

A Novel Framework for Improving Bandwidth Utilization for VBR Video Delivery over Wide-Area Networks

Junli Yuan*, Sujoy Roy, Qibin Sun
Institute for Infocomm Research (I²R), 21 Heng Mui Keng Terrace, Singapore 119613

ABSTRACT

There are great challenges in streaming variable-bit-rate video over wide-area networks due to the significant variation of network conditions. The utilization of the precious bandwidth of wide-area networks is often low in such streaming systems. In this paper, we propose a novel framework to improve the bandwidth utilization from a new perspective. Instead of focusing on the performance of each single media stream, we aim to improve the overall bandwidth utilization for video streaming systems. We try to exploit the unoccupied bandwidth in ongoing streams and using it to deliver some prefetched data which can be used to facilitate future streaming. Preliminary results show that our mechanism has great potential to improve both the overall bandwidth utilization and the caching performance of the proxy servers in the streaming systems.

Keywords: VBR video, streaming, delivery, bandwidth utilization, prefetching

1. INTRODUCTION

Streaming video over Wide Area Networks (WAN) is gaining popularity rapidly in recently years. The underlying network environments for streaming video over WAN are typically heterogeneous networks which typically consist of many Local Area Networks (LAN) and a WAN such as the Internet (because the Internet is the most widely used WAN, so we will use the term “WAN” and “Internet” interchangeably in this paper). The backbone of the WAN is usually shared by a large number of users, and the bandwidth of the WAN is generally considered as a critical and costly resource. Therefore, improving the overall bandwidth utilization of the WAN backbone is an important objective in the design of streaming video delivery systems.

A WAN, e.g. the Internet, is typically a best-effort network which does not provide a fixed quality of service, neither in terms of bandwidth nor of latency [1]. To effectively delivery time-critical streaming video content over such networks is a very challenging problem. Variable-bit-rate (VBR) encoded video content further complicates the delivery problem because VBR video exhibits significant burstiness [2] [3]. To efficiently deliver the bursty VBR video content, researchers have proposed several techniques. The most important and intensively studied mechanism are the bandwidth smoothing techniques [4] [5], which have been shown to be able to effectively reduce the burstiness of the traffic caused by VBR video. However, smoothing techniques are more for the networks with guaranteed quality of service such as the bandwidth and latency. They are not suitable for a WAN environment like the Internet. This is because smoothing techniques require guaranteed network bandwidth and real-time monitoring of the fill state and restrictions of the client buffer, which are difficult to achieve in a best-effort network. Another technique, called “video staging”, also tries to improve the effectiveness of VBR video delivery over WAN [6]. This mechanism partitions the data of a VBR video along the rate axis using certain cut-off rate, prefetches the upper part of the data and caches them on the proxy server. Later on, when a user requests a video, only the lower part of the data needs to be delivered from the origin server. It has been shown that video staging can reduce the backbone WAN bandwidth requirement considerably. However, this mechanism uses dedicated network connection and bandwidth to deliver the prefetched data, which does not improve the overall bandwidth utilization of video streaming systems.

In this paper, we present a new framework for improving the overall bandwidth utilization of video streaming systems over WAN/Internet. Since the WAN/Internet is only a best-effort network, it is very difficult (if not impossible) to provide guaranteed quality of service such as fixed bandwidth and latency. So, instead of focusing on the performance of

* junli@i2r.a-star.edu.sg; phone +65 6874-8810; fax +65 6774-4998; www.i2r.a-star.edu.sg

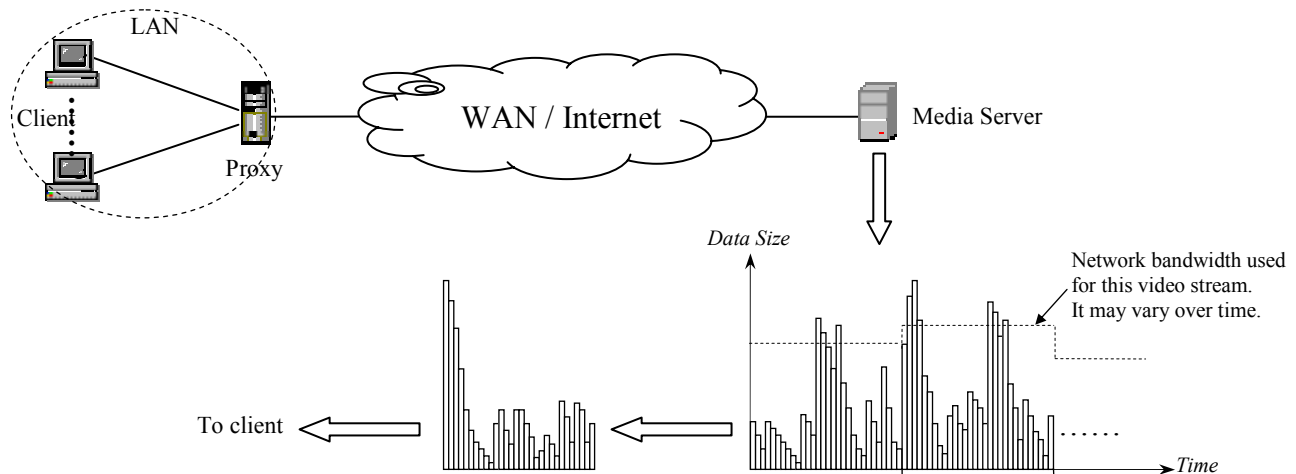


Fig. 1. System structure for variable-bit-rate video delivery over wide-area networks

each single media stream, we try to improve the performance of streaming systems from another perspective, that is, to improve the overall bandwidth utilization of the streaming systems. The WAN/Internet bandwidth is generally considered as an expensive resource. If we can improve its utilization, it should also be regarded as a positive contribution to the design of streaming systems since it can help the systems to remain cost-effective. In our proposed framework, we try to make use of the “spare bandwidth” in the ongoing streams to prefetch some useful data for future use. This way, the overall bandwidth utilization can be effectively improved. Although it seems intuitively simple to conceive, to our knowledge this idea has not been looked at in the existing literature.

2. NEW FRAMEWORK: IMPROVING BANDWIDTH UTILIZATION USING PIGGYBACK PREFETCHING

In our study, we assume the presence of proxy servers in the streaming system. In video streaming systems over WAN, the use of proxy servers is a common way to improve the delivery performance over the underlying best-effort network. Proxy servers are also widely used in the multimedia content delivery networks (CDN) [7]. Therefore, in our study, we also assume the usage of proxy servers, which results in the client-proxy-server structure, as shown in the upper part of Fig. 1. We also assume that proxy servers have huge storage space which is capable of holding immense amount of data.

Since there is no way to provide guaranteed bandwidth and latency in a best-effort network, streaming systems over the WAN/Internet would just try to deliver video content to the best of its effort. In such an effort, streaming servers usually would just allocate the most appropriate amount of bandwidth according to its knowledge about the current network condition, and use that bandwidth to stream the video data that are needed by the user for the playback (see the lower part of Fig. 1). Note that the bandwidth allocated may vary over time (also indicated in Fig. 1). We noticed that the bandwidth allocated by streaming servers in their endeavor to fight against the fluctuating network condition is often in an over-provisioned state, although there may be situations where the bandwidth is under-provisioned. Since the WAN/Internet bandwidth is a precious resource, such over-provisioning is an undesirable waste.

We propose a new framework, called *piggyback prefetching*, to improve the utilization of the precious WAN/Internet bandwidth by making the best use of the often over-provisioned bandwidth in the ongoing streams. In streaming systems, the server usually has the a priori knowledge about the bit rates of the videos, and it can have certain mechanism to estimate the bandwidth [8]. When the server allocates certain bandwidth to deliver a segment of video data, it can compute whether and how much bandwidth will be unoccupied. Our piggyback prefetching mechanism tries to make use of this “spare space” in the allocated bandwidth by having the server to deliver some prefetched data using this unoccupied bandwidth, as shown in Fig. 2 (the “block” shown in this graph may be read as the volume of the data that can be transmitted by the server during the timeslot when the system is switched to process this video stream). From

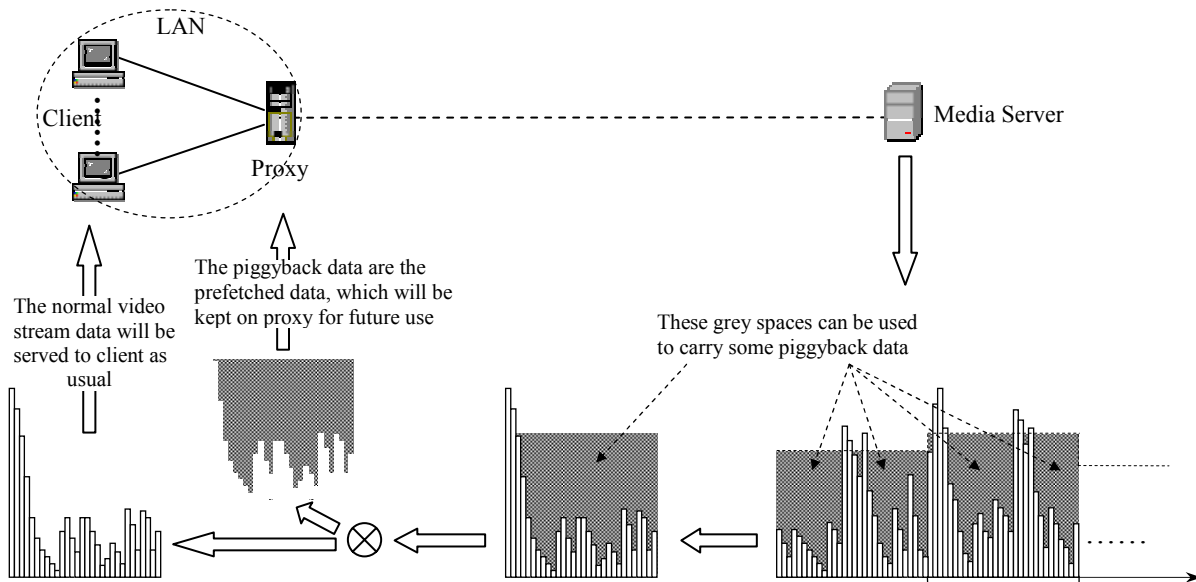


Fig. 2. The operation of piggyback prefetch for variable-bit-rate video delivery

another point view, these prefetched data piggyback onto the normal video data stream and are delivered to users simultaneously. As such, the overall utilization of the WAN/Internet bandwidth can be improved effectively.

Whenever the streaming server puts piggyback data together with the normal video data, it also puts special headers describing the details about the data, such as the object ID, offset and size of the data etc. The size of such descriptive headers is very trivial compared with the transmitted video data. When the data arrive at a proxy server, the proxy server decouples the piggyback data from the normal video data according to the header information. Then the proxy server passes the normal video data to users as usual, while keeps the prefetched data for future use. The prefetched data are supposed to be some useful data which can be used to facilitate future requests. This way, not only the unoccupied bandwidth can now be utilized positively, but the performance of proxy caching can also be improved due to the usefulness of the prefetched data.

It may be argued that streaming servers can use the unoccupied bandwidth to deliver the to-be-viewed portions of the current video as fast as possible, and in that way the bandwidth can be fully utilized. But this may not always be possible due to the limited buffer size at client side. In some cases, the unoccupied bandwidth could be quite big. If the server uses it to deliver to-be-viewed data, the amount of such data could be too big to be fitted into client's buffer. Furthermore, using the unoccupied bandwidth to deliver to-be-viewed data is not reliable. Due to the variation of the network condition, some to-be-viewed data may not be able to reach clients in time for playback, which is undesirable. Our proposed mechanism does not have these problems because we use the unoccupied bandwidth to deliver prefetched data and introduce the help of proxy servers. Since the data delivered using the unoccupied bandwidth are just prefetched data, they are not critical to users' current playback, therefore, it will not cause any problem to users' playback if they are delayed or even lost. Furthermore, the prefetched data in our mechanism do not need to be stored in clients' buffers. They will be stored in proxy servers, where huge storage space can be utilized.

3. VARIOUS APPLICATIONS OF THE NEW FRAMEWROK

The new piggyback prefetching framework can be applied in the streaming system in various ways according to the type of the prefetched data that are delivered using the unoccupied bandwidth. Theoretically, the unoccupied bandwidth can be used to deliver any type of prefetched data that can bring positive impact on the performance of streaming systems. Here we briefly examine four types of applications of our framework.

Application 1) Prefetching the later portions of the currently-viewed video object

In this application, we use the unoccupied bandwidth to deliver the later portions of the currently-viewed video object. This will result in smoother playback for the later portions of the video.

Actually, this application is similar to the idea discussed in the last paragraph of Section 2. However, our mechanism does not have the problems discussed in that paragraph. For the buffer overflow at client side, this problem does not exist in our mechanism because we store the prefetched data in the proxy server, not client buffers. As for the possible impact that packet delay and packet loss may have on users' playback, we suggest that the later portions to be prefetched should be far away enough from the currently viewed portion. So, the system will have enough time to wait for the delayed data or re-prefetch the lost data.

Application 2) Prefetching the data above certain cut-off rate line for the currently-not-viewed video objects

In this application, we partition the data of the currently-not-viewed video objects along the rate axis using certain cut-off rate, and deliver data above the cut-off rate line using the unoccupied bandwidth. Again, such prefetched data will also be stored on proxy servers for future use.

This application is very like the "video staging" mechanism, except that our mechanism "steals" some bandwidth from the ongoing stream to deliver the prefetched data while "video staging" uses dedicated network connection and bandwidth to deliver the prefetched data. So, our scheme not only enjoys the same benefits as video staging, it can also improve the utilization of the precious network bandwidth.

With proper arrangement, this application can also be extended to work for the currently-viewed objects, i.e. to prefetch the data above the cut-off rate line for the currently-viewed objects.

Application 3) Prefetching the whole body of the currently-not-viewed video objects

This application tries to prefetch the whole body of the currently-not-viewed video objects. If the whole body of a video object can be prefetched successfully, the resulting effect will be very attractive: smooth playback for the entire video object. However, because of the huge size of media objects, this scheme faces some limitation, e.g. the proxy storage may not be able to hold many of full media objects.

Application 4) Prefetching the initial segments of currently-not-viewed video objects

Many research works revealed that the initial segments of video objects are more important than other segments because the initial segments are critical to the startup delay perceived by users. Therefore, we may apply our piggyback prefetching framework to prefetch the initial segments and delivery them using the unoccupied bandwidth. This application will result in reduced startup delay for the video objects whose initial segments are prefetched.

From the above four applications, we see that our framework is applicable to both single media streams and multiple media objects. In contrast, most existing techniques in video smoothing and video staging are only applicable to single media streams.

4. EXPERIMENTAL RESULTS

We have done some preliminary experiments to evaluate our proposed mechanism. The data set we used is some MPEG-1 video files from TRECVID [9] (the statistical characteristics of this data set should also be applicable to most other formats of VBR-encoded videos). Statistics about those videos show that the data rates do change dramatically over time. The peak data rate can be 40 times bigger than the lowest data rate. Given such great variation of the video data rates, we expect that a considerable amount of video streaming bandwidth will be unoccupied.

As our proposed piggyback prefetching mechanism tries to deliver some prefetched data using the unoccupied bandwidth, we would like to examine how much data can be delivered using our mechanism. Assume a 1-hour VBR video is being streamed, Fig. 3 gives the amount of data (in terms of video playback time) can be prefetched under different bandwidth allocated. We see that the amount of data can be prefetched is substantial. For example, when the bandwidth allocated is 50% of the peak data rate, as much as 33.7 minutes of video data can be prefetched using the

unoccupied bandwidth. No matter this significant amount of data is exploited for which of the 4 applications described in Section 3, a high performance gain to the WAN-based streaming systems would be expected.

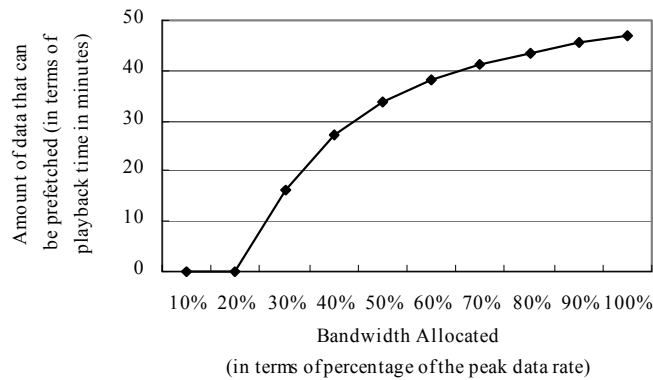


Fig. 3. Data amount that can be prefetched under different allocated bandwidths

5. CONCLUSIONS

In this paper, we propose a novel framework for improving the overall utilization of the precious network bandwidth for VBR streaming systems. The mechanism tries to exploit the unoccupied bandwidth in ongoing streams and using it to deliver some prefetched data which can be used to facilitate future streaming. Preliminary experimental results show that our mechanism has great potential to improve both the overall bandwidth utilization and the caching performance of the proxy servers for the WAN-based streaming systems.

REFERENCES

1. Jim W. Roberts, "Traffic theory and the Internet", *IEEE Communications*, January 2001.
2. T. V. Lakshman, A. Ortega, and A. R. Reibman, "Variable bit-rate video: Tradeoffs and potentials", In the *Proceedings of the IEEE*, vol.86, May 1998.
3. M. Garrett and W. Willinger, "Analysis, modeling and generation of self-similar VBR video traffic", In the *Proceedings of ACM SIGCOMM'94*, pp.269-280, London, England, Aug. 1994.
4. Wu-Chi Feng and Jennifer Rexford, "A comparison of bandwidth smoothing techniques for the transmission of prerecorded compressed video", In the *Proceedings of IEEE INFOCOM'97*, vol. 1, Kobe, Japan, Apr. 1997.
5. Subhabrata Sen, Jennifer Rexford, Jayanta Dey, James F. Kurose, and Don Towsley, "Online smoothing of variable-bit-rate streaming video", *IEEE Transactions on Multimedia*, pp. 37--48, March 2000.
6. Zhi-Li Zhang, Yuewei Wang, David H.C. Du, and Dongli Su, "Video Staging: A proxy-server-based approach to end-to-end video delivery over wide-area networks", *IEEE/ACM Transactions on Networking*, 8(4):429--442, August 2000.
7. Jian Ni and Danny H.K.Tsang, "Large-Scale Cooperative Caching and Application-Level Multicast in Multimedia Content Delivery Networks", *IEEE Communications Magazine*, May 2005.
8. Jacob Strauss, Dina Katabi, and Frans Kaashoek, "A Measurement Study of Available Bandwidth Estimation Tools", In the *Proceedings of the ACM SIGCOMM Internet Measurement Conference '03*, Miami, Florida, October 2003.
9. TRECVID home page, <http://www.itl.nist.gov/iaui/894.02/projects/trecvid>