Network Locality Positioning System in P2P Networks

Chun-Hsin Wang and Yen-Ting Wu
Department of Computer Science and Information Engineering
Chung Hua University, Hsinchu, Taiwan 30012, R.O.C.

Abstract— **In this paper, we propose a network locality positioning system (NLPS) to provide precise locality information of nodes in networks instead of designing new P2P systems. It can not only be used to solve the consistent problem whatever the P2P systems are unstructured or structured but also provide how to select the best candidates for downloading the desired resource from the searching result. The nodes in P2P system can be classified into logical clusters and assigned the same locality code to define their locality by the proposed NLPS. The approximate distance between two nodes is embedded in locality codes. To evaluate the performance of NLPS, DHT-based P2P systems cooperating with NLPS are simulated. The locality code of a node is associated with its DHT-based identifier (ID).The searching process is performed as normal operation in DHT-based P2P system, then searching results will include the locality code of each node which owns the desired resource. Therefore, best candidate node(s) can be selected from the searching result by their associated locality codes. Extended simulation results show that DHT-based P2P system cooperating with NLPS has better performance than the others DHT-based P2P systems.**

I. Introduction

The P2P technology has been widely applied to the integration and sharing of network resources such as files sharing, grid computing, multimedia video streaming, network TV, multicast applications ., and etc. It is easy to get the resource (ex. files) when a lot of nodes are participating in P2P networks for resource sharing. But P2P traffic costs much network bandwidth resource. Measurement studies in [1] consistently indicate that $50-70\%$ of Internet traffic is caused by popular P2P applications. Therefore, how to enhance the performance of P2P networks to avoid wasting network bandwidth is an emergency issue.

The key technology of P2P system is how to maintain where the network nodes are and which resources of them can be provided such that the shared resources can be acquired as soon as possible. The overlay networks [2] build on existing internet take the responsibility to organize network nodes which own the shared resources into a logical network topology. The directory of the shared resources can be centralized or distributed over the nodes on overlay networks. Then the resource can be inquired by the build overlay network. The performance of P2P system will depend on how overlay network is constructed well and maintained when network nodes are joining or leaving.

The well-known peer-to-peer system for file sharing in Napster [3] and Gnutella [4]. The directory of the shared files is managed by a centralized server in Napster. It makes the Napster system scalability and single-point failure problems. In Guntella system, the query message for locating data items is flooding over the overlay network built by the joining nodes. Therefore Guntella design is also criticized for being non-scalable. To overcome the scalability problem, some structured P2P system such as Chord [5], CAN [6], Tapestry [7], and Kademlia [8] are based on Distributed Hash Tables (DHT) to organize all nodes in the system into overlay networks. The delivery of a query message to destination can be guaranteed within $\bigcirc(logN)$ hops, where N is the number of nodes joining the P2P system.

Although the DHT-based P2P systems have the advantage of scalability, fault-tolerance and load balance for maintaining the information related shared resources, the locality of nodes are not considered. The overlay network topology is not congruent to physical network topology. The long-latency delay may exist between one single-hop on overlay networks. It is what is called the consistent problem in P2P system. Some researches such as [9-20], they try to build topology-aware overlay networks for the purpose of reducing the difference in average latency of two nodes between overlay networks and physical networks.

These related searches [9-20] can be classified into three categories according what the locality information of nodes are used. The first class [9-14] is based on some selected landmarks. The joining node measures round-trip latencies from itself to each landmarks and sorting the result. Nodes with the same measured result are viewed as neighboring nodes and have the same locality. The overlay network topology is constructed by locality information. But the sensitivity of locality is limited by how many the number of landmarks is and where the landmarks are located. The second class [15-17] tries to reduce traffic transmission crossover the different AS(Autonomous System), ISP (Internet Service Provider) or DNS. The AS number, prefix of IP address, or DNS of nodes is used as the locality of nodes. The number of nodes in an As (ISP or DNS) may be too many and the difference of nodes locality can not be distinguished precisely. The third class [18-20] adopts hierarchical overlay networks, super-network and sub-networks. In each sub-network, there exists one leader node. A joining node measures the latency to each leader nodes to decide which sub-networks will belong to. If the measured latency for a joining node is longer than the predefined time threshold, a new sub-network is created and the joining node will become the leader node. The super-network is composed of all leader nodes. The network topology of sub-network and super-network are organized by DHT-based methods. Since the size of sub-networks is variant, the natural load balance provided by DHT system may be destroyed.

We summarize some disadvantages of nodes's locality information described above as follows.

- Landmarks-based methods [9-14]: The sensitivity of nodes's locality depends on how many number of landmarks is and where landmarks are located.
- AS(IPS)-based methods[15-17]: The nodes's locality can not be distinguished in an AS (or ISP) which may contains a large number of nodes, even over 10000 nodes [21].
- Hierarchy-based methods[18-20]: They may destroy the natural load balance provided by DHT system.

To get the resources for a network node in P2P system, there are two necessary steps. The first step is searching where the desired resource is on the overlay network. The second step is to download the desired resource directly from nodes found by the first step. The locality of nodes play an important role to solve the consistent problem. Many researches [9-20] make efforts to improve the searching latency by building topology-aware overlay networks with the assistance of nodes's locality. But some disadvantages described above still exist due to the lack of more precise locality information of nodes. In addition, it is worth noting that there may be many nodes which own the desired resource in searching result. Which node(s) will be the best candidate(s) for downloading the desired resource? The time need to acquire the desired resource will depend on how close is between the selected node to requested node.

In this paper, we propose a network locality positioning system (NLPS) to provide more precise locality information of nodes in networks instead of designing new P2P systems. The proposed NLPS can provide precise locality information of nodes to solve the consistent problem whatever the P2P systems are unstructured or structured. It can also provide how to select the best candidates for downloading the desired resource from the searching result.

The nodes in P2P system can be classified into logical clusters and assigned the same locality code to define their locality by the proposed NLPS. The approximate distance between two nodes is embedded in their locality codes. The proposed NLPS can cooperate with any unstructured and structured P2P systems without destroying the characteristics provided by original P2P systems. To evaluate the performance of NLPS, DHT-based P2P systems cooperating with NLPS are simulated. The locality code of a node is associated with its DHT-based identifier(ID). The searching process is performed as normal operation in DHT-based P2P system, then searching results will include the locality code of each node which owns the desired resource. Therefore, best candidate node(s) can be selected from the searching result by their associated locality codes. Extended simulation results show that DHT-based P2P system cooperating with NLPS has better performance than the others DHT-based P2P systems.

The rest of the paper is organized as follows. The proposed NLPS is described in section II. Simulation result is presented in section III. Finally, some concluding remarks are given in section IV.

II. Network Locality Positioning system

This section describes the proposed network locality positioning system (NLPS). Each node joining the P2P system will be assigned a locality code provided by NLPS to define its locality. The relative positions of nodes in physical network topology are embedded in their locality codes. The locality codes of nodes can represent the distant relationship how close they are. The approximate distance between two nodes can be evaluated by their locality codes. Since nodes may join or leave in P2P system, the relationship of neighboring nodes in P2P system may be changes. Therefore the locality codes of nodes need to be updated as nodes are joining or leaving to match their relative positions in physical network.

Before describing our proposed system, we introduce the basic idea how to organize peers into clusters first. One node within a cluster is chosen to be the cluster leader. According to how close to cluster leaders is, a joining node can decide which cluster belongs to. The cluster leaders become the request candidates for a joining node and only the nearest one from the joining node to be selected. This phenomenon is very like the anycast protocol [22-24]. It motivates us to use anycast protocol to organize peers into clusters. The cluster leaders can be grouped into an anycast group sharing with an anycast address. A joining node can find the nearest cluster leader from it by anycast protocol. If the joining node is far from the found cluster leader, a new cluster will be created. Otherwise, it belongs to the found cluster. In this way, peers can be self-organized into clusters by anycast protocol.

The system initialization, nodes joining, nodes leaving, locality code assignment and update, and how to select the good node(s) by locality codes are described as follows.

A. system initialization

The joining nodes will be organized into logical clusters. In each cluster, there exists one cluster leader and backup leader for tolerance. The cluster leaders are grouped into an anycast group (say, G_{any}). In the system initialization, the first node joining P2P system will be the root node and become the cluster leader. An unique cluster identifier (CID) with length l bits will be assigned to the root cluster. The CID may be generated by some hashing functions. The anycast group G_{any} contains only one member initially (ie., the root node). Note the root node may be setup by the P2P system and it always stay in the system, not leaving.

B. Node joining

By anycast protocol, a new node joining a P2P system can send a query to the anycast group G_{any} to find which the nearest cluster leader is close to it. The round-trip delay will be measured between the new node and the cluster leader (say, node N) found by anycast protocol. To define a cluster which joining node belong to, a time threshold, t_ρ, is defined. If the measured round-trip delay is not longer than t_ρ, the

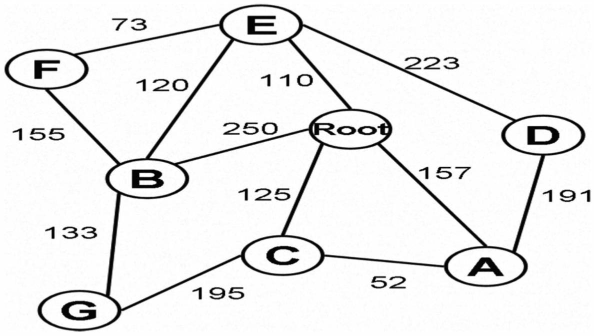

Fig. 1. The physical network topology of an example.

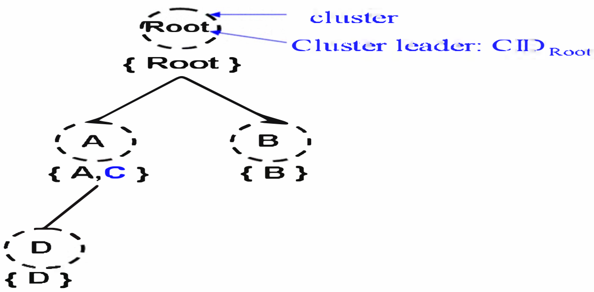

Fig. 2. The self-organized clusters in a P2P system.

new node is classified into the same cluster with N and being assigned the same locality code with N. Otherwise, the new node will be the cluster leader of a new cluster and joining in anycast group G_{any}. In this situation, the cluster including node N becomes the parent cluster of the new cluster. The new cluster will be assigned with an unique CID. Besides, each cluster leader has to maintain IP address of its parent cluster leader and round-trip delay time between them to keep neighboring relationship. Each node maintains the IP address of its cluster leader. Fig. 1 displays an example of a physical network topology with propagation delay in each link. The order of nodes joining the P2P system is Root, A, B, C, and D in sequence. The time threshold t_ρ equals to 100 ms. Fig. 2 displays the relationship of clusters. A circle of line segments is used to represent a cluster and the cluster leader in it. The members of cluster are showed in a set. For convenient to description, cluster is named with its cluster leader. For example, the cluster including the Root node is name "cluster Root". Its CID is CID_{Root} and only one member is in the set $\{Root\}$.

C. Node leaving

When a node is leaving P2P system, the leaving procedure will be performed by the cooperating P2P system. If the leaving node is not a cluster leader or backup cluster leader, no overhead of node leaving is added to original P2P system. The backup leader exists in a cluster for the fault tolerance when the cluster leader is leaving or crashed. The necessary information maintained by cluster leader will be synchronized with its associated backup leader in a periodic time. If the backup leader in a cluster is leaving or crashed, the new backup leader has to be elected again. One approach of selecting the backup leader in a cluster is to choose one of nodes which keeps living for a period of time longer than the others. Once the cluster leader is crashed or leaving, the backup leader will become new cluster leader and join to anycast group G_{any}. An another node in cluster will be selected to be the new backup leader.

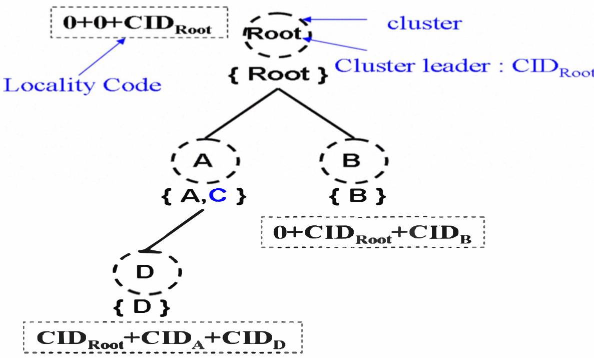

Fig. 3. Locality codes assignment in Fig. 2

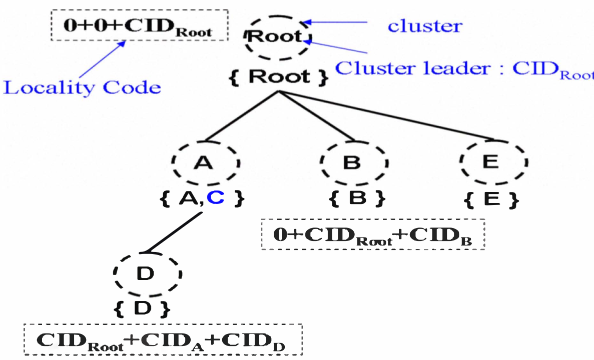

Fig. 4. Node E joins initially

D. Locality code assignment and update

The locality code of a joining node is inherited by its cluster leader. That is nodes in the same cluster will be assigned with the same locality code. To represent the relative position of nodes in physical network, the locality code of cluster leader is defined by concatenating $(k-1)$ CIDs of its ancestors and CID of itself. That is the locality code is compose of k CIDs. Since cluster leader maintains the IP address of its parent cluster leader, the CIDs of its ancestors can be known by recursively querying. Each node can know its locality code by requesting its cluster leader. One peer can know the relationship of its neighbors by maintaining large value of k in its locality code.

Fig. 3 shows the locality codes assignment of clusters with $k = 3$ for the example in Fig. 2. The locality code of cluster D is assigned by the concatenation of $CID_{Root} + CID_A + CID_D$. If the number of ancestors of a cluster is less then k, the CID of absent ancestor is set to zero. For example, the locality code of cluster B is $0 + CID_{Root} + CID_B$.

To keep actual relative position of peers in P2P system, we need to update the relationship of clusters and reassign locality codes of related clusters when a peers is leaving or a new node is joining. The cluster may disappear once the leaving

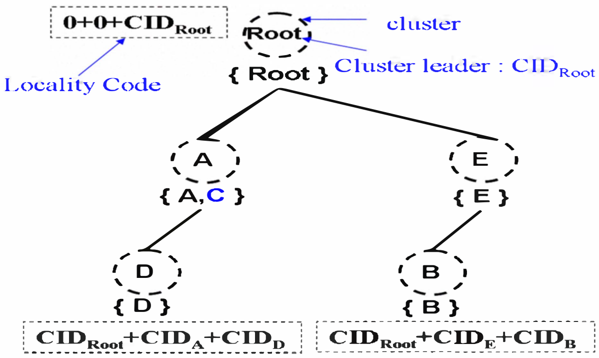

Fig. 5. Locality codes are updated after node E is joining

node is the cluster leader and no other nodes exist in it. In this situation, the son cluster(s) of the disappeared cluster will simply replace their new parent cluster with their grandparent cluster first and then update their locality codes. As a new node is joining to the P2P system, it may self-organize a new cluster and the neighboring relationship of existing clusters may be changed. For example, node E joins the P2P system in the example of Fig. 3. A new cluster E is created. Fig. 4 shows the relationship of clusters when node E joins the P2P system initially. The original sons cluster of cluster *Root* may need to update their neighboring relationship due to the new cluster E. The cluster E can know its sibling(s) from the cluster *Root* and then measure the round-trip delay time from it to them. The cluster B is near cluster E than cluster *Root*. Therefore, cluster B will be notified to be the son cluster of cluster E and update its locality code as showed in Fig. 5.

E. The neighboring hop distance between two locality codes

The locality code of peer is composed of $(k-1)$ $CIDs$ ancestor clusters and its CID . It means that one peer can keep the number of $(k-1)$ different level neighboring clusters starting from itself. We refer to each level as **neighboring hop distance**. If two peers have common ancestors, there must exist same part of CIDs in their locality codes, except the common ancestors are over $(k-1)$ neighboring hop distance. We can find the nearest common ancestor from two locality codes and know the neighboring hop distance between them. For example, the neighboring hop distance is 3 for two locality codes, $CID_{Root} + CID_A + CID_D$ and $0+CID_{Root}+CID_B$. Because the CID_{Root} is their nearest common ancestor, cluster D is two neighboring hop distance to cluster *Root* and cluster B is one neighboring hop distance to cluster *Root*. If no common ancestors exist for two peers, it means that they have more than $2(k-1)$ neighboring hop distance.

F. The priorities of selecting good nodes for downloading

Each peer in P2P system is assigned with a locality code provided by the NLPS. The NLPS can cooperate with structured or unstructuredP2P systems. In structured P2P system, locality codes of peers can be distributed with the published resources-sharing information. In unstructured P2P system, peers only need to reply their locality codes associated with IP

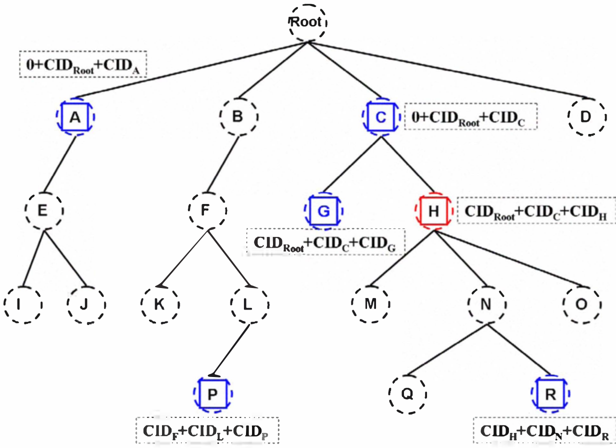

Fig. 6. The priorities of selecting nodes for downloading

address when queries are receiving and they have owned the requested resource. The searching process to query a required resource is performing by the cooperating P2P system.

One node in the cluster sends a query message for requesting a resource, the searching result will contain the locality code of each node which owns the requested resource. By computing the neighboring hop distance, the request node can evaluate the owners how close to it is. The good nodes can be selected for downloading by the following priorities.

1) The first priority is to select random nodes from the cluster which is the same with the request node. That is the neighboring hop distance between the selected node and the request node is zero.
2) The second priority is to select random nodes which are one neighboring hop distance from the request node.
3) The third priority is to select random nodes which are two neighboring hop distance from the request node.
4) If there is no first priority, second priority node and third priority nodes, random nodes are selected from the searching result.

For example, Fig. 6 displays the relationship of clusters and their locality codes. One node in cluster H sends a query by the P2P system cooperated with the proposed NLPS. The searching result is supposed that the owners of the requested resource are in cluster A, C, G, H, P, and R. The first priority nodes are in cluster H, the second priority nodes are in cluster C, the third priority nodes are in cluster G or R, and then the random selected nodes are in clusters A or P.

III. Simulation Results

In this section, simulations are performed to study the performance of the proposed NLPS P2P system. To simulate internet topology, the real-word internet topologies from the skitter database [25] is used to generate random graphs. In the system initialization, there are eight kinds of files resources to be shared and each of them are owned by three different random nodes in the system. The node may join or leave the P2P system in our experiment. The time of nodes joining the P2P system forms a poisson process. The lifetime of nodes

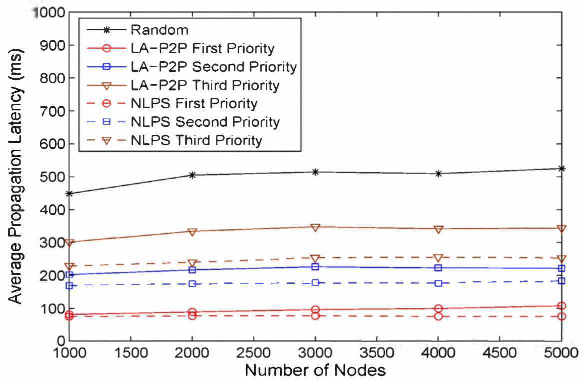

Fig. 7. Propagation latency for different priorities nodes

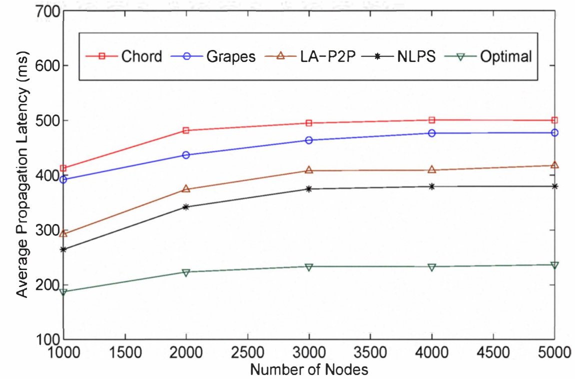

Fig. 9. Propagation latency with $t_\rho = 100$

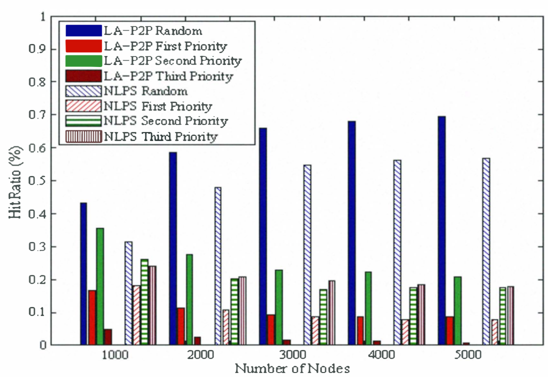

Fig. 8. Hit ratios for different priorities

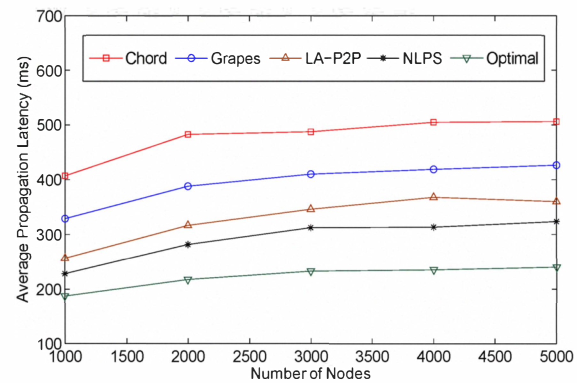

Fig. 10. Propagation latency with $t_\rho = 200$

in P2P system forms an exponential distribution. The mean lifetime of nodes is one hour. For each data point in our experiment, 20 random graphs are generated and 5000 queries for requesting a random resource from eight kinds of files in each graph are performed. The time of quires from random nodes in P2P system also forms a poisson process. The time threshold, t_ρ, equals to 100 ms.

The chord-based [5] system, Grapes [18] and our previous work LA-P2P [26] are implemented for comparison purpose. In Grapes system, the time threshold, 100 ms, is used to decide which sub-network belong to for a new joining node. Since we focus on studying the performance of downloading process in P2P system, the searching time to find the desired resource is not considered in our simulation. We want to observe the effect how does select good nodes from searching result on the performance of downloading process in previous works and our proposed NLPS P2P system.

Fig. 7 shows the average propagation latency from the request node to the selected nodes for different number of nodes in the networks. The average propagation latency for the first priority node has better performance than the others in both of LA-P2P and NLPS P2P system. This because the request node and the first priority node is within the same cluster and the maximum propagation latency should be bound to two times of t_ρ, i.e, 200 ms. The average propagation latency for random nodes selection from searching result such as chord-based system has longest latency than those for the others.

Form Fig. 7, we can also observe that the performance for each kinds of priority nodes in NLPS P2P is better than it in our previous work LA-P2P. Although the main idea of LA-P2P is similar with NLPS P2P system, LA-P2P can not provide precise relative position of peers because relative position of peers are not updated when nodes are joining or leaving. In addition, the approximate distance between peers can not be computed by their locality codes. Fig. 8 shows the hit ratios of the selected nodes in each kinds of priority nodes to the total selected nodes for downloading. The ratios of selected nodes in third priority for NLPS are large than those for LA-P2P and the ratios of selected nodes by random for NLPS are less than those for LA-P2P. Compared with LA-P2P, NLPS can select most of nodes nearby the request node for downloading. The simulation result of Fig. 8 can support the reason why the performance for NLPS is better than it for LA-P2P.

Fig. 9 and Fig. 10 shows the average propagation latency in different P2P systems with $t_\rho = 100$ ms and $t_\rho = 200$ ms respectively for different number of nodes in the network. The optimal propagation latency from the request node to the nodes which own the desired resource is calculated for comparison purpose. From these two figures, we can observe that the performance of the proposed NLPS P2P system is

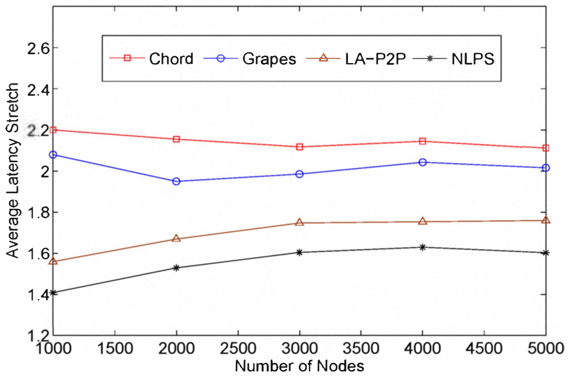

Fig. 11. Average latency strength with $t_\rho = 100$

closer to the optimal propagation latency than those for the others. Compared with $t_\rho = 100\ ms$, more nodes which own the desired resource may be included into the first three priorities when $t_\rho = 200\ ms$. The request node may increase the chance to select nodes for downloading nearby it instead of random selection. The performance for Grapes, LA-P2P, and NLPS is improved when $t_\rho = 200\ ms$. Note the performance of NLPS is still better than the others.

The latency strength is defined by the ratio of the measured propagation latency to the optimal propagation latency. Fig. 11 shows the average latency strength in different P2P systems for different number of nodes in the network. The proposed NLPS P2P system has lowest latency strength than the others. The performance for the chord-based system is still worst than those for the others.

IV. CONCLUSION

The proposed NLPS system can provide precise locality information of peers in networks. The peers can be classified into logical clusters and assigned the same locality code to define their locality. The approximate distance between two nodes is embedded in locality codes. The good nodes close to the request node can be selected for downloading from the searching result by their associated locality codes. Extended simulation result shows that the proposed NLPS P2P system has better performance than the others. In future, how to setup good topology-aware overlay networks by NLPS system for solving the consistent problems will be further studied.

REFERENCES

[1] R. Steinmetz and K. Wehrle, "P2P Systems and Applications", Springer Lecture Notes in CS, 2005.

[2] H. Wang, Y. Zhu, and Y. Hu, "To Unify Structured and Unstructured P2P Systems," In Proceeding of the 19th International Parallel and Distributed Processing Symposium (IPDPS05), Denver, Colorado, April 2005.

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

[4] Gnutella http://www.gnutella.com

[5] I. Stoica, R. Morris, D. Karger, M. F. Kaashoek and H. Balakrkshnan, "Chord: A Scalable Peer-to-peer Lookup Service for Internet Applications," In Proceedings of the 2001 conference on applications, technologies, architectures, and protocols for computer communications (SIGCOMM), pp. 149V160, San Diego, CA, 2001.

[6] S. Ratnasamy, P. Francis, M. Handley, R. Karp, and S. Shenker, "A Scalable Content Addressable Network," In Proceedings of the 2001 conference on applications, technologies, architectures, and protocols for computer communications (SIGCOMM), San Diego, CA, 2001.

[7] B. Y. Zhao, J. D. Kubiatowicz, and A. D. Joseph, "Tapestry: An infrastructure for fault-tolerant wide-area location and routing," Technical Report UCB/CSD-01-1141, UC Berkeley, Apr. 2001.

[8] Petar Maymounkov, and David Mazi?res, "Kademlia: A Peer-to-Peer Information System Based on the XOR Metric," The First International Workshop on Peer-to-Peer Systems, March 2002.

[9] S. Ratnasamy, M. Handley, R. Karp, and S. Shenker, "Topologically-aware overlay construction and server selection," in Proceedings of IEEE INFOCOM, 2002.

[10] Tein-Yaw Chung, Yang-Hui Chang, Chia-Yu Liu, and Yung-Mu Chen, "SSIF: Stable Self-Organized Infrastructure-based Peer-to-Peer Network," The 9th International Conference on Advanced Communication Technology (ICACT 2007), pp. 1441-1446, Feb. 2007.

[11] Oztoprak, K., and Akar, G. B.," Two-Way/Hybrid Clustering Architecture for Peer to Peer Systems," Second International Conference on Internet and Web Applications and Services (ICIW '07.), pp. 1457-1461, Feb. 2007.

[12] Wang Huijin, and Lin Yongting, "Cone: A Topology-Aware Structured P2P System with Proximity Neighbor Selection," Future generation communication and networking (FGCN 2007), pp.43-49, Dec. 2007.

[13] Xing Chen, and Qing Yang, "A Locality-Based LFH Cluster Strategy for Overlay Network," International Conference on Information Networking (ICOIN 2008), pp.1-5, Jan. 2008.

[14] X. Zhang, Z. Li, and Y. Wang, "A distributed topology-aware overlays construction algorithm," In Proceedings of the 15th ACM Mardi Gras conference, Jan. 2008.

[15] Pfeifer, G., Fetzer, C. and Hohnstein, T. "Exploiting Host Name Locality for Reduced Stretch P2P Routing," Sixth IEEE International Symposium on Network Computing and Applications (NCA 2007), pp. 134-144, July 2007.

[16] Guoqiang Zhang, and Guoqing Zhang, "Agent Selection And P2P Overlay Construction Using Global Locality Knowledge," IEEE International Conference on Networking, Sensing and Control (ICNSC.2007), pp. 519-524, April 2007.

[17] Weiyu Wu, Yang Chen, Xinyi Zhang, Xiaohui Shi, LinCong, Beixing Deng and Xing Li, "LDHT: Locality-aware Distributed Hash Tables," International Conference on Information Networking (ICOIN 2008), pp. 1-5, Jan. 2008.

[18] Shin, S. Lee, G.. Lim, H. Yoon, and J. S. Ma, "Grapes: Topology-based Hierarchical Virtual Network for Peer-to-peer Lookup Services," In Proceedings of the International Conference on Parallel Processing Workshops (ICPPW' 02), pp. 159-164, Aug. 2002.

[19] Richard Hsiao, and Sheng-De Wang, "Jelly: A Dynamic Hierarchical P2P Overlay Network with Load Balance and Locality," In Proceedings of the 24th International Conference on Distributed Computing Systems Workshops (ICDCSW '04), pp. 534-540, March 2004.

[20] Zhenyu Li, Gaogang Xie, and Zhongcheng Li, "Locality-Aware Consistency Maintenance for Heterogeneous P2P Systems," IEEE International Parallel and Distributed Processing Symposium (IPDPS 2007), pp.1 - 10, March 2007.

[21] Z. M. Mao, J. Rexford, J, Waang, and R. H. Katz, "Towards an accurate AS-level traceroute tool," In Proceeding of SIGCOM '03, pp. 365-378, 2003.

[22] S. Weber, and Liang Cheng, "A survey of anycast in IPv6 networks," IEEE Communications Magazine, pp. 127-132 Jan 2004.

[23] C. Metz, "IP anycast point-to-(any) point communication," IEEE Internet Computing, pp. 94-98, March-April 2002

[24] C. Partridge, T. Mendez, and W. Milliken, "Host Anycasting Service," RFC 1546 (1993).

[25] http://www.caida.org/tools/measurement/skitter/

[26] Chun-Hsin Wang, and Jyun-Cheng Ciou, "Locality-Aware Peer-to-Peer Networks(LA-P2P)" In The 9th International Conference on Advanced Communication Technology, (ICACT. 2007),pp.1462-1466, Feb. 2007.