



An ant-based rate allocation algorithm for media streaming in peer to peer networks: Extension to multiple sessions and dynamic networks

Hadi Goudarzi^{a,b}, Amir Hesam Salavati^{a,b,*}, Mohammad Reza Pakravan^{a,b}

^a Data Communications Research Lab, Advanced Communication Research Institute, Faculty of Electrical Engineering, Sharif University of Technology, Tehran, Iran

^b School of Electrical Engineering, Sharif University of Technology, Tehran, Iran

ARTICLE INFO

Article history:

Received 6 September 2009

Received in revised form

8 June 2010

Accepted 28 June 2010

Keywords:

Ant-colony optimization

Multimedia streaming

Peer to peer media streaming

P2P networks

Rate allocation

ABSTRACT

In this paper, we introduce a novel algorithm for rate allocation in media streaming P2P networks where multimedia contents are distributed among network members and streamed toward any requesting peer. The proposed algorithm is based on ant-colony optimization. It is capable of handling network dynamism, which is an inherent property of unstructured P2P networks. Another advantage of our algorithm is its ability to get over uncertainties in network state information, particularly the rate of supplying peers that could happen due to lack of accurate measurements. In addition, the suggested method does not rely on any information about the topology of the network.

We have investigated both single and multiple streaming sessions scenarios in which more than one peer is receiving media streams from media providers. We show that the suggested algorithm will reach the maximum achievable rate of the network quite fast. A key feature of the proposed algorithm is its low pass filter property, which makes it discriminate between transient and permanent network changes. If the changes are transient, the algorithm easily and rapidly compensates the temporary losses. In cases where the network changes last longer, the algorithm overcomes losses by employing other nodes that have the media stream available. The rate of adaptation is adjustable and must be carefully determined according to network conditions. Moreover, adaption rate is not constant and varies during the streaming session. This results in uninterrupted services for current users in cases where multiple sessions are present in the network.

Finally, since we have assumed that fountain codes are used to encode media streams in the P2P networks, the suggested algorithm does not require the user to receive different parts of the streams according to a predefined order and from a specific list of media suppliers. It suffices that the user gets as many stream chunks as necessary, regardless of their order or the fact that not all the media suppliers have all the parts available. In other words, using fountain codes enables us to overcome a big difficulty of P2P media streaming and that is to receive different parts of media streams according to a specific order.

© 2010 Elsevier Ltd. All rights reserved.

1. Introduction

Peer to peer networks have grown rapidly in recent years. Although P2P networks were originally designed for file sharing purposes, due to their inherent properties, they are perfectly suitable for other distributed network-wide activities such media streaming and voice communication. As a result of improved users access bandwidth in the past decade, numerous commercial softwares are currently available for each of these purposes such

as Skype (2009) for voice communications, Gnutella (2001) for content sharing, and SoPcast (2009) for video streaming.

The major difference between a general and a media streaming peer to peer system lies in the streaming process. While in a general P2P network users must download a file completely and then view it, in a media streaming architecture users view the media stream on the fly. The streaming bears some challenges to the network due to its real time nature and large bandwidth consumption. In addition, dynamic nature of peer to peer networks makes the streaming process even more difficult.

Since required bandwidth for receiving media streams is quite high and the upload bandwidth of users are limited, a single user usually cannot supply the necessary streaming rate for other peers. Therefore, in practice a subset of peers stream the media file to the requesting peers. Upon completion of the streaming session, the requesting peers become media suppliers. The sum

* Corresponding author at: Data Communications Research Lab, Advanced Communication Research Institute, Faculty of Electrical Engineering, Sharif University of Technology, Tehran, Iran. Tel.: +98 2166165922.

E-mail addresses: h_goudarzi@ee.sharif.edu (H. Goudarzi), salavati@ee.sharif.edu (A.H. Salavati), pakravan@sharif.edu (M.R. Pakravan).