

# Truncated Pyramid Peer-to-Peer Architecture with Vertical Tunneling Model

Zhonghong Ou<sup>1,2</sup>, Jiehan Zhou<sup>1</sup>, Erkki Harjula<sup>1</sup>, Mika Ylianttila<sup>1</sup>

<sup>1</sup>MediaTeam Oulu Group, Department of Electrical and Information Engineering  
Information Processing Laboratory, University of Oulu  
Oulu, Finland

<sup>2</sup>PCN&CAD Center, Department of Electronics Engineering  
Beijing University of Posts and Telecommunications  
Beijing, China

[tommy@ee.oulu.fi](mailto:tommy@ee.oulu.fi), [jiehan.zhou@ee.oulu.fi](mailto:jiehan.zhou@ee.oulu.fi), [erkki.harjula@ee.oulu.fi](mailto:erkki.harjula@ee.oulu.fi), [mika.ylianttila@ee.oulu.fi](mailto:mika.ylianttila@ee.oulu.fi)

**Abstract**—Peer-to-Peer (P2P) technologies have many advantages over traditional client/server technologies, including cost-effectiveness, scalability and robustness, due to their decentralized network structure. However, the performance has traditionally been an issue in P2P systems. Especially the higher lookup latencies, when compared with the traditional client/server systems, have been the bottleneck in many P2P systems. In this paper, we propose a truncated pyramid P2P architecture together with an enhanced model, Vertical Tunneling Model (VTM) for improving the lookup performance. The proposed architecture is built on Peer-to-Peer SIP (P2PSIP) network. VTM builds up vertical tunnels between the upper and lower sub-overlays to speed up the service lookup and decrease the session setup delay. Based on the performance analysis and results, it is shown that VTM has better performance compared with the existing systems in average lookup hops, which is about 1/3 of that of the existing systems, and the predominance is more evident as the network scale increases.

**Keywords**- *p2psip; overlay; distributed hash table; truncated pyramid architecture; vertical tunneling model*

## I. INTRODUCTION

Peer-to-Peer (P2P) technologies have attracted extensive attention of the academia, industry, and the media in the past few years because of the decentralized, no single-point-of-failure and scalable characteristics. Session Initiation Protocol (SIP)[1] gains the broad popularity in multimedia communication realm for its simplicity, flexibility and readability. Utilizing P2P paradigm to provide decentralized SIP services has raised a great deal of attention from the research and industry [2-4]. Internet Engineering Task Force (IETF) has formed a P2PSIP working group [5] for enabling the serverless use of SIP. However, alongside the advantages of the decentralized, server-free and scalable architecture, P2PSIP also has drawbacks.

Firstly, whereas the average session setup delay in conventional client/server SIP system is usually  $O(1)$  hops, in P2PSIP system it is  $O(\log N)$ ,  $O(\log_B N)$ ,  $O(dN^{1/d})$ , depending on the used Distributed Hash Table (DHT) algorithms [6-8]. Here,  $N$  stands for the total number of nodes in the system. Therefore decreasing the session setup delay has been one of the most critical design goals of P2PSIP system.

Secondly, node heterogeneity has not been considered adequately. The concept draft [9] of P2PSIP working group has some consideration on this issue, but it only partitions the nodes into peers and clients. To differentiate different peers, Le et al. presented a hierarchical and breathing P2PSIP system made up of more than two levels of sub-overlays [3]. In the un-oriented lookup, its minimum session setup latency is  $(\log N - k)$  hops, wherein  $N$  stands for the total number of nodes in the overlay and  $k$  stands for the number of sub-overlays. It uses Chord as an example to analyze the performance. Compared with the one level overlay, it decreases the session setup delay to some extent, but it still needs further optimization.

To solve these problems, we propose a Truncated Pyramid P2P (TPP) architecture, together with Vertical Tunneling Model (VTM) to accelerate the service lookup and decrease the network traffic for the overlay. The TPP has good scalability, can easily be scaled to multiple levels, but to make it clear and readable, this paper only considers a three levels architecture. The VTM builds up vertical tunnels between the upper and lower sub-overlays to speed up the service lookup and decrease the session setup delay.

The remainder of this paper is organized as follows. Section II introduces the TPP architecture, and VTM model. Section III analyzes the performance of TPP and VTM, and makes a comparison with the existing systems. Section IV presents the results with different network scales, Section V describes the future work and Section VI draws a conclusion.

## II. TPP ARCHITECTURE

In this section, the TPP architecture and the VTM model are introduced. In the concept draft [9] from IETF P2PSIP working group, there are all together two kinds of nodes in the reference model according to different functionalities: clients and peers. This paper does not discuss the functionalities of clients, just focus on the overlay made up of pure peers, no clients being included. The words “peer” and “node” are used interchangeably in this paper. There are also various mechanisms to classify  $N$  heterogeneous nodes into  $k$  sub-overlays [10]. We do not discuss the detailed mechanism for forming the  $k$  sub-overlays as well. Still there are a great number of DHT algorithms that can be used

---

The authors would like to thank TEKES, Nokia, Ericsson, Nethawk for financially supporting the research work in Decicom project.

to constitute the overlay. Chord [7] is chosen as an example to form the overlay in this paper. In reality, the TPP architecture and VTM are independent of DHT algorithms. They are general for P2P overlay and can be used in P2PSIP networks.

### A. Truncated Pyramid P2P (TPP) Architecture

In the TPP architecture, the network peers form multiple sub-overlays, the lowest sub-overlay has the most peers while the upmost one has the least peers, which makes it look like a pyramid being truncated. The upper sub-overlay peers duplicate all the resource indices and mapping relationships between, for instance, IP addresses and node Identifiers (IDs), for their lower sub-overlay peers whose keys fall into their key intervals.

As an example in Figure 1, resource indices and mapping relationships of nodes N1\_18 and N1\_23 from the first level of sub-overlay are stored in node N2\_22 from the second level of sub-overlay. In turn, indices of nodes N2\_22 and N2\_28 from the second level of sub-overlay are stored in node N3\_24 from the third level. Thus node N3\_24 stores the duplicate resource indices and mapping relationships of nodes N2\_22, N2\_28, N1\_18, N1\_23, N1\_27, and N1\_31, while node N2\_22 stores the duplicate copy of indices from nodes N1\_18 and N1\_23.

For the lookup procedure, take Figure 1 as the example, N1\_18 and N1\_23 constitute one SSO (denoted as S1\_3 in Figure 1) while N1\_27 and N1\_31 constitute another (S1\_4 in Figure 1), wherein N1\_18 is not able to detect the existence of N1\_27. If N1\_18 wants to lookup the service provided by N1\_27, it has to forward the service lookup request to N2\_22 at the second level of sub-overlay. Through the SSO located at the second level (S2\_2 in Figure 1), N2\_22 is able to find the location of N2\_28, through which the location of N1\_27 is able to be found. Then a response is sent back to N1\_18 from N1\_27. After that, they are able to set up a connection with each other directly. The basic principle of TPP is that service request is handled in the local SSO with a priority.

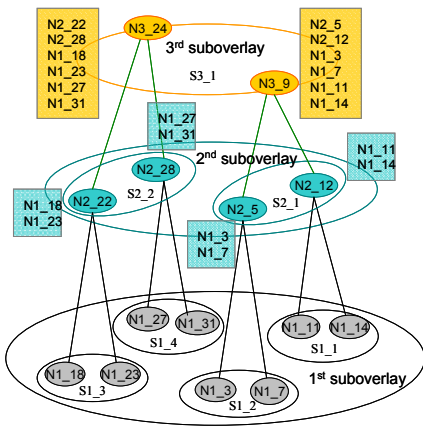


Figure 1. Truncated Pyramid P2P (TPP) Architecture

### B. Vertical Tunneling Model

To speed up the service lookup, Vertical Tunneling Model (VTM) is proposed in this section. Vertical tunnels are built up between upper and lower levels of sub-overlays to speed up the service lookup procedure.

As shown in Figure 2, VTM breaks up the normal TPP architecture to some extent. For the requests initiated from the first level of sub-overlay, two vertical tunnels are built up between the first and second sub-overlay, first and third sub-overlay respectively (the bold gray lines between N1\_18 and N2\_22, N1\_18 and N3\_24 in Figure 2). To make it clear, Figure 2 just shows two vertical tunnels. In reality, for each peer from the first level of sub-overlay, the same two vertical tunnels are built up between the first and second level of sub-overlays, first and third level of sub-overlays. For example, after receiving a lookup request, N1\_18 will lookup the service in the local SSO (S1\_3) with some probability. With the remaining probabilities, the request will be forwarded to the second and third level of sub-overlays directly along the vertical tunnels.

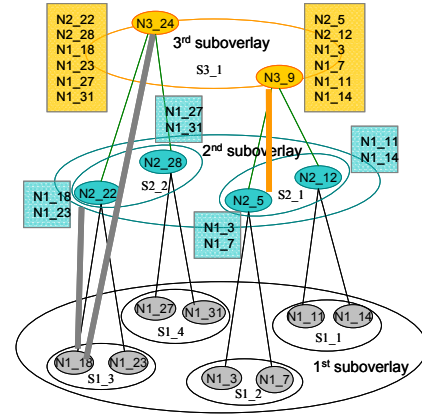


Figure 2. Vertical Tunneling Model (VTM)

For the request initiated from the second level of sub-overlay, one vertical tunnel is built up between the second and third sub-overlay, shown as the bold yellow line between N2\_5 and N3\_9 in Figure 2. After receiving a request, N2\_5 will lookup the service in the local SSO (S2\_1) with some probability, with the other probability, the request will be forwarded to S3\_1 directly.

For the request initiated from the third level of sub-overlay, as the peers from this level collectively store all the indices and mapping relationships of the whole system, the local SSO (S3\_1) is sufficient to find all the services in the system. In this way, the VTM can decrease the service lookup hops effectively through the vertical tunnels between the lower level and the upper level.

## III. PERFORMANCE ANALYSIS

In this section, a theoretical performance analysis of TPP and the VTM model is given. Firstly, we put forward some assumptions to make the performance analysis clearer.

(1) Each SSO from each level of sub-overlays has equally  $k$  nodes. The third sub-overlay has only one SSO, the second sub-overlay has  $k$  SSOs, while the first sub-overlay has  $k^2$  SSOs. Thus, the total number of nodes in the third sub-overlay is  $k$ , while the second sub-overlay  $k^2$ , and the first sub-overlay  $k^3$ .

(2) The service is distributed uniformly among all the peers in the system. Each node has the same probability to initiate a service request.

(3) The number of lookup hops for an overlay with  $k$  nodes is  $\log(k)$ .

The notations applied in the analysis are as follows:

The notations with '(T)' at the upper right corner stand for the counterparts in TPP, with '(V)' stands for the counterparts in VTM.

TABLE I. SYMBOLS AND MEANING

Symbol	Description
$N$	The total number of peers in the system equals to $k + k^2 + k^3$ .
$P_{ij}$	The probability of one lookup request which is terminated at the $j^{\text{th}}$ level of sub-overlay and initiated from the $i^{\text{th}}$ level of sub-overlay.
$\mathbf{P}$	The matrix made up of $P_{ij}$ .
$h_{ij}$	The number of lookup hops for one request which is terminated at the $j^{\text{th}}$ level of sub-overlay and initiated from the $i^{\text{th}}$ level of sub-overlay.
$\mathbf{H}$	The matrix made up of $h_{ij}$ .
$t_{ij}$	The probability of one lookup request being forwarded to the $j^{\text{th}}$ level of sub-overlay directly from the $i^{\text{th}}$ level of sub-overlay.
$\mathbf{T}$	The matrix made up of $t_{ij}$ .
$E(i)$	The average number of hops for one successful lookup initiated from the $i^{\text{th}}$ level of sub-overlay.
$Q(i)$	The probability of one node located at the $i^{\text{th}}$ level of sub-overlay.
$E_{\text{TPP}}$	The average number of hops for one successful lookup in TPP.
$E_{\text{VTM}}$	The average number of hops for one successful lookup in VTM.
$E_{\text{HA}}$	The average number of hops for one successful lookup in Hierarchical Architecture (HA) proposed in [3].
$E_{\text{FA}}$	The average number of hops for one successful lookup in Flat Architecture (FA) proposed in [7].

#### A. Average Number of Lookup Hops in TPP

Given the assumptions above, the following results exist for the lookup request initiated from the first level of sub-overlay:

$$\left\{ \begin{array}{l} p^{(T)}_{11} = \frac{k}{N} \\ p^{(T)}_{12} = \frac{k^2 + k - k}{N} = \frac{k^2}{N} \\ p^{(T)}_{13} = \frac{N - (k + k^2)}{N} = \frac{k^3}{N} \end{array} \right\} \quad (1)$$

Take Figure 1 as an example, and assume  $N1\_3$  wants to initiate a lookup request. Through one SSO ( $S1\_2$ ), it is able to find the locations of two peers in  $S1\_2$ . Through two SSOs ( $S1\_2$  and  $S2\_1$ ),  $N1\_3$  is able to find the locations of the peers from  $S1\_1$ ,  $S1\_2$  and  $S2\_1$ , that is  $2+2^2$  peers. But the peers from  $S1\_2$  can be found within only one SSO, so the number of peers which needs two SSOs to be found is  $2+2^2-2=2^2$ . This explains the items of  $-k$  and  $-(k+k^2)$  in (1).

Following the similar process, we can get the values of  $h^{(T)}_{ij}$ .

$$\left\{ \begin{array}{l} h^{(T)}_{11} = \log(k) \\ h^{(T)}_{12} = 2\log(k) + 1 \\ h^{(T)}_{13} = 3\log(k) + 2 \end{array} \right\} \quad (2)$$

The items of +1 and +2 stand for the forwarding hops. Take  $h^{(T)}_{12}$  as an example. It stands for the number of hops of one successful lookup through two SSOs for the request initiated from the first level of sub-overlay. Two SSOs require one forwarding hop from the first level of sub-overlay to the second level.

Similarly, the values of  $\mathbf{P}^{(T)}$  and  $\mathbf{H}^{(T)}$  are as follows.

$$\mathbf{P}^{(T)} = \begin{pmatrix} \frac{k}{N} & \frac{k^2}{N} & \frac{k^3}{N} \\ 0 & \frac{k+k^2}{N} & \frac{k^3}{N} \\ 0 & 0 & 1 \end{pmatrix} \quad (3)$$

$$\mathbf{H}^{(T)} = \begin{pmatrix} \log(k) & 2\log(k)+1 & 3\log(k)+2 \\ 0 & \log(k) & 2\log(k)+1 \\ 0 & 0 & \log(k) \end{pmatrix} \quad (4)$$

Thus, the values of  $E^{(T)}(i)$  can be calculated as follows:

$$E^{(T)}(i) = \sum_{j=1}^3 p^{(T)}_{ij} * h^{(T)}_{ij} \quad (5)$$

From Assumption (1), it is clear that the probability of one node located at the  $i^{\text{th}}$  level of sub-overlay has the following values:

$$\left\{ \begin{array}{l} Q^{(T)}(1) = k^3 / N \\ Q^{(T)}(2) = k^2 / N \\ Q^{(T)}(3) = k / N \end{array} \right\} \quad (6)$$

From the analysis above, we can get the value of  $E_{\text{TPP}}$ :

$$E_{\text{TPP}} = \sum_{i=1}^3 E^{(T)}(i) * Q^{(T)}(i) \quad (7)$$

### B. Average Number of Lookup Hops in VTM

In VTM, vertical tunnels are built up to allow for direct lookup request forwarding between upper and lower levels of sub-overlays. Because the upper level of sub-overlay has the duplicate indices and mapping relationships of lower level of sub-overlay, it does not have to forward lookup requests to peers at the lower level of sub-overlay, thus the values of  $\mathbf{T}^{(V)}$  have the following results:

$$\mathbf{T}^{(V)} = \begin{pmatrix} t^{(V)}_{11} & t^{(V)}_{12} & t^{(V)}_{13} \\ 0 & t^{(V)}_{22} & t^{(V)}_{23} \\ 0 & 0 & 1 \end{pmatrix} \quad (8)$$

Here,  $t^{(V)}_{ij}$  satisfies the following conditions:

$$\begin{cases} t^{(V)}_{11} + t^{(V)}_{12} + t^{(V)}_{13} = 1 \\ t^{(V)}_{22} + t^{(V)}_{23} = 1 \end{cases} \quad (9)$$

In VTM, for the requests initiated from the first level of sub-overlay, if it is firstly handled at the local SSO, then it follows the same procedure as the request initiated from the first level in TPP. If the request is forwarded to the second level directly, the processing procedure is the same as the request initiated from the second level in TPP, except the extra one forwarding hop. If it is forwarded to the third level directly, then it is similar to the request initiated from the third level in TPP, the difference is also the extra one forwarding hop. Thus,  $\mathbf{H}^{(V)}$  has the following value:

$$\mathbf{H}^{(V)} = \begin{pmatrix} E^{(V)}(1) & E^{(V)}(2)+1 & E^{(V)}(3)+1 \\ 0 & E^{(V)}(2) & E^{(V)}(3)+1 \\ 0 & 0 & E^{(V)}(3) \end{pmatrix} \quad (10)$$

The values of  $E^{(V)}(i)$  can be calculated as follows:

$$E^{(V)}(i) = \sum_{j=1}^3 t^{(V)}_{ij} * h^{(V)}_{ij} \quad (11)$$

The probability of one node located at the  $i$ th level of sub-overlay is the same as (6), thus it has the following results:

$$\begin{cases} Q^{(V)}(1) = k^3 / N \\ Q^{(V)}(2) = k^2 / N \\ Q^{(V)}(3) = k / N \end{cases} \quad (12)$$

The value of  $E_{VTM}$  can be got by the following formula:

$$E_{VTM} = \sum_{i=1}^3 E^{(V)}(i) * Q^{(V)}(i) \quad (13)$$

## IV. RESULTS ANALYSIS

In this section, the results analysis of the architectures proposed in this paper compared with the exiting architectures is given. In TPP and VTM, there are all together  $N = k + k^2 + k^3$  nodes providing services. Thus, if HA and FA are compared with them,  $N$  in HA and FA also equals to  $k + k^2 + k^3$ . The optimal average number of hops  $E_{HA}$  in HA equals to  $\log(N)-3$  according

to [3]. In FA with only one sub-overlay, the average number of lookup hops  $E_{FA}$  equals to  $\log(N)$ . As a conclusion, the following results exist:

$$\left. \begin{cases} E_{FA} = \log(N) = \log(k + k^2 + k^3) \\ E_{HA} = \log(N) - 3 = \log(k + k^2 + k^3) - 3 \end{cases} \right\} \quad (14)$$

We firstly analyze the average number of lookup hops in TPP, followed by the average number of lookup hops in VTM.

### A. Average Number of Lookup hops in TPP

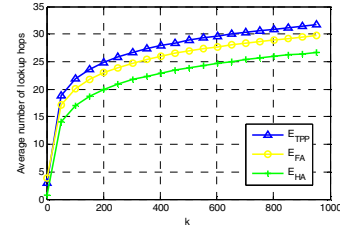


Figure 3. Average Number of Lookup Hops in TPP

Based on the (7) and (14), we can get the figures of TPP versus FA and HA, as shown in Figure 3. From Figure 3, it is more than clear that the average number of lookup hops for TPP is slightly larger than FA, while HA proposed in [3] has slight advantage in lookup hops compared with TPP and FA. Nevertheless, the advantage is not obvious; it is about 10% shorter lookup hops.

### B. Average Number of Lookup Hops in VTM

In this section, several cases regarding different values of  $\mathbf{T}^{(V)}$  are discussed.

$$(1) t^{(V)}_{13} = 1, t^{(V)}_{23} = 1$$

The values of  $\mathbf{T}^{(V)}$  have the following results:

$$\mathbf{T}^{(V)} = \begin{pmatrix} 0 & 0 & 1 \\ 0 & 0 & 1 \\ 0 & 0 & 1 \end{pmatrix} \quad (15)$$

In this case, the requests will be forwarded to the third level of sub-overlay directly no matter what level the requests are initiated. As the third level of sub-overlay has all the duplicate indices and mapping relationships from the first and second levels, the lookup hops in VTM can be minimized. The curve is as shown in Figure 4.

From Figure 4, we can see that the average number of hops in VTM is about 1/3 of lookup hops in FA and HA. In this case, as all the requests are forwarded directly to the third level of sub-overlay, the peers from the third level have to handle almost all the traffic, which is the drawback of this case.

#### (2) Forwarding probability related to the nodes distribution

In this case, we choose the forwarding probability related to the nodes distribution.  $\mathbf{T}^{(V)}$  has the following values. The curves are shown in Figure 5.

$$\mathbf{T}^{(V)} = \begin{pmatrix} \frac{k}{N} & \frac{k^2}{N} & \frac{k^3}{N} \\ 0 & \frac{k+k^2}{N} & \frac{k^3}{N} \\ 0 & 0 & 1 \end{pmatrix} \quad (16)$$

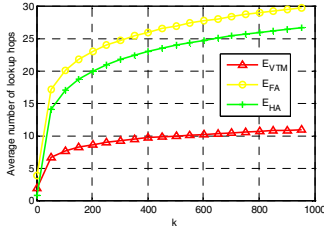


Figure 4. Average Number of Lookup Hops in VTM ( $t_{13}^{(V)}=1, t_{23}^{(V)}=1$ )

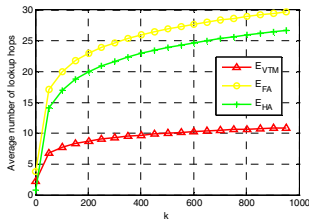


Figure 5. Average Number of Lookup Hops in VTM (forwarding probability related to the nodes distribution)

In this case, for the requests initiated from the first level of sub-overlay, with the probability of  $k/N$ , the requests will be handled firstly in the local SSO located at the first level; with the probability of  $k^2/N$ , the requests will be forwarded to the second level directly; with the probability of  $k^3/N$ , the requests will be forwarded to the third level directly. Comparing (16) with (3), we can find that they have the same values with each other. From Table 1, we know that  $P_{ij}$  stands for the probability of one lookup request which is terminated at the  $j^{\text{th}}$  level of sub-overlay and initiated from the  $i^{\text{th}}$  level of sub-overlay. It means whether the vertical tunnels and direct request forwarding exist or not, the request will have to be terminated at the  $j^{\text{th}}$  level of sub-overlay for one successful lookup. Consequently, if we make the values of  $\mathbf{T}^{(V)}$  equal to the values of  $\mathbf{P}^{(T)}$ , then VTM can not only decrease the number of lookup hops, but also alleviate the workload of the peers from the third level of sub-overlay.

From Figure 5, it is clear that VTM, which takes account of the nodes distribution, can achieve almost the same improvement in lookup hops as Figure 4. The improvement is about 1/3 of the lookup hops compared with HA and FA.

## V. FUTURE WORK

As a result, this paper provides a theoretical mathematical analysis of the proposed Truncated Pyramid Peer-to-Peer (TPP)

architecture with Vertical Tunneling Model (VTM). The used analysis tool is Matlab. In future, network simulations and possibly real-world prototypes will be implemented to analyze the maintenance overhead of TPP and VTM, and examine their relationship with different scales of SSOs. At the same time, the construction of the architecture and the influence of nodes joining and leaving will be analyzed.

## VI. CONCLUSION

In this paper we proposed a Truncated Pyramid P2P (TPP) architecture and an optimized model, Vertical Tunneling Model (VTM). Compared with the existing architectures, it was shown that VTM had better performance in average number of lookup hops, which was about 1/3 of the lookup hops in FA and HA. As shown in this paper, VTM decreased the average number of lookup hops by an improved service lookup mechanism. That was, taking account of the nodes distribution, VTM forwarded the requests to the second and third level of sub-overlay along the built vertical tunnels with some forwarding probability.

## ACKNOWLEDGMENT

The authors also would like to thank the anonymous reviewers for their valuable advice.

## REFERENCES

- [1] Rosenberg J. et al..SIP: Session Initiation Protocol. RFC3261,Internet Engineering Task Force, June 2002.
- [2] David A. Bryan, Bruce B. Lowekamp and C. Jennings. SOSIMPLE:A Serverless, Standards-based, P2P SIP Communication System. Proc. of International Workshop on Advanced Architectures and Algorithms for Internet Delivery and Applications, June 2005.
- [3] Le, L.; Kuo, G.-S.. Hierarchical and Breathing Peer-to-Peer SIP System. IEEE International Conference on Communications, 2007. ICC '07. 24-28 June 2007 Page(s):1887 – 1892.
- [4] Singh K., Schulzrinne H.. Peer-to-peer Internet Telephony Using SIP. Columbia University Technical Report CUCS-044-04, New York, NY, October 2004.
- [5] P2P SIP, <http://www.p2psip.org/>.
- [6] Ratnasamy S., Francis P., Handley M., Karp R., Shenker S.. A scalable content addressable network. SIGCOMM Symposium on Communications Architectures and Protocols, San Diego,CA, USA, Aug. 2001, ACM.
- [7] Stoica, I., Morris, R., Liben-Nowell, D., Karger, D.R., Kaashoek, M.F.,Dabek, F., Balakrishnan, H.. Chord: a scalable peer-to-peer lookup protocol for Internet applications. IEEE/ACM Transactions on Networking, Volume 11, Issue 1, Feb. 2003 Page(s):17 - 32.
- [8] Rowstron A., Druschel P.. Pastry: Scalable, distributed object location and routing for large-scale peer-to-peer systems. IFIP/ACM International Conference on Distributed Systems Platforms (Middleware), Heidelberg, Germany, Nov. 2001, Page(s):329 - 350.
- [9] Bryan D., Matthews P., Shim E., Willis D. (2007) Concepts and Terminology for Peer to Peer SIP. Internet Draft, draft-ietf-p2psip-concepts-01 (work in progress).
- [10] V. Lo, Zhou D.-Y, Liu Y.-H, Dickey C. -G and Li J. Scalable Supernode Selection in Peer-to-peer Overlay Networks. Hot topic in Peer-to-Peer System,2005