IP 视频会议系统中音视频同步的研究

曹 宁 胡建荣 马银松
(河海大学电子信息工程系,南京 210098)

摘 要 在 IP 视频会议系统中,音频流需要保持其连续性(媒体内同步),而视频流的变化应与音频信息保持一致(媒体间同步)。由于网络状态的变化给传输带来延迟抖动,因此接收方要如实地播放,就必须进行同步控制,以恢复数据之间的时间关系。为此探讨了在 MCU 集中管理模式下的视频会议系统中,音视频同步控制的问题,并结合 RTP/RTCP 协议给出具体可行的解决方案。

关键词 视频会议 音视频同步 RTP/RTCP
中图法分类号: TP393 TN91 **文献标识码**: A **文章编号**: 1006-8961(2005)02-0255-05

Research of the Audio-Video Synchronous Technology for Video Conference System over IP

CAO Ning, HU Jian-rong, MA Yin-song
(*Department of Electronic and Information Engineering, Hohai University, Nanjing* 210098)

Abstract Audio stream needs to keep continuous, and video stream movement should be consistent with audio stream in Video-Conference System over IP. Varieties of net conditions bring transmission delay jitter. The receiver must take synchronization control to resume time relation of stream before playing them according to the facts. This paper discusses the synchronization control in convergence MCU mode and present some specific and operatable solutions.

Keywords video-conference, audio-video synchronous, RTP/RTCP

1 引 言

考虑到现有协议、路由软件的支持程度以及数据传输的可靠性、保密性,基于 H.323 建议[1]的视频会议系统采用集中 MCU 管理模式,如图 1 所示。MCU 通过与系统数据库模块和媒体数据库模块的协同工作,有效地管理各个会议终端,实现音视频等媒体数据流的实时转发。媒体数据库用于保存与会各会议终端采集的音频,视频数据;系统数据库用于保存会议终端的各种信息。

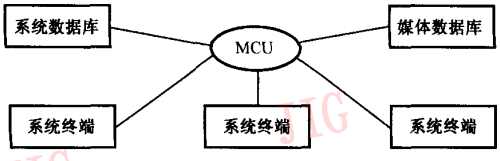

图 1 会议系统功能结构模型

Fig. 1 System function architecture model

在基于 H.323 建议的视频会议系统中,终端视频流和音频流分别直接发送到 MCU,两种媒体流通过独立的信道传输。这样可以根据对各媒体流的服务质量的要求,对相应的信道单独加以控制。由于经过压缩的视频数据量依然很大,分开音视频数据流传输的处理方法可以在带宽较窄的网络条件下优先保证音频数据的平滑传输,但这样带来了声像不同步的问题。因此,如何在目的端正确地恢复多种媒体间(这里主要是音视频流)的关系,即实现媒体间的同步,成为视频会议系统的关键技术之一。

收稿日期:2004-02-26;**改回日期**:2004-07-05

第一作者简介:曹宁(1962 ~),男,教授。1984 年于东南大学无线电系获电子信息工程学士学位,1990 年于东南大学无线电系获信号与信息处理硕士学位。研究方向为多媒体图像压缩及传输、数字信号处理。E-mail:caoning@vip.163.com

2 音视频同步方案

在视频会议系统中，除了音视频流能连续地在接收端播放，即实现媒体内同步外，音视频流同步的表现，图像和声音的同步程度也是一项重要的性能指标。

引起音视频流不同步的原因大致可分为两种：一种是终端处理数据引起的，发送终端在处理采样、编码、打包等模块和接收终端在处理解包、解压、回放等模块时，由于音频和视频的数据量以及编码算法各不同而引起的时间差；另一种是网络传输时延，网络传输时延是受到网络的实时传输带宽、传输距离和网络节点的处理速度等诸多因素的影响，在网络阻塞时，媒体信息不能保证以连续的“流”数据方式传输，特别是不能保证数据量巨大的视频信息的连续传输，从而引起媒体流内和流间的失步[2]，如图 2 所示。

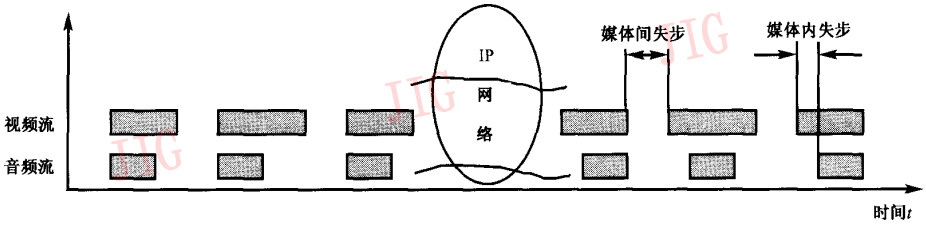

图 2 网络传输时延导致的媒体内失步和媒体间失步

Fig. 2 Intra and inter loss of synchronization resulted by transmission

因此，从媒体流间失步的原因来看，同步的解决主要分为

(1) 发送端同步：有效地控制源节点上音视频帧的发送时间，相同时间采集的音视频帧应当同时发送。但当网络传输时延抖动严重时，该方法难以取得较好的效果。

(2) 接收端同步：目标节点设置缓冲区，消除网络传输产生的抖动，使系统能够同步播放接收到的音视频帧数据。

(3) 根据网络状态，实时控制多媒体数据的发送量，及时有效地解决同步问题。

在设计的视频会议系统中，采取了图 3 所示的层次结构。

根据 H.323 建议，采用了 RTP/RTCP 协议[3]来保证音视频数据流的实时有效传送。

RTP 本身用于传送实时数据，其功能是提供净荷类型指示（数据类型和编码方法）、数据分组号、发送时戳、数据源指示。接收端根据这些信息可以正确地重组原始信号。RTP 本身只保证实时数据的传输，并不能为按顺序传送数据包提供可靠的传送机制，也不提供流量控制或拥塞控制，它依靠 RTCP 提供这些服务。

RTCP 则用以传送控制分组，提供 QoS 监视机制。在 RTP 会话期间，各参与者周期性地传送 RTCP 包，包中含有已发送的数据包数量、丢失的数据包数量等统计资料，可以利用这些信息动态地改变传输速率。RTP 和 RTCP 配合使用，能以有效的反馈和最小的开销使传输效率最佳化。RTCP 反馈可以直接作用于编码、发送，甚至协议选择环节。

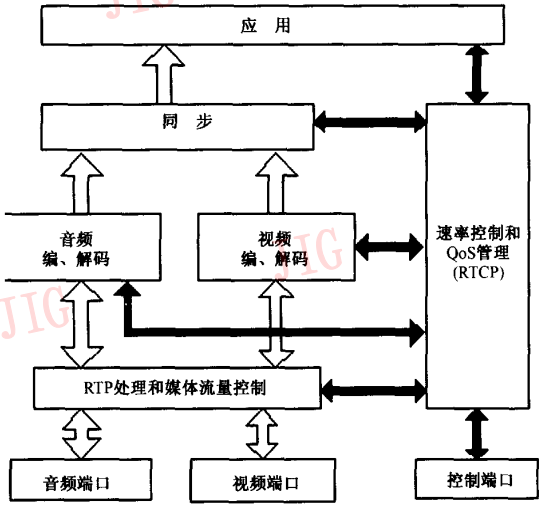

图 3 系统层次结构图

Fig. 3 System layer structure

由于音视频流作为不同的 RTP 会话传送，它们在 RTP 层无直接关联。尽管由一个数据源发出的不同的流具有不同的同步源标识（SSRC），为能进行流同步，RTCP 要求发送方给接收方传送一个唯一

的标识数据源的规范名(canonical name),应用层藉此关联音视频流,以便实现同步。

3　用 RTP 时间戳实现同步

网络畅通时,网络带宽能保证音、视频流按照预定的速率传输,网络传输时延基本恒定,抖动很小,发送端和接收端的音视频流帧间间隔基本保持一致,媒体数据基本没有丢失。因为系统中无全网同步的时钟可用,所以无法利用 RTCP 中 SR 分组包中的绝对时间戳 NTP 来实现音视频同步,所以此时同步问题主要利用 RTP 包头的时间戳字段 TimeStamp 来解决[4],如图 4 所示。

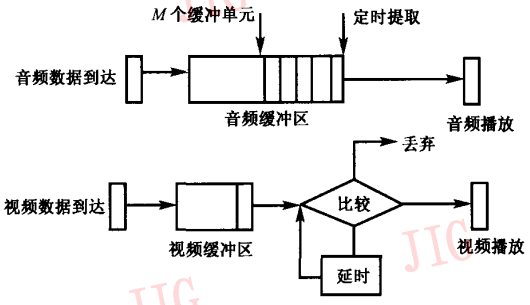

图 4　利用时间戳实现音视频同步

Fig. 4　Implement of synchronization based on timestamp

RTP 时间戳是一个长度为 32 比特的字段,它用来表示 RTP 数据字段中第一个字节的采样时刻。时戳的时间表示为线性单调递增的。如果若干个 RTP 数据包的数据是同时产生(如:一帧图像),那这几个 RTP 包会有相同的时间戳 。

在发送端,同一时刻采集的音视频帧打上相同的时间戳,尽可能地同时发送出去。这里可以把发送过程放到同一线程中处理,也即依次发送视频包、音频包、视频包……,这样可以保证发送端的同步。因为在一个视频帧的采样时间内采集了多个音频帧,所以一个音频包中有多个音频帧。

当 RTP 帧到达接收端后,随后被送至相应的解码器进行解码。由于每个数据帧的复杂度不相同,因此不可能精确知道解压缩所消耗的时间。为准确地控制回放时间,数据包在进行同步之前应由软件解码器进行解压缩。这样解压缩时间作为网络延迟的一部分,解压后的时间作为数据包的到达时间;但在下面将要讨论的计算延迟抖动时未进行解压,解压时间不作为网络延迟的一部分,这样延抖动才能真正反映网络拥塞情况。

系统中各媒体流采用的是虚轴(virtual axes)模型,即音视频流有各自的时间轴,发送时各帧打上相对时间戳(RTS)。考虑两个流,一个主流,一个从流,主流连续播放,从流的播放由主流的播放状态决定,从而实现同步。对于音频流和视频流,由于听觉对声音的不连续比视觉对图像不连续的敏感程度要高,因而选择音频流为主流,视频流为从流,音频流连续播放。

(1) 首先,为消除抖动带来的影响,采用基于缓存的同步方法,保证媒体流尤其是音频流的连续性。接收终端设置音频接收缓冲区(Jitter buffer)。缓冲区设定一个门限值 M,该值比预计最长抖动时间要大。开始时,数据在缓存中积累,直到缓存大小达到 M。此时,回放开始,定时从音频缓冲区中取出语音包,送入音频设备播放,并记录当前播放帧的时间戳 T_{play}。这样做的好处是考虑语音的敏感性(这里暂不考虑由音频播放的时间)。同时当新的音频数据分组到达,被加到缓存中时,只要没有分组的时延超过 M 个缓存单元,缓存中就有足够的数据用于连续播放。

(2) 视频帧到达时,把该帧的时间戳 T_v 与 T_{play} 比较。规定音视频帧不同步的容忍度为 T_{av} = 120ms。

若 $T_{play} - T_{av} \leqslant T_v \leqslant T_{play} + T_{av}$,就播放该视频帧。

若 $T_v < T_{play} - T_{av}$,视频帧滞后,就丢弃该帧。

若 $T_v > T_{play} + T_{av}$,视频帧超前,等待下一次定时读取音频帧时再处理。

(3) 这里音频的播放速率一定,不考虑终端工作负载的变化。若用缓存的方式来读取视频数据,当画面出现高速运动的物体时,数据会大量缓冲,造成画面出现严重的块效应,因此,正常情况下,采用到达事件驱动的方式播放视频流,而不采用定时读取的方式。为了防止音频流失步时,出现视频流数据丢弃较为严重的现象,系统中必须优先保证音频流的同步,并且能够连续播放。

视频处理算法如下:

```
HandVideoFameArrival: //视频帧到达
    if(getVideoFrame(v)//取出视频帧
    {
      if (Tv < Tplay - Tav) //滞后,丢弃;
      else if (Tv > Tplay + Tav) // 超前;
      {
```

```
            VideoFrameWaitNum ++; //缓存区待处理数据包数目加 1
            }
            else play(v) //音视频同步,播放视频
        }
```

音频处理算法如下:

```
    HandleAudioFrameTime: //定时时间到
        getAudioFrame(a);//取出音频帧
        A_play = getCurrentPlayingAudioTimestamp; //得到正在播放的音频帧时间戳
        if (VideoFrameWaitNum! =0) //若视频缓存不为空
            doHandleVideoFrame();//提取视频缓存中的帧处理,处理流程与视频帧到达处理 HandVideoFrameArrival 类似
        play(a); //送入音频设备播放
```

4 RTCP 对 QoS 反馈控制

当网络环境较差,而网络无法为系统提供 RSVP 时,音、视频流不能按原定的传输速率传送,音、视频信息包丢失严重。这时需要由 QoS 控制模块实现反馈控制。

H.323 协议利用 RTCP 的发送报告 SR 和接收报告 RR 包监测 QoS。接收终端将 RR 包发送给源端,该报告包含用来估算分组丢失和分组延迟抖动等必要信息。源端根据这些信息控制媒体数据的发送量,及时有效地解决同步问题。

按照协议,接收报告 RR 定义如下:

```
struct ReceiverReport {
    u_int32 ssrc; /* data source being reported */
    BYTE fraction; /* fraction lost since last SR/RR */
    BYTE lost[3]; /* cumulative number of packets lost */
    u_int32 last_seq; /* extended last sequence number received */
    u_int32 jitter; /* interarrival jitter */
    u_int32 lsr; /* last SR packet from this source */
    u_int32 dlsr; /* delay since last SR packet */
} RR;
```

4.1 网络状态评价指标与算法

根据 SR/RR 包中包含的累计计数的数据项,可以计算任意两个报告间网络状况的差别。传输质量的测算分为长时指标和短时指标。

4.1.1 长时指标

(1) 间隔内丢失的包数目 通过计算两个到达的包之间的丢失包的累数目(cumulative number of packets lost)的差值,可得到此间隔内丢失包的数目。

(2) 预期收到的包数目 在一段时间内预期收到的包数目,可以用这段时间内前后两个包的最大序列号(extended last sequence number received)相减得到。

(3) 间隔内包丢失比率 即为间隔内丢失的包数目与预期收到的包数目的比率。该指标用来判断网络的长期性堵塞。如果是连续两个包,则该比率等同于包丢失率。

4.1.2 短时指标

这里主要为间隔抖动:输入为 r->ts,表示到达的 RTP 包中的时间戳。arrival 代表接收到这个包的时间。指针 s 是指向该源的结构指针。s->transit 表示前一个 RTP 包的传输时间即从发送方发出到接收方收到这段时间。s->jitter 表示估计的抖动(浮点数)。rr 为接收报告。当每个 RTP 数据包到达时,抖动估计值将被更新。该指标用来判断网络的短时性堵塞。

算法如下:

```
int transit = arrival - r->ts;
int d = | transit - s->transit|;
s-> transit = transit;
if (d<0) d = -d;
s->jitter + = (1.0/16.0) * ((double)d - s->jitter);
rr->jitter = (u_int32)s->jitter;
```

4.2 QoS 调整

根据以上分析,当丢包率和抖动达到一定值时,就调整视频数据的发送量。

(1) 图像质量优先(privilege quality),即修改帧速,这样会造成图像不流畅现象(低于 15fps);

(2) 帧速优先(privilege frame rate),即保证图像的流畅。调整编码器的量化值,使用更松散的量化因子,降低图像精度,减少每帧图像的数据量。

下面以 MCU 转发终端 1 的媒体数据到终端 2 来说明如何利用 RTCP 的反馈实现 QoS 控制,如图 5 所示。

MCU 作为接收者:MCU 接收到终端 1 的媒体数据之后,向终端 1 发送一个接收者报告 RR 。此时 终端 1 分析从 MCU 过来的 RR 包,分析丢包率和延时抖动,调整视频数据发送量。

MCU 作为发送者:终端 2 接收到 MCU 转发过

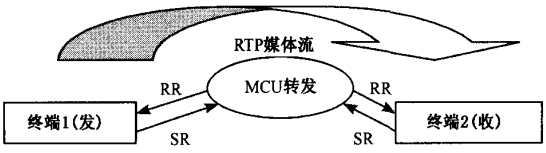

图 5　终端 1 数据由 MCU 转发给终端 2

Fig. 5　MCU forwards data from endpoint 1 to endpoint 2

来的媒体数据后,发送一个 RR 包;MCU 分析从终端 2 过来的 RR 包,分析丢包率和延时抖动,调整转发给终端 2 的数据量。

考虑到 RTCP 的发送需要占用一定的用户带宽,为避免 RTCP 包泛滥,应有效控制 RTCP 发送间隔。通常,RTCP 的流量应不大于媒体流量(RTP)的 5%,其中 1.25% 安排给发送者(SR 包),3.75% 安排给接收者(RR 包)。传送周期不超过 5s,以便能够实时地动态调整视频数据发送模块,使视频质量(帧速和图像质量)到达接收者满意的程度,实现良好的声像同步。

5　MCU 及媒体数据库在同步控制的作用

MCU 设置视频解码模块,当网络阻塞时,可同终端一样调整编码器参数,丢弃一些不重要的帧,控制转发给终端的数据量。虽然终端之间的媒体数据通过 MCU 多点单播转发,会带来数据的冗余以及较高的带宽利用,但可以通过 MCU 控制转发给该接收终端的视频数据量,而不影响转发给其他接收终端的数据量,当然这样会增加 MCU 的负担。因此当一定比例(如 50%)的接收终端的可见网络出现阻塞时,MCU 就通知源端降低视频数据发送量,而多点混音模块设置在终端。来自同一源端的音视频流同步后,音频流再经过混音模块混音后,最后才送往设备播放。

这里引入媒体数据库这个概念,简单实现是在 MCU 设置音视频接收、发送缓冲池模块,缓冲池的大小动态可变,可控制媒体数据的发送速率和时间,以获得较为理想的媒体内同步和媒体间同步(即先在 MCU 同步一次),但因此引入的延时会影响视频会议的实时性,所以应尽可能地兼顾系统的实时性和同步性。

6　结　论

鉴于现有路由协议和网络条件,讨论了基于集中 MCU 模式 IP 视频会议中的音视频同步的方案,并针对带宽资源共享,动态调整的网络环境提出如何应用 RTCP 协议对流媒体 QoS 进行控制,以增强对对网络的适应性。

多媒体同步有很多有关表现质量的要求,收端很难达到和本地回放几乎一样的效果。另外实现媒体间同步必须考虑媒体内的同步,尽量减少同步带来的延时,以达到流的实时连续播放。下一步将在实践中研究传输网络的数学模型,设计出更好的音视频同步解决方案。

参考文献(Reference)

1　ITU-T Recommendation H.323 V4-2000. Packet-based Multimedia Communication System[S].

2　Zheng Li-ming, Zhang Hui-ting, Liu Wei-ping, *et al*. Research and implemen of the video-sound synchronous technology of the distributed IP videoconferencin system [J]. Computer Engineering and Aplication, 2002, **11**:227 ~ 229. [郑力明,张会汀,刘伟平等. 分布式 IP 视频会议系统中声像同步技术的研究与实现[J]. 计算机工程与应用,2002, **11**:227 ~ 229.]

3　RFC 1889-1996, RTP: A Transport Protocol for Real-time Applications[S].

4　Yoshitaka Shibata, Naoya Seta, Shogo Shimizu. Media synchronization protocols for packet audio-video system on multimedia information networks[A]. In:Proceedings of the 28th Annual Hawaii international Conference on System Sciences [C], Kihei, Maui, Hawaii, USA, 1995:594 ~ 601.