MTcast: Robust and Efficient P2P-Based Video Delivery for Heterogeneous Users

Tao Sun\textsuperscript{1}, Morihiko Tamai\textsuperscript{1}, Keiichi Yasumoto\textsuperscript{1}, Naoki Shibata\textsuperscript{2}, Minoru Ito\textsuperscript{1}, and Masaaki Mori\textsuperscript{2}

\textsuperscript{1} Graduate School of Information Science, Nara Institute of Science and Technology, Ikoma, Nara 630-0192, Japan
\{song-t, morihi-t, yasumoto, ito\}@is.naist.jp
\textsuperscript{2} Department of Information Processing and Management, Shiga University, Hikone, Shiga 522-8522, Japan
\{shibata, mori\}@biwako.shiga-u.ac.jp

Abstract. In this paper, we propose a new video delivery method called \textit{MTcast} (Multiple Transcode based video multicast) which achieves efficient simultaneous video delivery to multiple users with different quality requirements by relying on user nodes to transcode and forward video to other user nodes. In MTcast, each user specifies a quality requirement for a video consisting of bitrate, picture size and frame rate based on the user’s environmental resource limitation. All users can receive video with the specified quality (or near this quality) along a single delivery tree. The main characteristics of MTcast are in its scalability, high user satisfaction degree in received video quality, short startup latency and robustness against node failure. Through simulations, we have confirmed that MTcast can achieve much higher user satisfaction degree and robustness against node failure than the layered multicast method.

Keywords: video multicast, transcode, QoS, service overlay networks.

1 Introduction

There is a demand for an efficient video delivery method for \textit{heterogeneous user nodes} which have different computation powers, display sizes and available bandwidths. There are several approaches for simultaneously delivering video to multiple users with different quality requirements. In the multiversion technique \cite{1}, multiple versions of a video with different bitrates are prepared in advance so that the best one can be delivered to each user, within resource limitation. In the online transcoding method \cite{2}, an original video is transcoded at a server or an intermediate node (i.e. proxy) to videos with various quality, according to receivers’ preferences, and forwarded to the receivers. In the layered multicast method \cite{3,4}, video is encoded with layered coding techniques such as in \cite{5} so that each user can decode the video by receiving arbitrary number of layers. Since each layer is delivered as an independent multicast stream, each user can receive as many layers as possible within his/her resource limitation. In this method, as the number of users increases, more layers are required in order to improve user satisfaction degree. However, decoding video from many layers consumes large processing
power and buffers. In [3], a method for optimizing bit rate of each layer to maximize user satisfaction degree is proposed. In the multiversion method, the control mechanism is simple, but not efficient in terms of server storage and network bandwidth usage. In the multiversion and layered multicast methods, there can be a large gap between the requested quality and the delivered quality if there are not enough number of versions or layers. The online transcoding method can satisfy all the above requirements since it can transcode original video to arbitrary quality video. But, large computation power required for transcoding can be a problem.

There are many studies on video streaming in peer to peer networks. [6] has proposed the Overlay Multicast Network Infrastructure (OMNI). In OMNI, each user node works as a service provider as well as a service user, and a multicast tree is composed of user nodes so that the video delivery service is provided to all the user nodes through the tree. OMNI can adapt to the change of the user node distribution and the network conditions. [7] has proposed CoopNet where traditional client-server based streamings are augmented when the load of the video server exceeds its limit. In CoopNet, user nodes cache parts of stream data, and deliver them through multiple diverse distribution trees to the user nodes while the server load is high. OMNI and CoopNet aim at adapting the video delivery service depending on the dynamic change of network conditions, server load and so on. However, they do not treat video delivery to user nodes with different quality requirements.

We propose a new video delivery method called MTcast (Multiple Transcode based video multicast) which achieves efficient simultaneous video delivery to multiple heterogeneous users by relying on user nodes to transcode and forward video to other user nodes. In MTcast, each user specifies a quality requirement for a video consisting of bit rate, picture size and frame rate based on the user’s environmental resource limitation. All users can receive video near specified quality along a delivery tree. Each user can change the quality requirement each time segment or each video shot.

We have considered the following criteria: (1) high scalability for accommodating a large number of users, (2) high user satisfaction in the sense that the delivered quality is close to the required quality, (3) small resource consumption within available resource of each user node, (4) short startup latency to start playing back video quickly, (5) reasonable number of transcoding times for keeping good video quality as well as short delivery latency, and (6) high robustness for continuing video delivery service even with node/link failures.

In order to achieve the above (1) to (3), a delivery tree called transcode tree whose root is the sender of a video content, is constructed as a variation of a perfect n-ary tree, where user nodes with higher quality requirements are located near the root of the tree, and nodes with lower quality requirements are located near leaves. Nodes are placed according to their computation power, available downstream and upstream bandwidths. Each node in the tree receives a video stream, transcodes it to lower quality video in real time and forwards it to its children nodes. In order to achieve the above (4) to (6), nodes are grouped so that each group has k members with similar quality requirements. These groups are called layers. All nodes in a layer receives the video with the same quality from their parent nodes along the transcode tree. We let the representative node of each layer keep the complete information of the tree. This allows