Journal of Zhejiang University SCIENCE
ISSN 1009-3095
http://www.zju.edu.cn/jzus
E-mail: jzus@zju.edu.cn

Control DHT maintenance costs with session heterogeneity*

ZOU Fu-tai (邹福泰), WU Zeng-de (吴增德), ZHANG Liang (张 亮), MA Fan-yuan (马范援)

(*Department of Computer Science and Engineering, Shanghai Jiaotong University, Shanghai 200030, China*)

E-mail: zoufutai@cs.sjtu.edu.cn; wuzengde@cs.sjtu.edu.cn; zhangliang@cs.sjtu.edu.cn; fyma@cs.sjtu.edu.cn

Received Jan. 10, 2004; revision accepted May 4, 2004

Abstract: The maintaining overheads of Distributed Hash Table (DHT) topology have recently received considerable attention. This paper presents a novel SHT (Session Heterogeneity Topology) model, in which DHT is reconstructed with session heterogeneity. SHT clusters nodes by means of session heterogeneity among nodes and selects the stable nodes as the participants of DHT. With an evolving process, this model gradually makes DHT stable and reliable. Therefore the high maintaining overheads for DHT are effectively controlled. Simulation with real traces of session distribution showed that the maintaining overheads are reduced dramatically and that the data availability is greatly improved.

Key words: Peer-to-peer (P2P), Distributed Hash Table (DHT), Finite element method, Session heterogeneity, Topology model
doi:10.1631/jzus.2005.A0378 **Document code:** A **CLC number:** TP393

INTRODUCTION

Distributed Hash Table (DHT) is a very promising topology model for peer-to-peer (P2P) network. The academic community has paid great attention to DHT systems such as Chord (Stoica *et al*., 2001), Pastry (Druschel and Rowstronand, 2001), Tapestry (Zhao *et al*., 2004), CAN (Sylvia *et al*., 2001) for their efficient, scalable, and robust P2P infrastructures. DHT is also the foundation of the new generation architecture of large-scale distributed applications in the IRIS project (IRIS, 2004).

However, there are few systems based on DHT successfully deployed in the Internet until now while systems such as Morpheus (2004) and Limewire (2004) etc. which are based on Gnutella protocol have millions of Internet users. From the standpoint of topology, there are two main shortcomings as follows:

(1) DHT has high maintaining overheads. While DHT systems implement global routing and location by means of each node's keeping a partial routing table, they are very sensitive to the node's join and leave. Generally speaking, for one node's join or leave, it needs messages to maintain the right P2P network topology (Balakrishnan *et al*., 2003). If the DHT system was deployed in a dynamic environment where nodes continuously join or leave, the maintaining overheads would be very huge, seriously affecting the performance of the system (Ledlie *et al*., 2002). However, DHT systems have to face the dynamic environment. In P2P systems, each node is free to leave and join which results in a highly dynamical network. The experimental result in Saroiu *et al*.(2002) showed that there were almost 50% nodes whose session time was less than one hour. Therefore, in such a real situation, the overheads to be maintained are excessively huge so that the performance of DHT systems is greatly affected.

(2) DHT ignores the heterogeneity of peer nodes. In initial DHT systems, nodes are treated as homogeneity and each node burdens the same responsibility. In fact, nodes in an open Internet are far from equal (Ratnasamy *et al*., 2002). There exist vastly different session time, host capacities and bandwidth among different pairs of nodes, varying by orders of

*Projects supported by the Science & Technology Committee of Shanghai Municipality Key Technologies R & D Project (No. 03dz15027) and the Science & Technology Committee of Shanghai Municipality Key Project (No. 025115032), China

magnitude (Saroiu *et al*., 2002). For example, some nodes may be server with high performance and some nodes may be home PC with low performance. However, they are assigned to take the same responsibility in the DHT system. Therefore, two cases would happen. On the one hand, the former may waste their capacities due to too little assigned responsibility; on the other hand, the latter may become bottleneck due to their low capacities. So it is necessary to take the heterogeneity into account during DHT design.

This paper proposes a novel model SHT (Session Heterogeneity Topology) to overcome these shortcomings. The SHT model is based on DHT but utilizes the clustering technique and the session heterogeneity among nodes in the dynamic network such as the Internet. The main contribution of the SHT model is to control maintaining overheads so as to make P2P systems more scalable and robust.

SHT MODEL

Saroiu *et al*.(2002) experimentally studied the session distribution of Napster and Gnutella. Their results (Fig.1) clearly showed that 50% of the nodes had session time of less than one hour, which represents the great session heterogeneity existing in the overlay nodes.

Fig.1a is the session CDF (Cumulative Distribution Function) curve and Fig.1b is the session PDF (Probability Density Function) curve. It can clearly be observed that:

(1) There is great heterogeneity of session time. That means peer nodes are not really peer.

(2) There is uneven distribution of session time among nodes. While many nodes' session time is very short, only a few nodes' session time is long.

This observation can explain why structured P2P systems have huge maintaining overheads. Many nodes whose session time is short would join and then leave the P2P network so frequently as to disturb the construction of network topology, rapidly increasing the maintaining overheads of structured P2P topology. Considering this fact, the intuitive idea is to exclude these nodes from joining the normal DHT topology. Therefore, we utilize the great heterogeneity of session time to construct the SHT model. The stable nodes would construct the DHT circle (We generalize the topology which is constructed by DHT technique such as Chord, Pastry, CAN and Tapestry as a ring space structure, called DHT circle), the other nodes would be out of the DHT circle. That means dynamic unstable nodes are clustered to the outside of the DHT circle so as to reduce the dynamic changes of the DHT circle. With these stable nodes on the DHT circle, the maintaining overhead for DHT circle can be reduced greatly and thus the total maintaining overheads can be reduced a lot. In this paper, the stability of the node is evaluated by the online session time. The node whose session time is the longest in the cluster would be selected onto the DHT circle (Refer to Theorem 1). Maybe more arguments such as CPU power, bandwidth etc. can be tradeoff when selecting the node onto the DHT circle; however, the session time is still the dominating one among factors which affect the maintaining overheads. So we do not discuss the tradeoff in this paper because of the limitation of space.

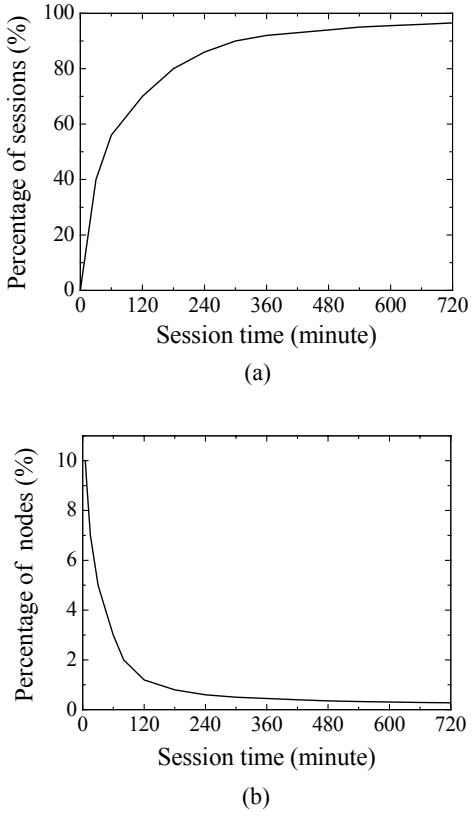


Fig.1 Session distribution in P2P network
(a) The curve of CDF; (b) The curve of PDF

The nodes on the DHT circle are called father node, while the others out of the circle but around a father node are called son node. One father node and its son nodes are called a cluster. Father is only one and son may be many around the father in a cluster. For a network with n nodes, we limit the number of sons for each father node by m. The expectation is that with a reasonable size m, a highly stable node can be found to become the father node so as to form a stable DHT circle. Therefore, a general SHT topology is composed of the cluster set $S=\{C_1, C_2, C_3, \ldots, C_i, \ldots\}$, where C_i is a cluster and $i\in[1,n]$. For $\forall C_i\in S$, $C_i=\{f, s_1, s_2, s_3, \ldots, s_i, \ldots\}$, where f is the father node, and s_i is a son node with $i\in[1,m]$. All father nodes constitute a DHT circle, that is, $D=\{f_1,f_2,f_3, \ldots,f_i, \ldots\}$ and $i\in[1,n]$. The core of SHT is to select stable nodes as father nodes onto the DHT circle. So each father node would maintain the son node list L, which includes the registered information (nodeID, IP, session time) on all son nodes. Father is selected from L and it gradually forms a stable DHT circle with an evolving approach.

Fig.2 presents the topology structure of the SHT model. In Fig.2, the filled circle of DHT circle stands for father node and the pane linked to the filled circle stands for son node.

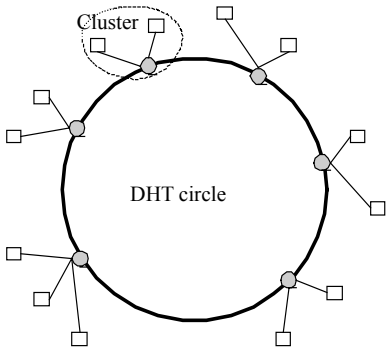


Fig.2 The topology of the SHT model

Node joins

The process for an arbitrary node α to join the network is described as follows:

1. Node α contacts a known introducer node I. Then it sends the joining request through node I to add itself into a randomly selecting cluster, say C.

2. If the size of the cluster C is over m, then cluster C is split into two clusters:

(1) Select node β whose session time is the longest from all son nodes of cluster C, and add node β into the DHT circle and then create new cluster called C'. Node β is the father node of the cluster C'.

(2) Migrating $\lfloor m/2 \rfloor$ son nodes from cluster C to cluster C'.

Node leaves

In the SHT model, we have different ways to deal with the leaving of the different types of nodes. When a son node leaves, a notification of updating son node list is sent to its father node. When a father node leaves, the stable node is selected from its son node list as the new father node, and then the new father node is added onto the DHT circle. The dynamic maintenance of son node list and the process of father selection will be described in the next section.

Stable DHT circle with evolving approach

Though the join and leave of nodes are very simple, it can form the stable DHT circle with an evolving approach. Here we say that an evolving process means that the topology is gradually changed towards the desired topology. The father node is selected from the cluster in two cases: (1) the size of the cluster is over m; (2) the fault events of the father node happen (e.g. node leaves or power off). Hence it insures that the father is the most stable node in the cluster. The selection is very critical for a stable DHT circle. To form a stable DHT circle, there are still two questions:

(1) How to keep a feasible and reliable selection?

(2) Whether these father nodes can make DHT circle reach the desired state?

First, we deal with Question 1 through maintaining the son node list. For a cluster, father node keeps the son node list which is an ordered list of father node candidates: the longer a node remains, the better a father node candidate it becomes. Furthermore, to deal with some exceptional failure conditions such as power off, broken network, etc., this list is sent periodically to each son node of the cluster. When a father node fails or disconnects, the node in the list whose session time is the longest becomes father node and joins into the DHT circle. Through this approach, the selection would be feasible and reliable.

Second, let us consider Question 2. A desired

stable state of DHT circle is defined as follows:

Definition 1 Let *node.session* be the session time of the node. Let U be the set of all son nodes set. For a network with n nodes and cluster size m, if it satisfies the two conditions:

(1) $D=\{f_1, f_2, ..., f_v\}, v=\lceil n/(m+1)\rceil$

(2) $\forall f \in D, \forall u \in U,\ f.session > u.session$

then it is the stable state of the DHT circle.

The clustering technique would finally leave v father nodes on the DHT circle. Thus it satisfies Condition 1. Let F be the father node set, where $F=D=\{f_1, f_2, \ldots, f_v\}$ and let the responding cluster set be C, where $C=\{C_1, C_2, \ldots, C_v\}$.

For $\forall f_i \in F$, $f_i.session$ is the maximum of cluster C_i based on the selection strategy. However, it may be less than the maximal session of son nodes set in other clusters. Thus, we use a random algorithm to adjust it. Fig.3 shows the pseudo code of the algorithm. The function adjust (C_i) would adjust the DHT circle to satisfy Condition 2, where $C_i \in C$. Lines 1~3 in Fig.3 define if there exists a random selected cluster whose father's session is less than the maximal session of son node set in C_i. If it is true, then lines 4~5 would update the father of the random selected cluster.

Adjust (C_i)

(1) Randomly select a cluster c, where $c \in C \cap c \neq C_i$.
(2) Select node t whose session is the maximum of the son nodes set in C_i
(3) If ($t.session > c.f.session$)
(4) $C_i = C_i - \{t\} + \{c.f\}$ // adjusts Cluster C_i
(5) $c.f = t$ // updates the father of cluster c with t.

Fig.3 The random adjustment algorithm

By the random adjustment algorithm, $f_i.session$ ($\forall f_i \in F$) would be the maximum not only in its own cluster but also in any other clusters. Therefore, by Definition 1, the desired stable DHT circle is formed. Due to the extremely heterogeneous distribution of session time, stable nodes can be selected rapidly so a stable DHT circle is rapidly formed. It can be observed from our experimental results.

Publishing and search

It is very important for peer nodes to publish and search documents. DHT is a good model to publish and search documents. SHT needs to retain the merits of DHT while overcoming its shortcomings. So SHT uses the stable DHT circle to store index metadata and search documents. Each node no matter whether it is father or son can publish its documents and search documents; but the storage and the routing must be based on the DHT circle[1].

For a father node, it lies on the DHT circle so it is normal operation just like DHT algorithms to publish and search documents. However, son nodes should operate with a proxy, that is, their father node. When publishing the documents of a son node, the request is sent to its father and the father node would publish related metadata and store them in the DHT circle. There is no need for son nodes to know how and where the metadata is stored. Similarly, the son node should deliver the search request to its father who would then lookup the DHT circle with the DHT routing algorithm and transfer the final results to the son node.

To avoid abundant invalid metadata information being stored in the DHT circle which will adversely influence the search efficiency, we adopt the soft state technique to publish the documents. The root node of the documents refreshes metadata information through periodically republishing it. If the node storing the metadata information finds some not recently refreshed metadata, it would delete them.

The availability of data would be improved because the DHT circle trends to be stable.

Analysis

The size of cluster m is a critical argument for the SHT model. It determines the stability of the DHT circle and controls maintaining overheads. In this section, we will analyze how the value of m affects the stability and the maintaining overheads. Our analysis is based on the stable DHT circle described in the previous section and the distribution of session time is derived from trace data in Saroiu *et al*.(2002).

Definition 2 Let C be the CDF (Cumulative Distribution Function) of session time of the node and let P be the PDF (Probability Density Function) of session time of the node. Let T be the time set.

Then, it is obvious to get the formula as follows:

$$P(t)=\lim_{\Delta t \to 0}\frac{C(t+\Delta t)-C(t)}{\Delta t},\ t \in T \qquad (1)$$

[1] To publish documents means to provide the index metadata of the documents so that these documents can be searched

Definition 3 Let G be the set of nodes. If the size of the cluster is m, then G is divided into two sub sets: G_{circle} and G_{outer}, where $G_{\text{circle}}=\{x|x.session\geq t_m,\ x\in G\}$ and $G_{\text{outer}}=\{x|x.session<t_m,\ x\in G\}$. The t_m is the session point that partitions the two sets.

In fact, by observing Fig.1a, we can learn that m is the ratio of the number of the nodes among those on the DHT circle and outside of the DHT circle. The formula is:

$$m=\frac{C(t_m)}{1-C(t_m)},\ C(t_m)=\frac{m}{m+1} \tag{2}$$

Let us see the relationship between the fault rate and m.

Definition 4 The fault rate of the node is the number of nodes which join or leave during Δt, when Δt approaches 0. Let $f(t)$ be the fault rate of the node at time t.

Let M be the total number of the nodes. Then we have the following formula

$$f(t)=\lim_{\Delta t\to 0}\left(\frac{1}{Dt}\frac{MC(t+\Delta t)-MC(t)}{MC(t)}\right)=\frac{P(t)}{C(t)} \tag{3}$$

By Eqs.(2) and (3), we can plot Fig.4 with the session distribution of trace data. Fig.4a shows that $f(t)$ decreases as the time goes on. Fig.4b shows that $f(t_m)$ decreases with the increase of m.

Theorem 1 The fault rate of the node would decrease as the time increases.

Proof

$$\because f(t)=\frac{P(t)}{C(t)},\ P(t)=C'(t)$$

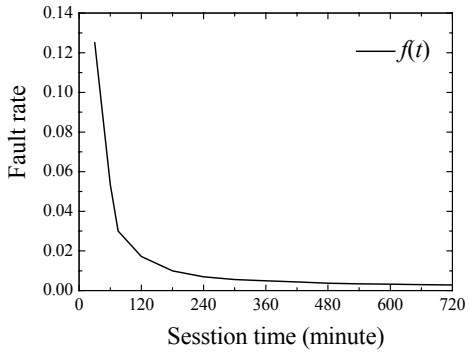
$$\therefore f'(t)=\frac{P'(t)C(t)-P(t)C'(t)}{C^2(t)}=\frac{P'(t)C(t)-P^2(t)}{C^2(t)}$$

$$\because P'(t)<0,\ 0\leq C(t)\leq 1$$

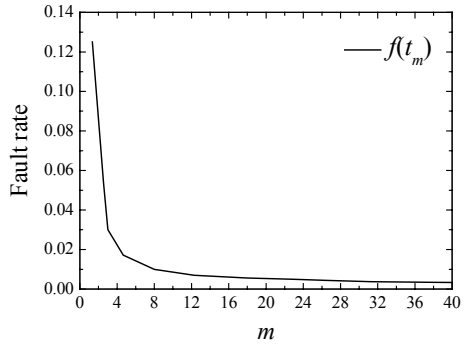
$$\therefore f'(t)<0$$

Thus $f(t)$ is a monotonously decreasing function. The fault rate of the node decreases as the time increase.

Definition 5 The stability of the system is evaluated by the average fault rate of the node in the system. Let S stand for the average fault rate.



(a)



(b)

Fig.4 The fault rate
(a) Fault rate vs session time; (b) Fault rate vs *m*

The stability is described as follows:

$$S=\frac{1}{M}\int_0^t MP(t)f(t)\mathrm{d}t=\int_0^t\frac{P^2(t)}{C(t)}\mathrm{d}t \tag{4}$$

The stability of the DHT circle can be evaluated by S_{circle}, where $S_{\text{circle}}=\int_{t_m}^{t}\frac{P^2(t)}{C(t)}\mathrm{d}t$.

Lemma 1 S_{circle} decreases with the increase of m.

Proof From Eq.(2), it is easy to know that t_m increases with the increase of m. Because S_{circle} is the integral in the area $[t_m, t]$, S_{circle} decreases accordingly with the increase of t_m. That is, with the increase of m, S_{circle} decreases.

Definition 6 The in/out rate is the average rate of joining and leaving the network in unit time. Let R be the in/out rate.

Definition 7 The maintaining overheads F comprise the cost of maintaining the structure of the topology. Let F_{DHT} be the maintaining overheads of DHT. Let F_{SHT} be the maintaining overheads of SHT.

For a network with n nodes using DHT model, we assume that the in/out rate is R_{DHT}. That is, the average number for node to join and leave the network in unit time is R_{DHT}. After clustering using the SHT model, the rate R_{SHT} would actually be divided into two parts: one is on DHT circle as represented by R_{circle}, the other is outside the circle as represented by R_{outer}.

Lemma 2 With the increase of m, R_{circle} decreases.

Proof Let T_u be the unit time. By Definitions 4, 5, and 6, $R=ST_u$. Then $R_{circle}=S_{circle}T_u$ and $R_{outer}=S_{outer}T_u$. By Lemma 1, S_{circle} decreases with the increase of m. So does R_{circle}.

Lemma 3 $R_{DHT}= R_{circle}+R_{outer.}$

Proof By Definition 3, $G=G_{circle}+G_{outer}$ and S_{outer} is the expectation of the fault rate in G_{outer}. Let the expectation of S in G, G_{circle}, G_{outer} be $E(G)$, $E(G_{circle})$, $E(G_{outer})$ respectively. Then $E(G)=E(G_{circle}+G_{outer})=E(G_{circle})+E(G_{outer})$. This means that $S=S_{circle}+S_{outer}$, so $R=R_{circle}+R_{outer}$. Because $R_{DHT}= R_{SHT} = R$, $R_{DHT}= R_{circle}+R_{outer}$.

Theorem 2 There is minimum of maintaining overheads with a reasonable cluster size m_t.

Proof The maintaining overheads for two models are listed respectively as follows (T is the running time):

$$F_{SHT}=R_{DHT}TO(\log^2 n) \tag{5}$$

$$F_{SHT}=R_{DHT}TO(m)+R_{circle}T\{O(\log^2[n/(m+1)]+O(m)) \tag{6}$$

By Lemma 3, Eq.(6) can be simplified to:

$$F_{SHT}=R_{DHT}TO(m)+R_{circle}TO(\log^2[n/(m+1)]) \tag{7}$$

Now, Let

$$F_{outer}(m) = R_{DHT}TO(m) \tag{8}$$

$$F_{circle}(m) = R_{circle}TO(\log^2[n/(m+1)]) \tag{9}$$

Then

$$F_{SHT} = F_{outer}(m) + F_{circle}(m) \tag{10}$$

Obviously, $F_{outer}(m)$ is an increasing function of m. According to Lemma 2, $F_{circle}(m)$ is a decreasing function of m. Thus there exists a reasonable value m_t to make F_{SHT} be minimal.

Theorem 3 The maintaining overheads reach the approximate minimum with a small cluster size $m_t=\min\{m|R_{circle}-R_{circle}(m+1)<e, m=1, 2, 3\ldots\}$, where e is the smooth threshold, and then the maintaining overheads are greatly reduced compared with DHT model.

Proof By Lemma 3, R_{circle} decreases with the increase of cluster size. Especially, R_{circle} dramatically decreases and then becomes smooth after a small cluster size m, which can be derived from Eq.(4) and the session distribution. Let e be the smooth threshold, with e being a positive real number small enough to reflect the stability degree of the DHT circle. Suppose $m_t=\min\{m|R_{circle}-R_{circle}(m+1)<e, m=1, 2, 3\ldots\}$. Then R_{circle} would reach the approximate minimum with m_t. By Theorem 2, F_{SHT} has a minimum. From Eq.(7), R_{circle} is the dominant factor for the maintaining overheads because m is far smaller than n. Thus F_{SHT} is the approximate minimum with the cluster size m_t. Let H be the reduced value of maintaining overheads. We have the expression:

$$H = F_{DHT}-F_{SHT} = R_{DHT}T(O(\log^2 n)-O(m)) - R_{circle}TO(\log^2[n/(m+1)]) \tag{11}$$

Since R_{circle} is a small factor and m is a small value with $m= m_t$, H would be a relatively huge value. That is, the maintaining overheads are greatly reduced compared with DHT model.

EXPERIMENTS

We designed and implemented a simulator prototype where nodes follow the steps outlined in previous section to form clusters and perform search. In particular, the construction of the DHT circle is based on Chord protocol and has the same stabilizing mechanism as Chord does (30 seconds as stabilizing interval time). Here we call it SH-Chord. Of course, we may also construct DHT circle based on CAN, Pastry, Tapestry, etc. However, it does not change the essential effects of the SHT model.

Our experiments had two parts. First we observe how the size of the cluster affects R_{circle}. Second, we observe the improved performance with the SHT model.

R_{circle} vs cluster size

We add 2000 nodes into the network. To obtain a

practical effect, the distribution of session time follows the trace data (Saroiu *et al*., 2002). To observe how the size of the cluster affects R_{circle}, the cluster size m was changed from 1 to 40. As all metadata information are published and stored in the DHT circle and search is finished within the DHT circle, R_{circle} embodies the stability of the SHT model. We measured the number of nodes each minute and calculate the average value as R_{circle}. The result (Fig.5) shows that R_{circle} first dramatically decreases but becomes smooth after m reaches about 10 as m increases. By Theorem 3, m_t is 10 when the maintaining overheads are near the minimum. Also, the m_t value could be automatically obtained in the running time of SHT systems with a feedback algorithm. This is a part of our future work.

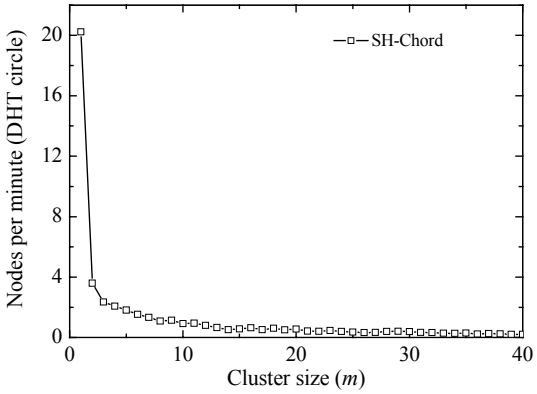


Fig.5 R_{circle} vs cluster size of 1 to 40

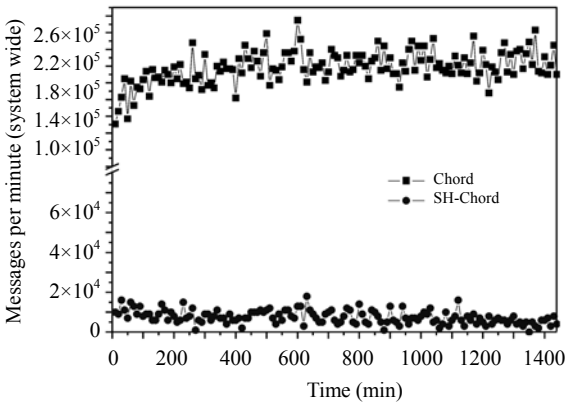


Fig.6 Comparison of maintaining overheads between SH-Chord and Chord

Performance of SHT

The SHT model reduces the maintaining overheads and makes the data more available. We designed the experiment as follows. From Experiment 1, we can observe that SHT is enough stable for a small cluster size. We constructed SH-Chord. The distribution of session time also accorded with the trace data in Experiment 1 and the cluster size was 10 according to the result of Experiment 1. Now 20000 nodes are joined to the network at the rate of one per second. To observe how the SHT model affects the maintaining overheads, this high rate is our desired network environment. The experiment was run 24 h and the system performance was recorded each 10 min during the running time.

1. Maintaining overheads

We evaluated the maintaining overheads' messages per minute in the whole system width. Fig.6 shows that the maintaining overheads in SH-Chord are about 5 thousand messages per minute, but those in Chord were 0.2 million messages per minute. Because there are 20 000 nodes in the system, it is 0.25 messages per minute per node for SH-Chord on average and 10 messages per minute per node for Chord on average. And thus the maintaining overheads have been reduced to only 2.5% of it in Chord with the clustering technique of the SHT model. Due to the clustering with m_t, DHT circle is very stable. So the maintaining overheads are greatly controlled. The comparative result is in accord with Theorem 3.

2. Lookup failure rate

The lookup failure rate and fitting curve by SH-Chord and Chord respectively are shown in Fig.7, where the dotted line is the fitting curve. It can be observed that the rate fluctuates as the time goes on, which is in accord with the fluctuation of the query executed in the extremely dynamic network (one node per second). From the fitting curve, the lookup failure rate is 0.005 by SH-Chord and 0.04 by Chord. The rate by SH-Chord being one order of magnitude less than that by Chord is due to the stable DHT circle with the SHT model. This result showed that the SHT model has better data availability than the DHT model.

RELATED WORK

The high dynamic and heterogeneous characteristics of P2P network were confirmed by the expe-

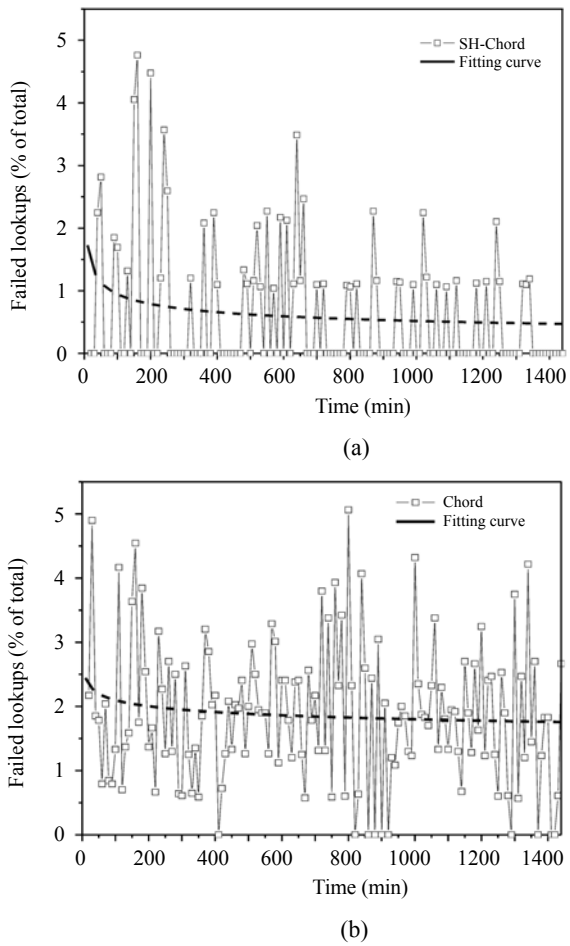


Fig.7 Lookup failure rate and fitting curve (a) SH-Chord; (b) Chord

rimental observation in recent researches (Ledlie *et al*., 2002; Bhagwan *et al*., 2003; Markatos, 2002). Since Ratnasamy *et al*.(2002) proposed the question if the performance of P2P network could be improved by exploiting the heterogeneity, some ongoing researches yielded good results by using the different heterogeneity characteristics of bandwidth (Saroiu *et al*., 2002), host (Hazel and Wiley, 2002), and geography (Wu *et al*., 2004). This paper is the first to suggest using the session heterogeneity to derive the inner heterogeneity of P2P network from a new angle and evaluate its performance by experiments.

Liben-Nowell *et al*.(2002) analyzed the dynamic characteristics of P2P network and researched the maintaining process of the topology when nodes join and leave. Mahajan *et al*.(2003) proposed a model for controlling and reducing maintaining overheads of DHT. The key idea was to adjust the maintenance measures so as to smooth the dynamic change of P2P network. In essence, it is an inactive method and maintaining overheads in highly dynamic environment may be unnecessarily excessive because too much unstable nodes have to be dealt with. In contrast, the SHT model can actively control the dynamic effect through filtering out the unstable nodes. Thus SHT can build a stable DHT circle so as to efficiently control the cost of maintaining topology.

The SHT model makes good use of the rewarding efforts of session heterogeneity. Though our analysis and experiments were based on the real trace from Saroiu *et al*.(2002) which is the earliest study we have found of session heterogeneity in P2P systems, the idea is actually applicable to a class of P2P systems with session heterogeneity. Otherwise, we must point out that the SHT model might be not so effective for systems without great heterogeneity. Nevertheless, a series of experimental results such as Bhagwan *et al*.(2003) and Sen and Wang (2002) showed that there is great session heterogeneity in currently deployed P2P systems. Therefore, the SHT model has its practical significance and would be an effective approach to make these systems more scalable and robust with only a few modifications.

CONCLUSION

DHT technique has been widely applied in P2P systems because it provides reliable services. However, great overheads are inevitable for maintaining the structure of DHT topology, which limits its application especially in highly dynamic environments. In this paper, we proposed a novel SHT model to control maintaining overheads by means of session heterogeneity in the P2P network. The SHT model uses a simple but effective clustering technique, whose main merits are:

(1) The management of clusters is very natural and simple with the evolving process. As it does not rely on any additional preconditions, it can be directly applied to current DHT algorithms.

(2) It achieves approximately optimal effect even with a small cluster size, which means that it has better scalability and makes DHT more adaptive in the dynamic networks compared with other clustering

techniques existing in current systems.

Simulation results showed that maintaining overheads can be reduced dramatically and that the lookup failure rate can be greatly decreased. Briefly, the SHT model enhances the stability of P2P systems and the data availability as well.

References

Balakrishnan, H., Kaashoek, M.F., Karger, D., Morris, R., Stoica, I., 2003. Looking up data in P2P systems. *Communications of the ACM*, **46**(2):43-48.

Bhagwan, R., Savage, S., Voelker, G.M., 2003. Understanding Availability. The 2nd International Workshop on Peer-to-Peer Systems. Berkeley, CA, USA.

Druschel, P., Rowstronand, A., 2001. Pastry: Scalable, Distributed Object Location and Routing for Large-scale Peer-to-Peer Systems. Proceedings of the 18th IFIP/ACM International Conference on Distributed Systems Platforms. Heidelberg, Germany, p.329-350.

Hazel, S., Wiley, B., 2002. Achord: A Variant of the Chord Lookup Service for Use in Censorship Resistant Peer-to-Peer Publishing Systems. Proceedings of the 1st International Workshop on Peer-to-Peer Systems (IPTPS'02). MIT Faculty Club, Cambridge, MA, USA.

IRIS Project, 2004. http://project-iris.net/.

Ledlie, J., Taylor, J., Serban, L., Seltzer, M., 2002. Self-Organization in Peer-to-Peer Systems. The 10th ACM SIGOPS European Workshop.

Liben-Nowell, D., Balakrishnan, H., Karger, D., 2002. Analysis of the Evolution of Peer-to-Peer Systems. Proceedings of the Twenty-First Annual Symposium on Principles of Distributed Computing. ACM Press, p.233-242.

Limewire, 2004. http://www.limewire.com.

Mahajan, R., Castro, M., Rowstron, A., 2003. Controlling the Cost of Reliability in Peer-to-Peer Overlays. The 2nd International Workshop on Peer-to-Peer Systems. Berkeley, CA, USA.

Markatos, E.P., 2002. Tracing A Large-Scale Peer to Peer System: An Hour in the Life of Gnutella. The 2nd IEEE/ACM International Symposium on Cluster Computing and the Grid. IEEE Computer Society, Washington, DC, USA.

Morpheus, 2004. http://www.musiccity.com.

Ratnasamy, S., Shenker, S., Stoica, I., 2002. Routing Algorithms for DHTs: Some Open Questions. Proceedings of the 1st International Workshop on Peer-to-Peer Systems (IPTPS'02), Cambridge, MA, USA.

Saroiu, S., Gummadi, P.K., Gribble, S.D., 2002. A Measurement Study of Peer-to-Peer File Sharing Systems. Proceedings of Multimedia Conferencing and Networking. San Jose, CA.

Sen, S., Wang, J., 2002. Analyzing Peer-to-Peer Traffic Across Large Networks. Proc. of ACM SIGCOMM Internet Measurement Workshop. ACM Press, New York, NY, USA.

Stoica, I., Morris, R., Karger, D., Kaashoek, F., Balakrishnan, H., 2001. Chord: A Scalable Peer-to-Peer Lookup Service for Internet Applications. Proc. of ACM SIGCOMM. San Diego, CA.

Sylvia, R., Paul, F., Mark, H., Richard, K., Scott, S., 2001. A Scalable Content-Addressable Network. Proc. ACM SIGCOMM. San Diego, CA, p.161-172.

Wu, Z.D., Ma, F.Y., Rao, W.X., 2004. Super-proximity routing in structured P2P networks. *Journal of Zhejiang University SCIENCE*, **5**(1):16-21.

Zhao, B.Y., Huang, L., Stribling, J., Rhea, S.C., Joseph, A.D., Kubiatowicz, J., 2004. Tapestry: A resilient global scale overlay for service deployment. *IEEE Journal on Selected Areas in Communications*, **22**(1):41-53.