Darmstadt University of Technology

P2P Streaming using Hierarchically Layer Encoded Video

Michael Zink

KOM Technical Report 01/2003
Version 1.0
January 2003

E-Mail: zink@kom.tu-darmstadt.de

Multimedia Communications (KOM)

Department of Electrical Engineering & Information Technology
& Department of Computer Science
Merckstraße 25 • D-64283 Darmstadt • Germany

Phone: +49 6151 166150
Fax: +49 6151 166152
Email: info@KOM.tu-darmstadt.de
URL: http://www.kom.e-technik.tu-darmstadt.de/

1. Introduction

Recently, peer-to-peer applications have become very popular in the Internet. In most cases they are used for file sharing purposes, like the exchange of MP3 files introduced by Napster [1] and evolved by its successors (e.g. Gnutella [2]). The mechanisms these applications are based on work fine for file sharing due to the elastic traffic characteristics. This is not the case for on-demand video streaming where data has to arrive at a certain point in time to guarantee a pleasant perception of the streamed content. Realizing video streaming in environments where senders and receivers are mainly connected via asymmetric links (e.g. ADSL or cable modems) becomes even harder due to the bandwidth constraints of the up-link. Another limiting factor is the limited resources at the peers and the quite large storage space requirements of the video objects. Here, we propose a new technique in P2P networks that allows video streaming that makes use of layer encoded video to overcome the constraints of asymmetric access networks and available resources at the peers. We will show that the proposed technique will also increase the fault-tolerance of such a P2P streaming system.

2. Assumptions

One significant characteristic of asymmetric access networks is the different amount of bandwidth that is available on the up- and down-link. For example, let us assume an ADSL service that is usually offered with an up-link bandwidth of 128 kBit/s and the a down-link of 768 kBit/s[1]. Given a scenario as shown in Figure 1, a video could only be streamed from sender *Peer 1* to the receiver *Peer 4* if its bandwidth requirement is less or equal to 128 kBit/s. It is actually possible to stream video with such a low bit rate but only at a very poor quality. E.g. MPEG-4 allows bit rates between 5 kBit/s and 1 GBit/s. However, the down-link of *Peer 4* allows the reception of video having a bandwidth that is 6 times higher and, thus, the video could be perceived in a much better quality. Hierarchically layer encoded video is an encoding scheme where the video is split into a base layer and several enhancement layers. Such an encoding scheme allows to adapt to the available bandwidth by dropping or adding enhancement layers. Nevertheless, in the given example for ADSL only the base layer might be streamed from *Peer 1* to *Peer 4* due to the constrained up-link bandwidth of *Peer 1*. To circumvent this problem, each layer of the video could be streamed from a different peer (as shown in Figure 1) allowing a much better quality at the receiver than in the case where only one peer is used as sender.

To realize such a distributed video streaming applications the following problems have to be solved:

- Sender-to-layer assignment
- Ordering at the receiver
- Transport
- Layer segmentation

1. Typical ADSL service offered by German Telekom

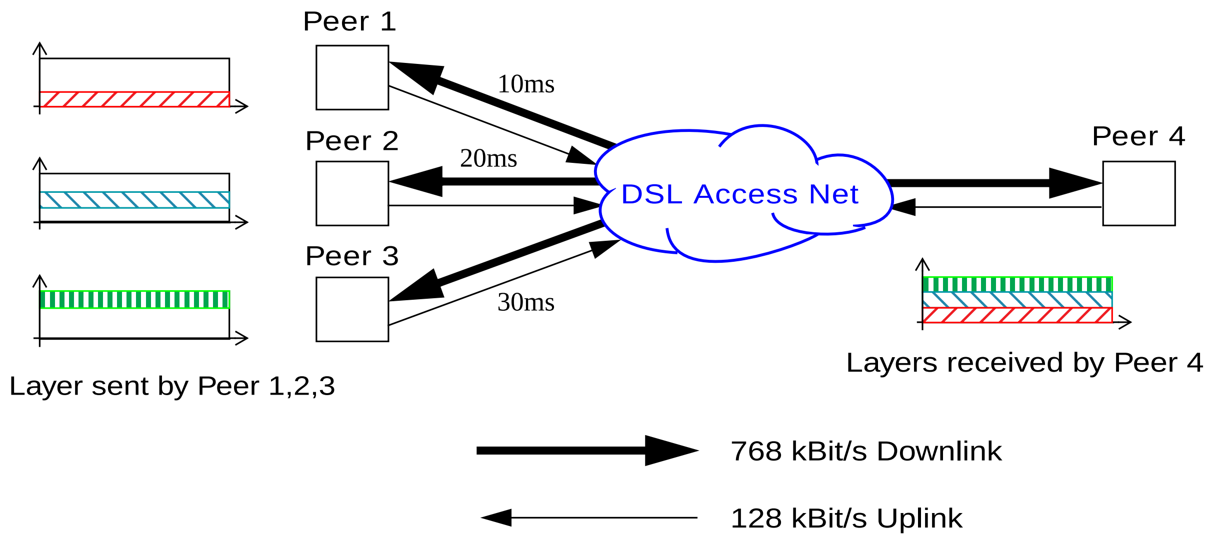

Figure 1: P2P streaming of layer encoded video via asymmetric access links

3. Sender-to-Layer assignment

A drawback of hierarchically layer encoded video is the fact that the base layer is always needed in order to be able to render the complete frame. Enhancement layers of level n that are received without the underlying layers (base layer or enhancement layer n-1) are worthless. Thus, the application should make use of a layer allocation algorithm that decides which layers should be streamed from which peer. Such an algorithm should meet the following requirements:

- The sender should be chosen for the streaming of the base layer which connectivity characteristics are the best among all the possible senders and the receiver. Criteria for the connectivity quality might, e.g., be the delay between sender and receiver or the amount of routers on that link up to the sender. For the latter case, a route with less hops should always be preferred for the transmission of the base layer because it can be assumed that a shorter distance between sender and receiver will lead to less packet losses. Delay, on the other hand, is important due to the rendering constraints at the receiver. Data of a video frame have to be at the receiver at a certain point in time to allow its rendering. Choosing the sender with the smallest delay for the base layer transmission can assure that at least those data of a frame will arrive at the receiver in time and thus allow a rendering of the frame even though in the lowest possible quality.
- The possible senders should be sorted by this criteria and the resulting list determines which layer is streamed by which peer. E.g., in Figure 1 *Peer 1* is ranked first, followed by *Peer 2* and *Peer 3*, respectively. Thus, the base layer will be streamed from *Peer 1*, enhancement layer 1 from *Peer 2*, and enhancement layer 2 from *Peer 3*.
- Running this algorithm also during the streaming phase can increase the fault-tolerance of the system. Thus, the receiver must monitor statistics about the quality of each single stream as it would, e.g., be possible based on the RTCP [3] information. This would allow the receiver to dynamically

change the sender-to-layer assignment, and allow a non disruptive service in the case of a sender failure. E.g., if we assume that *Peer 2* also locally stores the base layer, the receiver could signal to *Peer 2* to stream the base layer instead of enhancement layer 1, in the case that *Peer 1* or the connection to *Peer 1* fails. The dynamic behavior of the sender-to-layer assignment is important with respect to the fact that the senders are rather unreliable and might become off-line at any point in time.

4. Ordering at the receiver

Although the algorithm described above does its best to ensure that at least a certain quality at the receiver is provided, this quality can be increased if a buffer is used at the receiver. If we assume that the different layers are streamed from the senders as described above, data from Peer 2 and Peer 3 always arrive later than data from Peer 1 caused by the different delay on each link. Thus, a buffer would allow to synchronize the streams and allow the rendering of all layers of a frame. Without such a buffer, only the base layer of each frame could be rendered because of the fact that data from the enhancement layers would arrive too late. In addition, such a buffer can be used to reduce impairments that are caused by delay jitter on the single links. If the receiver has enough memory, the buffer size can be chosen large enough in order to allow the retransmission of packets that were lost, e.g., due an overflowing queue at one of the intermediate routers.

5. Transport

Video data is usually streamed via RTP/UDP, and the application-level framing (ALF) is defined by RTP profiles. That means, those profiles define how video data is packetized into RTP packets. All existing profiles assume that a video is always transported completely (e.g., all layers of a layer encoded video) from one sender to a receiver. This is not the case in the described scenario for P2P streaming. In this case, every layer will be transported as a single RTP/UDP stream. Thus, the receiver needs additional information in order to identify the layer to which the data transported in the arriving RTP packet belongs to. Additional information is needed to allow the receiver to identify the layer of incoming RTP packets. In [4] we propose such an extension for the transport of layer encoded video in RTP.

If only data that belongs only to a single frame should be transported in an RTP packet depends on the size of the data. In case the data units of the application become too small several application packets should be packetized in a single RTP packet to be more efficient, as it is done in the case of audio streaming.

6. Layer segmentation

In the case of MPEG-4 the two profiles that are defined for layer encoded video so far are simple scalable and fine grained scalability (FGS), respectively. Both consist of a base and an enhancement layer. While in the first case the rate of the enhancement layer is fixed, the rate for the FGS enhancement layer can vary. Yet, both cases are not well suited for the case of P2P streaming in combination with asymmetric access networks. E.g. in the case of an ADSL network with the characteristics as described above, ideally, a layer encoded video would consist of 6 layers with a bit rate of 128 kBit/s for each layer. With the two profiles offered by MPEG-4 this can not be realized because the layer encoded video could only be sent from a maximum of two senders and, thus, a rate of 156 kBit/s could not be exceeded. A better suited layered encoding scheme is, e.g., the one presented in [5] where the encoded video can consist of several layers.

7. Summary

Using hierarchically layer encoded video in P2P application where receivers are mostly connected via asymmetric access networks is a viable solution to allow streaming. The major challenge here is to coordinate the streams from different senders in order to obtain the best possible quality of the video at the receiver. This issue of distributed streaming has been investigated by [6] and [7] but not within the scope of layer encoded video and asymmetric access networks.

8. References

[1] Napster. http://www.napster.com.

[2] Gnutella Protocol. http://www.clip2.com/GnutellaProtocol04.pdf.

[3] H. Schulzrinne, S. L. Casner, R. Frederick, and V. Jacobson. RFC 1889 - RTP: A Transport Protocol for Real-Time Applications. Standards Track RFC, January 1996.

[4] M. Zink, C. Griwodz, J. Schmitt, and R. Steinmetz. Scalable TCP-friendly Video Distribution for Heterogeneous Clients. In *Proceedings of SPIE/ACM Conference on Multimedia Computing and Networking (MMCN), Santa Clara, USA*. SPIE, January 2003.

[5] J.-Y. Lee, T.-H. Kim, and S.-J. Ko. Motion Prediction Based on Temporal Layering for Layered Video Coding. In *Proceedings ITC-CSCC'98*, pages 245–248, July 1998.

[6] J. Apostolopoulos, T. Wong, S. Wee, and D. Tan. On multiple description streaming with content delivery networks. In *Proceedings of the 21th Annual Joint Conference of the IEEE Computer and Communications Societies (INFOCOM'02)*, pages 1736–1745, June 2002.

[7] T. Nguyen and A. Zakhor. Distributed video streaming with forward error correction. In *11th International Packet Video Workshop (PV2002), Pittsburgh, PA, USA*, April 2002.