Simulation Modelling Practice and Theory 19 (2011) 649–661

Contents lists available at ScienceDirect

Simulation Modelling Practice and Theory

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

AntP2PR: An ant intelligence inspired routing scheme for Peer-to-Peer networks

Fotis Loukos [a,*], Helen D. Karatza [a], Constandinos X. Mavromoustakis [b]

[a] Department of Informatics, Aristotle University of Thessaloniki, 54124 Thessaloniki, Greece
[b] Department of Computer Science, University of Nicosia, 46 Makedonitissas Avenue, P.O. Box 24005, 1700 Nicosia, Cyprus

ARTICLE INFO

Article history:
Received 6 June 2010
Received in revised form 11 September 2010
Accepted 4 October 2010
Available online 13 October 2010

Keywords:
Peer-to-Peer networking
Peer-to-Peer routing
Ant intelligence
Biologically inspired computing

ABSTRACT

Unstructured Peer-to-Peer networks consist of an infrastructure-less overlay on top of another network. Most of them use distributed algorithms for all operations, such as resource discovery or connectivity control. Research has shown that a considerable amount of the generated traffic is due to signaling messages. Furthermore, another challenge when implementing a Peer-to-Peer network is avoiding free riders, i.e. users trying to profit from the network without sharing their resources. In this paper a new approach to routing packets in such networks is presented using ant intelligence. Success messages are used as agents and the biological procedure of pheromone trails is used for forwarding new packets used in resource discovery. These agents carry an amount of pheromone which will be added to a pheromone table representing routes to other peers. This approach enables the network to adjust to the dynamic nature of Peer-to-Peer networks where new nodes connect and disconnect continuously. Peers that are free riding will be ultimately isolated from the rest of the network by limiting the number of messages directed to them. The authors have simulated an unstructured Peer-to-Peer network, such as Gnutella, that uses this method and the results are very promising. The amount of traffic used solely for resource discovery is greatly reduced enabling the users to use more bandwidth for transferring content.

1. Introduction

Peer-to-Peer technology is one of the most frequently cited expressions when it comes to networking. The term Peer-to-Peer is used to denote a network of peers which forms an overlay structure on top of another infrastructure, such as the Internet [1,2]. In contrast to the conventional client–server model, these peers are equal to each other and can act both as a client and a server [3].

The first and primary use of Peer-to-Peer networks is file sharing. However, since large-scale environments based on such networks have emerged, research on other applications commenced. By utilizing many useful properties of Peer-to-Peer networking, new protocols have been proposed for other uses such as process execution. An example is the OSIRIS [4] framework which combines many different technologies to provide new value-added services.

Peer-to-Peer networks can be divided into two main categories based on the location where the directory of content or available services is stored, *centralized* and *decentralized*. The first popular Peer-to-Peer network for file sharing was Napster [5], which had a centralized architecture. The directory of all users and the files they share is stored at a central

* Corresponding author. Tel.: +30 6979779108.
E-mail addresses: fotisl@csd.auth.gr (F. Loukos), karatza@csd.auth.gr (H.D. Karatza), mavromoustakis.c@unic.ac.cy (C.X. Mavromoustakis).

1569-190X/$ - see front matter © 2010 Elsevier B.V. All rights reserved.
doi:10.1016/j.simpat.2010.10.004

server. Whenever a node wants to search for a keyword it makes a query to the central server which replies with the peers, their addresses and the matching files. In order for a node to download a file it must contact directly the other peer and node the central server, which distributes the load. This is a much faster approach that the conventional client–server model and can scale better. However, there exists a single point of failure. A lawsuit by RIAA forced Napster to shutdown the operation of the central server which essentially shut down the whole network.

In order to overcome this problem, a more decentralized approach was used. The Gnutella network [6] uses a completely decentralized architecture with each node holding a catalog of its own shared files. A number of mechanisms, such as local and web cache, are used in order for a peer to discover a number of initial nodes to connect to, and then receive information about other peers in the network to update its local cache. Whenever a node needs to search for a file, it creates a query message and sends it to all of its neighbors. When a node receives such a message it checks the catalog of its shared files and forwards the message to the rest of the peers it has connected to, essentially flooding the network. This flooding is limited by the time-to-live parameter of all messages which specifies the maximum number of hops that the message may pass through. However, as the network grows in size the number of messages increases exponentially and it is apparent that this approach is not scalable.

Networks such as Freenet [7] have tried to solve this problem by creating a structured network using some form of Distributed Hash Tables (DHT). By using DHTs, a hash is generated for every shared file and each hash is assigned to a specific node. Whenever a peer wants to search for a file, it uses its hash value and a special routing protocol is used for forwarding the request. The majority of DHT Peer-to-Peer networks require a knowledge of $O(\log n)$ nodes and can route the query message to the responsible node in $O(\log n)$ hops. However, DHTs cannot be used always as a replacement for unstructured networks since they require the knowledge of the hash values for the files a peer wants to download.

Zhang and Hu have used concepts from Distributed Hash Tables at unstructured Peer-to-Peer networks in order to provide a more efficient search algorithm [8]. The proposed *Assisted Peer-to-Peer Search with Partial Indexing* uses a partial index of shared files in order to provide hints for peers sharing the same interests. Furthermore, information about unpopular files is provided to improve the success rate in locating them.

The Kazaa network [9], based on the FastTrack technology, does not make any use of algorithms from structured networks but instead introduces the concept of supernodes, i.e. nodes that have higher bandwidth connectivity and more available resources. Supernodes are used to store a catalog of the files of all non-supernode neighbors. This catalog is often updated to reflect any changes to the local catalog of every node. Whenever a supernode receives a query, it has knowledge of the files that its neighbors share and can reply for them. A caching scheme like this has been also included at a revised version of the Gnutella protocol. This approach has better results than the original flood algorithm, however it still does not allow the network to scale.

The authors propose an algorithm based on Ant intelligence. Ant intelligence is part of biologically inspired computing. Biological processes have inspired a lot of algorithms and methods for solving advanced computer problems. The study of the nervous system and its functions has fortified computer scientists with the necessary tools to create neural networks. Furthermore, the dynamic organization of many different living organisms and their adaptiveness to continuously changing conditions has featured biology as one of the main themes from which researchers can borrow ideas.

Some of the organisms that exhibit such useful properties are social insects. Insect swarms are similar to a fully distributed network when it comes to computer science. Babaoglu et al. [10] have described a number of different design patterns for solving problems in distributed networks using procedures borrowed from ant colonies. These design patterns are based on diffusion, replication and finally stigmergy. A number of different problems are illustrated together with their biologically inspired solutions.

The paper is organized as follows: In Section 2 the authors discuss the related work on the same subject. Section 3 presents some of the challenges that turn up when implementing Peer-to-Peer networks in detail. In the next section, the pheromone based routing algorithm is presented. Section 5 draws the simulation results and, finally, conclusions and future work insights are given in Section 6.

2. Related work

One problem that can be solved with ant intelligence is IP based routing. One of the most notable routing schemes is AntNet [11], a Monte Carlo system based on agents. It uses the stigmergy function of ants for exchanging information between the agents. In AntNet the ants are represented as agents which continuously explore the network and exchange information. This information is pheromone values that each node keeps for all of its neighbors and that create the routing tables. At regular intervals, new forward ants are generated which follow a specific route and generate backward ants that help update the pheromone trail values. This dynamic update helps adjust the routing to any changes that might occur at the network topology. The authors have evaluated their algorithm and compared it to six different routing algorithms. The results under many different conditions have been very promising.

Ant based routing has also been used to offer quality of service at various types of networks. Mavromoustakis and Karatza [12,13] have described the Split Agent Routing Technique (SART) for efficient capacity allocation and bandwidth reservation. This technique uses pheromone tables in order to obtain the desired results. In nature, ants are using pheromones in order to mark the shortest path to a destination and the quality of food source. Pheromones are chemicals deposited by ants in order