间的优化分配和利用率问题;三是最小裁减量计算问题.

2 问题描述

在端到端或对等点等流视频应用中,发送端或点已知待发送的整个或一段 FGS 视频数据,接收端或点提供一个缓冲区(如解码器的 VBV 缓冲器),用来缓冲和接收视频数据.

在 MPEG 中视频的解码时间是以帧单元为基准的(如 MPEG-4 中某 VOP 的一次呈现),所以将网络传输过程进行帧时间离散化,若播放帧率为 25 fps,则帧时间为 1/25 s,不失一般性,以下令帧时间为 τ ,文中其余要用到的参数如表 1 所示:

Table 1 List of Symbols

表 1 基本变量说明

Symbol	Definition
$l_b[i]$ and $l_e[i]$	The i th frame size of base layer and enhancement layer
$\tilde{l}_e[i]$	The i th truncated frame size of enhancement layer
$l[i]$ and $\tilde{l}[i]$	$l[i]=l_b[i]+l_e[i]$ and $\tilde{l}[i]=l_b[i]+\tilde{l}_e[i]$
$L_b[i]$ and $L_e[i]$	The sum of all frame size from frame 1~ i
$\tilde{L}_e[i]$	The sum of all truncated frame size from frame 1~ i
$L[i]$ or $\tilde{L}[i]$	$L[i]=L_b[i]+L_e[i]$ or $\tilde{L}[i]=L_b[i]+\tilde{L}_e[i]$
$\{l[i]\}$	A variable list, $i \in \{1, \dots, N\}$, N is total frame number of a video
$\tilde{s}[i]$ or $\tilde{s}[i]/\tau$	Actual send video data or rate in the i th time τ
$\tilde{S}[i]$	Actual send total video data from time $\tau \sim i\tau$
$C(t)$	Available network bandwidth in some times
n, m	Start and end frame number of a video or truncated window
$c(t[k])$	Send rate of the k th linear segment ends at the time $t[k]$

接收端/点(如视频解码器)每到一个 τ 时间就从接收缓冲区移出一帧数据去视频解码,令取出一帧数据的时间是瞬时的,那么移出量或解码量 $\tilde{L}(t)$ 是一个阶梯函数:

$$\tilde{L}(t) = \begin{cases} 0, & 0 \leq t < \tau, \\ \tilde{l}[i], & t = \tau, \dots, i\tau, \dots, N\tau, \\ \tilde{l}[i-1], & (i-1)\tau < t < i\tau. \end{cases} \quad (1)$$

若接收端/点的缓冲能力为 b (可共用解码器的 VBV),各分层编码视频动态共享一个缓冲区,那么 t 时刻缓冲区中能接纳的最大数据总量 $\tilde{U}(t)$,由式(2)计算:

$$\tilde{U}(t) = \begin{cases} b \mid b \geq \max \{l[i]\}, & 0 \leq t \leq \tau, \\ \min \{ \tilde{L}(t-\tau) + b, \tilde{L}[N] \}, & \tau < t \leq N\tau. \end{cases} \quad (2)$$

设定每个 τ 时间内发送端/点的发送率为 $\tilde{s}[i]/\tau$,那么任意时刻的实际发送函数 $\tilde{S}(t)$ 由式(3)表示:

$$\tilde{S}(t) = \tilde{S}[\lfloor t/\tau \rfloor] + \tilde{s}[\lceil t/\tau \rceil] \times (t - \lfloor t/\tau \rfloor \cdot \tau) / \tau. \quad (3)$$

在流视频服务期间,当网络有效带宽 $C(t)$ 小于视频需求的实际带宽时^[12],需要裁减一些视频编码,那么一个可行的发送序列 $\{\tilde{s}[i]\}$ 应满足条件式(4),其传输率应小于有效可用带宽,同时要保证缓冲区不存在上下溢出问题.

$$\begin{cases} \tilde{s}[i]/\tau \leq C(t) \\ \tilde{L}(t) \leq \tilde{S}(t) \leq \tilde{U}(t) \end{cases}, i \in \{1, \dots, N\}, \quad (4)$$

$$0 \leq t \leq N\tau.$$

图 1 给出了式(1)~(4)裁减传输的直观表示,其中虚线表示增强层数据量曲线 $L_e(t)$,点划线表示基本层数据量曲线 $L_b(t)$,下位实线表示实际解码曲线 $\tilde{L}(t)$,由于网络带宽的限制,一些视频帧中的增强层编码信息将被裁减.

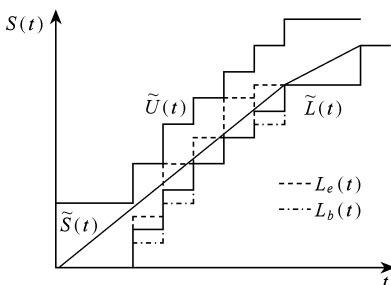


Fig. 1 A feasible transmission scheme with truncated bit stream.

图 1 截断比特流传输图

从图 1 可以看出,能够满足条件式(4)的可行发送序列 $\{\tilde{s}[i]\}$ 有许多(即图 1 中的 $\tilde{S}(t)$ 曲线). 寻找满足一些优化性能的 $\{\tilde{s}[i]\}$ 是这类裁减传输算法要研究的问题.

3 缓冲优化的带宽适应性传输算法

当网络动态带宽有时不能提供足够带宽但可以传输视频其中的基本部分(如系统信息和基层编码数据)时,则要考虑截断传输. 考虑到视频的变比特率编码性质、瞬时或短时的带宽需求可能超过有效传输带宽,需要进行裁减传输,但相对于一定长的时间段来看,其平均需求带宽未必超过此时有效传输带宽. 从这种观点看,可能不需要裁减或仅少量裁减. 缓冲或预取等一类传输技术可用来规划这种长时间段,从而实现数据少量或不裁减. 本文称这种长时间段为一个裁减窗口.

定义 1. 裁减窗口 $W_{n,m}$. 裁减窗口大小定义为包含连续视频帧的数目. 规定一个裁减窗口内的传输率是恒定的,这样其裁减量能做到最小. 按照图 1 的表示,一个直线传输段便是一个裁减窗口. 在需要裁减的情况下,窗口越大,则帧之间裁减分量的分配越灵活,以实现其他优化指标,如帧之间的质量平滑.

定义 2. 最短路径率平滑函数 $\{c(t[k])\} = SPS(n, m, \{\tilde{l}[i]\}, b)$, 其中 b 为给定接收点的缓冲区. SPS 算法^[5]是将一组传输率变换成另一组更平滑的传输率即图 1 中的直线段,当 $t[k-1] < i \leq t[k]$ 时, $s[i] = c(t[k])$ 是指该直线段内视频数据的发送率,一个直线传输段由一些连续相同 $s[i]$ 单元构成,按这种发送序列来传输原视频序列,可以极大地缩减原视频峰值带宽的需求.

基于一个裁减窗口,当网络带宽 $C(t) < c(t[k])$ 时,则要在该裁减窗口内根据某种规则裁减一些帧中增强层数据. 算法的关键是如何根据有效带宽调整裁减窗口以及在窗口内如何进行裁减,来满足极大视频质量的连续平滑. 不同于其他研究方法中的分层考虑缓冲策略, BAT (bandwidth adaptation transmitting) 算法中实现共享缓冲区,这样可以达到缓冲区的高利用率. 为简化 BAT 算法的书写,下面先给出两个裁减函数的描述.

定义 3. 裁减函数 $\{\tilde{l}_e[i]\} = TruncateBit(\{\tilde{l}_e[i]\}, \Delta_{n,m})$. $\Delta_{n,m}$ 为窗口内的最小裁减量,函数在最小裁减总量控制下,根据某种裁减法则给 $\{\tilde{l}_e[i]\}$ 分配裁减分量,更新 $\{\tilde{l}_e[i]\}$.

定义 4. 回填函数 $\{\tilde{l}_e[i]\} = \text{RefillBit}(\{l_e[i]\}, \{\tilde{l}_e[i]\}, \Delta_{n,m})$, $\Delta_{n,m}$ 为该裁减窗口内要回填的最大总量, 函数由回填总量根据某种回填法则, 从 $\{l_e[i]\}$ 中分配回填分量到相应的 $\{\tilde{l}_e[i]\}$ 上, 更新 $\{\tilde{l}_e[i]\}$.

动态网络带宽 $C(t)$ 的计算可实时分段进行, 并离散化, 记 $C(t)$ 为 $C(i)$. 算法要求 $C(t)$ 原则上要大于基本层传输要求. BAT 算法描述如下:

输入: 基本层和增强层帧序列 $\{l_b[i]\}$ 和 $\{l_e[i]\}$ 以及共享缓冲区 b .

输出: 增强层截断传输量 $\{\tilde{l}_e[i]\}$ 和发送端实时发送序列 $\{\tilde{s}[i]\}$.

- 1) 对于给定视频, 初始化序列 $\{s[i], \tilde{s}[i] = 0\}$, $\{\tilde{l}_e[i] = l_e[i]\}, n = i = 0, m = N$;
- 2) 初始化空栈 $\text{Heap}(c, t)$; /* 栈元素含需求带宽和终止时间两个值域 */
- 3) 执行 $\{c(t[k])\} = \text{SPS}(n, m, \{\tilde{l}_e[i]\}, b)$;
- 4) 按照 $k = K, \dots, 1$, 窗口参数 $c(t[k])$ 和 $t[k]$ 进堆栈 $\text{Heap}(c, t)$;
- 5) 若空栈, 结束算法, 否则, 取栈顶窗口参数, 使 $(c, m) = \text{getTop}(\text{Heap}(c, t))$;
- 6) 若 $n = m$ 窗口时间终止, 回到第 5) 步, 否则, 探测网络 $C(i)$ 分 3 步做:
 - 6.1) 若 $c = C(i)$, 则输出发送单元 $\tilde{s}[i] = c, n++$, 回到第 6) 步;
 - 6.2) 若 $c < C(i)$, 分两步做:
 - 6.2.1) 若窗口内区间 $(n, m]$ 存在 $\{\tilde{l}_e[i]\} = \{l_e[i]\}$, 则输出发送单元 $\tilde{s}[i] = c, n++$, 回到第 6) 步;
 - 6.2.2) 否则, 计算窗口内区间 $(n, m]$ 要回补的总量 $\Delta_{n,m} = (C(i) - c)(m - n)\tau$, 执行 $\{\tilde{l}_e[i]\} = \text{RefilledBit}(\{l_e[i]\}, \{\tilde{l}_e[i]\}, \Delta_{n,m})$, 设定 $\tilde{S}(n)$ 和 $\tilde{S}(m)$ 为新传输段端点, 回到第 3) 步;
 - 6.3) 若 $c > C(i)$, 则计算窗口内区间 $(n, m]$ 要裁减的总量 $\Delta_{n,m} = (c - C(i))(m - n)\tau$, 执行 $\{\tilde{l}_e[i]\} = \text{TruncatedBit}(\{l_e[i]\}, \Delta_{n,m})$, 设定 $\tilde{S}(n)$ 和 $\tilde{S}(m)$ 为新传输段端点, 回到第 3) 步;

4 算法分析

SPS 算法有极小极大和极大极小的性质, 即极小的最大传输率和极大的最小传输率. 这个性质用到 BAT 算法上, 提供了最小裁减量的理论保证.

最小裁减量分析:

让序列 $\{s[i]\}$ 表示 SPS 算法产生的传输, $\{s'[i]\}$ 表示其他可行传输规划, 令 $s_{[j]}$ 和 $s'_{[j]}$ 分别代表 $\{s[i]\}$ 和 $\{s'[i]\}$ 序列中的第 j 个最大单元 (相同大小的单元排序不分), 则序列 $\{s[i]\}$ 的平滑性质中存在关系: $\sum_{j=1}^K s_{[j]} \leq \sum_{j=1}^K s'_{[j]}$, 其中 $K \in [1, N]$.

推论 1. 对于一个给定的编码视频, 其中任何一个给定的传输时间段 $[n, m]$ 内也存在关系: $\sum_{j=1}^K s_{[j]} \leq \sum_{j=1}^K s'_{[j]}$, 其中 $k \in [1, m - n + 1]$, $s_{[j]}$ 和 $s'_{[j]}$ 分别代表时间段 $[n, m]$ 内 $\{s[i]\}$ 和 $\{s'[i]\}$ 序列中的第 j 个最大单元.

推论的结果容易理解, 一个算法作用于整个视频和作用其中一段视频其结果的性质是相同的. 按照推论, 若某个时间段内网络有效带宽为 $C(t)$, 且 $C(t) \leq \max \{s[i]\}$, 则总裁减量存在关系 (只求和正数项): $\sum_{j=1}^K (s_{[j]} - C(t)) \leq \sum_{j=1}^K (s'_{[j]} - C(t))$. 说明 SPS 框架下总裁减量是最小的, 而 BAT 算法中回填策略, 确保最小裁减量的实施.

BAT 算法的结果为两个序列: 增强层传输序列 $\{\tilde{l}_e[i]\}$ 与发送序列 $\{\tilde{s}[i]\}$, $\{\tilde{l}_e[i]\}$ 和最大视频质量连续平滑直接相关, 而 $\{\tilde{s}[i]\}$ 则直接影响裁减量, 一般情况下 $\{\tilde{s}[i]\} \neq \{\tilde{l}_e[i]\}$. BAT 算法计算了一个有效直线传输段内应裁减的最小裁减总量 (带宽有效利用的保证) 和提供一个时长尽可能长的裁减窗口, 以便充分协调窗口内各帧的裁减分量. 至于如何分配裁减分量, 可以借用其他文献的裁减方法, 本算法提供了一个通用的裁减平台.

裁减函数的工作方式是透明的, 在总裁减量的框架下可以根据某种原则分配各帧的裁减分量, 若能做到窗口内各帧之间视频质量连续平滑, 则裁减函数是质量平滑优化的. 图 2 演示了两种裁减分量分配方案, 其中图 2(b) 一次调用裁减函数即满足带宽传输要求, 图 2(c) 一次裁减分配后需要进行二次或多次处理, 前半传输段需要继续裁减, 后半传输段则需要回填.

回填函数的工作方式为裁减函数的逆过程, 发生在动态带宽的实时增加或图 2(c) 等情况下, 根据需要将一些原被截断的编码数据再追补到被裁减的部分序列 $\{\tilde{l}_e[i]\}$ 上, 这是最小裁减量的保证措施, 以实现带宽的最大充分利用.

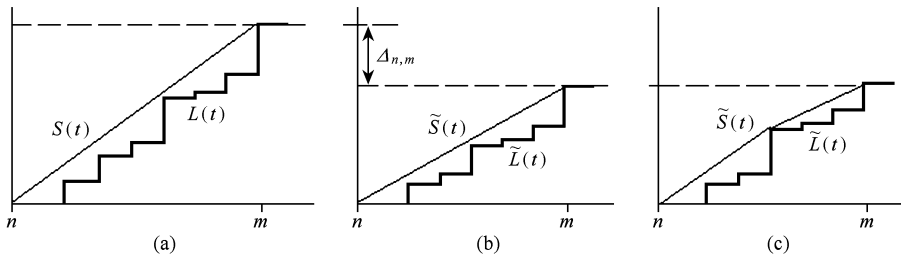


Fig. 2 Show two truncation methods.

图 2 两种裁减示意图

实时流调度算法时间复杂性非常重要,BAT 算法中的第 1 次调用 SPS 算法可以在离线完成,并作为视频的元数据一起存放,以后调用只计算当前直线传输段,实现分段计算传输,可以在近线性时间内完成,不影响后面已计算的调度序列,而文献[6]等一类算法需分层计算整个视频文件.

5 实验结果

图 3 给出了共享 80 KB 缓冲空间时 FGS 视频 Thefirn12000 帧起的一段原始帧数据和 SPS 预取平滑后数据发送图,下面给出本文算法的一个应用

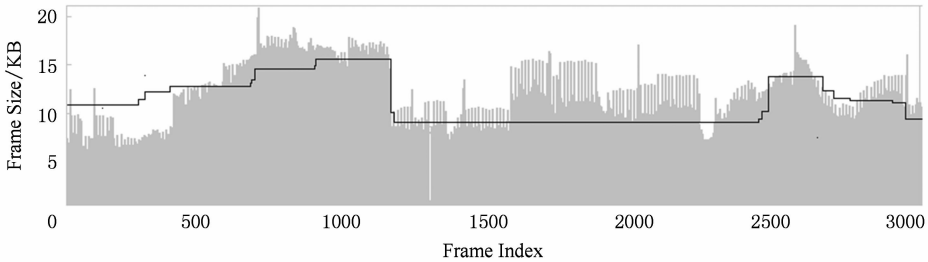


Fig. 3 Prefetching scheme of all layers video transmitted at total 80 KB buffering.

图 3 共享缓冲为 80 KB 时所有层数据的流式预取图

BAT 算法理论上不管是分层缓冲还是共享缓冲,在各自的环境下均有最小的总裁减量,所以实验中只比较一下两种环境下的裁减差别以及在给定裁减函数后,裁减窗口内的裁减次数对视频质量连续平滑的影响.

图 4 中网络最低可用带宽分别为 7.5 KB,8 KB,9 KB,9.5 KB 时,分层缓冲空间按 3 : 5 分配,水平轴为总缓冲,纵轴为裁减比较量,其计算方法是:(分层总裁减量-共享总裁减量)/共享总裁减量.结果显示,绝对的裁减比较量受到网络动态性、编码率的变比特性影响,但总的相对趋势说明共享缓冲要好于分层缓冲,特别是当网络带宽较低或缓冲空间较

小时非常明显.

视频 Thefirn 增强层编码数据由 8 个位平面编码层构成(MSB0~MSB7),编码类型有 I,P 和 B 三种帧,裁减顺序是在一个裁减窗口内先 BPI,同一编码类型先裁减播放时间靠前的,而帧内位平面是先裁减 B 中的 MSB7,MSB6...等位平面层,然后是 P 的,最后是 I 的,依此类推,裁减的最小单位是位平面内的一个比特.网络动态带宽由一个随机数产生,规定每秒一次,并记住随机产生的带宽序列,以便实验比较.设定带宽参考范围为 7 KB(SPS 算法在 30 KB 缓冲时基本层被传输的最小需求)~15 KB(所有数据被传输的最小需求).

小时非常明显.

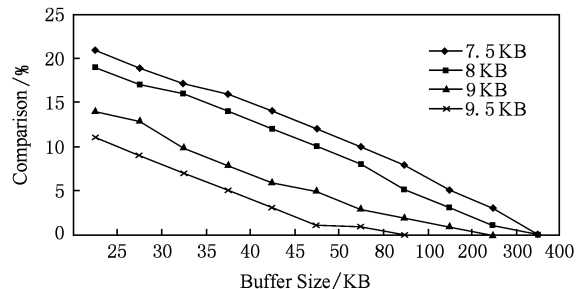


Fig. 4 Comparison of the minimum of truncated video data between sharing buffer and single layer buffer.

图 4 共享缓冲与分层缓冲最小裁减量之比

最小裁减量的计算是理论上的保证,但目标性能取决于具体裁减方法,按照上述方法,经 BAT 算法确定裁减窗口(可协调的范围大)后,一次裁减成功率越高目标性能连续性应越好.实验比较了当数据的属性相同时,解码时间在前的先裁减和解码时间在后的先裁减两种情况,发现前者一次裁减率明显好于后者,且一次成功率有 64%,二次成功率为 30%.

6 总 结

本文主要研究可伸缩编码在动态网络中的传输问题.当可伸缩编码的变比特率特性和网络传输带宽的动态性失配时,要保持流视频服务则需要裁减原编码视频,并传输其中的部分编码数据.不像其他算法主要关注如何裁减,本文算法在理论上主要解决最小裁减量的问题,但实际裁减量能否取得或接近理论上的效果还要取决于具体裁减算法.本研究的进一步研究是要解决裁减量与裁减方法的一致性及在实际网络环境下的应用问题.

参 考 文 献

- [1] Li W. Overview of fine granularity scalability in MPEG-4 video standard [J]. *IEEE Trans on Circuits System, Video Technology*, 2001, 3(11): 301-317
- [2] Wu F, Li S, Zhang Y Q. A framework for efficient progressive fine granularity scalable video coding [J]. *IEEE Trans on Circuits System, Video Technology*, 2001, 11(3): 332-344
- [3] Ohm J R. Advances in scalable video coding [C] //Proc of IEEE. Los Alamitos, CA: IEEE Computer Society, 2005: 42-56
- [4] Salehi J D, Zhang Z L, Kurose J, *et al.* Supporting stored video: Reducing rate variability and end-to-end resource

requirements through optimal smoothing [J]. *IEEE/ACM Trans on Network*, 1998, 6(4): 379-410

- [5] Xie Jianguo, Jiang lingmin, Chen Songqiao. The shortest path rate smoothing transmission algorithm of the VBR streaming videos [J]. *Chinese Journal of Computers*, 2004, 27(3): 357-364 (in Chinese)
(谢建国, 姜灵敏, 陈松乔. VBR 流式视频的最短路径率平滑传输算法[J]. *计算机学报*, 2004, 27(3): 357-364)
- [6] Kim T, Ammar M H. Optimal quality adaptation for MPEG-4 fine-grained scalable video [C] //Proc of INFOCOM 2003. Piscataway, NJ: IEEE, 2003
- [7] Cuetos P, Reisslein M R, Ross K M. Streaming FGS-encoded video: Insights from a large library of rate-distortion traces, RR-03-078 [R]. Sophia Antipolis, France: Institut Eurecom, 2003
- [8] P Cuetos, K M Ross. Adaptive rate control for streaming stored fine-grained scalable video [C] //Proc of NOSSDAV. New York: ACM, 2002: 3-12
- [9] Guo M, Ammar M. Scalable live video streaming to cooperative clients using time shifting and video patching [C] //Proc of INFOCOM 2004. New York: IEEE, 2004: 1501-1511
- [10] Chou P A, Miao Z R. Rate-distortion optimized streaming of packetized media [J]. *IEEE Trans on Multimedia*, 2006, 4(8): 390-404
- [11] Chakareski J, Apostolopoulos J, Girod B. Low-complexity rate-distortion optimized video streaming [C] //IEEE ICIP. New York: IEEE, 2004: 2055-2058
- [12] Tunalia T, Anar K. Adaptive available bandwidth estimation for Internet video streaming [J]. *Signal Processing: Image Communication*, 2006, 21: 217-234



Xie Jianguo, born in 1964. Ph. D. and professor. His main research interests include computer network, video streaming and audio-video system encoding.

谢建国, 1964年生, 博士, 教授, 主要研究方向为计算机网络、流媒体、音视频系统编码与传输。

Research Background

In recent years, video streaming application has led to a great growth. But dynamic behavior of the Internet's transmitting bandwidth makes it difficult to provide perceptually good quality of streaming video. FGS and SVC coding are proposed to deal with these problems by distributing the data in enhancement layers over a wide range of bit rates. However, encoded video exhibits significant data rate variability to provide a consistent quality video. This variability is meanwhile faced with the problem of the available bandwidth variability in network transmission. This paper focuses on the mismatch caused by both the available bandwidth variability and the FGS video variability. The author considers how many bits stream of the FGS video should be truncated when network bandwidth isn't enough, and develops a bandwidth adaptation scheme through minimizing to truncate encoded bits stream. The work is partially supported by the National Natural Science Foundation of China(60333020).