Journal of Theoretical and Applied Information Technology
20th May 2013. Vol. 51 No.2

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

A KIND OF ROUTING MODEL IN PEER-TO-PEER NETWORK BASED ON SUCCESSFUL ACCESSING RATE

[1]TAO LIU, [2]JI-JUN XU

[1]College of Information Science and Technology, Zhengzhou Normal University, China

[2]School of Mathematics and Statistics, Zhengzhou Normal University

E-mail: [1]liutao770910@126.com

ABSTRACT

The peer-to-peer network is based on location mechanism of distributed objects, which is conducive to sharing of resources among users. The commonly using overlay network topology method based on interesting group in the existing peer-to-peer network should be accomplished collaboratively by users, which increases system traffic and occupies large bandwidth. The paper brought out a kind of routing model in peer-to-peer network based on Successful Accessing Rate (SAR). The model organizes user group based on SAR and conducting resource searching in the high-low order of SAR. In the model, each node runs independently, which decrease system maintenance cost and provides new idea to solve free-rider problem. Simulation results show that the model effectively improve recall rate of system and also solve fairness problem.

Keywords: *Peer-To-Peer Network, Routing; Successful Accessing Rate*

1. INTRODUCTION

The emergency of peer-to-peer (P2P) network comes from requirements of users on sharing and adequately usage of Internet resources. The most prominent character of P2P network is that users share resources directly, the core of which is positioning mechanism of distributed objects. There are several routing methods currently as following. The first is centralized indexing manner as Napster system [1]. It has problems of single point failure as well as performance bottlenecks, so it cannot be applied in large-scale network. The second is structural P2P system based on distributed Hash table DHT, such as Chord and CAN systems [2, 3]. It can achieves high efficiency searching but only support searching based on accurate object key value and lack of fuzzy searching ability. It also does not support complex indexing based on semantic. On the contrary, unstructured P2P network as Gnutella [4-7] can better support various forms of information indexing. However, the searching strategy based on flooding or random roar increase messages on P2P network, which restricts scalability of P2P network.

In order to improve routing performance of unstructured P2P network, the method to construct overlay topology based on interest group is mainly used now [8-10]. Its main basis is that nodes with same interest has common request on resources and storage. The users with same interests are linked together by overlay network. In the searching, it is easier to find resources meet requirements in local interest group so as to improve successful rate for resource positioning. The construction of overlay network based on interest group needs users publish own interest or more specific file character in the network, and then establish interest group by indexing or self-organization. We can see that the method can only be accomplished by cooperation among users, which increases system communication amount and occupies large bandwidth.

The successful accessing rate-based (SARB) peer-to-peer network routing model in this paper organizes based on node SAR. It looks for resource in the high-low order of successful accessing rate. Nodes in the model operate independently, which decreases system maintain cost and provide new idea to solve free-rider problem. Simulation results show that the model can effectively improve system recall and better solve system fair problem at the same time. The paper is organized as follows: section 2 describes SARB model; section 3 gives specific SARB algorithm; section 4 conduct performance analysis and simulation on SARB; section 5 concludes our work.

Journal of Theoretical and Applied Information Technology
20th May 2013. Vol. 51 No.2

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

2. SARB MODEL DESCRIPTION

The starting point of SARB is that more times of node storage resources been successfully accessed, the node has higher SAR. Other nodes have higher probability to find needed resources on this node. Therefore, organize nodes with similar successful accessing rate in the system together. In case of searching, firstly access to node with larger SAR so that the resource recall increases.

Assume there are N nodes in the system marked as $P_1, P_2, \cdots, P_N$. Use D_i to represent successful accessing rate of node P_i, the initial value of D_i is 0. In the system operation process, other nodes $P_j (0 \le j \le N, j \ne i)$ successfully access resources from node P_i, then D_i increases correspondingly. Divide node successful accessing rate into K levels $0, 1, \cdots, K-1$ based on value. The boundary value is $D^0, D^1, \cdots, D^{K-1}$. If $D^m \le D_i < D^{m+1}$, d_i is the m-th level and $d_i = d(D_i) = m$. Using T as time interval, compute current successful accessing rate D_i of current node P_i and determine its level.

To facilitate description, the cyclic structure is used as basic structure of SARB. After node enters into system, assign a node sign for each one to identify it uniquely. Taking node i as example, when the node i enters into system, it uses the manner of normal cyclic structure. At intervals of time T, compute successful accessing rate D_i of node at current time and divide its level. Here we assure level d_i is m. Find nodes in the system. If there is other nodes whose successful accessing rate level is m, the node i also enter into cyclic structure of this level. In this way, each node actually has two logic positions in the system. It also has two routing selection information. The final structural diagram is shown in Fig. 1.

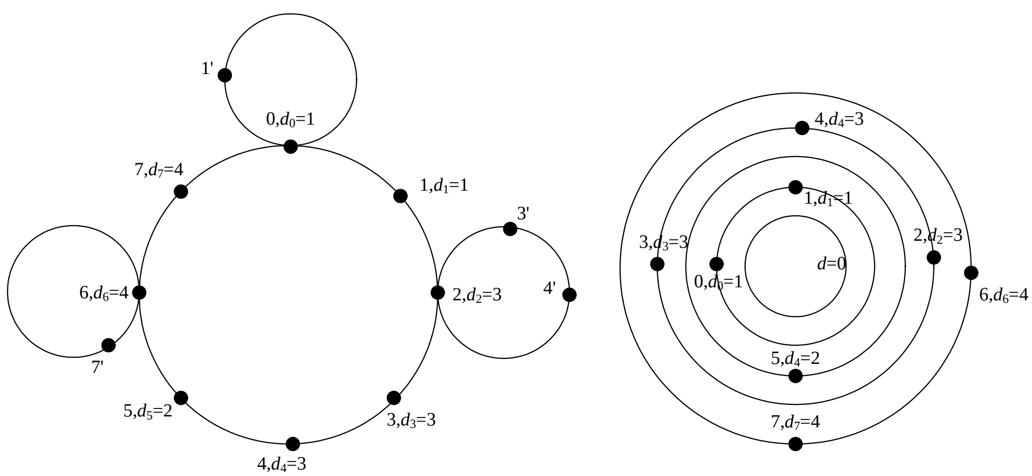

Figure 1: SARB logical structure with cyclic structure

As shown in Fig. 1 in the system with 8 nodes, the successful accessing rate level where node 5 locates only has the node itself, while node (0, 1), (2, 3, 4) and (6, 7) respectively belong to same successful accessing rate. $1', 2', 3', 7'$ respectively represent another logical position of node 1,2,3 and node 7.

3. SARB ROUTING ALGORITHM

3.1 Routing Table Structure

Each node in the SARB model keeps two sets of routing information. One is routing information of node in the whole system, which is called system routing table. Another is routing information of successful accessing rate level where the node locates, which is called level routing table.

The system routing table should add an item to represent successful accessing rate level of the item successor. In addition, the successor in system routing table item is not the first existing node of this area, but the node most closely to node sign with maximum successful accessing rate level and the node identifier closest node. The routing table structure of node 0 in Fig. 1 is shown in Table 1.

Journal of Theoretical and Applied Information Technology
20th May 2013. Vol. 51 No.2

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

Table 1: Routing table structure of node 0 in Fig. 1

System routing table (finger table)		
int.	succ.	*d*
[1, 2)	1	d_1=1
[2, 4)	2	d_2=3
[4, 0)	6	d_6=4
Level routing table (finger table) *d*=1		
[1, 0)	1	

3.2 Node Status Changes

The node position in system of SARB model is determined commonly by node identifier and node successful accessing rate level. The node identifier will no longer change after node enters system, while node successful accessing rate dynamically change according to response of node on other nodes and then periodically conduct level transition. Therefore, the node status change in system mainly includes three kinds of join, leave as well as successful accessing level transition. Here these three statuses are described as following.

3.2.1 Identification of subsections

Step1: Determine own node identifier that assumed to be *p*.

Step2: contact with another member node p' in the system. The node will act as bootstrap node of new joined node *p*.

Step3: The node p' initiates looking for node *p* and find the node whose identifier only below than *p* as its previous node.

Step4: Initiate system routing table of *p*. The node successful accessing rate level d_p=0.

Step5: Update system routing table of related nodes needed for modification because of join of *p*. Under circumstance of system node number *N*, the node number should be updated is $O(\log N)$.

Step6: Based on own system routing table, the node *p* find nodes with same successful accessing rate level and join level cyclic structure to construct level routing table.

3.2.2 Process of node leaves system

Assumed node identifier *p*, and the level of successful accessing rate d_p=*m*.

Step1: Conduct leaving operation according to level routing table in the cyclic structure whose successful accessing rate level is *m*. Update level routing tables of other related nodes correspondingly.

Step2: Complete leaving operation according to system routing table in the system and update system routing tables of related nodes correspondingly.

3.2.3 Process of node level transition

Assume *p* is the current node and successful accessing level d_p=*m*. The level to be transited is $d_p' = n$.

Step1: Conduct leaving operation according to level routing table in the cyclic structure whose successful accessing rate level is *m*. Update level routing tables of other related nodes correspondingly.

Step2: Modify successful accessing rate level of node *p* to *n*.

Step3: Update system routing tables of related nodes correspondingly. In the Fig. 1, assume node successful accessing rate level of node 7 transits from level 4 to level 5, then routing table update of node 0 updates as Table 2 shows.

Step4: The node *p* finds nodes whose successful accessing rate level is n according to own system routing table. Join this level cyclic structure and build level routing table.

3.3 Routing Algorithm

The basic idea of SARB routing algorithm is as following. Source node send query message with initial TTL value to network. Firstly send message to node with highest successful accessing rate level according to system routing table. Then search in the same level based on level routing table, and

Journal of Theoretical and Applied Information Technology
20th May 2013. Vol. 51 No.2

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

return query result to source node after found corresponding target.

Table 2: Routing table of node 0 after node 7 transit level

System routing table (finger table)		
int.	succ.	d
[1, 2)	1	d_1=1
[2, 4)	2	d_2=3
[4, 0)	7	d_6=5
Level routing table (finger table) d=1		
[1, 0)	1	

As the node system routing table stores nodes with highest successful accessing rate level in the whole system area, the system routing table of node sending request can find the node with highest successful accessing rate level in the system. Therefore, it only needs one step from source node to node cluster with highest successful accessing rate level. The algorithm is shown in Fig. 2.

Algorithm 1

Assume P is current node. Input is query message Q=*query*(*message*, *Src*, *TTL*). Where, *message* is query semantic description; *Src* is query message initiate node; *TTL* is maximum lifetime of this query message. The output is target result R to *Src*.

```
if (P==Src)
    Look for node with maximum level successful accessing rate in local system routing table and forward query message;
else
    Set Q.TTL=Q.TTL-1;
    Execute local query, return query result to Src;
    if (Q.TTL>0)
        Send query to other nodes in same successful sccessing rate elvel with P according to level routing table;
    end if
end if
```

Figure 2: Specific steps of algorithm 1

From the algorithm 1 we can see that the nodes with highest successful accessing rate level are accessed in each time of query, which will increase overhead of these nodes. However, the nodes with lower level only have fewer times been accessed, which also not conductive for increase of node level. Therefore, the paper brought out SARB routing algorithm 2.

In the query process of algorithm 2, source node adds successful accessing rate level of it in the query packet. The query was carried out on same success rate level with source node except for highest level. The contribution level of routing is equal to or lower than node request from query node. On the contrary, it can refuse the request with certain probability. The algorithm 2 is shown in Fig. 3.

The node j accepts resource request from node i with probability $p(d_i - d_j)$ and the probability $1 - p(d_i - d_j)$ to refuse the request. The computation of probability $p(d_i - d_j)$ is as following:

$$p(d_i - d_j) = \begin{cases} 1 & d_i \geq d_j \\ \dfrac{1}{\lambda \left| d_i - d_j \right|} & d_i < d_j \end{cases} \qquad (1)$$

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

Algorithm 2

Assume P is current node and d_p is successful accessing rate level of node P. Input is query message Q=$query(message, Src, TTL, d_{src})$. Where, *message* is query senmatic description; *Src* is query message initiate node; *TTL* is maximum lifetime of query message; d_{src} is successful accessing rate level of source node. Output target result is R to *Src*.

```
if (P==Src)
    Forward query message according to local level routing table;
    Look for node with maximum level successful accessing rate in local system routing table and forward query message;
else
    set Q.TTL=Q.TTL-1;
    Execute local query with probability p(dp-dsrc) and return query result to Src;
    if (Q.TTL>0)
      Send query to other nodes in same successful sccessing rate elvel with P according to level routing table;
    end if
end if
```

Figure 3: Specific steps of algorithm 2

Where, λ is constant coefficient factor and its value is in the interval of (1, 2]. Therefore, if the successful accessing rate level d_i of node i is small, the node with higher level than node i likely refuse request from node i.

We can see that the above two algorithms use abler way. In the algorithm 2, higher level node can refuse response to query request for decreasing overhead. Meanwhile, it also needs to constantly improve own level by responding to query requests. The node with lower level can only access to opportunity to response request of nodes in same level by sending request, so as to increase successful accessing rate level. Therefore, algorithm 2 is a kind of routing algorithm with reward mechanism. In addition, the above two kinds of routing algorithms may not find necessary resources, which is determined by query on resources by resource node according to unstructured P2P network flooding or random walk way.

3.4 Supervise Mechanism

Nodes in the SARB routing algorithm automatically compute its own successful accessing rate and perform level transition according to value of success rate. When the node in algorithm 2 receives query request, it determines whether to answer this request from difference between successful accessing rates from send and receive node. It may have forge behavior for node initiated query to obtain higher response rate. In view of this situation, the successful accessing rate neighborhood monitoring mechanism (NMM) was proposed.

The main idea of NMM is to constrain node behavior relaying on monitoring mechanism from neighbors. The specific description is as following. Each node keeps all neighbor statuses. The neighbor status is determined by answer situation of neighbor on query request, which can be divided into two kinds of met or un-met. The node dynamically maintains neighbor based on recorded neighbor status. If a neighbor is in met status, normally routes it. Otherwise, unlink connection to this neighbor. If all adjacent nodes unlink connections with this node, it is in isolated status in system. So it cannot perform querying or routing, so as to avoid forgery behaviors.

The method does not need large amount of information interaction among nodes in EigenTrust algorithm. Furthermore, dynamic changing feature of node successful accessing rate level enable neighbor nodes not easily form collusion. It is simple for implementation and has good scalability [11-13].

4. PERFORMANCE ANALYSIS

4.1 Evaluation Indexes

The paper uses following two indexes to evaluate its semantic query performance, namely recall $T(Q)$ and average message number generated in querying.

Recall is the proportion of searched documents number met conditions in all related documents, the formula of which is as follows:

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

$$T(Q) = \frac{D_{getcorrect}}{D_{total}} \quad (2)$$

Where, D_{total} is set of all related documents distributed on P2P network aiming at some query; $D_{getcorrect}$ is the actually obtained document set.

The queried average message number $\bar{M}$ means average generated message number in the propagation process on P2P network. Its size determines network overhead change caused by query operation, which is important index to measure network structure.

4.2 Algorithm Analysis

Assume there are N nodes in the system that marked as $P_1, P_2, \cdots, P_N$. The node successful accessing rate level can be divided into K levels $(0,1,\cdots,K-1)$ according to value. The successful accessing rate level of node P_i d_i=m. The node number in this level is *num*(*m*). The highest level of node successful accessing rate level in current system is *n*. Node number in this level is *num*(*n*).

When the node P_i initiate query Q, the average message number generated by queries by algorithm 1 is $\bar{M} = 1 + \log(\text{num}(n))$. In the algorithm 2, the average message number generated by queries is $\bar{M} = 1 + \log(\text{num}(n)) + \log(\text{num}(m))$. We can see that the average query message number is larger than that of algorithm 1.

Clearly, the routing algorithm under SARB structure is significantly lower than flooding mechanism in Gnutella in the aspect of query average message number.

4.3 Performance Simulation

In the simulation experiment, GT-ITM topology generator was used to generate random network topology. The transit-stub (TS) model that is more representative of Internet structure was also used. Network node size is 1000. Distribute documents for each node according to Zipf distribution with parameter $\alpha = 1.2$. The required resources in query also initiates according to Zipf distribution with parameter $\alpha = 1.2$. Assume each node has at most 100 documents in the test. All the results are average value under 100000 times of queries. The comparison of recall between two algorithms and Gnutella with different TTL is shown in Fig. 4.

From the figure we can see that the recall of algorithm is bigger than that of algorithm 1 in lower TTL value as it use the strategy that synchronous query of own level with highest level. However, with increase of TTL value, as algorithm 2 has situation that high level refuse to access lower level service, the recall is lower than that of algorithm 1. Under circumstance of TTL<6, the routing algorithm based on SARB structure proposed in the paper is significantly higher than Gnutella, which shows its superiority.

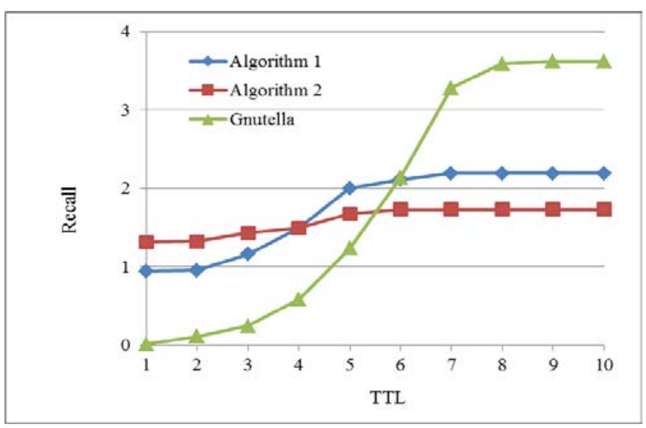

Figure 4: Comparison of recall in algorithms under different TTL conditions

5. CONCLUSION

In order to improve routing performance of unstructured P2P network, the method to construct overlay topology based on interest group is mainly used now. The method can only be accomplished by cooperation among users, which increases system communication amount and occupies large bandwidth. The peer-to-peer network routing model SARB based on successful accessing rate just organize nodes with similar successful accessing rate together. It is firstly conducted in nodes with larger successful accessing rate so that the resource

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

recall can be increased. The paper also gives basic structure of SARB as well as node status maintenance and two kinds of routing algorithm. Simulation experiments show that this structure can significantly improve recall under smaller TTL compared with Gnutella.

REFRENCES:

[1] 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*, Vol. 11, No. 1, 2003, pp. 17-32.

[2] Sylvia Ratnasamy, Paul Francis, Mark Handley, Richard Karp, Scott Shenker, "A scalable content addressable network", *In Proceedings of ACM SIGCOMM*, 2001, pp. 161-172.

[3] Christos Gkantsidis, Milena Mihail, Amin Saberi, "Random walks in peer-to-peer networks", *In Proceedings of IEEE INFOCOM*, 2004, pp. 120-130.

[4] Arturo Crespo, Hector Garcia, "Random walks in peer-to-peer networks", *In Proceedings of the Third international conference on Agents and Peer-to-Peer Computing*, 2004, pp. 1-13.

[5] Sepandar D. Kamvar, Mario T. Schlosser, Hector Garcia-Molina, "The eigentrust algorithm for reputation management in P2P networks", *In Proceedings of the 12th international conference on World Wide Web*, 2003, pp. 640-651.

[6] Wang Yan-liang, "Research on Small World Model in P2P Networks", *Advances in Information Sciences and Service Sciences*, Vol. 4, No. 2, 2012, pp. 93-100.

[7] Shi-dong Zhang, Geng-yu Wei, Bai Wang, De-yu Yuan, "An Optimization Method for P2P Resource Scheduling", *International Journal of Digital Content Technology and its Applications*, Vol. 6, No. 4, 2012, pp. 159-165.

[8] Chuan Xu, Hong Tang, CongCong Xing, "P2P Traffic Localization Based on Peer-List Caching", *International Journal of Advancements in Computing Technology*, Vol. 4, No. 5, 2012, pp. 272-279.

[9] Zhenling Wang, "A Novel Peer to Peer Traffic Identification Approach based on Hybrid Feature Selection Algorithm", *International Journal of Digital Content Technology and its Applications*, Vol. 6, No. 8, 2012, pp. 204-212.

[10] Hung-Wen Yang, Chou-Chen Yang, Yi-Ju Lin, Woei Lin, "A Novel Authorized File-Sharing Protocol for P2P Networks", *Advances in Information Sciences and Service Sciences*, Vol. 4, No. 8, 2012, pp. 76-83.

[11] Zhigang Chen, Limiao Li, Jingsong Gui, "Fuzzy Theory for the P2P Subject Trust Evaluation Model", *International Journal of Advancements in Computing Technology*, Vol. 4, No. 8, 2012, pp. 67-74.

[12] Jiaqi Liu, Zhigang Chen, Deng Li, Zhong Ren, "RELO: a Robust, Efficient and Low-cost Peer-to-Peer Overlay for Autonomic Communication", *International Journal of Advancements in Computing Technology*, Vol. 4, No. 6, 2012, pp. 50-58

[13] Ju-Yin Chuang, Pei-Ling Hsu, Yi-Shin Chen, "Improving Broadcast Processes in Peer-to-Peer Systems", *Advances in Information Sciences and Service Sciences*, Vol. 4, No. 8, 2012, pp. 187-198