

P2P Model Based on Isolated Broadcast Domains

Jianchun Li, Gongbing Guo, Jianyong Li, Daoying Huang

School of Computer & Communication Engineering, Zhengzhou University of Light Industry, Zhengzhou, China

Email: lijianchun@zzuli.edu.cn

Received March 19, 2012; revised April 20, 2012; accepted April 29, 2012

ABSTRACT

Resource location is the most important issue for Peer-to-Peer (P2P) system and flooding is the fundamental mechanism for unstructured P2P systems. Redundant messages will exponential growth with flooding scope increasing which severely influences the scalability of the system. In this paper, a new P2P model based on isolated broadcast domains is given to reduce the amount of redundant messages by limiting the radius of messages transmitted. Analysis and experiments show that this model can guarantee coverage of nodes and significantly reduce the amount of redundant messages generated.

Keywords: P2P Model; Isolated Broadcast Domains; Redundant Messages

1. Introduction

P2P is an application mode built on Internet to share resource includes files, hardware and other information among nodes and emphasizes the equal status of nodes in the system. Relationship between peers is loose and the system keeps absolute anarchy structure that means every node can't control others and not be controlled by others. For each node to join and leave system absolute free which bring troubles of management.

The emergence of P2P has impact Internet greatly. Over the past decade, applications based on P2P thought have been developed rapidly. Network traffic produce by P2P system on Internet has been the majority and still grow rapidly. Studies show that the proportion of traffic produced by P2P in the backbone of Internet has increased from 10% to 80% [1-3].

Systems based on P2P have different resource location mechanisms. Napster is the first P2P system which depends on a central index server to locate resource. When node in Napster wants to search a resource, it initiates a query to ask the server who has the resource and the server will tell the node where to find the resource. The question of Napster is the existence of central index server which makes it is not a pure P2P system and depends on the server to locate resource. The server will be a bottleneck when there are many queries need to be answered. If there are copyrights of resource, the server will be controlled, such as Napster was closed for copyrights. Some P2P systems, such as Chord [4], CAN [5] and Pastry [6] etc. depend on DHT (Distributed Hash Table) to locate resource. DHT is a distributed storage method without server, in order to achieve the DHT network addressing and storage, each peer is responsible for a small range of routing and a small part of the data

stored. Instead of maintaining indexes in centralized system or mapping resource to keys, node in decentralized system such as Gnutella [7], broadcasts query to its neighbors and this way to propagate messages is flooding. Flooding is the fundamental mechanism to search resource in unstructured P2P system. If a node in Gnutella initiates a query, it will broadcast the query to its neighbors in the system and neighbors received the query will repeat the same action until find the aim node or reach the limit of search. Nodes in flooding system may receive the same query many times which will generate lots of redundant messages. The existence of large number redundant messages makes flooding mechanism have serious bottleneck. Many solutions have been proposed to improve the bottleneck question. Yang and Garcia-Molina proposed iterative deepening method and breadth-first search (direct BFS) [8]. Based on nodes' degree distribution in Gnutella network presents a small world and power law distribution characteristics [9], algorithm of use the power-law distribution of nodes' degree properties to search resource was proposed by Adamic *et al.* in literature [10,11]. LightFlood algorithm was proposed in [12] according to the result of experiment that most redundant generated by last few hops. In this paper we give the search algorithm based on isolated broadcast domain (SAOIBD) to reduce redundant messages.

2. Search Algorithm Based on Isolated Broadcast Domains

2.1. Search Algorithm Based on Isolated Broadcast Domain

Search algorithm based on isolated broadcast domain (SAOIBD) is intended to reduce the amount of redundant

messages by control the scope of flooding. The basic idea of SAOIBD is node only sends queries to certain nodes which are in the same domain in its neighbors' information table, by this limit, queries are controlled in small scope when they propagated by flooding.

During a query propagated, more and more nodes receive the query and transmit it to their neighbors, thus, the probability of nodes to receive the same query increases rapidly, which is the reason described in [12] that most redundant messages generated by last few hops. To reduce the impact of last few hops, some measures have been taken in [8,12].

An obvious method to reduce redundant messages is to shrink scope of flooding to reduce the probability of nodes receiving same queries.

In SAOIBD, we divide nodes in the system into different groups, messages mostly propagated in groups, inside groups, flooding is used to search resources.

When nodes receive and transmit a query to their neighbors, they only transmit the query to neighbors which are in the same groups or neighbors which not belong to its group, but are super nodes for other groups, that is to say in groups, only those super nodes can receive queries form other nodes outside their groups, common nodes only receive messages from members in their groups. Suppose there are n nodes in a P2P system based on flooding mechanism, these nodes will be divided into m groups, each group has n/m nodes, messages will be flooding in each group and the radius of flooding can be reduced and redundant messages will be also reduced. Since messages can only be flooded in groups, we call groups as isolated broadcast domain (IBD). P2P system divided into IBD shows as in **Figure 1**. Given an id as GID to each group, some groups in the system may have the same GID but not be connected together, so nodes have the same GID are not necessarily belong to the same group, in fact, they belong to different IBD. If all nodes in system belong to one domain or each domain has only one node, the system will equal to system without domain divided.

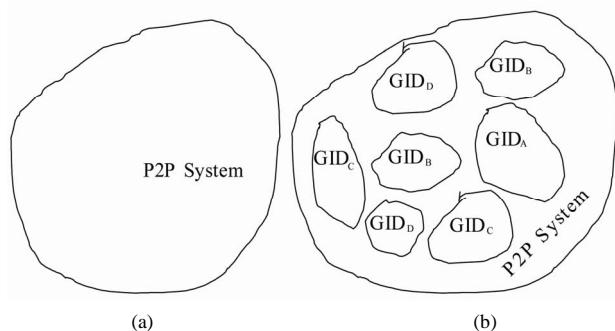


Figure 1. Figure for isolated broadcast domains: (a) System not divided; (b) System divided into group.

2.2. P2P Model Based on SAOIBD

2.2.1. Nodes Join into System

When node A joins into system, it will generate a group id at random, such as GID_A . If there are no neighbors belong to group GID_A , then A declares it is the super node of group GID_A . If A find group GID_A , it computes the distance to the super node in GID_A to decide whether it will join GID_A . If it finds it is far away from super node in group GID_A , which means more hops would be needed when messages transmit from super node and more redundant messages would be generated, then it would search another nearest super node and join into the group, else it will join into group GID_A and exchange its neighbors' table with its neighbors. If the node can not find a satisfied group, then it generates a new id again, and repeats the above action. If the node finds that there are many groups has same id can be joined into, it will choose one that has nearest super node.

2.2.2. Nodes Leave System

Pure P2P systems are distribute systems and nodes in systems have equal roles. Each node has no right to control others and nodes join and quit randomly. If a super node leaves normally, it will select the neighbor which has biggest degrees as super nodes to replace it and notice nodes in the domain. If common nodes leave system, it notices its neighbors to update their neighbors' information table. If super node quits abnormally, then the IBD loses its connection to outside world and needs to choose a new super node for the domain to receive messages from outside nodes. In order to monitor the super node leave system abnormally, it needs to keep communication regularly between super nodes and common nodes. While some nodes find super node has no response to their keep messages, each of these nodes announces it is the super node, if there is in conflict on who will be the super node, the question will be solved during the follow query propagated and a new super node will be elected.

2.2.3. Routing Algorithm on IBD

Every node has two kinds of neighbors, one kind is belongs to the same domain and the other is not. When nodes receive a query, it search in its resource list to find the resource described in the query, if it finds the resource, the query will be discard and not be transmitted again. If nodes can not find the resource in its resource list, it will transmit the query to some of its neighbors. Nodes received a query will search its neighbors information table and transmit the query to neighbors belong to the same domain or neighbors which are super node for other domains. If node searches its neighbor information table and not find neighbors in same domain and neighbors are super node for other domain, then it will

discard the query and not transmit again.

In order to accelerate the propagation of queries and improve coverage, the propagation can be divided into two stages. As show in [12], in the forth four steps of flooding, redundant messages generated is very few, most of them generated by the last few steps. So an acceptable method is to use pure flooding at first stage and flooding in isolated broadcast domain at second stage. So when node initiates a query, the query will be propagated by pure flooding at first. If the query has been transmitted four times, it will not be propagated by pure flooding again and will be propagated by SAOIBD.

2.3. Analysis of the Model Based on SAOIBD

2.3.1. Redundant Messages Generated

Set the average degrees of the system is d_v , degrees of node i is d_i and messages generated by a query propagated to reach n nodes is M in flooding system. It is easy to implement adding a limit to pure flooding that nodes don't transmit query to neighbors it receives the query from. The network can be transformed into a graph with n nodes and nd_v edges, when a query propagated, it will be transmitted on each edge only one time that generates one message, so the number of messages generated in the network is equal to edges' amount, thus:

$$M = \frac{1}{2}nd_v = \frac{1}{2}\sum_{i=1}^n d_i \quad (1)$$

In Equation (1), it shows that the amount of messages has relation with the number of nodes and degrees of nodes. Ideally to access n nodes only needs n messages, set redundant messages is R , thus:

$$R = \frac{1}{2}\sum_{i=1}^n d_i - n \quad (2)$$

Equation (2) shows for the same coverage, to decrease the degree of nodes will reduce the redundant messages. In SAOIBD systems, by dividing the coverage into different IBD and allowing nodes to transmit most messages only in domain cut off many edges connection between different domains, so messages transmitted among domains are eliminated. Set d'_i is the average number a node will transmit messages to its neighbors and messages generated when a query propagated to reach n nodes is M' in SAOIBD system, thus:

$$M' = \frac{1}{2}\sum_{i=1}^n d'_i \quad (3)$$

Since nodes in SAOIBD system transmit messages to neighbors in same domain or super nodes of other domains, it is obvious that for each node i in SAOIBD system there is $d_i \geq d'_i$, so $M \geq M'$.

2.3.2. Compatibility with Flooding

SAOIBD is based on IBD. The system is divided into

many domains, one extreme case is each node is a domain and another is all nodes belong to the same domain, in both cases, SAOIBD equals to flooding.

From the above analysis we can know that SAOIBD is well compatible with flooding system. Too many or few domains division will reduce efficiency of SAOIBD. The reason is that more nodes in one domain will enlarge the scope of flooding, so more redundant messages generated and less nodes in one domain leads to more domains in system and cause more super nodes in neighbors' information table.

2.3.3. Cost of SAOIBD

Nodes join into system need to find a group or generate a new group, for this action only need to communicate to its neighbors and not trouble others, so it has little impact on system.

Nodes leaving system includes two conditions, for normally leave, the cost is very limit. If the super node leaves the domain, it need time to elect a new one. When a new super node is not elected, the domain may have no super node or many. If there is no super node, the domain will lost connection with other domains, some queries can't be received. If there are many super nodes, nodes in the domains will received the same query many times. In order to monitor the existence of super node, communication will be kept.

So the cost of SAOIBD is more than pure flooding, but the cost will be limited in local area and has little impact on the system.

2.3.4. Coverage of Nodes

If there is no domain loses its super node, the coverage of SAOIBD will not be reduced.

3. Experiment

We simulate P2P system by two stages which use flooding at initial four hops and SAOIBD at follow hops. In **Figure 2** the line gives the percentage of nodes access at the end of i -th hop in different scale domain divided, such as 50, 100, 150 and 200. It shows in **Figure 2** that flooding has no obvious advantage on nodes' coverage. **Figure 3** shows the ratio of redundant messages generated by different domain scale and flooding. It shows the advantage of SAOIBD on reducing redundant messages is obvious in **Figure 3**.

4. Conclusion

Flooding is a fundamental mechanism in unstructured distribute networks. Such as in unstructured P2P systems, flooding is used to search files, in some routing algorithm to propagate routing information and especially in martial application, if the network is severely damaged, in order to ensure the robustness of network, flooding is

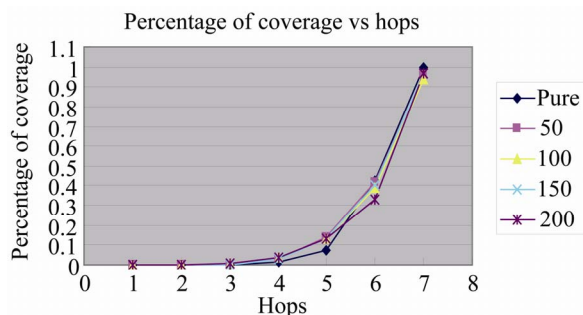


Figure 2. Percentage of coverage at the end of i -th hop.

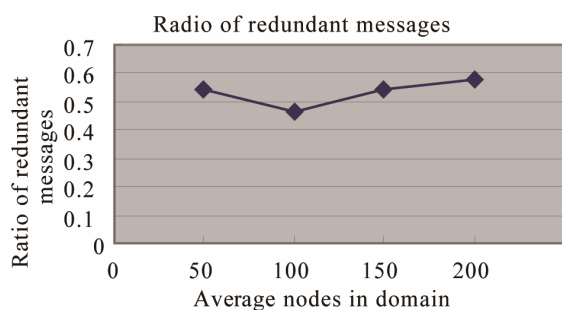


Figure 3. Ratio of redundant messages generate by SAOIBD and pure flooding.

needed, so the study of flooding is meaningful. The way of flooding propagating messages is similar to broadcast which will generate many redundant messages during messages transmitting and limit the scalability of the system. To reduce the amount of redundant messages, in this paper, we propose the algorithm SAOIBD by dividing nodes in system into different IBD and limiting messages only propagated by flooding in domains. Smaller scope of flooding will generate less redundant messages. Experiment and analysis show that SAOIBD is effective to reduce redundant messages and maintains the coverage of nodes. While one IBD includes one node or all nodes, SAOIBD will transform to flooding. The scale of domain will influence the efficiency, so the best scale of domain in given system using SAOIBD will be studied.

5. Acknowledgements

This work was supported by the National Natural Science Foundation of China under Grant 60974005 and 60704004, the Specialized Research Fund for the Doctoral Program of Higher Education under Grant 20094101120008, the Nature Science Foundation of Henan

Province under Grant 092300410201 and Zhengzhou Science and Technology Research Program under Grant 0910SGYN12301-6.

REFERENCES

- [1] A. Prker, "The True Picture of Peer-to-Peer Filesharing," 2005. <http://www.cachelogic.com/resarch/slide9.php>
- [2] T. Karagiannis, A. Broido, M. Faloutsos, *et al.*, "Transport Layer Identification of P2P Traffic," *Proceedings of the 4th ACM SIGCOMM Conference on Internet Measurement*, Taormina, October 2004, pp. 121-134.
- [3] T. Kragiannis, A. Broido, N. Brownlee, *et al.*, "Is P2P Dying or Just Hiding?" *Global Telecommunications Conference*, Riverside, 29 November-3 December 2004, pp. 1532-1538.
- [4] I. Stoica, *et al.*, "Chord: A Scalable Peer-to-Peer Lookup Service for Internet Applications," *Proceedings of the 2001 Conference on Applications, Technologies, Architectures, and Protocols for Computer Communications*, San Diego, 27-31 August 2001, pp. 149-160.
- [5] S. Ratnasamy, P. Francis, M. Handley, R. Karp and S. Shenker, "A Scalable Content-Addressable Network," *Proceedings of the 2001 Conference on Applications, Technologies, Architectures, and Protocols for Computer Communications*, San Diego, August 2001, pp. 161-172.
- [6] A. Rowstron and P. Druschel, "Pastry: Scalable, Distributed Object Location and Routing for Large-Scale Peer-to-Peer Systems," *Proceedings of Middleware 2001*, Heidelberg, November 2001.
- [7] Gnutella Website. <http://www.gnutella.com>
- [8] B. Yang and H. Garica-Molina, "Improving Search in Peer-to-Peer Networks," *International Conference on Distributed Computing*, Stanford, 2002, pp. 5-14.
- [9] A. Mihajlo and B. S. Jovanovi'c, "Modeling Large-scale Peer-to-Peer Networks and a Case Study of Gnutella," University of Cincinnati, Cincinnati, 2000.
- [10] L. A. Adamic, R. M. Lukouse, A. R. Puniyani and B. A. Huberman, "Search in Power-Law Networks," *Physical Review E*, Vol. 64, 2001, Article ID 046135. [doi:10.1103/PhysRevE.64.046135](https://doi.org/10.1103/PhysRevE.64.046135)
- [11] L. A. Adamic, R. M. Lukouse, A. R. Puniyani and B. A. Huberman, "Local Search In Unstructured Networks," In: S. Bornholdt and H. G. Schuster, Eds., *Handbook of Graphs and Networks*, Wiley-VCH, Berlin, 2003.
- [12] S. Jiang, L. Guo and X. D. Zhang, "LightFlood: An Efficient Flooding Scheme for File Search in Unstructured Peer-to-Peer Systems," *International Conference on Parallel Processing*, Williamsburg, 9 October 2003, pp. 627-635.