A PEER-TO-PEER VIDEO-ON-DEMAND SYSTEM USING MULTIPLE DESCRIPTION CODING AND SERVER DIVERSITY

Xiaofeng Xu, Yao Wang, Shivendra S. Panwar, Keith W. Ross

Polytechnic University, Brooklyn, NY

ABSTRACT

We propose a video streaming system that uses many ordinary computers (peers) as servers, which is less costly and more scalable than using a single or a few dedicated servers. . To circumvent frequent peer going-downs and limited peer up-link bandwidth, each video is coded into multiple descriptions, which are distributed over multiple peers. The system serves a client request by streaming multiple descriptions of the requested video from separate peers. Using the MD-FEC coding scheme, and simple models for peer going-down, coming-back and for searching replacement serving peers after a peer going-down, we simulated video streaming in such a system. Our results show significant benefits in using large number of descriptions and serving peers.

1. OVERVIEW OF THE PROPOSED SYSTEM ARCHITECTURE

In a conventional video –on demand system, compressed videos are stored in one or several servers. When a client requests a video, a server that has the corresponding file is found, and this server starts to send the video to the client. In the streaming mode, the client starts to play the video after a certain play-out time. One problem with this architecture lies in the tremendous cost for setting up the servers, which must have very large storage space, ultra-fast bandwidth, and high reliability. To alleviate the above problem, we propose to use many ordinary computers as servers and store the video content over these computers. In the context of peer-to-peer (P2P) networking, these computers are called peers or nodes. We assume that the service provider set up a contract with participating nodes so that the participating nodes, in return, will be paid in the form of certain incentives, such as reduced cost when requesting a video. Each node has a certain storage space and uplink bandwidth reserved for participating as a serving node. Currently we assume all participating nodes can function both as servers and clients. That is, each node can request video from others, and also serve others. But this can be generalized to have only some nodes designated as possible servers. Compared to the central server approach, the proposed system architecture requires very low initial set-up cost and can be more scalable and reliable.

To circumvent the frequent peer going-down events and limited uplink bandwidths at the peers, we propose to use multiple description coding and distribute different descriptions of the same video over separate peers. The system serves a client request by streaming multiple descriptions of the requested video from separate peers. When a serving node goes down in the middle of a streaming session, the system will look for a replacement peer that has the same video description and sufficient uplink bandwidth.

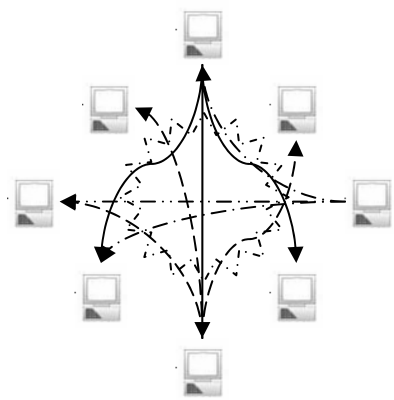

Figure 1. Proposed System Architecture.

The benefit from coding a video into multiple descriptions and distributing these over separate nodes are several folds: 1) When a serving peer goes down, it only causes temporary loss of a single description, which has limited impact on receiving video quality. 2) Each description has a rate much lower than the total rate of a video, thus reducing the required uplink bandwidth at peers. 3) Splitting a video file across multiple nodes helps to reduce the load on each serving node, which is an important design factor in a P2P application. 4) Finally, from the service provider point of view, it is desirable not to store any video entirely in one node, to prevent the node to have illegal access to this video.

Recently, several video multicast systems making use of peer coordination have been studied [1-5]. But in these systems, the video content is stored in a central server, the peers merely help to relay a video originated from the server, by forming one or several multicast trees, and peers only help each other when they watch the same video. These systems also assume each node has enough uplink bandwidth to support the transmission of an entire video. However, the measurement study reported in [6] reveals that the most popular forms of Internet access for peers are cable modem and DSLs. Both of them are asymmetric in the uplink and downlink bandwidths. The uplink bandwidth limits one node to serve only part of a video. Also, the proposed system does not require the construction of multicast trees. Each node stores parts of several video files. Every node uses parallel downloads to take advantage of the lateral bandwidth among peers.

The remainder of this paper is organized as follows. Section 2 describes our network models and FEC-based multiple description source coding (MD-FEC) scheme. Section 3 presents our simulations results and investigates the impact of description number and network parameters on video quality. Section 4 concludes the paper and suggests directions for future work.

2. SYSTEM MODELS AND ASSUMPTIONS

2.1 Network Model

We assume that there are a total of N homogeneous nodes (peers), each with storage capacity C, uplink bandwidth R_u, downlink bandwidth R_d. Note that C and R_u represent the storage space and uplink bandwidth reserved for the peer-to-peer video streaming service by each node. The total storage space and uplink bandwidth of each node can be much larger than these numbers. We further assume a "STAR" network topology. As shown in Figure 1, each node is connected to a backbone network that is assumed to have infinite bandwidth and no congestion. The communication channel (in terms of delay, bandwidth, loss) between two nodes is only limited by the uplink and downlink (bandwidth and availability) of the two nodes to/from the backbone, no matter what is the distance between two end nodes.

2.2 Node ON/OFF models

We assume that each participating peer alternating between "on" and "off" status. The ON time is how long a node would stay in the network as a participating node.[1] We model it as an exponentially distributed random variable $T_{on} \sim \lambda e^{-\lambda t}$. Similarly, the OFF time is the time period when a node could not provide the streaming service to others. We assign it another exponentially distributed random variable $T_{off} \sim \beta e^{-\beta t}$. For every node, we draw a random number T_{on} based on the probability density function (PDF) $\lambda e^{-\lambda t}$. After time T_{on}, the node is turned down. Another random number T_{off} is chosen from the exponential distribution with parameter β. Repeating the operation, we generate a sequence of $T_{on}(1)$, $T_{off}(1)$, $T_{on}(2)$, $T_{off}(2)$…….This sequence represents the time fragments in which the same node is on or off.

2.3 Modeling of video request and serving node selection

The number of new requests within a given time interval (e.g., 1 second) is modeled by the Poisson arrival process.

When a node requests a video, we assume that a central manager will try to find M serving nodes that have the M descriptions of the video and each node also has sufficient surplus uplink bandwidth to serve one description. The central manager chooses M serving nodes based on the two criteria: (1) the chosen nodes are those who have the maximum uplink bandwidth among all ON nodes in the network; (2) Each node could serve only one description.[2] Since we enforce that a node must download different descriptions from M distinct nodes, the entire service load is evenly distributed among all nodes. Also, the going-down of one chosen serving node in the middle of the streaming session will affect only one description.

2.4 Modeling of description loss rate

We assume that the uplink and downlink of each node have negligible bit error rate and packet loss rate, based on the STAR network model. Furthermore, we assume a node will request a streaming video, only if it has sufficient unused downlink bandwidth, and a node is chosen for serving a description only if it has sufficient surplus uplink bandwidth. Under these assumptions, there will be no loss incurred by congestion at the serving node or receiving node. Once a node has been chosen to deliver a description upon the start of a streaming session, the packets in this description are delivered error free, until the node becomes unavailable (because it is disconnected from the network, or because it has to run other higher priority tasks). The node going-down rate is λ according to section 2.2. Once a node becomes unavailable, the server will try to find a replacement node that stores the same description and has sufficient surplus uplink bandwidth. Assuming the time to find a replacement node is an exponential random variable, with mean $1/\gamma$. For a given substream, the receiving node will see alternating periods of receiving the stream and receiving nothing. We can model the above process as a two state (receiving vs. non-receiving) Markov process. The transition probability from receiving to non-receiving is λ; the transition probability from non-receiving to receiving is γ. The steady state probability in the non-receiving state is $\varepsilon = \dfrac{\gamma}{\gamma + \lambda}$, which is the probability that a description is lost at any time instant.

Recall that we assume the M descriptions of the same video is stored in M different nodes. Assuming the going-down and finding replacement events at different nodes are independent, the probability that receiving m out of M descriptions at any time is:

$$P(m, M) = \binom{M}{m} (1-\varepsilon)^m \varepsilon^{M-m} \qquad (1)$$

2.5 The MD-FEC video coder

One simple way of generating M equally important and equal rate descriptions is by combining a scalable coder and forward error correction (FEC) codes [7]. In the most general case of this coder, in each time interval T_{seg} (usually a group of frames or GOF), one partitions all the coded bits for this interval (R bits) into M groups with the <u>m</u>-th group ending at R_m bits (m = 1,2, ……, M). Then each group of R_m-R_{m-1} bits is further divided into m sub-groups of equal length. Next a Reed-Solomn (M,m) channel code is applied across the m subgroups, to generate M sub-groups. Description m for this time interval is formed by

[1] A node may be on, but is busy with other higher priority tasks and hence not available as a server.

[2] Unless less M nodes are available who have sufficient surplus bandwidth, in which case some chosen nodes may serve more than one description.

concatenating subgroups *m* from all *M* groups and is put into a separate packet. This is illustrated in Figure 2.

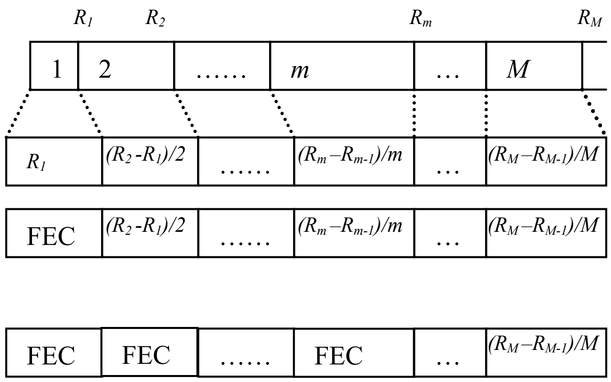

Figure 2 MD-FEC with *M* descriptions

With the above FEC and packetization scheme, receiving *m* descriptions will enable correct decoding of up to R_m bits, with distortion $D_s(R_m)$. Here we use $D_s(R)$ to represent the operational distortion-rate function of the underlying scalable coder. The total rate after using FEC depends on the rate partitions ($R_1, R_2, \ldots, R_M$).To obtain the optimal rate partitions R_m, we can minimize the expected distortion

$$D(R) = \sum_{m=0}^{M} p(m,M) D_s(R_m)$$

subject to the rate constraint:

$$M \sum_{m=1}^{M} (R_m - R_{m-1})/m = R_t$$

In our implementation, we obtain $p(m,M)$ either based on the model in (1) or from network simulations. Given $p(m,M)$ and $D_s(R)$, we use the algorithm in [7] to solve the above constrained multivariable optimization problem.

3. SIMULATION RESULTS

We simulated the proposed peer-to-peer video streaming system, using the node ON/OFF model and the replacement time model, described in Sec. 2. At present, we have not explored how to distribute descriptions from multiple video sources with different popularity over all participating nodes. The present simulation considers one video source only and every node stores all descriptions of this video. We varied the number of descriptions and the mean replacement time, to see the impact of these parameters on receiving video quality.

3.1 Simulation setting

(a) Video Data

We coded the "Foreman" video sequence in CIF (352x288) resolution into a scalable bit stream using the MPEG-4 FGS codec[8], at a base layer rate of 150 Kbps. Each GOF has the duration of T_{seg}=1 second and comprises of 15 frames. The output bits from each GOF are converted to *M* descriptions using the MD-FEC method, where M is varied from 4 to 64. The total rate of a video after MD-FEC is constrained to be 768 Kbps. Thus the rate of each description is $\frac{768}{M}$.

(b) Network settings

Our simulated network has *N*=500 nodes. The uplink bandwidth at each node is R_u=250 Kbps. The average on/off time ($1/\lambda$ or $1/\beta$) of a node is 20/5 minutes. The arrival rate of new requests is 1 per second. The replacement time ($1/\gamma$) is varied from 10 seconds to 1 minute. These parameters are chosen so that the network is not overloaded in the sense that M distinct serving nodes can be typically found for each new request. The total simulation time is 1800 minutes. We assume the length of each streaming session is 20 minutes. Although the length of a typical video file in video-on-demand applications can be much longer, we choose a shorter time, so that within our network simulation time, more streaming sessions can be invoked.

3.1. Impact of the description number

We first examine how the video quality vary with the description numbers in a fixed network setting, with the average on/off time set to 20 and 5 minutes, respectively, and the average replacement time 1 minute. Figure 3 shows the average PSNR for all nodes as a function of time for *M*= 4, 8, 16, 32, 64, respectively. From this figure, we can observe that the PSNR improves as the number of description increases, but the improvement gradually saturates as M becomes very large (beyond 32).

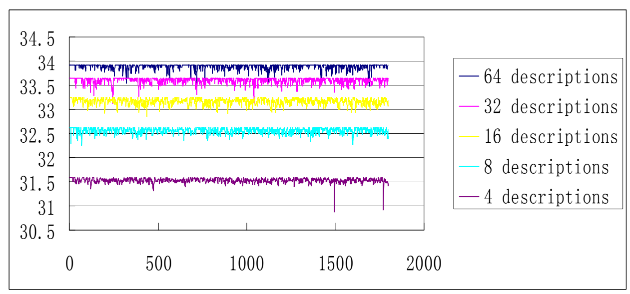

Figure 3 Average PSNR across nodes versus Time. Replacement time=60 seconds

One reason that increasing the description number is always better is that the MD-FEC redundancy is adaptively determined based on the description loss rates determined from the network simulation. When *M* is large, MD-FEC can apply unequal error protection to each GOF more effectively, thus requiring lower total redundancy. For example, when *M*=4 four levels of redundancy (redundancy= R_t / R_s) are possible (4, 3, 2, 1) and the possible redundancy is only the mixture of these four. But when *M*=8, eight levels of redundancy (8, 7, 6, 5, 4, 3, 2, 1) are possible. The total allocated redundancy thus reduces as M increases, for a fixed network setting. Figure 4 shows the total redundancy determined by MD-FEC for different values of *M*.

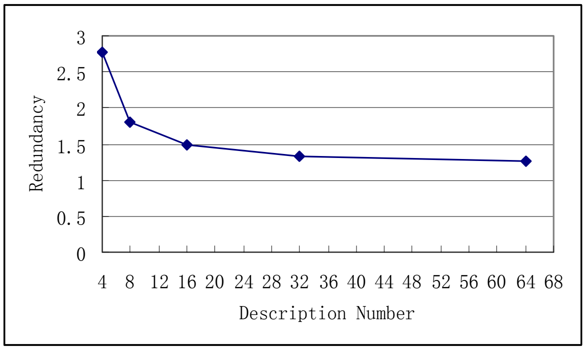

Figure 4 Redundancy vs. Description Number

3.2. Impact of Replacement Interval

The results so far are for a fixed replacement time (1 minute). Figure 5 examines the impact of replacement time on video quality, where the PSNR is averaged over all nodes over the entire simulation time. We can see that the average PSNR increases as the replacement time becomes shorter. This is as expected, as the replacement time is the time interval a lost description due to a node going-down stays unavailable.

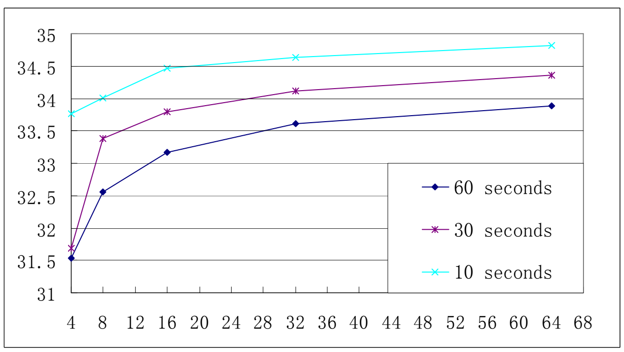

Figure 5 Average PSNR vs. Description Number vs. Replacement Time (for optimal video coding parameter)

4. CONCLUSIONS AND FUTURE WORK

We described a video-on-demand system using peers as servers. By using multiple description coding and streaming different descriptions of a requested video from separate peers, the system is resilient to data losses due to frequent and unpredictable peer going-downs. We have investigated the impact of the description number and the replacement interval on the receiving video quality. We have found that increasing the description number can improve the system performance significantly due to greater path diversity and flexibility in redundancy allocation by MD-FEC.

The present network simulations are based on simple models about peer behaviors (going-down and coming-back). Also, we have only simulated the network in under-loaded situations. In overloaded situations, an admission policy that governs the minimal number of available descriptions necessary before starting a session, and the associated blocking probability needs to be studied. Furthermore, one needs to study, after a new node becomes "ON", how to distribute its uplink bandwidth evenly among requesting nodes that are still searching for replacement nodes. Finally, a real VOD system must accommodate many video files with varying popularities. The video placement problem, which deals with how to distribute the descriptions from different videos among peers, is still an open question.

5. REFERENCES

[1] Y. Chu, S. Rao, S. Seshan and H. Zhang, "A Case for End System Multicast", *IEEE Journal on Selected Areas in Communication (JSAC), Special Issue on Networking Support for Multicast*, To Appear.

[2] H. Deshpande, M. Bawa, and H. Molina. "Streaming Live Media over a Peer-to-Peer Network", *Stanford Database Group Technical Report (2001-20)*, Aug. 2001

[3] V. Padmanabhan, H. Wang, P. Chou, and K. Sripanidkulchai, "Distributing Streaming Media Content Using Cooperative Networking", *NOSSDAV '02*.

[4] D. A. Tran, K. A. Hua, and T. T. Do. "ZIGZAG: An Efficient Peer-to-Peer Scheme for Media Streaming". In *Proceedings of IEEE INFOCOM 2003*, March 30-April 3 2003, San Francisco, CA, USA.

[5] Y. Guo, K. Suh, J. Kurose, D. Towsley "P2Cast: Peer-to-peer Patching Scheme for VoD Service" *Proceedings of the 12th World Wide Web Conference (WWW-03)*, Budapest, Hungary, May 2003

[6] S. Saroiu, P. Gummadi and S. D. Gribble, "A Measurement Study of Peer-to-Peer File Sharing Systems", *Technical Report UW-CSE-01-06-02*

[7] R. Puri and K. Ramchandran. "Muliple description source coding through forward error correction codes". In *Proc. Asilomar Conference on Signals, Systems, and Computers,* Asilomar, CA, October 1999. IEEE

[8] MoMusys code, "MPEG4 verfication model version 18.0" *ISO/ IEC JTC1/SC29/WG11 Coding of Moving Pictures and Audio*, January 2001.