

基于 OpenFlow 的 SVC 流媒体时延自适应分级传输方法

黄韬¹, 张丽¹, 张云勇^{1,2}, 刘韵洁¹

(1. 北京邮电大学 网络与交换国家重点实验室, 北京 100876; 2. 中国联通研究院, 北京 100048)

摘要: 针对当前互联网流媒体传输的时延敏感性问题, 提出一种基于 OpenFlow 的 SVC (scalable video coding, 可分级视频编码) 流媒体时延自适应分级传输方法, 该方法有效结合 SVC 流媒体可分级和 OpenFlow 灵活可编程的特性, 在网络带宽受限和链路拥塞的复杂网络环境下, 通过构建基础层和增强层 2 个独立路由, 实现了动态网络下 SVC 流媒体分级自适应高效传输。仿真结果表明, 该方法在提升 SVC 流媒体传输效率和质量, 改善用户体验方面有重要作用。

关键词: OpenFlow; 可分级视频编码; 流媒体传输; 时延自适应

中图分类号: TP393

文献标识码: A

文章编号: 1000-436X(2013)11-0121-08

Delay adaptive transmission method for scalable video streaming based on OpenFlow

HUANG Tao¹, ZHANG Li¹, ZHANG Yun-yong^{1,2}, LIU Yun-jie¹

(1. State Key Laboratory of Networking and Switching Technology, Beijing University of Posts and Telecommunications, Beijing 100876, China;

2. China Unicom Research Institute, Beijing 100048, China)

Abstract: For the current internet streaming media transmission delay sensitivity issues, a delay adaptive method for SVC (scalable video coding) streaming media transmission based on OpenFlow was proposed. This method combines SVC streaming scalable and flexible programmable features of OpenFlow effectively, and implements SVC streaming hierarchical adaptive transmission in a dynamic network, by building the base layer and the enhancement layer separate routes under complex network environment with limited network bandwidth and link congestion. Simulation results show that this method has an important role in improving efficiency and quality of the SVC streaming, so it can improve the user experience effectively.

Key words: OpenFlow; scalable video coding; media streaming; delay adaptive

1 引言

近年来, 随着全球互联网视频、移动流媒体和 IPTV 等数字媒体业务蓬勃发展和应用规模的持续增长, 不断增加的海量视频内容传输需求与不断提升的用户视听需求, 使得基于互联网的流媒体传输面临日益严峻的挑战^[1]。由于流媒体传输具有非常敏感的时延特性, 当网络带宽受限时, 现有互联网 TCP/IP 模式的数据传输方式将会造成链路拥塞, 最

终导致视频回放时间增加以及视频流播放不连续等问题^[2,3]。

基于可分级视频编码(SVC, scalable video coding)的流媒体传输技术作为一类解决思路应运而生^[4]。SVC 技术可通过视频编码的方式将流媒体数据分为一个基础层(base layer)和若干个增强层(enhancement layer), 其中, 基础层数据提供最基本的观看质量, 而增强层能够在时间、空间和图像质量 3 个维度上增强视频质量。增强层依赖基础层数据解码, 因此基础层

收稿日期: 2013-07-26; 修回日期: 2013-09-25

基金项目: 国家重点基础研究发展计划(“973”计划)基金资助项目(2012CB315801); 国家自然科学基金资助项目(61300184, 61302089)

Foundation Items: The National Basic Research Program of China (973 Program)(2012CB315801); The National Natural Science Foundation of China(61300184, 61302089)

数据更重要, 对时延敏感性也更高^[5]。

SVC 流媒体传输系统最大的优势在于能够在传输过程中对不同层级的视频数据采用不同的传输策略和处理方法。近年来, 学术界在 SVC 流媒体传输及相关领域也做了诸多研究。其中, 为了提高互联网上 SVC 流媒体视频的传输质量, 文献[6]通过基于 RTP 封装的差错隐藏方法提高视频解码质量, 而文献[7]将网络编码应用于 SVC 视频传输, 通过合理的网络编码方法提高 SVC 视频传输的可靠性和顽健性, 两篇文献均主要从视频编码和封装方法方面进行改进, 而没有设计高效的传输方法, 从而间接抵抗 SVC 流媒体时延敏感性, 降低网络传输条件对 SVC 流媒体质量的影响。文献[8]为流媒体视频传输提出子流的概念, 在时延约束条件下, 为不同子流分配最佳传输资源并选择传输策略, 采用了类似 SVC 的可分级特性, 实现 HCCA(HCF controlled channel access)无线网络中流媒体高效传输。文献[9]利用 SVC 的可分级特性, 优先为基础层数据选择高信噪比的子载波传输, 实现了分级传输, 然而文中仅仅考虑传输差错控制而没有考虑流媒体传输时延特性。文献[10]采用 P2P 的传输方式达到降低 SVC 流媒体传输时延的目的, 但是此方案针对 SVC 分级传输需维护多个多播树, 造成系统开销增加, 此外由于 SVC 的分层数据量相差较大, 此方案解决多个多播树数据传输同步性问题也会增加方案的复杂性。文献[2,3,11]在 OpenFlow 网络环境中, 为 SVC 流媒体传输设计了 QoS 路由方案, 综合考虑网络分组丢失和时延条件, 为基础层数据传输提供优先 QoS 保证, 并利用 LARAC (lagrange relaxation based aggregated cost) 算法解决了限制最短路径问题, 实现了分级路由和部分时延自适应功能, 然而 QoS 方案可能会对整个网络吞吐量产生影响。

然而, 上述研究均未针对 SVC 流媒体的抗时延敏感性提出高效传输方案, 不能根据业务和媒体特点提供灵活可控的时延自适应传输路径, 造成接收端 SVC 流媒体播放质量下降。以 OpenFlow^[12]为代表的软件定义网络 (SDN) 技术, 具有实时感知网络状态以及强大的灵活可编程能力, 因此可根据应用需求完成数据流的智能路由选择和动态调度。本文针对上述问题, 提出一种基于 OpenFlow 的 SVC 流媒体时延自适应分级传输方法, 通过周期性探测网络时延状态变化, 完成网络实时拥塞情况感

知, 进而实现 SVC 流媒体传输的时延自适应分级路由计算与选择, 有效保证了 SVC 流媒体高质量的实时连续传输。

2 SVC 流媒体传输质量评估模型

为了设计高效的时延自适应分级传输方法, 首先对 SVC 流媒体分层码流进行重要性区分, 并据此定义本文在仿真研究中所需的 SVC 流媒体传输质量评估模型。

SVC 流媒体数据一帧表示一幅图像, 根据 SVC 的结构属性, 定义流媒体视频帧效用值 $eff(i)$, 用于表示接收端数据帧的完整度即接收质量, 计算公式如式(1)所示。其中, 每帧均由基础层 $base(i)$ 和增强层 $enhance(i)$ 组成。 $base(i)$ 和 $enhance(i)$ 取二进制数值 0 或 1, 取 0 表示接收超时或者数据不完整, 取 1 表示按时接收并且完整。考虑到基础层数据与增强层数据对视频图像恢复的重要性不同, 用 a 表示基础层数据的重要性指数, β 表示增强层数据的重要性指数。这里假设发送端传输的原完整帧效用值为 100。

$$eff(i) = \begin{cases} base(i) \times a + enhance(i) \times b, & base(i) > 0 \\ 0, & base(i) = 0 \end{cases} \quad (1)$$

subject to $a \geq 0, b \geq 0, a + b = 100$

对于整个传输文件, 定义 eff_aver 表示所有数据帧的平均效用值, 用于衡量 SVC 流媒体文件的平均接收质量。

$$eff_aver = \frac{1}{n} \left[\sum_{i=1}^n eff(i) \right] \quad (2)$$

为便于网络侧根据已收到的视频数据进行动态自适应调整, 同时定义用于前 i 帧数据总体接收质量的衡量指标 $eff_sum(i)$ 。

$$eff_sum(i) = \left[\sum_{j=1}^i eff(j) \right] = \sum_{j=1}^i base(j) \times a + enhance(j) \times b$$

subject to $a \geq 0, b \geq 0, a + b = 100$ (3)

定义 eff_std 为传输文件的帧效用值均方差, 表示接收数据帧的质量波动。

$$eff_std = \sqrt{\frac{\sum_{i=0}^n (eff(i) - eff_aver)^2}{n}} \quad (4)$$

由于 SVC 流媒体文件所有增强层数据解码必须参照其对应的基础层数据，因此必须保证基础层数据接收完整的情况下，此帧图像才能正确播放，效用值才会是非零值。对于流媒体数据接收播放而言，接收端通常设置播放缓存 T ，定量 (M) 数据帧只有在指定时间 (T) 内完整到达才能保证连续播放。综上，将 SVC 流媒体传输质量评估模型设计如下。

表 1 给出了基础层数据和增强层数据接收时延 4 种不同组合情况。其中， D_base_i 或 $D_enhance_i > iT$ 表明第 i 组对应类型帧数据超时，此时定义 $base(i)$ 或 $enhance(i)$ 为 0；同理， D_base_i 或 $D_enhance_i < iT$ ，表明第 i 组对应类型帧数据按时完整接收，此时定义 $base(i)$ 或 $enhance(i)$ 为 1。

表 1 SVC 流媒体的帧效用计算方法

序号 (i)	基础层数据接收时延	增强层数据接收时延	效用值 $eff(i)$
1~ M	$D_base_1 > T$	$D_enhance_1 > T$	0
$M+1 \sim 2M$	$D_base_2 < 2T$	$D_enhance_2 > 2T$	$eff(i) = a$
$2M+1 \sim 3M$	$D_base_3 > 3T$	$D_enhance_3 < 3T$	0
$3M+1 \sim 4M$	$D_base_4 < 4T$	$D_enhance_4 < 4T$	$eff(i) = 100$
...

注： D_base_i 表和 $D_enhance_i$ 表示对应数据帧接收时间， M 表示 T 内要求播放帧数。

在 SVC 流媒体数据接收端处统计效用值，接收端会设置接收零点时间 D_0 ，接收到的所有数据帧根据零点时间标记各自的接收时间 D_base_i 或 $D_enhance_i$ ，然后根据统计数据按照式(2)~式(4)进行平均效用、帧效用和以及效用均方差的计算，据此说明传输质量的好坏。其中，文件平均效用值越大，表明接收端 SVC 流媒体质量高，即传输效果越好；均方差值越小，表明接收端 SVC 流媒体质量波动小，说明传输方案自适应能力强。

3 基于 OpenFlow 的 SVC 流媒体传输系统架构

OpenFlow 技术核心思想是将传统网络的数据平面和控制平面进行分离，一个最基本的网络包含一个控制器和多个支持 OpenFlow 协议的转发交换机。控制器通过下发流表通知转发交换机如何转发数据。通过 OpenFlow，可以在开放可编程的控制器中为不同的数据构造不同的流表，控制数据通过不同的路径转发^[12]。本文针对传统 SVC 流媒体传

输方案的不足，提出一种基于 OpenFlow 的 SVC 流媒体自适应分级传输方案。

首先考虑单控制域的场景，假设网络中有 N 个 OpenFlow 交换机，它们均由控制器直接控制。任意 2 个交换节点 i 和 j 之间，若存在直连链路则表示为 L_{ij} ，其中， $i=1,2,\dots,N, j=1,2,\dots,N$ ，并且 $i \neq j$ ，若不存在直连链路但存在 N_{ij} 条不相关路由，则标记为 r_{ij}^x ，其中， $x=1,2,\dots,N_{ij}$ 。令 C_{ij} 表示任意一条链路容量，即链路带宽， PL_{ij} 表示任意一条链路分组丢失率。令 D_{ij}^x 表示任意一条路由 r_{ij}^x 的时间长度， D_{ij}^x 采用 r_{ij}^x 所经历的所有直连链路延时时间为标度测量得到，即路径时延，计算式如式(5)所示， $D_{m(m+1)}$ 表示在路由 r_{ij}^x 上 2 个相邻转发节点间直连链路的时延标度值。

$$D_{ij}^x = \sum_{m=i}^j D_{m(m+1)} \tag{5}$$

为保证研究条件和逻辑分析的严谨性，在不背离整体最优效果的前提下，本文对研究之外的影响因素进行了假设，将研究问题集中到基于链路时延的自适应路由选择。因此，将问题做如下假设和简化：1)网络中链路带宽一致，即 C_{ij} 取得定值 C ，并且需要两条路径同时传输才能满足 SVC 流媒体传输的带宽要求；2) 链路无分组丢失，即 $PL_{ij}=0$ ；3) 链路时延状态动态随机变化，即 $D_{m(m+1)}$ 是变量，那么端到端的路径时延 D_{ij}^x 也随机变化。

由于 SVC 流媒体中各层数据对恢复视频质量的影响程度不同，可以为各层数据采用不同的路由协议、传输方式甚至选择不同的传输路径。为对抗 SVC 流媒体时延敏感性，对基础层数据优先选择时延最优路由，这样既能保证接收端恢复基本质量的视频，增强 SVC 流媒体传输的稳健性和接收端流媒体正常播放的可靠性，又能通过接收的增强层信息来尽可能改善视频质量。

基于上述思路，本文提出基于 OpenFlow 的 SVC 流媒体时延自适应分级传输方法，其系统架构如图 1 所示。主要实现步骤包括：1) 链路时延信息收集，由动态时延探测模块实现，包括链路发现和拓扑管理，它是自适应分级路由的计算依据。通过信息收集，控制器实时感知网络中的链路时延状态，并获得网络分布信息；2) 自适应分级路由，由自适应分级路由模块实现，包括路由计算和路由管理。根据步骤 1 中收集的信息，自适应分级路由设

计有效的资源调度算法，在带宽约束条件下，最小化端到端的路径时延，抵抗 SVC 流媒体传输时延敏感性，实现最佳质量传输，给出优化问题为

$$\min(D_{ij}^{SVC}) = \min(D_{ij}^{base} + D_{ij}^{enhancement}) \quad (6)$$

从数学上，可转化为离散问题，即

$$r = (r_{base}, r_{enhancement}) = \{\min_{r_j}(D_{ij}^x), \text{second_min}_{r_j}(D_{ij}^x)\} \quad (7)$$

其中， i, j 分别表示 SVC 流媒体的发送节点与接收节点， r_{base} 表示基础层数据传输路径， $r_{enhancement}$ 表示增强层数据传输路径。 D_{ij}^{SVC} 表示 SVC 流媒体整体传输时延， D_{ij}^{base} 表示基础层数据传输时延， $D_{ij}^{enhancement}$ 表示增强层数据传输时延。

由于带宽受限，单条路径传输不能保证 SVC 流媒体正常连续播放，因此需要首先为最重要的数据（SVC 基础层数据）选择时延最优路径，若该层数据分配完毕，且时延最优路径带宽有余，则补充增强层数据，否则将增强层数据直接分配到时延次优路径。最后，控制器下发流表，控制相应数据自适应选择路径传输。在所有功能模块之下，安全层主要使用 SSL(secure sockets layer, 安全套接层)接口，实现控制器与转发交换机之间的安全通信^[2,3]。

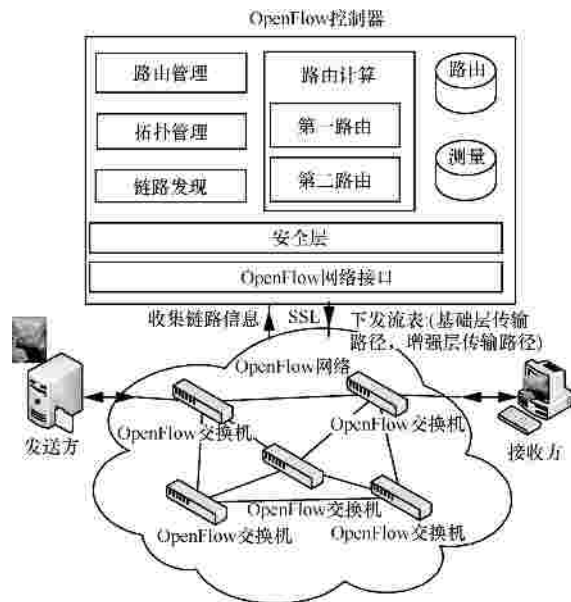


图 1 基于 OpenFlow 的 SVC 流媒体分级传输系统架构

4 SVC 时延自适应分级传输算法

SVC 流媒体具有较强的时延约束，根据表 1 分析，接收端数据分组丢失根本原因是传输时延超过

预设值，因此，SVC 流媒体传输时，如果可以保证尽量减小数据的时延，则可以降低用户分组丢失率，并最终达到抵抗 SVC 流媒体时延敏感性的目的，进而提高流媒体接收质量。本文的时延自适应分级传输方法，首先通过动态时延探测收集链路时延信息，然后由自适应分级路由模块计算传输端之间路径时延，并据此完成最优时延路径与次优时延路径的分级选择和自适应切换功能。

4.1 动态时延探测

动态时延探测是指使用自定义探测分组(testpacket)，由控制器定时发送到交换机，并控制数据分组转发流向，最终获得交换机之间的直连链路时延信息。控制器周期性定时发送探测数据分组并更新存储信息，因此能够动态感知网络链路时延变化。而由于每帧数据发送传输时都可能出现网络状态的变化，因此本文的探测周期(MT)选择依据的是 SVC 发送帧率。动态时延探测模块在控制器上构造 2 个探测数据分组，第一组用于探测控制器与交换机之间的链路时延，如图 2 所示的 testpacket3；第二组在第一组测量结果的基础上，用于探测交换机之间的直连链路时延，如图 2 所示的 testpacket34。

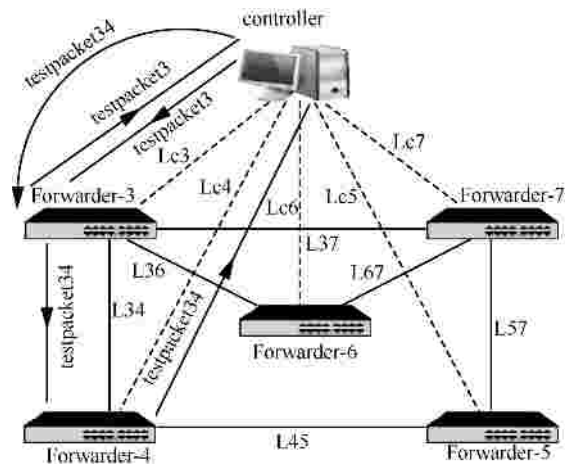


图 2 基于 OpenFlow 的动态时延探测机制

首先，控制器同时发出第一组测量数据分组，记录发出时间 ST_i ，其中， $i=3,4,\dots,7$ 。以图 2 为例，testpacket3 到达交换机 Forwarder-3 之后，立即被转发返回控制器，控制器记录接收时间 RT_3 。控制器根据记录按式(8)计算控制器与交换机 Forwarder-3 之间链路时延 D_{c3} 。同理计算得到所有 D_{ci} ，其中， $i=3,4,\dots,7$ ，并存储到全局测量数据库 $D_{measurement}$ 。

$$D_{c3} = \frac{RT_3 - ST_3}{2} \tag{8}$$

然后，控制器同时发出第二组数据分组，并对应记录发出时间，以图 2 为例，testpacket34 流经 Forwarder-3 被转发到 Forwarder-4，再转发返回控制器，记录接收时间 RT_{34} ，然后控制器通过计算式(9)得到交换机 Forwarder-3 和 Forwarder-4 的直连链路 L_{34} 的时延 D_{34} 。同理，可计算得到所有交换机之间直连链路 L_{ij} 时延，并存储至测量数据库 $D_{\text{measurement}}$

$$D_{34} = RT_{34} - ST_{34} - D_{c3} - D_{c4} \tag{9}$$

最终，动态时延探测模块将通过这种测量方式，表征 OpenFlow 网络中所有链路时延状态信息 D_{ij} ，即通过非常简便的方法获得了网络实时状态信息，并将此作为路由计算的依据，提供自适应分级路由模块调用。根据探测原理可知，第一组探测需计算 N 次，第二组探测为避免重复，探测节点按照标号顺序展开，即要求 L_{ij} 中 $i < j$ ，因此第二组探测计算复杂度 $O(N^2)$ ，最终时延探测机制的计算复杂度为 $O(N^3)$ 。

4.2 自适应分级路由算法

自适应分级路由算法是指控制器根据 SVC 流媒体数据转发请求消息，识别待转发数据类型，根据数据类型按照式(7)进行自适应分级路由选择，为基础层数据和增强层数据分别选择第一路由和第二路由的路由计算模块，计算所得路由信息并存储到路由数据库。路由管理模块实时取出路由信息下发至 OpenFlow 网络中的转发交换机。每当时延探测结果更新，则第一路由和第二路由需重新计算路径，否则直接使用路由数据库中的路径信息。图 3 详细描述了自适应分级路由算法的流程。

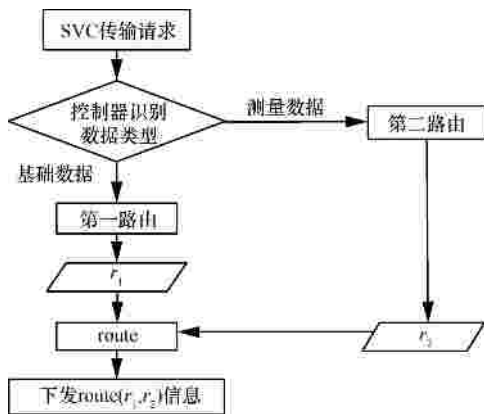


图 3 自适应分级路由算法流程

其中，第一路由模块根据动态时延探测得到链路

时延信息 D_{ij} ，计算单源端到端最优时延路径^[13]。第一路由模块采用 Dijkstra 算法思想，返回源点 s 至目的点 d 的时延最优路径信息 $r_1 = \min_{ij} (D_{ij}^x)$ 。第二路由模块计算单源端到端次优时延路径，算法思想与第一路由模块保持一致。第二路由算法首先利用第一路由取得源节点 s 到目的节点 d 的最优路径 r_{sd}^{\min} 枚举 r_{sd}^{\min} 上的链路，加入集合 $L = \{L_{ij} | L_{ij} \in r_{sd}^{\min}\}$ ，然后枚举删除集合 L 中的链路，重复调用第一路由模块计算删除链路之后的 s 到 d 最优路径，得到路径集合 $r = \{r_{sd}^{\min(E-L_{ij})} | L_{ij} \in r_{sd}^{\min}\}$ 。其中， E 表示实验网络链路集合， $r_{sd}^{\min(E-L_{ij})}$ 表示删除链路 L_{ij} 之后计算得到的最优路径；对应记录每次求出的最优路径时延消费，得到路径消费集合 $D = \{D_{sd}^{\min(E-L_{ij})} | L_{ij} \in r_{sd}^{\min}\}$ ，其中， $D_{sd}^{\min(E-L_{ij})}$ 表示删除链路 L_{ij} 之后计算得到的最优路径时延。最后，比较路径消费集合得出最小值， $\min_delay = \text{Min} \{D_{sd}^{\min(E-L_{ij})} | L_{ij} \in r_{sd}^{\min}\}$ ，其对应的路径 $r_{sd}^{\min(E-L_{ij})}$ 即为整体时延次优路径 $r_2 = \text{second_min}_{ij} (D_{ij}^x)$ 。

由于第一路由采取 Dijkstra 算法思想，在运行时要执行两重嵌套的循环语句，因此其总的计算复杂度是 $O(N^2)$ 。而第二路由模块在此基础上，多次局部调用了第一路由算法，因此计算复杂度为 $O(UN^2)$ (U 表示第一路由模块计算所得最短路径的链路数目)。

5 实验与分析

本文首先构建一个简单的 OpenFlow 网络实验环境，如图 4 所示，转发交换机选取 OpenVSwitch，采用 OpenFlow 1.0 版本标准^[14]，文件发送端和接收端分别使用 Linux 虚拟主机上的 Java 客户端 sender 和 receiver。控制器选择开源 Floodlight0.9 版本，并通过 SSL 与 OpenFlow 交换机远程交换控制信息。

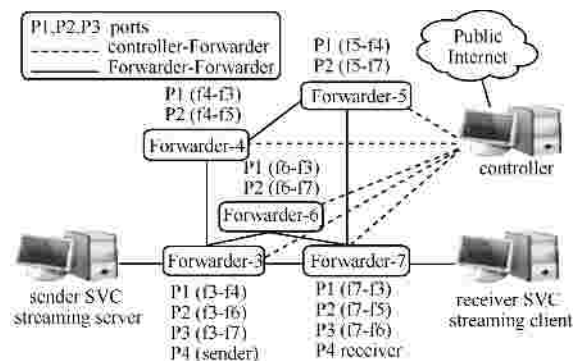


图 4 实验组网拓扑

假设发送帧率为 10 帧/秒可以形成动态效果，则要求客户端每秒钟至少收到 10 帧的 SVC 流媒体数据，所以本文设置播放缓存 $T=1\text{ s}$ ，缓存帧数 $M=10$ ，时延探测周期 $MT=100\text{ ms}$ 。设置链路带宽 $C=3\text{ Mbit/s}$ ，SVC 流媒体发送端根据可用传输带宽以适当的速率发送数据，数据速率 5 Mbit/s ，单路传输不能满足流媒体数据传输要求，以表示网络带宽瓶颈。定义路径 R1 为 f3-f7，R2 为 f3-f6-f7，R3 为 f3-f4-f5-f7，其中，路径 R1 和 R2 的链路时延不定量随机变化，以表示链路拥塞导致时延状态的变化，详细状态如表 2 所示。接收端分别接收并统计基础层和增强层数据，进行实验对比分析。

表 2 实验中网络链路时延状态变化

拓扑编号	路径时延/ms		
	R1	R2	R3
1	50	100	150
2	250	200	150
3	300	400	150
4	500	400	150
5	600	700	150

在图 4 的 OpenFlow 网络环境下，本文进行了 2 种场景的实验。每种场景下，均进行传统传输方案与本文方案的对比实验。传统方案基于最短跳数路由算法的路径传输，SVC 流媒体的基础层和增强层数据传输路径均相对固定，不提供基础层数据最优时延路径传输的保证。新方案基于链路时延的自适应最优路径传输，在网络状态变化的情况下，通过动态自适应路由的计算和切换功能，根据 SVC 流媒体基础层和增强层数据重要性不同实现分级传输，优先保证基础层数据最优时延路径传输。

场景一将 SVC 文件固定为 50 MB，进行传输实验与性能分析，实验结果如图 5~图 8 所示，图 5 和图 6 结果表明，对于相同帧号的基础层和增强层数据，本文提出的传输优化方法有效降低了对应数据的平均传输时延，提高了 SVC 流媒体传输效率。

图 7 描述的是传统方案与新方案的数据帧效用分布对比。数据统计显示，传统方案与新方案按时接收到的基础层数据量分别为 53 和 56；传统方案与新方案按时接收到的增强层数据量分别为 48 和 51。传统方案接收的 SVC 流媒体文件效用平均值

51.5，均方差 49.16；新方案 SVC 流媒体文件效用平均值 54.5，均方差 48.98。图 8 描述了 2 种方案帧效用和随接收帧数量的增长趋势。实验结果表明，新方案比传统方案接收到的 SVC 流媒体文件总体价值和平均质量均得到提高，质量波动范围小，用户体验得到有效改善。

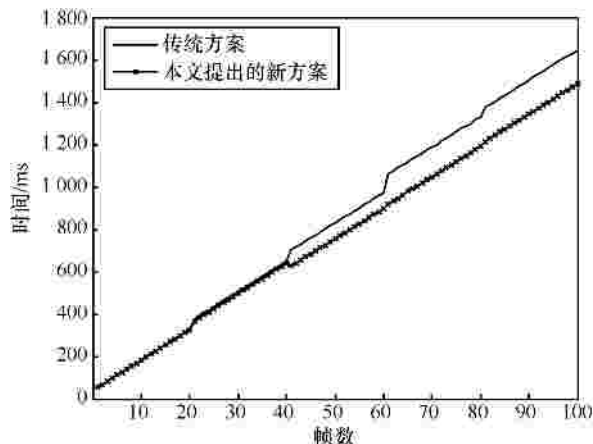


图 5 SVC 基础层数据传输时延对比

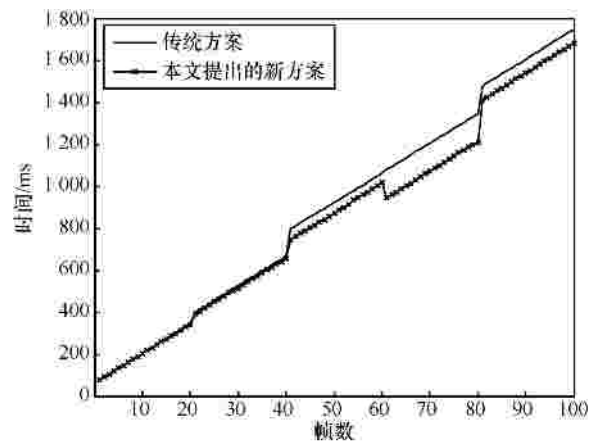


图 6 SVC 增强层数据传输时延对比

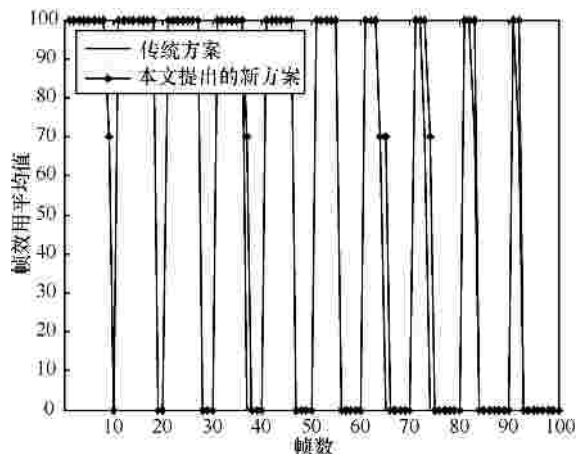


图 7 帧效用分布 ($a=70, T=1\text{ s}$)

表 3 不同 SVC 文件大小传输性能分析

文件大小/MB	传统方案			新方案			2 种方案基础层有效帧数量之差*	2 种方案增强层有效帧数量之差*	2 种方案的总效用差
	基础层有效帧数量	增强层有效帧数量	有效帧效用和	基础层有效帧数量	增强层有效帧数量	有效帧效用和			
0	93	90	9 150	98	93	9 650	5	3	500
100	181	175	17 650	198	190	19 560	17	15	1 910
150	276	267	26 910	298	289	29 530	22	22	2 620
200	375	345	36 180	398	386	39 440	23	41	3 260

这里的“*”表示有效帧数量之差为新方案超过传统方案的数量。

场景二将 SVC 文件大小递增变化,选取 50 MB、100 MB、150 MB、200 MB 的文件,分别进行传输实验,并分析传输优化性能,实验结果如表 3 所示。从统计结果可以看出,新方案的有效帧数量、文件整体效用总是大于传统方案,且文件越大新方案与传统方案的差值越大,即表明传输文件越大,新方案的传输优势越明显,传输效果越好,这一特点使得本方案在解决未来大数据量流媒体的传输将具有重要价值。

上述 2 个场景的实验结果充分证明了本文提出的 SVC 流媒体时延自适应分级传输方法,能够针对延时敏感、数据量大的 SVC 流媒体传输进行优化,有效提高流媒体内容传输效率和质量。需要说明的是,图 7、图 8 和表 3 是在参数设置为 $a=70$, $\beta=30$ 的情况下得到的结果,实验中也分析了参数 a 和 β 取其他值的情况,实验结论基本相同,即参数 a 和 β 取值大小不影响本文提出的传输方法的结论和创新性。

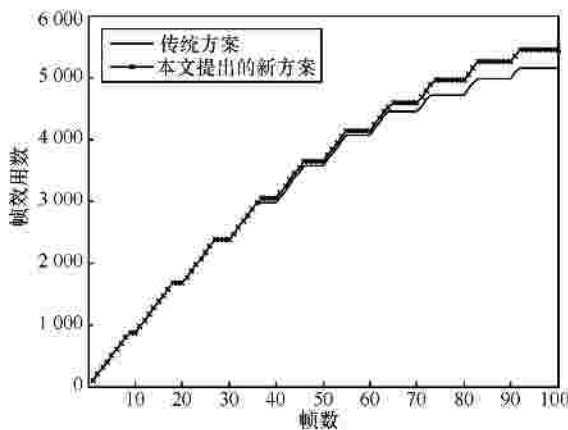


图 8 帧效用和 ($a=70, T=1s$)

6 结束语

流媒体视频通信是网络应用的研究热点,广泛用于在线直播、视频会议、网络电台、电子商务、

视频点播、多媒体新闻发布、远程教育等网络服务各领域。而另一方面,由于当今网络技术领域,尤其是新型网络体系结构的发展如火如荼,其中 OpenFlow 引起了高度关注,成为未来网络重要方向之一。本文借助 OpenFlow 的灵活可编程能力,提出一种基于多路径的 SVC 流媒体时延自适应分级传输方法,在网络带宽不足以及链路时延随机变化的复杂网络环境下,实现了 SVC 流媒体数据高质量传输,有效提高了流媒体连续正确传输的可靠性。

本文基于 OpenFlow 的 SVC 流媒体传输的研究尚处于初级阶段,但是必将对流媒体传输和未来网络应用领域产生重要影响。此外,本文着重关注了单控制域 OpenFlow 有线网络下单播形式的流媒体传输,随着研究的推进,多控制域 OpenFlow 网络、信道条件恶劣的无线网络以及多播模式的流媒体传输,将是后续进一步研究的重要方向。

参考文献:

- [1] 章东轶.流媒体技术的现状和发展前景[J]. 今日科技, 2003(7): 10-11.
ZHANG D Y. Streaming media technology present situation and development prospect[J]. Today Science and Technology, 2003(7):10-11.
- [2] EGILMEZ H, GORKEMLI B, TEKALP A M. Scalable video streaming over openflow networks: an optimization framework for QoS routing[A]. IEEE International Conference on Image Processing[C]. Sept. 2011.
- [3] GORKEMLI B, TEKALP A M, CIVANLAR S. A QoS-enabled openflow environment for scalable video streaming[A]. Globecom Workshops[C]. 2010.
- [4] SCHWARZ H, MARPE D, WIEGAND T. Overview of the scalable video coding extension of the H.264/AVC standard[J]. IEEE Transactions on Circuits and Systems for Video Technology, 2007,17(9): 1103-1119.
- [5] 毕厚杰,王键. 新一代视频压缩编码标准—H.264/AVC[M]. 北京: 人民邮电出版社,2009.
BI H J, WANG J. A New Generation of Video Compression Coding

- Standard—H.264/AVC[M]. Beijing: Posts & Telecom Press, 2009.
- [6] 柳伟, 陈旭, 梁永生. H.264/SVC 的 RTP 封装算法及其应用[J]. 计算机工程与应用, 2010,46(19):27-32.
LIU W, CHEN X, LIANG Y S. H.264/SVC RTP encapsulation algorithm and application[J]. Computer Engineering and Applications, 2010,46(19):27-32.
- [7] 张楠. 基于网络编码的视频传输技术研究[D]. 西安: 西安电子科技大学, 2010.
ZHANG N. Research on Video Transmission Technology Based on Network Coding[D]. Xi'an: Xidian University, 2010.
- [8] SCHAAR M V, REOPOULOS Y, HU Z P. Optimized scalable video streaming over IEEE 802.11a/e HCCA wireless networks under delay constraints[J]. IEEE Transactions on Mobile Computing, 2006, 5(6): 755-768.
- [9] 池琛, 张璇, 薛永林. 基于 H.264 SVC 和 OFDM 的可分级视频传输方法[J]. 电视技术, 2011, 35(3):61-63.
CHI C, ZHANG Y, XUE Y L. Scalable video transmission scheme method based on H.264 SVC and OFDM[J]. Video Engineering, 2011, 35(3):61-63.
- [10] BACCICHET P, SCHIERL T, WIEGAND T. Low-delay peer to peer streaming using scalable video coding[A]. Proc of International Packet Video Workshop[C]. Lausanne, Switzerland, 2007.
- [11] EGILMEZ H E, CIVANLAR S, TEKALP A M. An optimization framework for QoS-enabled adaptive video streaming over OpenFlow networks[J]. IEEE Transactions on Multimedia, 2013,15(3):710-715.
- [12] MCKEOWN N, ANDERSON T, BALAKRISHNAN H, OpenFlow: enabling innovation in campus networks[J]. ACM SIGCOMM Computer Communication Review, 2008,38(4):69-74.
- [13] 陈益富, 卢潇, 豪杰. 对 Dijkstra 算法的优化策略研究[J]. 计算机技术与发展, 2006, 16(9):73-78.
CHEN Y F, LU X, HAO J. Optimized strategy research over algorithm of Dijkstra[J]. Computer Technology and Development, 2006 16(9): 73-78.
- [14] MCKEOWN N, PFA B, HELLER B, *et al.* OpenFlow Switch Specification[S]. Version 1.0.0, December 31, 2009.

作者简介:



黄韬 (1980-), 男, 重庆人, 北京邮电大学副教授, 主要研究方向为路由与交换、软件定义网络、内容分发网络等。

张丽 (1987-), 女, 河北保定人, 北京邮电大学硕士生, 主要研究方向为计算机网络、软件定义网络、媒体传输技术等。

张云勇 (1976-), 男, 江苏盐城人, 中国联通研究院平台与云计算中心主任, 北京邮电大学兼职教授, 主要研究方向为云计算、移动互联网、融合网络技术等。

刘韵洁 (1943-), 男, 山东烟台人, 中国工程院院士, 主要研究方向为未来网络体系架构、网络融合与演进等。