Optimization of Prefetching in Peer-to-Peer Video on Demand Systems

Parag Bafna and B. Annappa

Department of Computer Science and Engineering
National Institute of Technology Karnataka, Surathkal, India - 575025
bafnaparag@gmail.com, annappa@ieee.org

Abstract. In Peer-to-Peer Video on Demand System like Video Cassette Recording (VCR) various operations (i.e. forward, backward, resume) are found to be used very frequently. The uncertainty of frequent VCR operations makes it difficult to provide services like play as download. To address this problem, there exist algorithms like random prefetching, distributed prefetching, etc. But each such algorithm has its own advantage and disadvantages. So to overcome the problem of prefetching we propose optimize prefetching for Peer-to-Peer(P2P) Video on Demand systems.The simulation result proves that the proposed prefetching algorithm significantly reduces the seeking delay as compared with the random prefetching scheme.

Keywords: Peer-to-Peer, Video on Demand, Distributed Prefetching, Random Prefetching, Seeking delay.

1 Introduction

Multimedia communication has been in continuous state of evolution over the past few decades. After the application level multicast, video streaming is boosted by Peer-to-Peer(P2P) networks. Multimedia streaming can be categorized into live streaming and on-demand streaming. In live streaming systems, the source servers broadcast videos, and all the clients are synchronous.Successful examples include CoolStreaming[1], and PPLive[2]. On the other hand, On-demand streaming or Video on Demand is an interactive multimedia service, which delivers video content to the users on his/her demand. Differing live streaming, for on-demand streaming, clients mostly demands different videos or different parts of the same video[10].

It is found that Video on Demand (VoD) is increasingly popular with Internet users. VoD is compelling because it provides users with video stream control, such as pause and random seek which results in increase seeking delay and stress on streaming server.The uncertainty of frequent VCR operations makes it difficult to provide high quality real-time streaming services. To address this problem, prefetching is proposed. In a data prefetching, peers prefetch and store various portions of the streaming media ahead of their playing position, which grants peers the ability to overcome the departure of source peer and to smooth

D.C. Wyld et al. (Eds.): NeCoM/WeST/WiMoN 2011, CCIS 197, pp. 100–106, 2011.

video playing experience. The prefetched portion of content can also serve to other peers on the network which requires some additional bandwidth and storage.Considering the increasing stress on bandwidth, storage capability on local peers nowadays, it certainly offers a more desirable tradeoff between quality and cost. In this paper we proposed an optimized prefetching strategy and compared our strategy with strategies like random prefetching strategy.

The rest of this paper is organized in different sections as follows. In section 2, we discuss the related work. In section 3, we discuss our proposed optimize prefetching strategy. Section 4 illustrates the performance evaluation and comparison of our strategy with random prefetching strategy. Section 5 presents final conclusion.

2 Related Work

In random prefetching[6] it has been observed that it randomly start to select the media chuck which is actually not required for proper functioning of on demand video streaming .The behavior of user while accessing the video chunk is unpredictable so random prefetching does not provides effective solution to such problem. Where as, in case of distributed prefetching[5], user viewing behavior logs are maintained by a tracker server which improves hit ratio by considering users access patterns, however extracting a user viewing pattern from user viewing behavior logs require large computation to be performed by tracker server.

3 Optimization of Prefetching

In order to overcome drawbacks of various prefetching techniques discussed in earlier section, we proposed a new prefetching technique called optimize prefetching strategy. In this technique, every peer node maintains the record of playback media chunks (video divided into media chunks) by other peers in the same session-interval. This information is obtained through mutual sharing basis. After collecting state information from all peers (in same session-interval), a table of available media chunks is constructed by each peer in that particular session-interval. Table 1 shows peers obtained state information.

Table 1. Information received by peer I

Peer ID	Records
I	1 2 5 8 11 20
J	5 6 7 8 9 11
K	1 2 8 15 16 17
L	6 8 9 11 12 17
M	9 10 11 12 13
N	4 5 6 7 8 9 20
O	2 3 4 6 7 8
P	3 4 5 6 7 8 9