Computer Communications 32 (2009) 41–50

Contents lists available at ScienceDirect

Computer Communications

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

A peer-to-peer file search and download protocol for wireless ad-hoc networks ☆

Hasan Sözer [a,*], Metin Tekkalmaz [b], Ibrahim Korpeoglu [b]

[a] Faculty of Electrical Engineering, Mathematics and Informatics, University of Twente, Enschede, The Netherlands
[b] Department of Computer Engineering, Bilkent University, Ankara, Turkey

ARTICLE INFO

Article history:
Received 15 March 2007
Received in revised form 26 August 2008
Accepted 4 September 2008
Available online 11 September 2008

Keywords:
Wireless ad-hoc networks
Peer-to-peer networks
File sharing

ABSTRACT

Deployment of traditional peer-to-peer file sharing systems on a wireless ad-hoc network introduces several challenges. Information and workload distribution as well as routing are major problems for members of a wireless ad-hoc network, which are only aware of their immediate neighborhood. In this paper, we propose a file sharing system that is able to answer location queries, and also discover and maintain the routing information that is used to transfer files from a source peer to another peer. We present a cross-layer design, where the lookup and routing functionality are unified. The system works according to peer-to-peer principles, distributes the location information of the shared files among the members of the network. The paper includes a sample scenario to make the operations of the system clearer. The performance of the system is evaluated using simulation results and analysis is provided for comparing our approach with a flooding-based, unstructured approach.

1. Introduction

Peer-to-peer networks have been very popular since their first emergence. Several peer-to-peer file sharing systems have been deployed and are functional on the Internet, such as Napster [1], Gnutella [2] and FastTrack [3]. Similar systems currently serve many users who are able to share files located on their personal computers. Together with the new users of the Internet and the emergence of different types of files to be shared (documents, audio/video files, etc.), the number of users of peer-to-peer file sharing systems increases every day.

In the meantime, mobile devices and wireless communication technologies have evolved and become very popular. Both areas have experienced rapid improvements during the last few years, which have led to the development of high-performance products. Today, personal digital assistants (PDAs) have almost the same abilities of ordinary desktop computers despite their small size and weight. On the other hand, new wireless technologies enable handheld devices to communicate and form ad-hoc networks easily and automatically. Bluetooth [4], for instance, is one such technology that uses short-range radio communication and interconnects handheld electronic devices ranging from cellular phones to PDAs.

Today, high-performance handheld devices can communicate with each other in a wireless ad-hoc network (WANET). Such an environment provides the possibility to share files. Moreover, peer-to-peer systems that are often employed for file sharing are also suitable for WANETs, since they do not require any infrastructure. However, the deployment of traditional peer-to-peer file sharing systems on a WANET introduces several challenges. Such networks can be formed anytime/anywhere without requiring any infrastructure, and the nodes of the network may change their locations. In addition, a peer-to-peer file sharing system that is running on the Internet relies on the network layer (IP) for communication between nodes and for downloading files. A WANET needs to run an ad-hoc routing algorithm to provide these services. Several protocols [5,6] have been proposed to route packets in a WANET, and some have been standardized [7]. However, we still lack a widely accepted common routing protocol that is implemented and deployed. Moreover, the standardization efforts tend to keep the WANET routing protocols simple (due to the heterogeneity of mobile systems), which can be inadequate to provide the services needed for specific applications [8].

The work presented in this paper takes a cross-layer design approach where the lookup functionality and the routing functionality are unified. We propose a file sharing system, which determines both *from where* and *how* to obtain a file in a WANET. The system works in a peer-to-peer manner and it distributes the location information of the shared files among the members of the network. To store and maintain the location information together with the routing information, the system uses a distributed hash table and a tree-structure based on the topology of the network. The system also employs and adapts dynamic source routing [5] and peer-to-peer location lookup techniques [9,10].

We present a sample scenario to describe how the set of system operations work together to accomplish file sharing among the

☆ This work is partially supported by TUBITAK The Scientific and Technological Research Council of Turkey (Project n. EEEAG 104E028), and by the European Commission in the framework of the FP7 Network of Excellence in Wireless COMmunications NEWCOM++ (contract n. 216715).

* Corresponding author. Tel.: +31 53 489 5682; fax: +31 53 489 3247.
E-mail addresses: sozerh@ewi.utwente.nl (H. Sözer), metint@cs.bilkent.edu.tr (M. Tekkalmaz), korpe@cs.bilkent.edu.tr (I. Korpeoglu).

0140-3664/$ - see front matter © 2008 Elsevier B.V. All rights reserved.
doi:10.1016/j.comcom.2008.09.004

members of a WANET. We also provide the simulation results of the system and we elaborate on the messaging requirements to maintain the distributed index, to perform file queries, and to access files. Simulation results showed that the high mobility of the nodes leads to poor performance due to frequent updates of the distributed index. For this reason, we also make a trade-off analysis and we compare the bandwidth efficiency of our system with that of a flooding-based, unstructured approach. Results showed that our system performs much better as the network size increases. In a WANET with 100 nodes, less than one file query per node is enough to amortize the cost of an index update.

The remainder of this paper is organized as follows. In the next section, related previous studies are summarized. In Section 3, an overview of the system is given, which is followed in Section 4 by a detailed description of each operation supported by the system. In Section 5, a working scenario of the system is described to show how each operation updates and maintains the distributed location and routing information stored in the system. Next, in Section 6, simulation results and discussions on them are presented. Finally, in Section 7 conclusions are given and some future work issues are discussed.

2. Related work

As far as peer-to-peer (P2P) file sharing is considered, Napster [1] appears to be one of the earliest and most popular applications. Napster, in its initial form, enabled file sharing among computers on the Internet, which is by nature unpredictable, since it is hard to predict when the computers connect and disconnect. The main idea behind Napster is a central server that stores index information (i.e. filename and address pairs), which is used to answer queries about where the files are stored on the Internet. Once the location of a file is determined, file transfers are carried out in a P2P manner. Although the actual file transfers are P2P, index information is accessed using a client–server paradigm. Napster enables easy location lookup by using a central server, but it is affected by the typical weaknesses of centralized systems. Several studies have been carried out in recent years to cope with problems posed for P2P file sharing by the dense, highly dynamic and lowly aware nature of the Internet. More recent works aim for fully distributed P2P systems; therefore, they store index information in a distributed manner. One such distributed system, Content-Addressable Network (CAN) [9], is based on a fully distributed hash table. In CAN, filenames are hashed and mapped to points in a d-dimensional space. The d-dimensional space is divided into chunks and distributed among the members of the network where each member is responsible for one portion of the space (i.e. a chunk). Along with a chunk, each node stores some information about the neighboring nodes, which makes searching of files possible by providing the location information for files and an overlay network-level routing. Chord [11] is another well-known fully distributed P2P system in which a ring shaped overlay network is applied. Each node on this ring maintains pointers to other nodes at various distances. To gather the location information of a file, these pointers are followed in a manner that shortens the access path as much as possible. Ref. [10] surveys P2P content distribution technologies and it can be referenced for other significant P2P file sharing methods.

The P2P file sharing approaches mentioned so far are mainly designed for the Internet. The wireless ad-hoc networks (WANETs) counterpart of the same problem comes with several difficulties due to the dynamic nature of the WANETs as stated previously. The first work on P2P file sharing on WANETs is 7DS [12]. 7DS allows nodes with an intermittent Internet connection to browse the web, in which, whenever a node fails to connect to the Internet, it can search for the required data among its peers. Ref. [13] is based on partial flooding where the searches are carried by queries broadcasted several hops ahead, and where flooding the entire network is prevented by mechanisms like caching and selective routing. ORION, described in [14] and being another P2P file sharing approach for WANETs, also employs flooding for the file queries. The query results are returned selectively, hence duplicate results are avoided for the same file. During the file transfer phase, the caches constructed while querying the files are used for routing. A more recent work on ad-hoc P2P file sharing, which is described in [15], is also based on flooding. Various heuristics and techniques like replication, query filtering and limiting the number of hops for message forwarding are applied in order to reduce the overhead of the flooding. [16] employs context-awareness toward the same aim. Approaches based on flooding work fine for small WANETs but as the network gets larger they cause traffic overhead and the probability of finding a file in the network reduces. In our work, rather than utilizing a flooding-based protocol and introducing techniques to reduce its overhead, we propose a novel, cross-layer system, which combines a location information service (see [17] for more information on location information services) and routing functionality. Our system is designed and specialized to provide a deterministic way to locate and access files (i.e. if a file is shared in the WANET, its location can be determined and it can be accessed).

Virtual ring routing (VRR), proposed in [18], is one of the most recent studies on routing which can successfully work on WANETs. Like many overlay routing protocols, VRR employs distributed hash tables (DHT), but it is directly implemented over the link layer and thus does not require an underlying network routing protocol. VRR performs better than many existing routing protocols working on WANETs since it requires neither network flooding nor translation between fixed identifiers and location-dependent addresses. Although VRR is proposed as a routing protocol, it can also provide DHT functionality in which keys can identify application objects (e.g. file locations) instead of routes to the nodes. But in order to provide this functionality, VRR requires additional messaging, whereas our algorithm employs a cross-layer approach, hence distributing the key-value pairs is enough both for DHT and routing functionality.

Both our algorithm and VRR make use of overlay networks, although they have different designs. The overlay network used by our algorithm is actually a spanning tree of the graph representing the connectivity of the nodes in the network, hence if two nodes are adjacent in the overlay network, they are in the communication range of each other. On the other hand, adjacent nodes in the virtual ring of VRR are not necessarily so in the physical topology of the network. Since the messages are routed through the overlay tree in our algorithm, the paths between the nodes are not always the shortest ones on the actual topology. However our design provides significant advantages over VRR as far as the space and time requirements as well as the maintenance costs of the routing tables are considered. As described in [18], VRR requires $rp + k$ routing table entries per node on the average, where r is the number of virtual neighbors, p is the average path length, and k is the number of physical neighbors. On the other hand, the number of routing table entries of the proposed algorithm is at most k. If p is assumed to grow with $\sqrt{n}$ as in [18], where n is the number of nodes in the network, and k is assumed to be constant for a given node density, the routing table size is $O(r\sqrt{n})$ for VRR, whereas it is constant, that is $O(1)$, for the proposed algorithm. Hence, VRR requires much more space to store the routing tables and processing power to search on the tables for each packet routed. Furthermore, the proposed algorithm requires much less communication overhead to construct the routing tables since the newly connected node only communicates with its parent in the overlay tree, which is in its communication range. On the other hand, in VRR, whenever a new node connects, multihop communication is required between the new node and all of its virtual neighbors in order to update the routing tables and DHT to reflect the new topology.