

# 移动网络视频传输速率自适应控制技术的研究

陈 飞<sup>1</sup>, 黄仁乐<sup>2</sup>, 李 艺<sup>1</sup>, 王莹煜<sup>3</sup>

(1. 北京华电天益信息科技有限公司, 北京 102200;

2. 国网北京市电力公司, 北京 100031;

3. 北京科东电力控制系统有限责任公司, 北京 100085)

**摘 要:**目前移动终端上的视频传输与交互技术得到了广泛应用,但由于移动网络带宽波动等原因,网络中的可用带宽往往会小于视频的传输速率,造成视频在传输时出现丢包的情况,对视频传输与交互的质量产生了较大影响。因而如何根据网络状态的变化实时地调整视频的传输速率,从而降低传输过程中的丢包率,保证视频传输与交互的质量,具有至关重要的意义。针对视频传输对于移动网络环境的特殊要求,设计了一种适用于移动网络视频传输的自适应速率控制方法。该方法将 TCP 协议拥塞控制方法进行了改进,在视频传输的过程中依据网络状态的变化情况动态地调整视频的传输速率。仿真实验结果表明,该方法能够保证传输速率在一定范围内平稳变化,提高了网络可用带宽的利用率,降低了传输过程中的丢包率,基本能够满足移动网络环境下对于视频传输与交互质量的要求。

**关键词:**移动网络;视频传输;速率控制;自适应;反馈控制

中图分类号:TP311.1

文献标识码:A

文章编号:1673-629X(2016)10-0177-07

doi:10.3969/j.issn.1673-629X.2016.10.039

## Research on Self-adaptive Control Technology of Video Transmission Speed in Mobile Networks

CHEN Fei<sup>1</sup>, HUANG Ren-le<sup>2</sup>, LI Yi<sup>1</sup>, WANG Ying-yu<sup>3</sup>

(1. Beijing Huadian Tianyi Information Technology Corporation Limited, Beijing 102200, China;

2. State Grid Beijing Electric Power Company, Beijing 100031, China;

3. Beijing Kedong Electric Power Control System Co., Ltd., Beijing 100085, China)

**Abstract:** Nowadays video transmission and interaction on mobile terminals is widely used. However, because the bandwidth of mobile network often fluctuates, it may lead the available bandwidth lower than the transmission speed and packets loss transmitted in network, which in turn affects the quality of video transmission and interaction. Thus how to change the transmission speed timely according to the change of network status, decreasing packet loss ratio in transmission, guaranteeing the quality of transmission and interaction, is very crucial. Aiming at the specific request of video transmission in mobile network environment, a self-adaptive speed control method applicable for mobile network video transmission is put forward. It improves the congestion control method used for TCP, and adjusts transmission speed dynamically according to the variation of network situation during the process of video transmission. The result of simulation shows that this method can guarantee the transmission speed change placidly within limits, which improves utilization ratio of network available bandwidth, and reduces packet loss ratio during transmission. Thus it can basically meet the requirement of video transmission and interaction quality in mobile networks.

**Key words:** mobile network; video transmission; speed control; self-adaption; feedback control

## 0 引 言

随着移动网络及移动终端设备的飞速发展及普及,移动网络视频应用越来越受到广大用户的欢迎。

通过移动网络视频应用,用户不仅可以随时随地点播与收看视频节目,还可及时与家人、朋友及商业伙伴等进行沟通与交互,这也使得近年来人们对于移动网络

收稿日期:2015-11-27

修回日期:2016-03-16

网络出版时间:2016-08-01

基金项目:国家电网公司总部科技项目(0711-130TL21501237-033)

作者简介:陈 飞(1986-),男,博士,研究方向为电力智能化技术、电力信息安全;黄仁乐,硕士,教授级高工,研究方向为智能电网、电力智能化技术。

网络出版地址: <http://www.cnki.net/kcms/detail/61.1450.TP.20160801.0904.034.html>

视频应用的需求越来越大,对于其服务质量的要求也越来越高。然而当大量的视频数据在网络中传输时,会占据大量的网络带宽,而移动网络的可用带宽本身较窄,且带宽波动较大,导致网络中经常出现拥塞,这对于移动网络视频的正常传输与质量保证产生了很大的影响,可能造成视频画面出现延迟或音视频数据不同步等现象<sup>[1-4]</sup>。因而如何根据移动网络的状态动态地调整视频传输的速率,保证所传输的视频数据的完整性和质量,已成为近年来广泛研究的一个课题。

目前在传输速率控制领域,应用最广泛的是 TCP 协议的拥塞控制机制。TCP 协议的拥塞控制机制的核心思想就是“线性增加,乘积减少”,即当网络中尚未出现拥塞时,发送速率呈线性增加,而当网络中出现拥塞时,发送速率以一定倍数减小<sup>[5]</sup>。

对于 TCP 协议来说,由于其在每个环回时间内就会对窗口进行调整,所以这种速率控制策略可以快速地调整窗口以适应网络的变化。然而对于视频传输应用来说,由于其采用 RTP 协议进行数据传输,而 RTCP 协议的发送者报告每 5 s 才会对网络状况进行一次反馈,因而如果单纯地将 TCP 协议的速率控制机制照搬到视频传输应用中,会使得视频传输速率的抖动过于剧烈,并且会加剧网络的拥塞,降低网络带宽的有效利用率<sup>[6]</sup>。

针对移动网络视频传输速率控制方面存在的问题,在对移动网络的特点进行研究的基础上,文中提出一种视频传输速率自适应控制的方法。该方法将 TCP 协议的拥塞控制机制进行了改进,并将其应用于移动网络视频传输中,使其能够根据移动网络状态的改变而动态地对视频传输速率进行调整。仿真结果表明,与简单地将 TCP 协议的拥塞控制机制应用到移动网络视频传输中的方法相比,该方法能够更为高效地利用网络中的可用带宽,实现了传输速率的平稳变化,并降低了视频传输过程中的丢包率,可以有效满足用户对于移动网络视频传输质量的要求。

## 1 移动网络在数据传输方面的缺陷及其对视频传输造成的影响

### 1.1 移动网络在数据传输方面的缺陷

当前的移动网络主要包括蜂窝网络、无线局域网、蓝牙、卫星通讯等几种形式。与传统的有线网络相比,移动网络在数据传输方面主要存在以下缺陷<sup>[7-8]</sup>。

(1)网络带宽较窄。在有线网络环境中,与视频传输有关的应用的可用带宽通常为 70 ~ 700 Mbps。而在移动网络环境中,同类型应用的可用带宽通常仅为 5 ~ 25 Mbps,因而移动网络的可用带宽与有线网络相比明显较窄。数据,移动网络的上行带宽与下行带宽

也不相同,传输过程中的速率也会随时发生变化。

(2)数据传输延迟较高。由于在移动网络环境中物理层的传输信号不稳定,这给数据链路层的数据控制带来了更多的开销,影响了实际的传输速率。因而与有线网络相比,移动网络环境下数据包在传输过程中的延迟时间要更长,并且延迟时间经常会发生波动。

(3)传输过程中的丢包率较高。在有线网络环境中,出现数据包丢失的主要原因是由于网络中出现拥堵,因为在传输过程中发生误码而导致数据包丢失的可能性较小。然而在移动网络环境中,因为传输链路误码导致数据包丢失的概率与有线网络相比要大得多,加之移动网络环境的可用带宽较窄,导致移动网络环境下传输过程中的丢包率大大增加。

(4)网络稳定性较差。由于接入移动网络的终端设备通常都处在位置不断变化的状态之下,尤其在移动终端快速移动的时候,移动终端可能会在多个蜂窝区内进行切换,这可能会导致数据传输链接在发生蜂窝区切换时出现链接断开和重连的情况。因而移动终端在快速移动中,通信链路可能经常会出现中断。

(5)网络拓扑结构较为复杂。尽管移动网络中接入的终端通常处于无线网络环境下,然而其业务服务器通常处于有线网络环境中,因而一般的移动网络业务的数据传输需要穿越无线网络和有线网络,从而导致其网络拓扑结构与有线网络业务相比更为复杂。

### 1.2 移动网络环境对视频传输造成的影响

鉴于移动网络自身存在的缺陷,移动网络中的设备在进行视频传输时会受到以下几个方面的影响。

(1)视频传输将受到网络带宽波动的影响。从理论上讲,当前的移动网络对于视频媒体的传输是可以满足要求的,然而在实际应用中,由于移动网络在传输过程中的可用带宽会发生较大波动,因而会给视频的传输带来很大的问题。

(2)传统的针对有线网络的数据传输拥塞控制机制将不能很好地应用到移动网络视频传输过程中。在目前的传输拥塞控制机制中,应用最广泛的就是 TCP 协议所采用的拥塞控制机制。该机制在网络空闲时呈线性增加其发送速率,而在网络中出现拥塞时,则呈乘性减小其发送速率。这种拥塞控制机制在有线网络中应用时效果较高,然而在移动网络环境中,由于网络的可用带宽的变化往往非常剧烈,这势必将引起视频传输速率的剧烈变化,影响视频播放的最终效果。因而在移动网络环境中,需要对现有的拥塞控制机制进行改进。

(3)移动网络视频传输中的其他问题。在移动网络进行视频传输时还会受到其他因素的影响,如移动终端的电源和能量等问题。

## 2 传输速率自适应控制技术及其控制算法

### 2.1 传输速率自适应控制的总体解决方案

文中提出的传输速率自适应控制技术采用如下的解决思路:首先对视频传输过程中由接收端返回给发送端的 RTCP 数据包进行分析,从中提取与网络状态有关的参数并进行分析,由此构建当前的网络状态,并对后续时间段内网络状态的变化进行估计。随后对于 TCP 协议所采用的拥塞控制机制进行改进,将原有的定常数提高与降低发送速率的方法替换为变常数提高与降低发送速率,并且发送速率提高或降低的幅度随着时间的推移逐渐变小,从而在发送速率可以对网络状态的变化及时做出响应的同时,避免速率的波动过于剧烈。

### 2.2 RTCP 协议

#### 2.2.1 RTCP 协议概述

实时传输控制协议 (Real-time Transport Control Protocol, RTCP) 是与实时传输协议 (Real-time Transport Protocol, RTP) 一道被定义在 1996 年提出的 RFC 1889 中的,与 RTP 协议一起工作的控制协议<sup>[9-10]</sup>。RTCP 单独运行于 UDP 协议之上,由 UDP 协议对其提供数据与控制包的复用。在 RTP 会话过程中,每个 RTP 会话参与者周期性地向所有其他参与者发送 RTCP 控制信息包,如图 1 所示。对于 RTP 会话或者广播来说,RTCP 协议通常会使用单个多目标广播地址,属于这个会话的所有 RTP 和 RTCP 信息包都会使用这个多目标广播地址,由于在会话过程中 RTP 协议和 RTCP 协议使用不同的 UDP 端口号,因而通过使用不同的端口号即可把 RTP 数据包和 RTCP 数据包区分开来<sup>[11]</sup>。

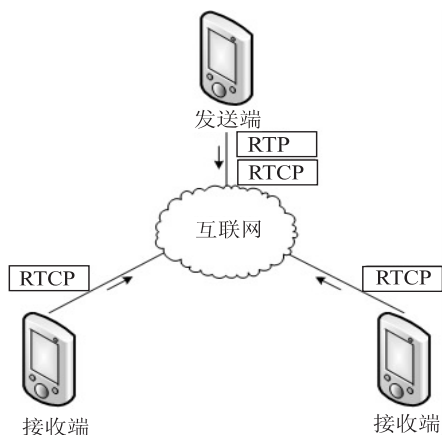


图 1 RTCP 协议工作原理示意图

#### 2.2.2 RTCP 协议的功能

在视频传输及交互过程中,RTCP 协议主要具有以下功能<sup>[12]</sup>。

(1) 为参与会话的应用提供会话质量保证或对会话质量信息进行数据播。

RTCP 协议的主要功能是为参与会话的应用提供会话质量的保证,以及广播会话质量信息。在 RTCP 数据包不对需要传输的音频或视频数据进行封装,而是封装发送端与接收端会话质量的统计信息。这些信息包括发送的数据包的数量、丢失的数据包的数目、传输过程中的延时以及传输延时的抖动情况等。这些信息反映了网络的当前状态,可以为发送端、接收端程序或者网络管理程序调整传输或管理策略提供参考。

(2) 识别与定位会话中的用户。

RTCP 协议为参与到会话中的每个用户提供了一个规范名称 (Canonical Name, CNAME) 的全局唯一的标识符。通过这一标识符,即可对一个 RTP 会话过程中的参与者进行定位。与此同时,会话中的接收者也可以利用 CNAME 在相关 RTP 连接的几个数据流之间建立联系,例如实现音视频同步等。

(3) 对 RTCP 数据包的发送速率进行控制。

由于在会话过程中,每个参与者都会定期地发送 RTCP 数据包,当会话的参与者数量较大时,网络中传输的 RTCP 数据包将占用过多的网络资源。为了防止网络拥塞的发生,必须对 RTCP 数据包的发送速率进行控制,使其所占的带宽不超过可用带宽的 5%。由于会话中任意两个参与者之间都会互相发送 RTCP 数据包,因此比较容易计算出参与者的数量,进而对 RTCP 数据包的发送速率进行调整。

#### 2.2.3 RTCP 数据包格式

RTCP 数据包主要有以下 5 种类型<sup>[13]</sup>。

(1) 发送者报告 (Sender Report, SR)。由 RTP 会话的发送端发送,其中主要包含发送端的信息、媒体间的同步信息、累计发送的数据包的数量以及累计发送的字节数等。

(2) 接收者报告 (Receiver Report, RR)。由 RTP 会话中的接收端发送,其主要内容是对数据传输过程中接收质量的反馈。针对为其发送数据的每个发送端,接收者报告都为其提供丢失报文的数量、已接收报文的最大序列号、平均时延抖动、接收最后一个发送者报告的时间、接收最后一个发送者报告的延时等信息。

(3) 信源描述符 (Source Description Items, SDSE): 包含对发送端进行描述的信息,包括 CNAME 等。

(4) BYE: 参与者结束会话。当会话过程中有参与者离开时,会发送一个 BYE 类型的 RTCP 数据包。

(5) APP: 该类型的 RTCP 数据包通常包含具有特定功能的应用函数。

RTCP 数据包主要由数据包头部分和结构化的报告内容组成,数据包的内容根据类型的不同而有所差异。文中以发送者报告为例进行说明,其数据包结构如图 2 所示。



V	P	RC	PT	Length	首部
SSRC of sender					发送者信息部分
NTP timestamp					
RTP timestamp					
Sender's packet count					
Sender's octet count					
SSRC_1 (SSRC of first source)					接收报告块部分
Fraction lost	Cumulative number of packets lost				
Extended highest sequence number received					
Interarrival jitter					
Last SR (LSR)					
Delay since last SR (DLSR)					↓
SSRC_2 (SSRC of first source)					
.....					↓
Profile-specific extensions					

图 2 发送者报告数据包格式示意图

通常一个发送者报告报文由三部分构成。

(1) RTCP 数据包头部部分。

V:版本号,长度为 2 bit,当前版本为 2。

P:填充标志,长度为 1 bit,如果该标志的值为 1,则在数据包的尾部填充一个八位组。

RC:接收报告计数,长度为 5 bit,指接收报告的块的个数。

PT:报文类型,长度为 8 bit,SR 类型为 200。

Length:报文长度,长度为 16 bit,以 32 位为单位。

SSRC of sender:长度为 32 bit,表示发送该发送者报告的信源的同步标识符。

(2) 发送者信息部分。

NTP timestamp:长度为 64 bit,指该发送者报告发送时的全局网络时间。

RTP timestamp:长度为 32 bit,其所指示的时间与 NTP timestamp 一致,但所使用的时间单位与 RTP 数据包中的相同。

Sender's packet count:长度为 32 bit,表示发送者从开始发送 RTP 数据包到该发送者报告产生所经过的时间间隔内总共发送的 RTP 数据包的数量。

Sender's octet count:长度为 32 bit,表示发送者从开始发送 RTP 数据包到该发送者报告产生所经过的时间间隔内总共发送的字节数。

(3) 接收报告块部分。

SSRC\_n:长度为 32 bit,表示在该时间间隔内曾向该参与者发送过 RTP 数据包的发送者的序号。

Fraction lost:长度为 8 bit,表示在该时间间隔内从 SSRC\_n 接收到的 RTP 数据包的丢包率。

Cumulative number of packets lost:长度为 24 bit,表示从开始接收 RTP 数据包到现在为止累计丢失的数据包数量。

Extended highest sequence number received:长度为 32 bit,表示从 SSRC\_n 接收到的 RTP 数据包的最大序列号。

Interarrival jitter:长度为 32 bit,表示从 SSRC\_n 接

收 RTP 数据包的时延的抖动情况。

Last SR (LSR):长度为 32 bit,表示从 SSRC\_n 接收到的最近一个发送者报告中所记录的该发送者报告发送的时间。

Delay since last SR (DLSR):长度为 32 bit,表示从 SSRC\_n 接收到的最后一个发送者报告到发送该报告之间的时间差。

2.3 视频传输过程中网络状态的测量与预测

2.3.1 网络丢包率的计算与预测

网络丢包率是指传输过程中丢失的数据包的数目与所传输的数据包的总数的比值。一般来说,在网络中的链路与物理设备均工作正常的情况下,网络中出现拥塞是造成传输过程中数据包丢失的最主要原因。当网络中数据包丢失的状况持续出现时,说明网络持续处于拥塞状态,因而网络丢包率可以反映网络在一段时间内的拥塞情况。依据 2.1.3 节所描述的发送者报告,通过提取其中的累计丢失报文数 (Cumulative number of packets lost) 以及接收到的报文的最高序号,即可计算出某一时间间隔内视频传输过程中的丢包率,从而对网络中的拥塞状况进行长时间测量<sup>[14-15]</sup>。令第  $i$  个时间间隔内的丢包率为  $L_i$ ,则  $L_i$  的计算方法如式(1)所示。

$$L_i = \frac{\text{cum\_lost}_i - \text{cum\_lost}_{i-1}}{\text{hi\_num\_recv}_i - \text{hi\_num\_recv}_{i-1}} \tag{1}$$

其中,  $\text{cum\_lost}_i$  表示从视频传输过程开始到第  $i$  个传输时间间隔内累计丢失的 RTP 数据包总数;  $\text{hi\_num\_recv}_i$  表示从视频传输过程开始到第  $i$  个传输时间间隔内所接收到的 RTP 数据包的最大序列号。

由于在发送者报告中只是给出了某一个时间间隔内丢失包的数目,在视频传输过程中,如果简单地根据式(1)中计算出的丢包率对传输速率进行控制,会导致传输速率的波动。为了减小波动,采用一次指数平滑法对预测的丢包率进行平滑,其计算方法如式(2)所示。

$$L'_{i+1} = \alpha L_i + (1 - \alpha) S_{L_i} \tag{2}$$

其中,  $L'_{i+1}$  是第  $i + 1$  个时间间隔内丢包率的预测值;  $S_{L_i}$  是前  $i$  个时间间隔内丢包率的平均值;  $\alpha$  为权重系数,其取值范围为 (0,1)。

2.3.2 网络平均时延抖动的计算

时延抖动指的是网络传输时延的变化情况。与网络丢包率主要对网络中长期的拥塞状况进行描述不同,平均延时抖动 (Inter arrival jitter) 主要是对网络中短期的拥塞状况进行描述。此外,时延抖动的变化可能预示着网络拥塞即将出现。依据发送者报告,第  $i$  个时间间隔内的平均时延抖动  $J_i$  可以直接从发送者报告中获取,即  $J_i = \text{jitter}$ 。

### 2.3.3 网络状态预测量的定义与计算

由于在视频传输过程中,移动网络的网络状况会不断发生变化,因而在传输过程中发送速率需要根据网络状态的变化进行动态调整,以实现视频传输速率的平稳变化。当发送者报告反馈出现 RTP 数据包丢失时,说明此时网络中已经出现了拥塞,而这种拥塞可能是长期的,也可能是暂时的。因此,当视频传输的发送端要对传输速率进行调整时,需要对当前网络中的丢包率以及时延的抖动情况进行统筹考虑,对下一时间间隔内的网络状态做出预测,并据此对传输速率进行调整。令第  $i+1$  个时间间隔内的网络状态预测量为  $P_{i+1}$ ,依据第  $i+1$  个时间间隔内的丢包率预测值  $L'_{i+1}$  以及第  $i$  个时间间隔内的平均时延抖动  $J_i$ ,即可对  $P_{i+1}$  进行计算,其计算方法如式(3)所示。

$$P_{i+1} = \beta L'_{i+1} + (1 - \beta) \frac{J_i}{J_{\max}} \quad (3)$$

其中,  $\beta$  为权重系数,取值范围为  $(0,1)$ ;  $J_{\max}$  为根据统计得出的网络中平均时延抖动的最大值。

在实际应用中,通过对权重系数  $\alpha$  和  $\beta$  的不断调整,可以最大程度地拟合移动网络中网络状态的变化情况,从而更精确地对发送速率进行控制,避免视频传

$$v_{i+1} = \begin{cases} \min \left\{ v_i + \left( 1 - \frac{\sqrt{\theta - \beta [\alpha L_i + (1 - \alpha) S_{L_i}] - (1 - \beta) \frac{J_i}{J_{\max}}}}{\beta [\alpha L_i + (1 - \alpha) S_{L_i}] + (1 - \beta) \frac{J_i}{J_{\max}}} \right) \times v_0, v_{\max} \right\}, & \text{若 } 0 \leq P_{i+1} \leq \theta \\ \max \left\{ \left( 1 - \frac{\sqrt{\beta [\alpha L_i + (1 - \alpha) S_{L_i}] + (1 - \beta) \frac{J_i}{J_{\max}} - \theta}}{\beta [\alpha L_i + (1 - \alpha) S_{L_i}] + (1 - \beta) \frac{J_i}{J_{\max}}} \right) v_i, v_{\min} \right\}, & \text{若 } \theta \leq P_{i+1} \leq 1 \end{cases} \quad (5)$$

通过式(5)可以看出,当  $P_{i+1} \leq \theta$  时,说明此时网络状态较好,网络中无拥塞现象或拥塞程度较低,此时可提升发送速率;而当  $P_{i+1} > \theta$  时,说明此时网络中拥塞较为严重,网络状态较差,此时应降低发送速率。并且由该公式的函数性质可知,  $v_{i+1}$  分别为  $L_i$  和  $J_i$  的减函数,也就是说随着  $L_i$  和  $J_i$  的增加,发送速率提升和降低的服务都趋于收敛,因而采用该公式对发送速率进行控制,可以降低发送速率的变化程度,使发送速率趋于平稳。另外,在网络传输过程中,发送速度要受到网络带宽的限制,因而根据网络状态预测量计算出的发送速率需要与预先设定的最大和最小发送速率进行比较,确保发送速率在可以接受的范围内。

文中提出的视频传输速率自适应控制算法的工作流程描述如下。

(1)将发送速率的初始值设置为  $v_0$ ,开始进行视频传输。万方数据

输过程中 RTP 数据包的丢失,并减小抖动的程度。

### 2.4 速率控制算法的工作流程

文中所提出的视频传输速率自适应控制算法利用 RTCP 作为传输反馈控制协议,引入网络状态预测量  $P$  作为判断当前网络状态的依据,并采用变常数提高与降低的方法来对视频传输过程中的发送速率进行调整。令  $v_{i+1}$  为第  $i+1$  个时间间隔内的发送速率,其具体的计算方法如式(4)所示。

$$v_{i+1} = \begin{cases} \min \left\{ v_i + \left( 1 - \frac{\sqrt{\theta - P_{i+1}}}{P_{i+1}} \right) \times v_0, v_{\max} \right\}, & \text{若 } 0 \leq P_{i+1} \leq \theta \\ \max \left\{ \left( 1 - \frac{\sqrt{P_{i+1} - \theta}}{P_{i+1}} \right) v_i, v_{\min} \right\}, & \text{若 } \theta \leq P_{i+1} \leq 1 \end{cases} \quad (4)$$

其中,  $\theta$  为决定在下一时间间隔内发送速率提高或降低的阈值,其取值范围为  $[0,1]$ ;  $v_0$  为视频传输开始时的发送速率;  $v_{\max}$  和  $v_{\min}$  分别表示预先设定的最大和最小发送速率。

把式(2)代入到式(4)得到式(5)。

(2)从发送者报告中提取前  $i$  个时间间隔内的累计丢失报文数以及接收到的报文的最高序号,根据式(2)计算第  $i+1$  个时间间隔内丢包率的预测值  $L'_{i+1}$ 。

(3)从发送者报告中提取第  $i$  个时间间隔内的平均时延抖动值  $J_i$ 。

(4)根据式(3)计算第  $i+1$  个时间间隔内的网络状态预测量  $P_{i+1}$ 。

(5)将  $P_{i+1}$  与阈值  $\theta$  进行比较,若  $P_{i+1} \leq \theta$ ,按照式(5)计算第  $i+1$  个时间间隔内的发送速率预测值  $v'_{i+1}$ ,并将其与预先设定的最大发送速率  $v_{\max}$  进行比较,取二者中较小的一个作为第  $i+1$  个时间间隔内的发送速率  $v_{i+1}$ ,并等待下一个发送者报告的产生。否则进行第(6)步。

(6)按照式(5)计算第  $i+1$  个时间间隔内的发送速率预测值  $v'_{i+1}$ ,并将其与预先设定的最小发送速率  $v_{\min}$  进行比较,取二者中较大的一个作为第  $i+1$  个时

间间隔内的发送速率  $v_{i+1}$ , 并等待下一个发送者报告的产生。

文中提出的视频传输速率自适应控制算法的工作流程如图 3 所示。

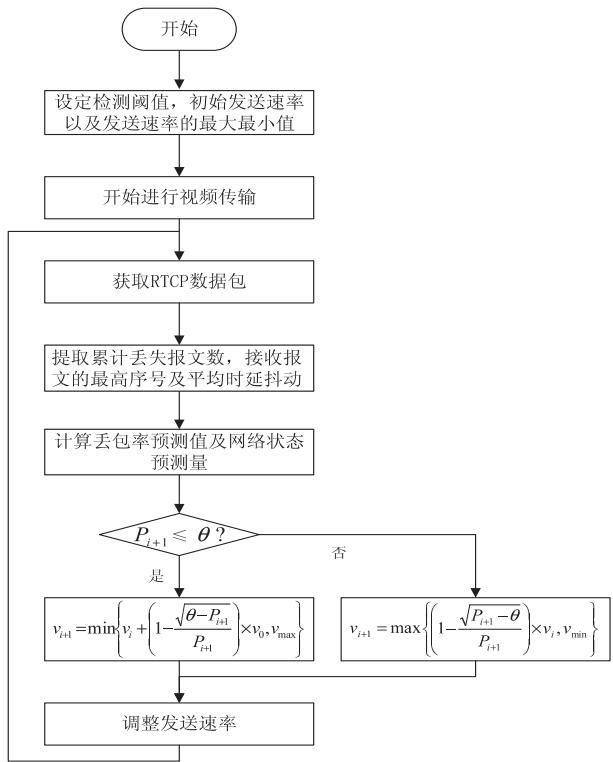


图 3 视频传输速率自适应控制算法工作流程示意图

### 3 实验与分析

为了对文中提出的视频传输速率自适应控制算法的有效性和实用性进行验证, 保证实验评价的客观性, 利用 OMNeT++ 仿真软件构建了一个典型的哑铃状单瓶颈链路网络, 并对上述算法进行仿真实验。OMNeT++ (Objective Modular Network Testbed in C++) 是一个专门为大型网络提供的开源的、基于组件的、模块化的、开放的、面向对象的离散事件网络仿真工具。OMNeT++ 具备非常强大和完善的图形界面, 并能够对图形界面中的参数进行配置, 同时具备编程、调试和跟踪支持的功能, 能够有效而便捷地对通信网络及其他复杂 IT 系统进行仿真。与此同时, OMNeT++ 提供对用户组件库的支持, 可以实现模块化的灵活重用, 允许对仿真内核提供扩展, 并且提供了图形化的网络编辑和网络数据流查看工具。与其他开源的网络仿真软件 (如 NS2 等) 相比, OMNeT++ 具有高度的模块化特性, 并可运行于多个操作系统平台, 具备更好的灵活性<sup>[16-17]</sup>。文中构造的哑铃状单瓶颈链路网络的拓扑结构如图 4 所示。

在图 4 中,  $S_1$ ,  $S_2$  和  $S_3$  为视频传输过程中的发送端,  $R_1$ ,  $R_2$  和  $R_3$  为接收端,  $N_1$  和  $N_2$  是网络中的两个中

间节点, 网络中链路的带宽均为 4 Mbps, 传输延迟均为 20 ms。发送端每 5 s 向所有接收端发送一次发送者报告, 设置网络状态预测量的检测阈值  $\theta$  为 0.43, 权重系数  $\alpha$  和  $\beta$  分别为 0.36 和 0.41。初始发送速率设定为 200 kbps, 最大和最小发送速率分别设定为 500 kbps 和 50 kbps。实验持续时间为 600 s, 由  $S_1$ ,  $S_2$  和  $S_3$  分别向  $R_1$ ,  $R_2$  和  $R_3$  传输视频数据。

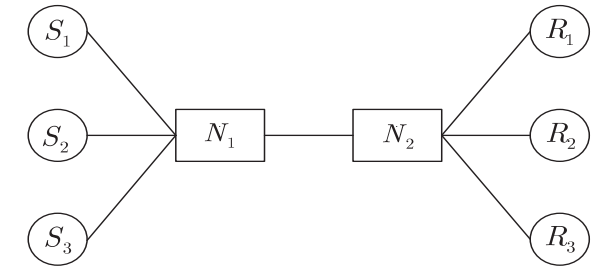


图 4 哑铃状单瓶颈链路网络拓扑结构示意图

实验结果显示, 简单地采用 TCP 协议拥塞控制的传输速率控制机制的发送速率在整个实验过程中的波动非常剧烈, 而文中所提出的传输速率控制算法的速率波动则明显较为平缓。在经过一段时间的调整之后, 视频的发送速率逐步稳定在 280 ~ 295 kbps, 有效提高了网络中有效带宽的利用率。实验中两种方法的速率变化情况见图 5, 以  $S_1$  发往  $R_1$  的视频数据为例。

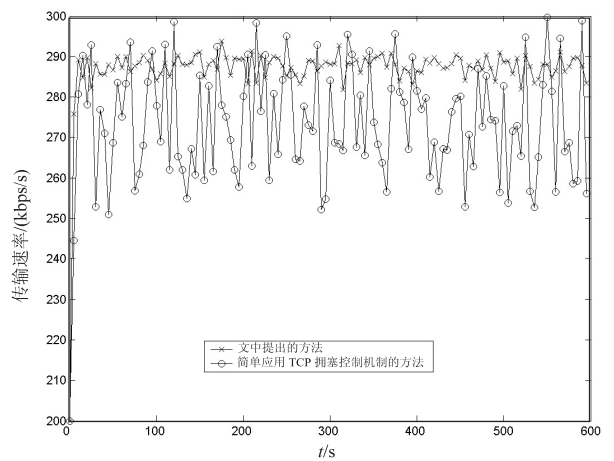


图 5 视频发送速率波动情况对比示意图

此外, 实验也对视频传输过程中的丢包率进行了统计, 统计结果如表 1 所示。

表 1 两种方法在视频传输过程中丢包率的比较		
数据流向	简单应用 TCP 拥塞控制机制的方法/%	文中提出的方法/%
$S_1 \rightarrow R_1$	2.09	1.46
$S_2 \rightarrow R_2$	1.95	1.39
$S_3 \rightarrow R_3$	1.98	1.47

结果显示, 文中提出的传输速率控制算法在整个实验过程中的丢包率被控制在 1.5% 以下, 与简单地采用 TCP 协议拥塞控制的传输速率控制机制相比有

了明显提高。由此可见,文中提出的传输速率控制算法在保证传输速率平稳变化的同时,降低了视频传输的丢包率,保证了接收端视频播放的质量,满足了移动网络应用对于视频传输质量的要求。

## 4 结束语

移动视频传输技术及交互应用的出现和普及,为人与人之间沟通与交流提供了极大的便利。然而视频数据在移动网络中的传输往往会受到移动网络带宽波动等不利因素的影响,导致视频数据无法正常传输,影响视频传输与交互的质量,因而根据网络状态实时对视频传输速率进行调整至关重要。现有的传输速率控制机制虽然对于有线网络及基于 TCP 协议的数据传输调控效果较好,但其调控幅度对于视频传输来说过于剧烈,反而会影响视频交互的效果。

针对目前视频传输在移动网络中存在的问题,文中提出了一种视频传输速率自适应控制方法。该方法对于 TCP 协议的拥塞控制方法进行了改进,通过对视频传输中产生的 RTCP 数据包的内容进行分析,提取与当前网络状态有关的信息,并以此来对网络状态的变化情况进行分析与预测,进而依据网络状态的变化动态地对视频传输过程中的速率进行调整。该方法在提高了网络带宽利用率的同时,降低了视频传输过程中的丢包率,提高了视频传输与交互的质量。在今后的工作中,可将该方法应用于移动视频交互应用中,从实践角度对该方法的正确性进行验证。

## 参考文献:

[1] 郑启鸿. 移动网络通信质量测评方案的分析与研究[J]. 信息技术与信息化,2015(2):47-48.

[2] Samanchuen T, Kiattisin S. Implementation and quality evaluation of video telephony using session initiation protocol[C]//Proc of 2014 Asia-Pacific signal and information processing association annual summit and conference. Chiang Mai, Thailand: Asia-Pacific Signal and Information Processing Association, 2014.

[3] Qi X, Yang Q, Nguyen D T, et al. LBVC: towards low-bandwidth video chat on smartphones[C]//Proceedings of the 6th ACM multimedia systems conference. Portland, USA: ACM, 2015.

[4] Aktas I, Schmidt F, Weingärtner E, et al. An adaptive codec switching scheme for SIP-based VoIP[M]//Internet of Things, Smart Spaces, and Next Generation Networking. Berlin: Springer, 2012: 347-358.

[5] Kuang T, Tang P. A new feedback streaming media congestion control algorithm[J]. Journal of Chemical & Pharmaceutical Research, 2014, 6(6): 2286-2289.

[6] 梁 皓, 骆新全. 一种基于 RTCP 的自适应流媒体拥塞控制算法[J]. 中国传媒大学学报: 自然科学版, 2014, 21(5): 28-31.

[7] 张成伟, 程文青, 黑晓军. 基于 Android 平台的 3G 移动网络测量研究及性能分析[J]. 计算机科学, 2015, 42(2): 24-28.

[8] 李晓城, 钱松荣. 3G 网络中流媒体点对点自适应传输策略[J]. 计算机工程, 2011, 37(20): 66-68.

[9] 许 宁. 基于 RTP 协议的移动视频监控系统的的设计[J]. 通信技术, 2014, 47(4): 455-458.

[10] Ghosh R, Goswami P. Implementation on adaptive transmission of real time multimedia data[J]. International Journal of Research in Engineering & Technology, 2014, 2(4): 175-184.

[11] Volkert T, Mitschele-Thiel A, Becke M, et al. Homer conferencing - a multimedia test bed for various experiments and measurements[C]//Proc of 7th international conference on computing and convergence technology. Seoul, Korea: IEEE, 2012: 224-229.

[12] Schulzrinne H, Casner S, Frederick R, et al. RFC3550 RTP: a transport protocol for real-time applications[S]. Fremont, USA: Internet Engineering Task Force, 2003.

[13] Xu Y, Hu H. Development of a network adaptive H. 264/AVC medical video transmission system[J]. International Journal of Future Computer and Communication, 2013, 2(4): 292-295.

[14] Jiang W, Meng L, Ying S, et al. A control mechanism of stream media based on 3G network[J]. Journal of Software Engineering and Applications, 2013, 5(12): 8-13.

[15] 刘国柱, 王洪林. RTCP 反馈和缓冲区的传输控制算法研究[J]. 计算机工程与设计, 2010, 31(12): 2704-2706.

[16] 杨 宁, 李满坡, 宋元成, 等. 基于 OMNeT 的电力调度数据网仿真[J]. 计算机系统应用, 2014, 23(6): 28-34.

[17] Dhobale J V, Kalyankar N V, Khamitkar S D. Computer network performance evaluation based on different data packet size using OMNeT++ simulation environment[J]. Global Journal of Computer Science and Technology, 2014, 5(5): 2041-2045.