SEOUL NATIONAL UNIVERSITY
College of Engineering
Technology Management, Economics and Policy Program

Technology Management, Economics and Policy Papers

A P2P File Sharing Network Topology Formation Algorithm Based on Social Network Information

Jörn Altmann and Zelalem Berhanu Bedane

TEMEP Discussion Paper No. 2010:49

서 울 대 학 교
공 과 대 학, 기술경영경제정책 대학원과정
151-742 서울시 관악구 신림동 산 56-1

Technology Management, Economics and Policy Program
College of Engineering, Seoul National University
San 56-1, Shillim-Dong, Kwanak-Gu, Seoul 151-742 ,KOREA
Phone: ++82-2-880-9140, Fax: ++82-2-880-8389

Technology Management, Economics and Policy Program (TEMEP) is a graduate program at the Seoul National University. It includes both M.Sc. and Ph.D. programs which are integrated into the graduate program at the College of Engineering.

TEMEP Discussion Papers is intended to serve as an outlet for publishing research about theoretical, methodological and applied aspects of industrial economics, especially those related to the institute's areas of specialization, namely management of technology, information and telecommunication, health and energy industries, as well as development economics infrastructure. In particular submission of studies analyzing current technology and industry related issues and discussion of their implications and possible alternative policies are welcome. The objective is to gain insights into important policy issues and acquiring a balanced viewpoint of policymaking, technology management and economics which enable us to identify the problems in the industries accurately and to come up with optimal and effective guidelines. Another important aim with the series is to facilitate communication with external research institutes, individual researchers and policy makers.

Research disseminated by TEMEP may include views on information and telecommunication, technology management, energy, health and development economics policy, but the institute itself takes no institutional policy positions. When appropriate the policy views of the institute are disseminated in separate policy briefs. Thus, any opinions expressed in TEMEP Discussion Papers are those of the author(s) and not necessarily the institutes.

Editor: Almas Heshmati
Professor of Economics

Technology Management, Economics and Policy Program
College of Engineering
#37-306, Seoul National University
San 56-1, Shilim-dong, Kwanak-gu
Seoul 151-742 Korea

Phone: 0082-(0)2-880-5845
Fax: 0082-(0)2-886-8220
E-mail: heshmati@snu.ac.kr

A P2P File Sharing Network Topology Formation Algorithm Based on Social Network Information

Jörn Altmann
Department of Industrial and Management Engineering
College of Engineering
Seoul National University
Seoul, South-Korea
email: jorn.altmann@acm.org

Zelalem Berhanu Bedane
Data Processing Technology
Department of Computer Science and Mathematics
FernUniversität Hagen
Hagen, Germany
email: abazebe@yahoo.com

January 2010

Abstract: This paper applies the theory of social networks to P2P systems, creating a social-network-based P2P network topology formation algorithm for file sharing. The algorithm extends the Gnutella P2P file sharing technology, which uses super nodes for searching and for relaying shared files between network leafs that are located behind Firewalls/NATs. The topology of the P2P network is based on the actual social relationship between peers (users). The idea is that users are willing to contribute their resources to a P2P network if they know that their resources directly benefit their friends and family. Following this approach, free-riding in P2P networks will be avoided by not providing better-than-basic service if peers do not reveal their social relationships. Within the paper, we simulate the proposed topology formation algorithm, considering the real characteristics of the Gnutella P2P network and realistic network topologies. The simulation shows the effectiveness of the topology formation algorithm and the high utility of nodes under this new file sharing scheme.

Keywords: *social network analysis; peer-to-peer; economics; network science; free-riding; file sharing; sociology; NATs; utility; Gnutella; Firewalls; network economics; incentive mechanisms*

JEL Classification Numbers: *C13, C14, C61, C62, C63, D01, D02, D82, D85, L96, L96, M15, M21*

1. Introduction

Efficient and effective resource sharing systems are invaluable in modern economic systems such as the Internet. Peer-to-Peer (P2P) systems are among the newly emerging Internet resource sharing mechanisms for sharing content files, processing cycles, disk storage, and bandwidth. The core objective of P2P systems is to create an efficient system through which users share Internet resources among a group of peers. Through P2P systems users are able to give and take resources depending on their terms of agreement.

Although P2P systems have huge potential, there are technical and non-technical challenges. Free-riding, security breaches, existence of firewalls/NATs, and non-ethical behavior of users are the main prevailing challenges. This paper proposes a solution to two of those issues (free-riding and existence of firewalls/NATs), using social network information [18]. It will use this information for formulating a topology formation algorithm that will help to mitigate or even avoid these issues.

The underlying research questions that are addressed within this paper are: (i) How do we integrate social network information into hybrid P2P networks like Gnutella to discourage free riders? (ii) How can we improve the use of super nodes as a relay machine to overcome the firewall/NAT problem? (iii) To what extent can the new system (the social-network-based P2P network topology formation algorithm) reduce inefficient performance without affecting the performance of the super node?

The paper is structured as follows. After this introductory section, the issues of free-riding and firewalls are discussed in more detail. Section 3 describes the new topology formation algorithm, based on a discussion about social network information in P2P systems. Section 4 provides simulation results and an analysis of the solution proposed. The paper concludes with a summery and recommendations in section 5.

2. P2P Issues

2.1 Free-Riding

Free-riding is the main issue for the further development and application of P2P systems. For instance, a study of a Gnutella network showed that 70% of the peers do not share any files at all. Furthermore, 63% of the peers, who share some files, do not get any queries for these files [1]. Another study of the Gnutella network showed that 85% of the users are free-riders [8]. Recently, it has been shown that free-riding is also a common problem in eDonkey. About 80% of the clients are found to be free-riders [7]. The studies indicate that free-riding exist in two forms. There are:

- Free riders, who do not share files at all, and
- Free riders, who share files with irrelevant content.

Consequently, relevant shared files are concentrated among a limited number of peers, causing network congestion. In addition to this, peers that are willing to share their resources experience performance degradation of their system and network.

In order to discourage free-riding, researchers conducted different studies and propose various mechanisms. Andrade et al. suggested an incentive mechanism based on a reputation scheme [3]. He argued that collaboration among peers can be enhanced through provisioning of an incentive scheme. Vishnumurthy et al. proposed a similar approach [16]. They suggested using an incentive mechanism based on a global or community-based reputation scheme instead of a local one. Idris and Altmann try to solve the free-riding problem by narrowing the concept of public goods into club goods [10]. In their approach, they group users into clubs with similar reputation and behavior.

2.2 Firewalls/NATs

Firewalls and Network Address Translators (NATs) create another issue for P2P networks. In P2P networks like Gnutella, file sharing is not possible if both nodes are behind NATs. This fact has two implications. First, peers cannot directly consume resources from other peers, and, second, they cannot provide their resources to other peers. Since more than 50% of the nodes are located behind firewalls/NATs [5] [6], firewalls have a serious impact on the adoption of P2P systems.

Generally, the NAT problem forces P2P systems to depart to some extent from its basic concept and adapt a central management concept. A weak form of this is the involvement of a third party (another peer) that is not behind a firewall. However, the involvement of a third party in the sharing process increases the cost for sharing, and, because of the methods applied, might discourage sharing of resources.

For instance, although Skype attempts to solve the NAT problem by using *super nodes* as a relay machine, users behind NATs experience low performance when transferring files. This is caused by the fact that Skype limits the file transmission speed to 5 kbps. The low transmission speed avoids overloading super nodes with forwarding packets and, therefore, avoids performance degradation of super nodes [2]. This is essential, since it has been shown that Skype traffic can even saturate 100Mbit/s connections. As a consequence, some universities with high-speed connections even tried to restrict Skype usage in their domains [20].

3. Topology Formation Algorithm

3.1 Social Network Information in P2P Systems

Besides the economics of P2P systems [21], the impact of social network information on P2P systems has been studied in [22] [23]. A survey with Skype users has been analyzed in [19]. In this survey, the researchers considered two scenarios: the regular Skype technology, which does not apply social network information (Scenario I) and Skype technology using Social Network Information (Scenario II). Fig. 1 shows both of the scenarios. Within Scenario I, the system automatically assigns super nodes to each node from a list of available super nodes. The selection of super nodes is done implicitly by the Skype system. The Skype system also restricts all nodes to use at most 5 kbps bandwidth of its super nodes. Within Scenario II, nodes can select their super nodes from their contact lists. There is no explicit restriction on the bandwidth that could be used for file transfer. Instead, the bandwidth that can be used depends on the social relationship that the nodes have with the super nodes. The researchers argue that users (i.e. super nodes) are willing to provide their resources to their contacts, depending on the capacity of their resources and the closeness of their relationship with their contacts.

The central research question of this survey has been “What makes a super node share its resources for the benefits of its contacts?” According to the survey results, on average, 31.30% of the contact lists contains family members, 36.28% close friends, 3.74% loose contacts, 22.56% work acquaintances, 2.16% friend of a friend, 3.68% people who sent invitations to make friends, and 0.28% belonged to other groups. Therefore, 67.58% of contacts are family members or close friends, having a trust relationship.

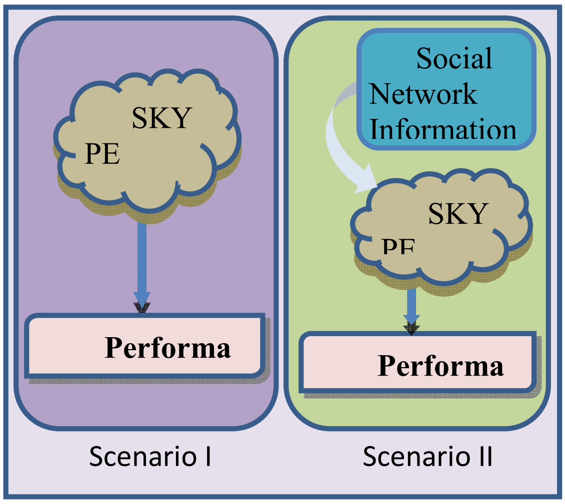

Figure 1. Skype with and without social network information.

In general, the analysis result of the survey indicates that the performance problem as a result of free-riding and NATs can be reduced by applying social network information in P2P systems.

3.2 SocialGnutella Algorithm

3.2.1 Components of SocialGnutella

The three components of the new topology formation algorithm are super nodes, ordinary nodes, and the contact list of the nodes:

- **Super Nodes (SN):** Super nodes in this architecture are used as a relay machine and routers. Super nodes will be selected based on their performance capacity and their Firewall/NAT status. SN must be free from any Firewall/NAT.

- **Ordinary Nodes (N):** Ordinary nodes are nodes that have no super node status. A node may or may not be behind a NAT/Firewall.
- **Contact List:** Each node has a contact list, which contains information about their contacts. These contacts (social relationships) are categorized as *Strong*, *Weak*, or *Visitor*. For example, friends and family in the contact lists get rated *Strong*, and nodes that are not best friends are classified as *Weak*. Any nodes that are not in the contact list are considered *Visitor*.

An example of a social network-based P2P network is shown in Fig. 2.

3.2.2 Protocol of SocialGnutella

The protocol is based on Gnutella v6.0, which runs on top of TCP/IP, and enables two nodes to communicate with each other and share files. The protocol is composed of the following three steps:

Step 1: Establishing a connection

First, the node establishes a connection with the currently active super node in the system. If the node has no friends, who are super nodes, or if its friend (super node) is inactive, they will join a super node as a Visitor with limited access. If friends are available that are active super nodes, the node sends a connection request to one of its super nodes in the contact list.

Super Nodes	Contact Lists	Visitors
SN1	N1, N2, N3, SN2	N4, N7
SN2	N5, N6, SN1	N4
SN3	N8, N9, N10, N11, SN1, SN4	N7, N12
SN4	N13, N14, N15, N16, N17, SN3	N7, N12

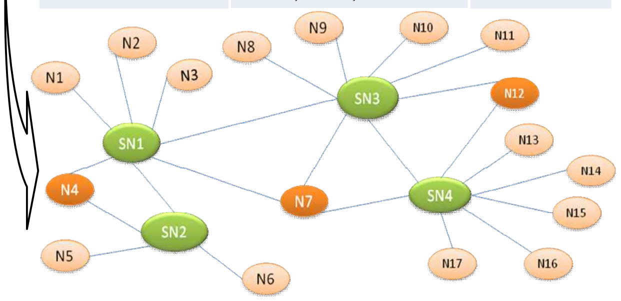

Figure 2. Example of a social-network-based P2P network.

Step 2: Searching for a file

After a node established a connection, it may send a QUERY (a request for a file) to the super node. The super node forwards/broadcasts the QUERY message to its neighboring super nodes and to the nodes, which are currently connected to it. Super nodes will forward the QUERY unless the Time-to-Live (TTL) has expired. If there are nodes in the system, which have the specified file, they will send a QUERYHIT (a response to the request) message back to the requestor. The requestor then selects one from the list of QUERYHITs, considering the NAT status of the responder and its distance in the network. This part of the protocol is identical to the one of Gnutella v0.6.

Step 3: Transferring file

After selecting the responder, the requestor will start the transfer of the file in one of the following three ways, where the first two are identical to the Gnutella v0.6 protocol.

- **Direct Download:** If the responder is not behind a NAT, then the requestor will download directly from the responder. In this case, a relay through the super node is not needed.
- **Push Download:** If the responder is behind a NAT and the requestor is not behind a NAT, then the requestor will send a PUSH message via the super node. It requests the responder to establish a direct connection with the requestor so that the responder can upload the file.
- **Relay Download:** If both the requester and the responder are behind the NAT, then the requestor will send a RELAY message to the super node. In the process of receiving the RELAY message, the super node will first check whether the responder is already connected to it or not (Fig. 3).

If the responder is already connected to it, it will start downloading automatically from the responder. When the file has been completely downloaded to the super node, then the super node sends a PULLHIT message to the requestor, allowing the requester requesting the file. Afterwards, the requester sends a PULL message to the super node to download the file.

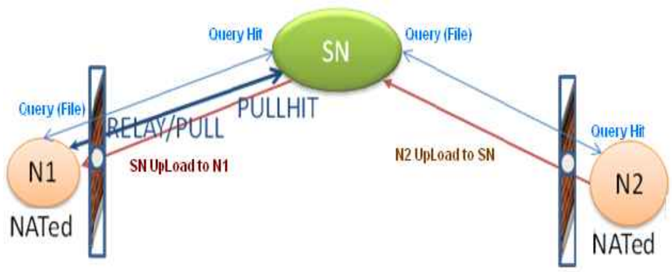

Figure 3. Both Nodes (N1, N2) are members of the same super node (SN)

If the responder is not found in its connection list, it will broadcast to the neighboring super nodes and it will expect a RELAYHIT message from a super node (Fig. 4).

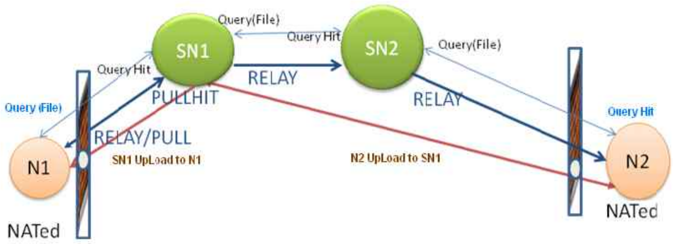

Figure 4. Both nodes (N1, N2) are members of different super nodes (SN)

The following sequence of instructions shows how a super node handles the RELAY message that comes from a requestor node:

if (SenderID $\subseteq$ Supernode.ContactList) {
    If (ResponderID $\subseteq$ Supernode.ContactList) {
        waitingForRelay.put(relayQuery);
    }
} else {
    waitingForConnection.put(relayQuery);
}

waitingForRelay is a list at a super node, where all relay queries (*relayQuery*) are lined up in a first-come-first-serve order and are kept until they are given bandwidth access for transferring the files. Relay queries, whose responder ID is not known at the super node, stay in *waitingForConnection* list of the super node until the responder establishes a connection to the super node of the requestor node.

The super node can control its bandwidth by restricting the share size in such a way that sharing will not significantly affect the super node. The following algorithms show how a super node downloads files from responder nodes and uploads files to the requestor nodes. The super node downloads the file that is at the top of the *waitingForRelay* list.

```
while (WaitingForRelay is not empty) {
	if (Supernode.relationWith(requestorNode)==WEAK) {
		// num is the number of currently transferred files that
		// have been requested by strong nodes
		if (((inBandwidth - (totalWeakBW + 5120)) / num) >=  minBWforSN) {
			// 5kps=5120
			totalWeakBW = totalWeakBW + 5120;
			availableBW = ((inBandwidth - (totalWeakBW + 5120)) / num);
			waitingForRelay.remove(relayQuery);
			downLoadFileWeak(descriptor_Id, fileid, filesize);
		}
	} else {
		// num - number of files currently transferred
		if (((inBandwidth - (totalWeakBW)) / num + 1) >= minBWforSN) {
			num = num + 1;
			availableBW =((inBandwidth - totalWeakBW) / num);
		}
		waitingForRelay.remove(relayQuery);
		downLoadFileNonWeak(descriptor_Id, fileid, filesize, nodeID);
	}
}
```

The super node uploads the file that is at the top of the list *upLoadWaitingList* if sufficient bandwidth is available for uploading.

```
if (upLoadWaitinglist.isEmpty==false) {
	if (Supernode.relationWith(requestorNode)==WEAK) {
		if (((inBandwidth - (totalWeakBW + 5120)) / num) >= minBWforSN) {
			totalWeakBW = totalWeakBW + 5120;
			availableBW =((inBandwidth - totalWeakBW) / num);
			upLoadFileWeak(descriptorID, fileID, fileSize);
		}
	} else {
		if (((inBandwidth - (totalWeakBW)) / num + 1) >= minBWforSN) {
			num = num + 1;
			availableBW =((inBandwidth - totalWeakBW) / num);
			upLoadFileNonWeak(descriptorID,fileID, fileSize);
		}
	}
}
```

4. Simulation

4.1 Simulation Environment

For the simulation, we extended the simulation program used by Idris and Altmann [10]. This simulator is written in Java and uses J-Sim. The simulation program comprises two main parts: The first part generates a P2P network topology, based on some input parameters. Based on the statistical information obtained from [5] [6], it is assumed that 68% of the nodes of this topology will be behind Firewalls/NATs. The remaining 32% of the nodes are assumed to have direct Internet connectivity. The P2P network topology is stored in a set of text files that can be read by the second part of the simulation program.

The distribution of files within the simulation system is based on the information found in [7] [8]. Accordingly, we assume that the files in the simulation system will be distributed only on 20% of the nodes in the system. The distribution of files to each of these 20% of nodes and the file size distribution follow a Gaussian distribution. The distribution of the relationship type in the contact list of a node and the size of the list is based on the study result in [19], as described in section III.A. The link capacity of nodes in the P2P network is characterized by the following distribution: 10% of the nodes have little bandwidth (modem), 60% have average broadband (DSL), and 30% have a high bandwidth connection (20 Mbps).

4.2 Simulation Parameters

The effectiveness of the algorithm is measured based on the utilization of the bandwidth of a super node, the utility of a super node, and the utility of nodes with a *strong* relationship to a super node. For calculating those values, the following parameters are used:

- **TWBW:** This is the total bandwidth consumption by *weak* nodes. It is calculated as the sum of the bandwidth consumption of all *weak* nodes that are currently using the super node as a relay.
- **TBW:** It represents the total bandwidth capacity of a super node.
- **NC:** It denotes the number of file transactions (downloads and uploads) by *friend* nodes (i.e. *strong* relationship) currently using the super node as a relay. This number depends on the number of contacts with a strong relationship to the super node and in need of a relay service.
- **BWAvailable:** It specifies the available bandwidth of a super node. If a request comes from a *strong* node, the available bandwidth is:

$$\mathrm{BW}_{\mathrm{Available}}(\mathrm{SN}) = \frac{TBW - (TWBW)}{NC + 1} \qquad (1)$$

If a request is from a *weak* node, the available bandwidth is:

$$\mathrm{BW}_{\mathrm{Available}}(\mathrm{SN}) = \frac{TBW - (TWBW + 5)}{NC} \tag{2}$$

Note: Weak nodes are allowed only to use 5 kbps for transferring files.

- **Utility of a Super Node:** The utility of a super node is measured based on the bandwidth available for its own use ($\mathrm{BW}_{\mathrm{Available}}$). The utility is one, if the super node could utilize the full capacity of its Internet connection. If the super node makes all of its bandwidth available to others then its utility is zero.
- **$\mathbf{TS_{Available}}$(FN):** This represents the available transmission speed for a *friend* node, FN (i.e. a node with a *strong* relationship).

$$\mathrm{TS}_{\mathrm{Available}}(\mathrm{FN}) = \frac{TBW - (TWBW)}{NC + 1} \tag{3}$$

- **Utility of a Node**: This utility is measured based on the available transmission speed for transferring a file through its connected super node ($\mathrm{TS}_{\mathrm{Available}}$). The utility of a node is one, if a node gets access to 50% of the super node's link capacity for transferring its file. If the node does not get access to the super node's bandwidth for transferring its file, the utility is zero.
- **$\mathbf{BW_{MN}}$:** This parameter specifies the minimum available bandwidth reserved for a super node's own use. This parameter guarantees that the experienced performance of a super node is not less than expected. In particular, the condition is as follows:

$$IF \quad \left(\frac{TBW - (TWBW)}{NC + 1} \geq BW_{MIN}(SN)\right) \quad THEN$$

$$\mathrm{BW}_{\mathrm{Available}}(\mathrm{SN}) = \frac{TBW - (TWBW)}{NC + 1}$$

$$\mathrm{TS}_{\mathrm{Available}}(\mathrm{FN}) = \frac{TBW - (TWBW)}{NC + 1} \tag{4}$$

ELSE

Put_the_query_for_transferring_the_file_into_the_
waiting_list_until_there_is_enough_bandwidth.

4.3 Simulation Results

To illustrate the effectiveness of the algorithm, we show how a super node is affected by the number of nodes in the contact list. The parameter settings that we used, i.e. the number of users in the contact list and their strength of relationship to the super node, are shown in Table 1.

Table 1. Average Available Bandwidth for Different Scenarios

Scenarios	Number of Contacts in the Contact list			Average Available Bandwidth (bps)
	Total	Strong	Weak	
1	2	1	1	1370612.922
2	4	2	2	1279330.177
3	8	4	4	0423674.100
4	16	8	8	0140887.642
5	32	16	16	0107763.740

The super node utility is calculated based on the average available bandwidth for each of the scenarios, as shown in Table 1. For each scenario, the link capacity of the super node is 1.544 mbps with 5% of the capacity always reserved for the super node's own consumption. The utility curve in Fig. 5 indicates, as expected, that, as the number of nodes in the contact list of a super node increases, the utility of the super node degrades as well. However, because of the reservation of 5% of the link capacity, the utility does not go below 0.05.

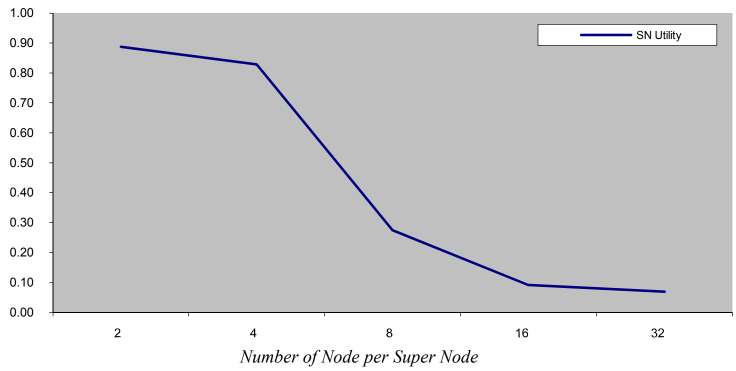

Figure 5. Super node utility for different number of contacts

Although, as Fig. 5 shows, the utility of super nodes decreases as bandwidth available for their own use decreases, the new social-network based algorithm (SocialGnutella) can avoid performance degradation of super nodes by limiting the bandwidth available for relay service. Otherwise, it could happen that a large number of NATed contacts consume up the entire link capacity available to the super node.

4.4 Comparison of Network Topology Formation Algorithms

Fig. 6 and Fig. 7 illustrate how the utility of super nodes and nodes compare for different network topology formation algorithms (i.e. Gnutella v0.6, Skype, and SocialGnutella). As the graphs indicate, the super node utility is highest for the Skype topology algorithm, while the utility of nodes for Skype is lower than the utility for the other two algorithms. The Gnutella nodes utility is highest and the super node utility is lowest when compared with the other two topology algorithms.

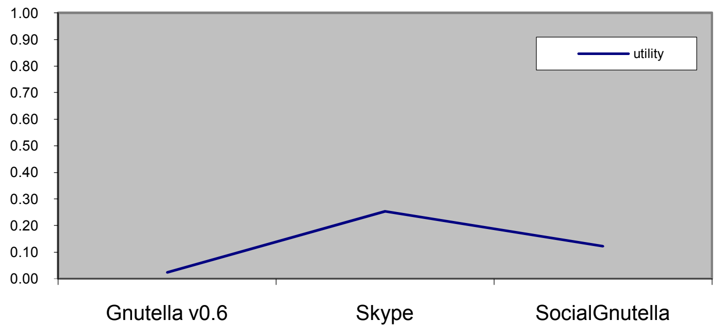

Figure 6. Utility of super nodes for different topology formation algorithms

As Fig. 6 and Fig.7 indicate, Gnutella maximize the utility of nodes without considering the utility of super node, leading users not to share their resources. Skype maximize the super node utility by limiting the nodes transmission speed to 5 kbps. However, this discourages sharing of resources between peers. The new topology algorithm, SocialGnutella, achieves to optimize the utilization of resources from the perspective of super nodes and regular nodes. This keeps super nodes sharing their resources with others, while maintaining their own benefit at the declared minimum level.

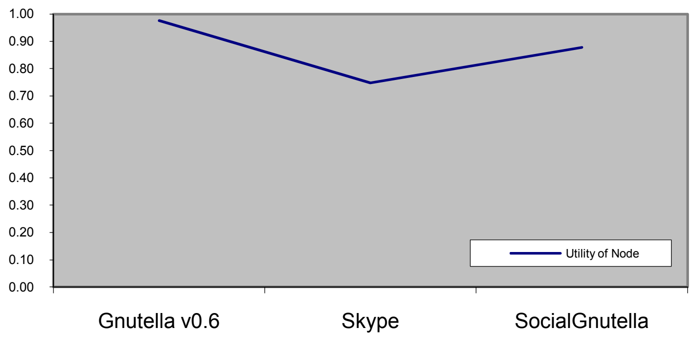

Figure 7. Total utility of nodes in different Topology algorithms

However, in the new topology formation algorithm, there will be more overhead for super nodes than for super nodes in Skype. Skype limits file transaction speed to 5 kbps for all users. In SocialGnutella, the transmission speed depends on the relationship between nodes and their super nodes.

5. Conclusion and Recommendations

It can be concluded that the new P2P network topology formation algorithm, SocialGnutella, presents a solution to the free-riding issue and has the potential to reduce the performance inefficiency of P2P systems, which is caused by NATs and free-riding. The new topology algorithm encourages resource sharing between peers by using a social-network-based relay service. The motivation for contributing resources comes from the already existing social relationship between peers. Free-riders are discouraged since they do not get better than basic performance in those systems. Since the network is established based on social relationships, better service is only possible if a strong relationship exists between the node and the super node. In this case, the super node (i.e. friend) is willing to carry the cost.

In addition to this, through the implementation of social network information, the anonymity, which free riders require, is not present anymore within SocialGnutella. If free-riders want to receive better-than-basic service, they have to reveal their social relationship to other peers.

The analysis of the simulation results confirms that the new P2P network topology algorithm is effective. Without performance degradation, it allows super nodes to give relay service based on the strength of the social relationship to its contacts.

Given the findings of this research, the following statements can be made: P2P systems can benefit by introducing the new, social-network-based topology formation algorithm into their system. In particular, in societies, where there is a high number of social interactions, there will be a conductive environment for the application of the new concept.

6. References

[1] E. Adar and B. Huberman, "Free riding on Gnutella," FirstMonday, vol.5, no.10, October 2000.

[2] J. Saarinen, "Skype Supernodes Sap Bandwidth," Computerworld, Auckland, New Zealand, October 2005.

[3] N. Andrade, F. Brasileiro, W. Cirne, and M. Mowbray, "Discouraging Free Riding in a Peer-to-Peer CPU-Sharing Grid," HPDC2004, 13th IEEE International Symposium on High Performance Distributed Computing, IEEE Computer Society, pp.129-137, 2004.

[4] R. Brunner, "A Performance Evaluation of the Kad-Protocol," Institut Eurecom, France, 2006.

[5] D. Eichhorn, "A Peer-to-Peer Network Framework with Network Address Translation Traversal," Master Thesis, Department of Informatics, IFI, University of Zurich, May 2006.

[6] Eyeball, "AnyFirewall Server," [cited 08.04.09], available from http://www.eyeball.com/technology/whitepapers/EyeballAnyfirewallWhitePaper.pdf , 2008.

[7] S. Handurukande, A. Kermarrec, F. Le Fessant, L. Massoulie, and S. Patarin, "Peer Sharing Behaviour in the eDonkey Network, and Implications for the Design of Serverless File Sharing Systems," Eurosys2006, EuroSys Conference, ACM SIGOPS Operating System Review, vol.40, no.4, pp.359-371, Leuven, Belgium, October 2006.

[8] D. Hughes, G. Coulson, and J. Walkerdine, "Free Riding on Gnutella Revisited: The Bell Tolls?" IEEE Distributed Systems Online, vol.6, no.6, June 2005.

[9] Z. Hu, "NAT Traversal Techniques and Peer-to-Peer Applications," Telecommunications Software and Multimedia Laboratory, Helsinki University of Technology.

[10] T. Idris and J. Altmann, "A Market-Managed Topology Formation Algorithm for Peer-to-Peer File Sharing Networks," ICQT2006, pp.61-77, LNCS, St. Malo, France, June 2006.

[11] S. Koolen, "Creating and Maintaining Relationships in Social Peer-to-Peer Networks," Delft University of Technology, 2007.

[12] S. ManLui and S. HoKwok, "Interoperability of Peer-To-Peer File Sharing Protocols,"

[13] B. Mitchell, "Introduction to Peer to Peer Networks," [cited 16.06.08], available from http://compnetworking.about.com/od/ basicnetworkingfaqs/a/peer-to-peer.htm, 2008.

[14] Peer-to-Peer File-Sharing Technology:Consumer Protectionand Competition Issues, June 2005

[15] Practically Networked, [cited 25.03.2008], available from http://www.practicallynetworked.com/sharing/firewall.htm, 2008.

[16] V. Vishnumurthy, S. Chandrakumar, E. G. Sirer, “KARMA: A Secure Economic Framework for P2P Resource Sharing,” Workshop on the Economics of Peer-to- Peer Systems, Berkeley, USA, June 2003.

[17] Wikipedia, "Network Address Translation,” [cited 25.06.2008], available from http://en.wikipedia.org/wiki/ Network_address_translation, 2008.

[18] S. Leinhardt, “Social Networks – A Developing Paradigm,” Academic Press, London, UK, 1977.

[19] J. Altmann, F. Zifa, and Z. Bedane, “Limiting Free-Riding in P2P Networks Through Social Network Information,” unpublished.

[20] R. Poe, “The Great Disappearing University Skype Ban,” [cited 08.04.2009], available from http://www.voip-news.com/feature/sjsu-skype-disappearing-ban-101706/, October 2006.

[21] M. Zghaibeh and K. G. Anagnostakis, “On the Impact of P2P Incentive Mechanisms on User Behavior,” NetEcon+IBC2007, Workshop on the Economics of Networked Systems and Incentive-Based Computing, San Diego, California, USA, June 2007.

[22] J. A. Pouwelse, P. Garbacki, J. Wang, A. Bakker, J. Yang, A. Iosup, D. H. J. Epema, M. Reinders, M. van Steen, H. J. Sips, “TRIBLER: A Social-Based Peer-to-Peer System,” vol.20, no.2, pp.127-128, February 2008.

[23] F. Zifa, “Using Social Network Information to Improve P2P the Performance of P2P Networks,” Master Thesis, Seoul National University, February 2008.

TEMEP Discussion Papers

2009-36: Jörn Altmann and Juthasit Rohitratana, “Software Resource Management Considering the Interrelation between Explicit Cost, Energy Consumption, and Implicit Cost: A Decision Support Model for IT Managers”

2009-37: Marcel Risch, Ivona Brandic, and Jörn Altmann, “Using SLA Mapping to Increase Market Liquidity”

2009-38: Amit K. Bhandari and Almas Heshmati, “Willingness to Pay for Biodiversity Conservation”

2010-39: Marcel Risch, Jörn Altmann, Li Guo, Alan Fleming, and Costas Courcoubetis, “The GridEcon Platform: A Business Scenario Testbed for Commercial Cloud Services”

2010-40: Junseok Hwang, Jörn Altmann, and Ashraf Bany Mohammed, “Determinants of Participation in Global Volunteer Grids: A Cross-Country Analysis”

2010-41: Ruzana Davoyan, Jörn Altmann, and Wolfgang Effelsberg, “Intercarrier Compensation in Unilateral and Bilateral Arrangements”

2010-42: Ruzana Davoyan, Jörn Altmann, and Wolfgang Effelsberg, “Exploring the Effect of Traffic Differentiation on Interconnection Cost Sharing”

2010-43: Ruzana Davoyan and Jörn Altmann, “Investigating the Role of a Transmission Initiator in Private Peering Arrangements”

2010-44: Ruzana Davoyan, Jörn Altmann, and Wolfgang Effelsberg, “A New Bilateral Arrangement between Interconnected Providers”

2010-45: Marcel Risch and Jörn Altmann, “Capacity Planning in Economic Grid Markets”

2010-46: Dang Minh Quan and Jörn Altmann, “Grid Business Models for Brokers Executing SLA-Based Workflows”

2010-47: Dang Minh Quan, Jörn Altmann, and Laurence T. Yang, “Error Recovery for SLA-Based Workflows within the Business Grid”

2010-48: Jörn Altmann, Alireza Abbasi, and Junseok Hwang, “Evaluating the Productivity of Researchers and their Communities: The RP-Index and the CP-Index”

2010-49: Jörn Altmann and Zelalem Berhanu Bedane, “A P2P File Sharing Network Topology Formation Algorithm Based on Social Network Information”