Engineering Applications of Artificial Intelligence 45 (2015) 180–191

Contents lists available at ScienceDirect

Engineering Applications of Artificial Intelligence

journal homepage: www.elsevier.com/locate/engappai

Towards conscientious peers: Combining agents and peers for efficient and scalable video segment retrieval for VoD services

Vladimir Rocha *, Anarosa Alves Franco Brandão [1]

Escola Politécnica. Av. Professor Luciano Gualberto, travessa 3, 158, CEP 05508-010, Cidade Universitria, University of São Paulo, Brazil

ARTICLE INFO

Article history:
Received 25 December 2014
Received in revised form
5 May 2015
Accepted 2 July 2015
Available online 23 July 2015

Keywords:
Multiagent system
Peer-to-peer
Video-on-demand

ABSTRACT

In the last years, Video-on-Demand (VoD) systems, such as Youtube, have become a very popular way to watch videos. Researches in Multiagent Systems (MAS) and Peer-to-Peer (P2P) for VoD have focused on using agents to assist peers to share video segments efficiently, by replicating them or creating the best path between peers to transfer such segments. A key challenge faced by these systems is to efficiently search and to retrieve which peers own certain segments in order to provide the streaming service. In this paper we propose an innovative architecture that combines MAS and P2P network for multimedia information retrieval in the VoD environment. This architecture, composed of a P2P and a MAS layer, supports streaming and random seeking while providing scalability and efficiency. The P2P layer is responsible for discovering the peers that own the video segments and share these segments among them. Nevertheless, if such search is time consuming, it is forwarded to the MAS layer. The MAS layer is responsible for monitoring groups of peers as well as for discovering agents that manage a group of peers sharing certain video segments that could not be found efficiently by peers. This is made for improving the sharing performance of its monitored peers. Experimental results, using several metrics, showed the viability of adopting our approach if compared with the most used approaches.

1. Introduction

With the advent of multimedia devices, such as mobile phones and tablets, and the increase of Internet access, Video-on-Demand (VoD) streaming services have become very popular since they allow users to play a video while it is being downloaded. Examples of this kind of streaming services are Massive Open Online Courses (MOOC) systems, reaching millions of students worldwide (Matkin, 2013) and YouTube (Youtube, 2013), a video-sharing system that has reached 1 billion users and more than 4 billion daily views (Youtube Statistics, 2013).

To provide VoD streaming, several strategies are currently used, including Content Distribution Networks (CDN) (Buyya et al., 2008), Cloud Computing (Payberah et al., 2012), and Peer-to-Peer Networks (P2P) (He and Guan, 2010).

In a VoD P2P network, a video is divided into *N* segments, each one represented by a unique and incremental (starting at 1) identifier, which are distributed and shared across the networked peers. Peers behave as both client and server, downloading video segments from other peers (client) and uploading them to peers that request it (server). Nevertheless, there are peers in such a network that always behave as clients. They are known as *free riders* and their behavior usually degrades the VoD streaming performance of the entire system (Mol et al., 2008; Karakaya et al., 2009).

Currently, BitTorrent is one of the most used P2P video sharing protocols (Ma et al., 2012), composed of one tracker and the peers. The tracker is a centralized server responsible for helping peers to find each other and to monitor their downloading and uploading statistics (transfer statistics). The peers, in turn, are able to share the segments in a swarm, which is a set of connected peers that shares all the video segments among them. In order to efficiently share the segments, BitTorrent provides the *rarest first* policy, in which the main idea is disseminating the less downloaded segment first (Cohen, 2008). Although the *rarest first* policy allows transferring the whole video very fast, it is not applicable to VoD services, in which streaming services need to provide the sequential delivery of segments instead of the less downloaded ones.

In order to solve this problem, many researchers proposed finding the sequential segments by modifying the BitTorrent segment selection policy (i.e., *rarest first*) or the BitTorrent swarm, using distributed structures. In the first case, the BitTorrent segment selection policy is modified to give higher priority for downloading to segments that are about to be reproduced by the peer (D'Acunto

* Corresponding author. Tel.: +55 1183247657.
E-mail addresses: vmoreira@ime.usp.br (V. Rocha), anarosa.brandao@usp.br (A.A.F. Brandão).

[1] Supported by grant #014/03297-7 Sao Paulo Research Foundation (FAPESP).

http://dx.doi.org/10.1016/j.engappai.2015.07.001
0952-1976/

et al., 2013). In the second case, distributed structures (such as AVL Trees, Liu and Zhou, 2006; Distributed Hash Tables (DHT), Yiu et al., 2007) are used to create swarms of peers that share specific segments, using efficient algorithms to find the segments about to be reproduced.

In both alternatives, one of the most important requirements is being efficient and scalable while discovering the requested video segments (Shen et al., 2011) to maintain the user quality of service while reproducing the segment (Hareesh, 2013). However, regardless of the alternative used, it is still a challenge to efficiently and scalably find peers that own specific video segments. Our work addresses this challenge by proposing a hybrid architecture which combines a multiagent approach with the already used P2P one for streaming service in the VoD context.

The architecture combines and improves existing solutions for video segment discovery and retrieval, by dividing responsibilities among peers and agents. For instance, peers are organized in swarms that were optimized by the way connections among peers are established. Also, agents are responsible for managing swarms, acting as distributed trackers and helping peers to find segments that they were not able to find. Therefore, scalability is balanced by the decision of agentifying (converting into agent) just one peer by swarm, reducing the number of agents in the system. In addition, efficiency is balanced by using distributed structures to optimize the agent's search performance.

The paper is organized as follows: Section 2 presents existing solutions for finding specific video segments and compares them with ours; Section 3 provides the broad picture of the solution proposed which is detailed in Sections 3.1 and 3.2. Section 4 presents how the architecture is built during the system lifecycle. The results of the experiments for evaluating the architecture viability are presented in Section 5. Finally, we conclude our work and point some future directions in Section 6. Additionally, a site was created to display all algorithms cited along the text, at http://github.com/AgentPeer/conscientious-peers.

2. Related work

Since our proposal combines several solutions already adopted for dealing with the problem of video segment retrieval, we begin the section presenting a table describing some approaches related to ours by pointing out their strengths and weakness. For reading the table, column 2 is composed of the name of the approach or the name of its authors, the technique adopted (MAS for multiagent systems, DHT for Distributed Hash Tables, BitTorrent for the video sharing protocol), the year it was proposed followed by the associate reference; column 3 states what the focus of the approach is and columns 4 and 5 state their strengths and weaknesses, respectively.

Our proposal uses a combination of existing approaches for profiting from their strengths and addressing their weaknesses. Thus, we propose an architecture that relies on solutions that use MAS, as well as the state-of-the-art P2P technologies for segment discovery and retrieval as well as strategies for dealing with free-riders. Additionally, it introduces new findings to improve them in their weaknesses, to provide more efficient and scalable services.

From the adoption of the MAS point, we provide a clear separation between peer and agent responsibilities, which helps to encapsulate and to differentiate the actions executed, depending on their current role (peer or agent). In addition, our architecture proposes that only a few peers should be converted into agents, making each agent responsible for managing one swarm of peers and addressing an important weakness of existing approaches. In fact, by doing so we are avoiding problems such as scalability and single point of failure. Existing works fail to deal with scalability concerns, while proposing one agent for each peer (Yang et al., 2006; OryCzak and Kotulski, 2011) or on dealing with the single point of failure, while proposing only one agent to deal with all peers (Teket et al., 2014). Considering the system efficiency, existing works in MAS (Savarimuthu et al., 2013; Ciarletta et al., 2013; Centeno et al., 2013) deal with free-riders identification and strategies to influence them to change this behavior for a cooperative one, by sharing their video segments. Our agents influence their swarms to avoid the connection with free-riders (purging them from the swarm) and disseminating this information with other agents.

From the adoption of state-of-the-art P2P technologies for video segment discovery and retrieval, considering BitTorrent, our modified peer behaves exactly as in the BiTos original version, focusing on improving the segment download performance. In addition, along with Waterfall, our swarms are organized in order to share previous, current and next segments among connected swarms, reducing video segment download time. Moreover, our architecture avoids the centralized tracker as a single point of failure (as in BitTorrent and Waterfall), while distributing its responsibilities (joining new peers and time-consuming searches for segments) through the agents of the MAS layer. This is made by using DHT for distributing our agents in the same way as peers in VMesh, TemporalDHT, and TDM, i.e., as part of a global information repository. Therefore, requests are distributed among all peers, increasing the system scalability. In addition, swarm monitoring is performed by agents, which learn from the swarm behavior to improve its efficiency.

3. Proposed architecture

Our proposed solution extends the well known P2P architecture by including functionalities to peers and building a multiagent layer above the P2P layer, through “agentifying” some peers. A sketch of it is shown in Fig. 1. In the multiagent layer, agentified peers act as distributed trackers for managing the swarms, monitoring their behavior and performing searches that are time-consuming within

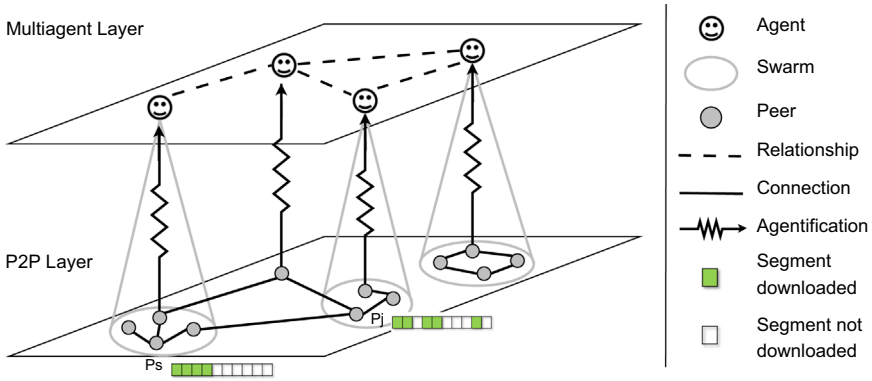

Fig. 1. MAS – P2P architecture for VoD.