Journal of Korea Multimedia Society Vol. 11, No. 12, DECEMBER 2008(pp. 1706-1715)

Traffic based Estimation of Optimal Number of Super-peers in Clustered P2P Environments

Jugyun Kim[†], Junsoo Lee[††]

ABSTRACT

In a super-peer based P2P network, the network is clustered and each cluster is managed by a special peer, which is called a super-peer. A Super-peer has information of all the peers in its cluster. This type of clustered P2P model is known to have efficient information search and less traffic load than unclustered P2P model. In this paper, we compute the message traffic cost incurred by peers' query, join and update actions within a cluster as well as between the clusters. With these values, we estimate the optimal number of super-peers that minimizes the traffic cost for the various size of super-peer based P2P networks.

Key words: P2P network, super-peer, cluster, traffic cost

1. INTRODUCTION

P2P(Peer-to-Peer) network is a distributed system in which the nodes called peers share part of their resources and act as clients and servers at the same time[1]. In a traditional client-server network, all services requested from clients are processed by a centralized server. However, this causes a lack of certain qualities of service if the performance of server is not good enough or receives too many service requests in a short period of time. Many P2P network models have been proposed to improve the inefficiency of these centralized server models. Currently, P2P network is one of the most important Internet service elements because it shares storages, processors, and media contents that reside in each peer[2].

One of the initial P2P network model is Napster, in which each peer connects to the central server, which manages the users and files of the whole network. The main difference between traditional client-server networks and Napster is that a server in Napster does not maintain the file itself, but the index of users and files only so that when it receives a query, it searches the index and helps clients connect directly to the peer[2,3].

Unlike the P2P networks with a central server, distributed P2P networks do not maintain the central server. Distributed P2P networks are classified into two categories: unstructured and structured [4]. Unstructured P2P network such as Gnutella [5] and Freenet [6] connect each peer in pyramid architecture and share data files unlimitedly without the central server. However, this type of P2P network must send data search requests to all the peers connected to this network. This broadcast causes extra duplicate traffic and completeness problem when the data search requests occur [7]. To solve these problems, the structured P2P networks incorporate structured property, and one of the examples is the DHT(Distributed Hash Table) based P2P networks [8-12]. However, unless the

※ Corresponding Author : Jugyun Kim, Address : Seoul, Yongsan-ku, chungpa-dong 53-12(140-742), TEL : +82-2-710-9430, FAX : +82-2-710-9296, E-mail : jgkim@sookmyung. ac.kr
Receipt date : July 15, 2008, Approval date : Nov. 7, 2008
† Computer Science Department, Sookmyung Women's Univ.
†† Computer Science Department, Sookmyung Women's Univ.
(E-mail : jslee@sookmyung.ac.kr)
※ This research was supported under sookmyung women's university research grants 2007

key value of data is completely known, searching in DHT P2P network is often difficult. Another problem is that it does not consider the physical structure of the network.

A super-peer based P2P network is another type of structured P2P network which improves the scalability of DHT based P2P network. It divides a network into clusters, and in each cluster there is one special peer, called a super-peer, which maintains information on each client peer [13,14]. Network traffic may decrease and searching is efficient in this type of network because queries are sent only to a small number of super-peers. While super-peer based P2P network has several advantages overall, efficiency depends on the number of super-peer in the network. If there is large number of super-peers, the traffic inside the cluster may decrease because each cluster has small number of members. However, message traffic between super-peers increase because each super-peer only maintains a small number of members.

This paper studies the traffic cost by varying the number of super-peers in the super-peer based P2P network and estimates the optimal number of super-peers according to the network size. Although computing traffic cost in the network is complex and should consider many different factors such as propagation delay, method of switching, and bandwidth, this paper assumes traffic cost as the quantity of message, which is obtained by multiplying the number of messages and the size of messages. Other factors will be considered in the future work.

Traffic costs based on the quantity of messages are composed of two parts. One is the case where a request from a peer is satisfied in its cluster. The other case is that a request from the peer is not satisfied in its cluster. Based on the total quantity of messages, the traffic cost is computed and we estimate the number of clusters in the network. In order to develop numerical model, we assume clusters of uniform size. However, the simulation results show that the traffic cost depends on the number of clusters not on the variation of size. As a result, we concluded that when one forms a network cluster, the sizes of individual clusters are not important.

The remainder of the paper is organized as follows. Section 2 describes related work focused on the structure and operating principles of the super-peer based P2P network. Section 3 presents the total traffic cost in a numerical equation and estimates the optimal number of super-peers. These values are compared against the results obtained from the simulation done under various cluster sizes. Section 4 shows the analysis results derived from the numerical model. Then, section 5 concludes and presents future works.

2. RELATED WORK

The DHT based P2P network applies a general hash table to the overlay network which is usually generated from the application layer and does not depend on the physical layout of the network [8]. Each peer stores part of the <key, value> pair and the whole network is represented as a big hash table. The key part in the pair is generated from a hash function applied to a file name and the value part with the information on the peer which owns the file.

One of the DHT based algorithm is Pastry [9]. Pastry is an algorithm that routes packets using the prefix of a peer's ID. Another DHT based algorithm is Tapestry [10] which routes packets using the suffix of a node's ID. Other DHT based algorithms are CAN [11], Chord [12], etc. The characteristics of these networks are guaranteeing O(logN) hop complexity if N nodes exist. Using the information table in each node about a network, the query is moving toward a final node which has the desired file key in O(logN) hops. Therefore, it incurs less overhead and provides a more accurate and fast search than the previous models [9-12].

However, DHT based P2P network models that search with the file's key value may fail unless the accurate file's key value is known. Another disadvantage is that a hop in the DHT based network can go through several underlying networks, but it does not take into account any overhead from these physical topologies [13].

Research on solving problems of DHT based P2P network by using super-peer based P2P network has been done by many researchers. Super-peer is a peer which maintains information of the client peers in a sub network called clusters. A super-peer exists for each cluster and maintains two tables. One table has the information of the client peers itself and shared files stored in peers. The other table includes a list of the super-peers in the whole network. A client peer must have the information of the super-peer it joins. When a client-peer joins a network, it registers to a super-peer, and then transfers the information on the files it wants to share.

One of the super-peer based P2P network is Grapes [13]. Grapes is a P2P network composed of nodes called leaders and sub network which is a collection of nodes physically adjacent to the super-peer. Grapes is faster than DHT based P2P network when it inserts, searches, and receives data in the network. However, Grapes only tracks execution time of each action, so it does not assess the change of network traffic in detail.

[14] proposed a super-peer network which reduces the excessive traffic existing in the unstructured P2P network. This shows various results obtained from the cost of traffic on completing search request and average number of hops to obtain final answer. [15] proposed a network model in which a search can be done in the O(1) time complexity. This method clusters Chord, which is a DHT based P2P network, and each super-peer that maintains a cluster exchanges messages while each connects with the other with ring type topology [15]. develops mathematical model to compute memory and traffic amounts for a super-peer to maintain clusters and search for data. The simulation in this paper also shows that their analytical model matches with their simulation results. Papers mentioned in this section are related with our work studying cost efficiency of the super-peer based P2P networks.

3. Traffic cost under the super-peer P2P network

Network traffic in the super-peer based P2P network is composed of three operations: Join, Update, and Query. Whenever a new peer joins a network, it must register to one of the super-peers in the following procedure. First, the new peer contacts a nearby existing peer, which sends information on the super-peer to the new peer. Using this information, the new peer sends its register request message to the super-peer. Then, the super-peer assigns ID to the new peer and stores it in the peer list table. Lastly, the super-peer sends its register completion message to the new peer. At this point, if the new peer wants to register files, then the super-peer processes this request [15,16].

In an update operation, the client-peer and super-peer check each other's status periodically. The status check message in this case is to confirm existence of the other party in the network [15].

In order to search for a file, a client-peer sends a file query request to the super-peer which registered this client-peer. If the super-peer knows where the requested file is, then it sends a tuple <filename, peer ID, peer address> back to the client. However, if the super-peer does not have the information on file, it sends query to other super-peers. If the other super-peers have the information on the requested file, they send back the answer to the original super-peer [15,16].

The total traffic cost can be obtained by adding two costs shown in below. The first cost is an amount of messages between the client-peer and the super-peer due to join, update, and query operation. The second cost is the amount of messages generated between super-peers when a super-peer receives a search request which cannot be answered by this super-peer.

$$Ctotal = Ccluster + Csuper$$

3.1 Numerical model

The basic parameters used in this paper are based on [13], and our assumptions in this paper are as follows.

- If each cluster contains n client-peers, then the super-peer has n entries in its file list table.
- The number of peers in each cluster is the same. In other words, the size of all the clusters is the same.
- There is no client-peer which acts independently without the supervision of a super-peer.
- There should be one super-peer in a cluster. If the super-peer fails, one of the client nodes in the cluster should be elected as a super-peer. Thus, in this paper the number of super-peers and the number of clusters are used interchangeably.
- Information about other super-peers are composed only with the address.
- For a given size of network, we assume an equilibrium state, in which a rate of node registration and a rate of node withdrawal are the same.

3.1.1 Cost of Traffic in a Cluster

The followings are variables used to compute traffic cost in this paper. Some of the environment variables used in here can be found in [15] and [16] selectively for some possible solution.

- N : Network size (= Total number of peers in the network)
- S : Number of super-peers (= Number of clusters).

And by above assumption, each cluster size is defined by N/S.

- *Ccluster* : Traffic cost in a cluster
- *Csuper* : Traffic cost between super-peers
- *Cq, Cj, Cu* : Costs for Query, Join, Update.
- *CQsend, CJsend, CUsend* : Quantities for sending message by Query, Join, Update.
- *CQrev, CJrev, CUrev* : Quantities for receiving message by Query, Join, Update.
- *Qrate, Jrate, Urate* : Rates of Query, Join, Update in a cluster / unit time (=sec).
- *Qpacket, Jpacket, Upacket* : Packet length for sending message by Query, Join, Update.
- *Qrec, Jrec, Urec* : Packet length for receiving message by Query, Join, Update.

Operations in a cluster to compute traffic cost are composed of Join, Query and Update. Thus, the intra cluster cost is the sum of the cost incurred from Join, Query, and Update operations.

$$Ccluster = Cq + Cj + Cu \tag{1}$$

Cq in Equation (1) can be expressed as follow.

$$Cq = CQsend + CQrev = Qrate \cdot Qpacket + Qrate \cdot Qrec \tag{2}$$

If a super-peer has the information of the file requested from a peer, it answers back to the client immediately. Otherwise, the super-peer sends the request to the other super-peers. Message traffic between super-peers is a cost outside of clusters, but once super-peer receives a response from another super-peer, it forwards responses to the client-peer. Therefore, the super-peer sends a response to the client at the rate of the receiving search query. If the query occurs at a rate of q during the unit time from the peer, the average query rate for each cluster becomes $Qrate = q \cdot (N/S)$ and Cq are equal to the equation (3). Equation (5) indicates expression for update cost, in which each super-peer sends update messages to the peers in the cluster and then receives the reply from the peer.

$$Cq = q \cdot (N/S) \cdot Qpacket + q \cdot (N/S) \cdot Qrec \tag{3}$$

$$Cj = CJsend + CJrev = Jrate \cdot Jpacket + Jrate \cdot Jrec \quad (4)$$

$$Cu = (N/S) \cdot (CUsend + CUrev) = (N/S) \cdot (Urate \cdot Upacket + Urate \cdot Urec) \quad (5)$$

Equation (4) and (5) can be expressed as (6) and (7) respectively because the size of *Jpacket* and *Jrec* ,and the size of *Upacket* and *Urec* are same as in <Table 1>.

$$Cj = 2 \cdot Jrate \cdot Jpacket \quad (6)$$

$$Cu = 2 \cdot (N/S) \cdot Urate \cdot Upacket \quad (7)$$

Therefore, if we develop (1) using (3), (6), (7), the traffic cost in clusters can be expressed as in (8).

$$Ccluster = q \cdot (N/S) \cdot Qpacket + q \cdot (N/S) \cdot Qrec + 2 \cdot Jrate \cdot Jpacket + 2 \cdot (N/S) \cdot Urate \cdot Upacket \quad (8)$$

3.1.2 Traffic cost between clusters

We assume one super-peer in each cluster, thus traffic cost between clusters is the same as the amount of messages generated between the super-peers for searching and update information. Messages between super-peers are generated when super-peers do not have the information on the requested file, so we need to consider the following probability.

- Pr[Qsuc] : Probability for a client-peer to find a file in its cluster

If every peer in the network shares more than one file, a cluster with a larger number of peers may have higher probability of success when it searches for a file because its super-peer maintains a larger amount of information on files. This probability increases in proportional to the number of peers, but we assume in this paper that every cluster has the same number of peers. Therefore, we compute Pr[Qsuc] and Pr[Qfail] as follows. For reference, each super-peer has a file which contains a peer list and this list grows proportionally to the number of peers.

Pr[Qsuc] = file size owned by one super peer / sum of the size of files owned by super-peers = (N/S) / N = 1/S

Pr[Qfail] = 1 - Pr[Qsuc]

The traffic cost between super-peers equals the sum of the cost of query requests from one super-peer to other super-peers due to query failure, the cost of reply, and the cost of update. Equation (9) shows the traffic cost between super-peers.

$$Csuper = Qrate \cdot Pr[Qfail] \cdot [Qpacket \cdot (S-1) + Qrec \cdot (S-1)] + 2 \cdot Urate \cdot Upacket \cdot (S-1) = q \cdot (N/S) \cdot (S-1)/S \cdot [Qpacket \cdot (S-1) + Qrec \cdot (S-1)] + 2 \cdot Urate \cdot Upacket \cdot (S-1) \quad (9)$$

3.1.3 Optimal Number of super-peers based on the quantity of message

The total traffic cost under the variation of super-peers can be obtained from the traffic cost between clusters and within a cluster as mentioned in the previous section. In this paper, we only consider the quantity of message, and the memory cost for the client node table in the super-peer or the table search costs for the query will be considered in the future work.

Optimal number of super-peers can be obtained from the expression of total traffic equal to the sum of a traffic cost in a cluster and the traffic cost between super-peers as (10).

$$Ctotal = Ccluster + Csuper$$

Table 1. Message size for each action

Action	Message Cost(Bytes)
Send query	82+query length
Send Response	80+28*address+76*number of result
Send Join	80+72*number of files
Receive Join	80+72*number of files
Send Update	152
Receive Update	152

$$= [q \cdot (N/S) \cdot Qpacket + q \cdot (N/S) \cdot Qrec + 2 \cdot Jrate \cdot Jpacket + 2 \cdot (N/S) \cdot Urate \cdot Upacket] + [q \cdot (N/S)(S-1)/S \cdot \{Qpacket \cdot (S-1) + Qrec \cdot (S-1)\} + 2 \cdot Urate \cdot Upacket \cdot (S-1)] \quad (10)$$

The minimum value of S satisfying expression (10) becomes the optimum number of super-peers. This value S is the same as the value S which is satisfying expression (11).

$$\frac{dCtotal}{dS} = 2 \cdot U_{rate} \cdot U_{packet} \cdot S^3 + (q \cdot N \cdot Q_{packet} \cdot Q_{rec} - 2 \cdot U_{rate} \cdot U_{packet} \cdot N) \cdot S - 2 \cdot q \cdot N \cdot Q_{packet} \cdot Q_{rec} = 0 \quad (11)$$

Because answers for the expression (11) can not be obtained by solving the equation directly, we should find an approximation value by applying the numerical method. <Table 2> shows minimum S values according to the various network sizes, and also corresponding graphs are shown in the next section.

Table 2. Optimal super-peer number for various rates

Network size(N)	Query rate(q)	Update rate ($Urate$)	Optimal super-peer number
1000(10^3)	0.005	0.1	29
		0.5	31
		1	31
	0.01	0.1	25
		0.5	30
		1	31
	0.05	0.1	4
		0.5	26
		1	29
10000(10^4)	0.005	0.1	90
		0.5	98
		1	99
	0.01	0.1	79
		0.5	96
		1	98
	0.05	0.1	4
		0.5	79
		1	90
100000(10^5)	0.005	0.1	285
		0.5	310
		1	313
	0.01	0.1	250
		0.5	304
		1	310
	0.05	0.1	4
		0.5	250
		1	285

3.1.4 Analysis of the traffic cost using numerical model

Based on the numerical expression in section 3.1, we apply real parameters to compute the traffic cost and analyze the result of the traffic cost variation when the network size and the number of super-peer vary. The amount of messages generated from the super-peer and the client-peer is shown in table 1. The values used here are referenced from the similar or same environment mentioned in the related works.

Send query is an action when you search for a file. The packet size of send query is the sum of the base packet length, which is 82 bytes, and length of query for searching for a file. Send Response is the reply from the query. Length of Send Response is the sum of the base packet length of 80 bytes, 28 times the address of a peer who has the requested file, and 76 times the number of result as in the <Table 1>. Send Join is an action when a new node arrives in the network, which sends packet to the super-peer. Its packet length is the sum of the base packet of 80 bytes and 72 times number of files it wants to share. Receive Join is an action when new node joins the network, and packet length of Receive Join is the same as that of Send Join. Send Update is an action that the super-peer checks connection status with the client-peers or other super-peers, and its packet length is 152 bytes.

Among the cost of each action, the query length is set to 12 in <table 1> by considering the average length, appeared in [13], and the address length is set to 32. The number of results and number of files is set to 1 with assumption that there exists only one copy in the network. As we have seen in expression (11), *Jrate* value is fixed as 1.0 since it does not affect the result. Applying the values in [15] such as q value to 0.005, *Jrate* to 1.0, and *Urate* to 0.5 in a network with 10^3, we obtain the graph in (Figure 1). As a reference, <Table 2> shows the minimum number of super-peers while varying q and *Urate* values in the size of 10^4 and 10^5 networks.

(Figure 1) shows the variation of total traffic, *Ccluster*, and *Csuper* cost in a network which has 1000 peers. The point where total cost is the minimum is the same as the optimal number of clusters. This point is equal to the optimal number of super-peers. When the number of super-peers increases the cost in a cluster decreases as shown in this figure. This is because if the number of super-peer increases, then the number of client-peer per cluster decreases. This means there is less message traffic in each cluster, but more message traffic cost between clusters.

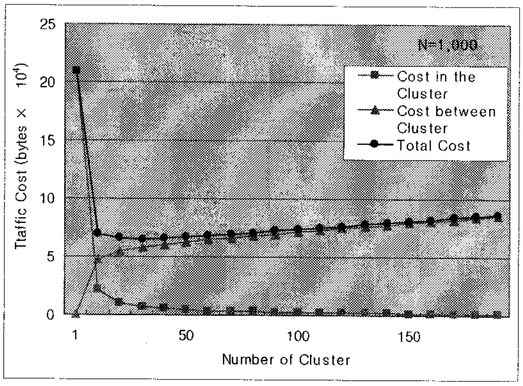

Fig. 1. Optimal number of cluster (N=1,000)

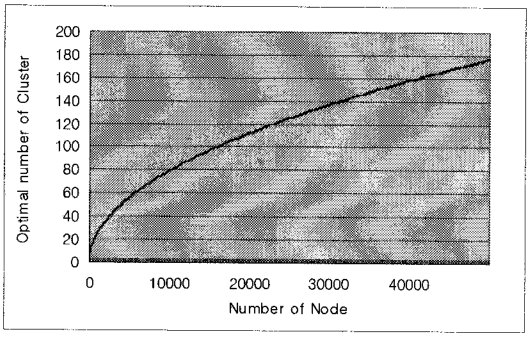

Fig. 2. Optimal cluster numbers under various network sizes

(Figure 2) shows the number of super-peers increase proportional to the network size. With this figure, the optimal number of super-peers can be easily estimated.

3.2 Simulation of various size clustering

We assumed that the number of client-peers maintained by each super-peer is the same for each cluster to develop a simple numerical model in section 3.1. However, in reality, there are various sizes of clusters in a network. To verify our model in section 3.1, we simulate various size clusterings and show that the total traffic cost depends on the number of clusters instead of the number of peers in a cluster.

As we showed in the expression of the section 3.1, one peer communicates with one super-peer. Thus our traffic model based on the amount of the messages depends on the rate of the message which increases proportionally when the total peer N increases in a network. In other words, the minimum cost is obtained not by dividing the network with various sizes of clusters but by the number of total peers. This result is verified by the simulation in this section.

The total number of peers is based on the 10^3 network as in the numerical model, and we applied 5 different models by using random number generation to assign peers to each cluster. For example, if N is 1000 and number of clusters is 10, then in the numerical model all clusters are the same size of 100. In a simulation, 10 clusters are generated with 5 different forms by applying the random number generation. This expresses 10 clusters with small size to large size. We applied the larger number of forms, but the result was similar.

Each peer assigned a unique number between 1 and the size of network so that we can identify mapping between a peer and a cluster. Also

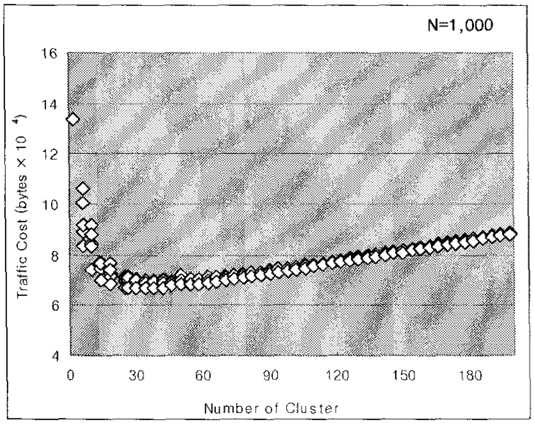

Fig. 3. Traffic cost under different number of cluster (N=1,000)

clusters are identified with unique number so that we can identify which peer is inside of which cluster.

To simulate query, we create two random values at the rate of the query, and if these two random values are in one cluster, then we consider this as an intra cluster cost. If these two values are different, it is considered as an inter cluster cost which reflects *Pr[Qfail]*.

(Figure 3) is a graph generated with the parameters in <Table 1> where, *q*, *Jrate*, and *Urate* is set with 0.005, 1.0, and 0.5, respectively. This shows cost per unit time by counting message and dividing the amount of messages by elapsed time. For each point on X-axis (number of clusters), there are 5 points which indicate each traffic cost. These represents the optimal traffic costs of each of the 5 models mentioned in the above under the same number of clusters and these values overlap more when the number of cluster becomes large.

Our numerical result in <table 2> shows the optimal number of super-peers is 31 when N equals 1000. The simulation result in (Figure 3) also indicates the similar value with our numerical result when N equals 1000. In addition, the comparison of results of numerical model and simulation matches when network size is 10^4 and 10^5.

4. ANALYSIS

There are many factors that affect network traffic cost. We develop numerical model which estimates traffic cost based upon the amount of message traffic in the previous section. Although we do not consider bandwidth, switching latency and the propagation delay because of the complexity developing numerical model, we believe that these can be considered in the future work. We also ran several simulations of various size of clustering and showed that these results confirm with analytical study. We concluded that computing the optimized traffic cost in a given network does not depend on the number of peers for each cluster, but the number of clusters. From the result of this, if the network size is given, and we apply our numerical model, we can infer optimized number of cluster.

If we look at the values in <Table 2> carefully, when the update rate increases, the number of clusters increases. If the update rate is the same, the number of clusters increases when the query rate decreases. When the network size varies, the optimal number of cluster increases as in the (Figure 2). This shows that the traffic cost which depends on the number of network cluster is sensitive especially when the number of network size is relatively small.

5. CONCLUSION

Important current research topic in P2P network is providing structured characteristic to the network for the fast and accurate query so that it incurs less network overhead. One of the structured P2P networks is super-peer based P2P network. This model divides the network into sub network called cluster, and in each cluster, a super peer which communicates with each other, maintains client peers. In this model, total traffic cost is determined by the number of super-peers.

This paper infers the optimal number of super-peers under the various size of network based on super-peer network. Experimental results show that when a size of network increases, traffic cost also have different values. Using numerical expression, we also found that generation rate, and values of a query affect the results. Moreover, simulation studies indicate that under a given network, minimum traffic cost does not depend on the number of peers per cluster but the number of the clusters. Conclusively, we can estimate optimal number of super-peers when the network size is varied by reflecting parameters considering Query, Join, Update.

Direction of future research is as follows. One may extend this numerical model by incorporating additional factors affecting cost computation such as propagation delay or processing cost, and simulation study might be followed.

REFERENCES

[1] R.Schollmeier. "A definition of peer-to-peer networking for the classification of peer-to-per architectures and applications," Proc.IEEE Conference P2P 2001, Linkoping Sweden. Aug. 2001.

[2] D.S. Milojicic, "Peer-to-Peer Computing," HP Technical Report, HP Labora-to-ries, Mar. 2002.

[3] Napster. http://www.napster.com

[4] D. Tsoumakos and N. Roussopoulos, "Adaptive Probabilistic Search for Peer-to-Peer Networks," Proc. of the 3rd IEEE International Conference on P2P Computing, Sep. 2003.

[5] Gnutella, www.gnutella.com

[6] I.Clarke, O. Sandberg, B.Wiley, and T, Hong. "Freenet: A Distributed Anonymous Information Storage and Retrieval System," Lecture Notes in Computer Science, 2009:46-66, 2001.

[7] S.Androutsellis-Theotokis, "A survey of peer-to-peer file sharing technologies," Technical Report WHP-2002-03, Athens Univ.of Economics and Business, 2002.

[8] U. Wieder, M.Dahlin, "A Simple Fault tolerant distributed Hash Table," IPTPS 2003, Berkeley CA, Feb. 2003.

[9] A. Rowstron and P. Druschel, "Pastry: Scalable, Distributed object location and routing for large-scale peer-to-peer systems," Proc. IFIP/ACM International Conference on Distributed Systems Platforms, Nov. 2001.

[10] B.Zhao, J.Kubiatowicz, and A.Joseph, "Tapestry: An infrastructure for fault-tolerant wide-area loation and routing," Technical Report UCB/CSD-01-1141, Computer Science Division, Univ. of California, Berkeley, Apr. 2001.

[11] S.Ratnasamy, P.Francis, M.Handley, R.Karp, and S.Shenker, "A Scalable content-addressable network," Proc. of ACM SIGCOMM, 2001.

[12] I.Stoca, R.Morris, D. Karger, F.Kaashoer and H. Balakrishnan, "Chord: A scalable peer-to-per lookup service for Internet application," in Proc. ACM SIGCOMM 2001.

[13] Kwangwook Shin, Seunghak Lee, Geunhwi Lim, H.Yoon, Joong Soo Ma, "Grapes : Topology-based Hierachical Virtual Network for Peer-to-peer Lookup Services," In Proceeding of the International Conference Parallel Processing Workshops, 2002.

[14] B.Yang, H.Garcia-Molina, "Designing a Super-Peer Network," Proc. of IEEE International Conference on Distributed Computing Systems (ICDCS), 2002

[15] A.T.Mizrak, Y.Cheng, V.Kumar, and S. Savage "Structured Superpeers: Leveraging Heterogeneity to Provide Constant-Time Lookup," Computer Science and Engineering Division, Unv. of California, San Diego, 2003.

[16] S.Jain, R.Mahajan, D.Wetherall, and G. Borriello, "Scalable Self-Organizing Overlays," Computer Science and Engineering Division, Unv. of Washington, 2001.

Ju Gyun Kim

1985 Seoul National University, Computer Science and Statistics (BS)
1985~1986 DEC Korea
1988 Seoul National University, Computer Science and Statistics (MS)
1992 Seoul National University, Computer Science and Statistics (Ph.D)
1992~Current Professor, Sookmyung Women's University Computer Science Department
Research Interests : Operating System, Performance Evaluation, Caching Strategies, Flash Memory Technologies.

Junsoo Lee

1990 Seoul National University, Computer Science and Statistics (BS)
1994 University of Pennsylvania, Computer Science (MS)
1994~1999 Samsung Electronics (Senior Researcher)
1999~2005 University of Southern California, Computer Science (Ph.D)
2005~Current Assistant Professor, Sookmyung Women's University Computer Science Department
Research Interests : Network , Embedded system, Sensor Network, Operating System