

# Model of Controlling the Hubs in P2P Networks

Yuhua Liu\*

Department of Computer Science,  
Huazhong Normal University,  
Wuhan, China  
Email: yhliu@mail.ccnu.edu.cn

Kaihua Xu

Research Center of the Digital Space Technology,  
Huazhong Normal University,  
Wuhan, China  
Email: xukh@phy.ccnu.edu.cn

Chun Yang

Department of Computer Science,  
Huazhong Normal University,  
Wuhan, China  
Email: c\_yang@yeah.net

Naixue Xiong

Department of Computer Science,  
Georgia State University,  
Atlanta, Georgia, USA  
Email: nxiong@cs.gsu.edu

**Abstract**—Research into the hubs in Peer-to-Peer (P2P) networks, and present a new method to avoid generating the hubs in the networks by controlling the logical topology structure of P2P networks. We firstly introduce the controlling ideas about hierarchizing the hubs. Then, we disclose and interpret the controlling model, and give out the concrete method to carry it out. Finally, we validate our controlling model via simulations and the simulation results demonstrate that our work is effective to control the hubs in P2P networks. Thus, this model can improve the network competence to defend against coordinated attacks, promote the network robustness, and ensure the network would develop continually and healthily.

**Index Terms**—P2P network; hubs; topology control; hierarchical

## I. INTRODUCTION

Peer-to-Peer(P2P) systems are distributed systems consisting of interconnected nodes able to self-organize into network topologies with the purpose of sharing resources such as content, CPU cycles, storage and bandwidth, which is an important transformation to the traditional client-server technology. Unlike client-server networking, P2P networks allow individual users (or nodes) to play the roles of both client and server at the same time [1].

P2P technology has gained rapid and wide popularity for it provides us many fast and convenient functions such as document search, file downloading, etc., at the same time, its development also brings itself some disadvantages. Research in [2] to [8] shows that there exist a small number of nodes that have a large number of connections from others in P2P network, and these nodes is what we called hubs. The existence of the hubs will do a lot of harms to the network. It could reduce the ability to defend against coordinated attacks, and increase the vulnerability of the whole P2P network [3]. If we

don't take actions against this phenomenon and let it grow, the number of hubs in P2P networks would become larger, then the hubs' failure or quit from the system would take the P2P network service systems into paralysis much easier [7]. Thereby, that how to let the hubs avoid being generated is one of the crucial problems to ensure P2P networks develop continually and healthily. In this paper, we present a strategy to achieve this aim by controlling the logical topology structure of P2P networks, and this controlling method can enhance the anti-vulnerability of P2P networks and ensure the networks develop continually and healthily.

## II. BACKGROUND

Recently, many researchers in P2P networks field have been focusing on free-riding, behavior characteristic of the nodes, network topology structure and some other aspects. After having researched on several large P2P systems such as Gnutella, Kazaa, BitTorrent, etc., researchers found that all of these P2P networks have a same critical feature, that is, the degree distribution of nodes in these systems obeys a distribution similar to power-law [5]. The node's degree is the number of its connections from other nodes, and the degree distribution obeys power-law distribution means that the relationship between the number of nodes whose degree is  $k$  and the numeral  $k$  could be approximately denoted by one power-function,  $P(k) = k^{-\lambda}$ , where  $\lambda$  is a constant. The curve of power-function is a slowly descending line, see Fig. 1, which shows the curve when  $\lambda = 2$ . Power-law degree distribution indicates that there exist a small number of hubs in P2P networks. It means that these hubs serve for a great many users. This is very harmful for that once one hub fails or quits from the network would be a hard blow to the P2P network system. Besides, the hubs are overloading, and its QOS could be descended. Thus, these hubs distributed in P2P networks

\* Corresponding Author: Yuhua Liu (Huazhong Normal University, China).

would definitely reduce the performance and the quality of the whole P2P systems [4], and increase the vulnerability of P2P networks. Therefore, the feature that there are a small number of hubs existed in P2P networks needs to be addressed by us.

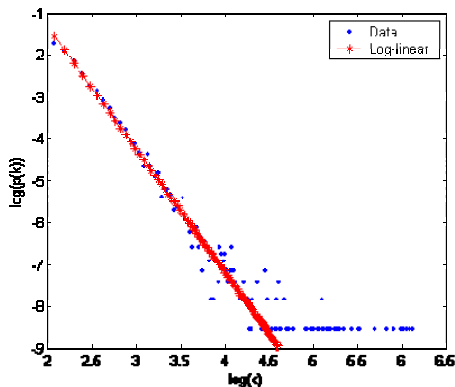


Figure 1. Curve of power-law distribution curve with  $\lambda=2.3$

At present, the controlling mechanisms on hubs are mainly about the aspect of restricting the user behavior characteristic of the nodes to restrain the free-riding and resolve the problem of the tragedy of the commons.

Free-riding means exploiting P2P networks resources (through searching, downloading objects or using services) without contributing to the P2P networks at desirable levels [9]. In 2000, an extensive analysis of user traffic on Gnutella in [4] shows a significant amount of free-riding in the system that nearly 70% of Gnutella users share no files, and nearly 50% of all responses are returned by the top 1% of sharing hosts. Furthermore, they found out that free-riding is distributed evenly between domains, so that no one group contributes significantly more than others, and that peers who volunteer to share files are not necessarily those who have desirable ones. Besides, it still goes on. The works [5], [6], [7] and [10] research into the large P2P networks respectively in 2002, 2004, 2005, 2006 and get the same conclusion that free-riding is getting worse.

Free-riding could lead to degradation of the system performance and add vulnerability to the system [4]. As a result of it, the number of files in the system becomes limited and the number of popular files may become even smaller as the time goes by, which could affect user's interest in the system and they eventually pull out of the system. When users who share popular files pull out, the system becomes even poorer in terms of the amount of files shared. This is a vicious cycle and it may eventually lead to the collapse of the system [7].

Since the users who are willing to share or provide services to others are few, all the downloading or other requests are directed towards those nodes that share information and resources, so those nodes are prone to overloading and congestion, leading to the tragedy of the commons problem [5].

The restrain mechanism is the main method to against free-riding in P2P networks, and many researchers restrain the free-riding with incentive mechanisms, game theory or social networks and economics models [11].

Ramaswamy L., Liu L. introduced the concept of utility function to measure the usefulness of peers to the system as a whole, and described a scheme based on that concept to control free-riding [12]. They identified three important utility factors for building fair incentives: the total number of files shared, the total size of data shared, and the popularity of the data shared. Accordingly, they defined three functions to adopt three factors respectively. Their experiment indicated that the utility-based free riding control could increase the lifetime of system by 10 times, even with a simple utility function. Then, Ahsan Habib and John Chuang [13] used the same concept in their incentive mechanism for media streaming and it still worked.

In reference [14], the authors considered the problem of fair resource sharing to optimize the performance of resource sharing in peer to peer systems. Resource sharing systems currently face rational peers which may exhibit a variety of strategies including: free-riding, and greedy behavior. An emergent property of a resource sharing system where nodes are neither free-riders nor greedy is fairness. They adopted a improved and appropriate utility function in their mechanism to guarantee fair resource sharing.

Karakaya M., Korpeoglu I. and Ulusoy O. A. in their work [15] proposed a distributed method with the aim of reducing the degree of free riding in P2P networks. In their model, each peer monitors its neighbors, makes decisions and takes appropriate actions. They specified three different free riding types and their symptoms observable from neighboring peers. The counter actions (TTL, neglect, disconnect) to be applied to free riders were defined in their mechanism. In fact, their work is an incentive mechanism because the free-riders were urged to share files.

The works [16] and [17] used game theory to study nodes' behaviors in peer-to-peer networks when nodes receive service based on their reputation. Reputation was used as a mechanism to inspire nodes to share resources and provide services to others. Game theory can be used by individual selfish nodes to determine their optimal strategy for participation level in such a system.

In [18], the authors reviewed the psychology and community behavior literature in order to provide an explanation for the reasons why people get motivated to participate in P2P systems and to contribute to the peer community. Based on individual factors and interpersonal factors, they developed incentive mechanisms which might help to induce peers to contribute and participate more and thus benefit the entire P2P system and community.

In paper [19], the authors studied the use of payments to implement an incentive mechanism that attaches a real monetary cost to low availability. Through a game theoretic analysis, they evaluated the effectiveness of such an incentive mechanism and found that peer availability could be significantly increased through the introduction of payments under many conditions.

Bohm K, Buchmann E in [20] described a protocol that renders free riding unattractive for one particular

P2P data structure. The protocol was based on feedback that adjacent nodes exchange. The protocol used proofs of work to deter free riding. To show that cooperative behavior dominates, they came up with a cost model that quantifies the overall cost of peers, depending on their degree of cooperativeness and many other parameters.

Zghaibeh M and Harmantzis FC [21] proposed a Lottery-based pricing mechanism to enhance the sharing level in P2P networks and help increase the number of objects disseminated. The scheme was an extension of the traditional micropayment mechanism. Their scheme provided higher payoff for peers who contribute to the P2P network and higher cost for peers who act selfishly and choose not to share resources.

Y. Liu, C. Yang, et al. in their works [22] and [23] proposed a rank-based fair incentive mechanism to restrain the free-riding, solve the tragedy of the commons problem and avoid generating hubs in the P2P networks. In their mechanism, each user had a rank according to its utility value and it can just visit the corresponding rating files with its rank in the system. The utility function in [22] took the absolute contribution value and the physical performance into account for the fairness and their value were calculated by Analytic Hierarchy Process. They constructed a pyramidal rank structure, and established the computation formula for node's transfer in each layer. The pyramidal rank structure enabled that the problem of hubs in P2P systems would be solved by user's self-management.

The conclusion of [24] is a little different from that of researches before. The analysis in this paper combined different areas of research: the private provision of public goods, social networks and peer-to-peer networks. Its analytic models showed that P2P networks could persist in spite of free-riding. At the same time, however, the models showed that without external incentives, the level of free-riding would be above the socially optimal levels. The authors also indicated that aggregate utility for network users could be improved by reducing free-riding.

However, all of the restrain mechanisms that have been proposed before may make the system no longer attractive to some of the free-riders. Because the restrain mechanisms were too strict for the free-riders, free-riders may drop the system since they can get few resources from it. Consequently, the number of users would decrease rapidly. But the amount of users is the key point to P2P systems, and too few users would lead to the failure of the systems. Especially in commercial operation, the number of users sometimes is the crucial ticket to decide whether the system is successful or not, and the number reduced predicates that the measure used to control the hubs is failing. Therefore, these controlling methods may act against the purpose to let the system develop more continually and healthily. Additionally, in [24], the authors found that moderate degree of free-riding would not do harm to the system, but could increase the throughput and improve the efficiency of the network. Accordingly, we should look for other methods to control the hubs in P2P network.

### III. BUILDING THE CONTROLLING MODEL

Except the mechanisms introduced above, there is another much more noteworthy research perspective to control the hubs in P2P networks--controlling the networks' logical topology structure to avoid generating the hubs. Besides, this method will not induce the former negative effect that the number of users may reduce. In the complex network research domain, Yuhua Liu and Shaohua Tao et al. [25] proposed two strategies, hierarchical and distributed, to deal with the hubs in scale-free network, and their strategies can optimize the topology and enhance the anti-vulnerability of the scale-free network. This paper consults that work's idea and hierarchizes each hub to alter the logical topology structure of P2P networks in order to avoid generating the hubs and improve the network's vulnerability brought by hubs.

#### A. The controlling idea

When catching out one node  $V_i$  in the network is going to become a hub, we should find out one resource  $s$  that have the most request connections from all the resources that  $V_i$  is sharing and search out other nodes that are sharing  $s$  in the network by starting from  $V_i$ . From these nodes, we sort out the most two excellent nodes that support more remainder connections than others as standby nodes of  $V_i$  on  $s$ , and record their IP address. The standby nodes and the original node  $V_i$  should be controlled to form a binary tree logically, and the original node is the root. Then, we could transmit the new request for  $s$  to one of the two standby nodes, so that standby node responds the request and sets up a direct connection with the request node. The transmission rule is that calculating the distances between the request node and the standby nodes respectively according to their IP addresses and transmitting the request to the standby node that has the shorter distance from the request node. If the standby node is going to be a hub as same as the original node  $V_i$ , we could use the process aforementioned on it to transmit the request again. In the end, the system would form a multilayer binary tree in logic (See Fig. 2.).

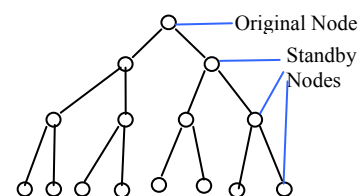


Figure 2. Logic binary tree

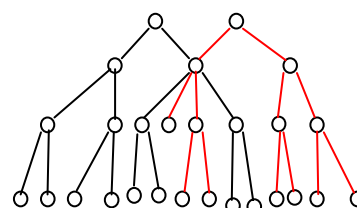


Figure 3. Admixture structure

If the trend of node  $V_i$  to be a hub is not changed by the treatment on  $s$ , we should take the process again and again on other resources of its sharing until the trend disappears. Finally, the whole system would be a mixed structure that many binary trees are intercrossed together (See Fig. 3.).

*B. The controlling model*

We first propose some definitions and assumptions for our model, and these definitions and assumptions are necessary.

*Definition I* The remainder capacity of supporting more connections ( $L(i)$ ) is the number of connections that the node  $V_i$  can support except the connections it is supporting. That is to say,  $L(i) = R(i) - D(i)$ , where  $R(i)$  is the threshold of  $V_i$ ,  $D(i)$  is the degree of  $V_i$ .

*Definition II* The IP address margin between  $VQ$  and  $VL$  ( $|IP(VQ) - IP(VL)|$ ) is the absolute value of  $IP(VQ)$  minus  $IP(VL)$  on every eight binary bits, in which  $IP(VQ)$  means the IP address of node  $VQ$  and  $IP(VL)$  means the IP address of node  $VL$ . The relation operations on IP address margins should be operated on every eight binary bits from left to right according to the principle that left is prior to the right, which means that the comparison should be started from the first eight bits to the fourth eight bits and once the two addresses are not equal, the comparison would be over. For example, suppose  $IP(VQ) = 128.11.3.31$ ,  $IP(VL) = 222.1.3.2$ ,  $IP(VR) = 128.11.3.9$ , then  $|IP(VQ) - IP(VL)| = 94.10.0.29$  and  $|IP(VQ) - IP(VR)| = 0.0.0.22$ , and because the first eight bits of  $|IP(VQ) - IP(VL)|$ , 94, is bigger than the  $|IP(VQ) - IP(VR)|$ , 0, we could know  $|IP(VQ) - IP(VL)| > |IP(VQ) - IP(VR)|$  without comparing the rest three eight bits.

*Assumptions* The threshold of node  $V_i$  in P2P network is  $R(i)$ , i.e., the number of connections that the node  $V_i$  can support at most is  $R(i)$ . At a certain time  $t$ ,  $V_i$  is sharing  $m$  ( $m \geq 1$ ) resources  $S$ , and  $V_i$  receives a new request  $Q$  about the resource  $s$  ( $s \in S$ ). At this moment, the degree of  $V_i$  is  $D(i)$ .

Now, we give out our model as below,

If  $D(i) < R(i) - 1$ , then  $V_i$  accepts and responds the request;

Else if  $D(i) = R(i) - 1$  and  $V_i$  has established the standby nodes on resource  $s$ , then  $V_i$  transmits the request  $Q$  to one of the standby nodes according to the transmission rule *a* below;

Else  $D(i) = R(i) - 1$  and  $V_i$  hasn't established the standby nodes on resource  $s$ , suppose there are  $m'$  ( $m' \leq m$ ) resources on which  $V_i$  hasn't established the standby nodes in  $V_i$ 's sharing  $m$  resources  $S$ , and the quantity relationship of the connections on these  $m'$  resources is  $C_{S(1)} \geq C_{S(2)} \geq \dots \geq C_{S(m')}$ , ( $C_{S(1)} + C_{S(2)} + \dots + C_{S(m')} + C_{S(m'+1)} + \dots + C_{S(m)} = D(i)$ ), then the node  $V_i$  searches the P2P network for the resource  $S(1)$  which has the most connections in  $m'$  resources. Suppose it searches out  $k$  nodes that are sharing  $S(1)$ .

*Case One:* If we can choose out two nodes that have the most two remainder capacities (not zero) of

supporting more connections, then mark them "VL" and "VR" respectively as the standby nodes of  $V_i$  on  $S(1)$ , and let  $V_i$  be the root,  $VL$  be the left child and  $VR$  the right child of  $V_i$  in order to form the logic binary tree. Next, record the IP addresses of  $VL$  and  $VR$  as  $IP(VL)$ ,  $IP(VR)$ . The last step is transmitting the request  $Q$  according to rule *a* or *b*.

*Rule a:* If  $s = S(1)$ , then calculate the IP address margins of  $VQ$  and  $VL$ ,  $VR$  respectively, i.e.,  $|IP(VQ) - IP(VL)|$  and  $|IP(VQ) - IP(VR)|$ , where  $IP(VQ)$  is the IP address of the client  $VQ$  that sends the request  $Q$ .  $V_i$  transmits the request of  $VQ$  to the standby node that has smaller margin than the other standby nodes. For example, if  $|IP(VQ) - IP(VL)| > |IP(VQ) - IP(VR)|$ , then  $V_i$  transmits the request to the standby node  $VR$ .

*Rule b:* If  $s \neq S(1)$ , then  $V_i$  intermits the latest connection  $L$  about the resource  $S(1)$  and answer the request  $Q$ . After that, we let  $Q = L$ , and treat  $Q$  with rule *a*.

*Case two:* If there exist only one node whose remainder capacity of supporting more connections is not zero, then record it as  $VL$ , and let  $VR$  be null,  $IP(VR) = 0.0.0.0$ . Now, it is turned into case one.

*Case three:* If there is no node whose remainder capacity of supporting more connections is not zero, then  $V_i$  searches the network again for the resource  $S(2)$  which has the second most connections in  $m'$  resources, and do the judgment and treatment again about the nodes searched out that are sharing  $S(2)$ . If it fails either, repeat the search process on  $S(3)$ ,  $S(4)$ , ...,  $S(m')$  orderly until find out one or more nodes whose remainder capacity of supporting more connections is not zero. If the treatments on  $m'$  resources are all failed, the system returns the information of waiting.

For standby node,  $VL$  or  $VR$ , if its number of connections get to its thresholds,  $R(L) - 1$  or  $R(R) - 1$ , take the searching process for it. Suppose the searching is about resource  $s'$ , sort out nodes that haven't established standby nodes on  $s'$  as the standby nodes of  $VL$  or  $VR$ .

For the exit of the nodes in logic binary tree, substitute it with its left child.

*C. The model descriptions*

*Description I* This model use the concept of the remainder capacity of supporting more connections ( $L(i)$ ) to estimate the comprehensive performance (CPU, bandwidth, etc.) of standby nodes.  $L(A)$  is bigger than  $L(B)$  indicates that the comprehensive performance of node  $A$  is greater than node  $B$  and the service capability of node  $A$  is also greater than node  $B$ .

*Description II* Because of the aggregation characteristic of the goals of address space management--address space should be distributed in a hierarchical manner, according to the topology of network infrastructure, and this is necessary to permit the aggregation of routing information by ISPs, and to limit the expansion of Internet routing tables [26] -- we can infer that the comparison of IP address margins could approximately reflect the physical position relations of

$VQ$  and  $VL$ ,  $VR$ . For instance, in the former example in definition II, we can see that  $VQ$  and  $VR$  are in the same network domain from their IP addresses and the physical path between  $VL$  and  $VQ$  is longer than  $VR$  and  $VQ$ . The discussions in [4], [5], [6] indicate that in P2P network, the better the application level topology matches the physical level topology, the less the network traffic will flow among the domains, moreover, the high-speed bandwidth of local LAN will be made full use of. The transmit rule of this model just considers the above research results and exerts to make the logical topology be coincident with the physical topology.

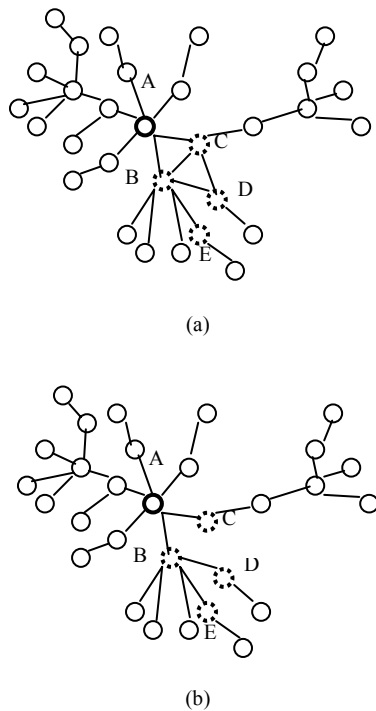


Figure 4. Two structures: (a) No control to form graph, (b) Control to form logic binary tree structure.

*Description III* This model controls and organizes the standby nodes into logic binary tree. It is because that if we don't control the structure, the standby nodes would go into the graph structure (See Fig. 4), and the time complexity of searching algorithm for graph is much higher than binary tree. Besides, P2P network is dynamic and the standby nodes may depart from the system at any time, so it will be much easier to maintain the system if we build the binary tree instead of the graph since there are mature algorithms for the nodes updating in binary tree. Thereby, it is suitable to control and organize the standby nodes into logic binary tree.

#### IV. CONTROLLING ALGORITHM AND TIME COMPLEXITY ANALYSIS

##### A. Controlling algorithm

According to the controlling model, we give out the controlling algorithm as below.

- STEP0  $V_i$  gets the request  $Q$  for  $s$  from  $VQ$ , obtain  $R(i)$   $D(i)$  and  $m$  of  $V_i$ .
- STEP1 if  $(D(i) < R(i) - 1)$ , accept; or, go to STEP2.
- STEP2 if existing super-cube for  $V_i$  on  $s$ , then get  $VL$  and  $VR$  of  $V_i$ , go to STEP7; or, go to STEP3.
- STEP3  $S = \{S(1), S(2), \dots, S(m)\}$ , sort( $S$ ), set  $j=1$ , go to STEP4.
- STEP4 if  $(j \leq m)$ , go to STEP5; else, wait to try again.
- STEP5  $k = \text{search}(S(j))$ . if  $(k \geq 2)$ , select out  $VL$  and  $VR$ , go to STEP6; else if  $(k=1)$ , select out  $VL$ , go to STEP9; else, go to STEP10.
- STEP6 form binary tree of  $V_i$  on  $S(j)$ . if  $(s=S(j))$ , go to STEP7; else, go to STEP8.
- STEP7 if  $VR \neq \text{NULL}$ , set  $\text{Temp} = \min\{|IP(VQ) - IP(VL)|, |IP(VQ) - IP(VR)|\}$ ; else, transmit  $Q$  to  $\text{Temp}$ , END.
- STEP8  $V_i$  intermits the latest connection  $L$  on  $S(j)$ , accept  $Q$  and set  $Q=L$ , go to STEP7.
- STEP9 set  $VR = \text{NULL}$ , go to STEP7.
- STEP10  $j++$ , go to STEP4.

##### B. Time complexity analysis

We use the three-tuple notation  $(s, VL, VR)$  to denote a node establishes its standby nodes,  $VL$  and  $VR$ , on one certain resource  $s$ . Each quasi-hub maintains a standby nodes list,  $\text{List\_children}$ , which records all of the three-tuples established by the quasi-hub. Each standby node maintains a parent list,  $\text{List\_parent}$ , which is a set of two-tuples,  $(s, \text{parent})$ .

In the model introduced in Section 3.2, when  $D(i) = R(i) - 1$ , it needs to search for  $s$  in  $\text{List\_children}$ . If searched out a three-tuple that includes  $s$ , return its  $VL$  and  $VR$ , or else, return false.

When node  $V_i$  establishes two standby nodes on resource  $s'$ , the new record  $(s', IP(V_iL), IP(V_iR))$  needs to be inserted into  $\text{List\_children}(V_i)$ . At the same time,  $V_iL$  and  $V_iR$  needs to add a new record  $(s', IP(V_i))$  to their list,  $\text{List\_parent}(V_iL)$  and  $\text{List\_parent}(V_iR)$ , respectively.

TABLE I. LIST\_CHILDREN

Resource	Left Child	Right Child
$s$	$IP(VL)$	$IP(VR)$

TABLE II. LIST\_PARENT

Resource	Parent
$s$	$IP(V_i)$

When node  $V_j$  in the binary tree departs from the system, for each resource in  $\text{List\_parent}(V_j)$ , query it in  $\text{List\_children}(V_j)$ . If succeed,  $V_j$  sends its left child information  $IP(V_jL)$  to its parent node corresponding to that resource, and that parent node substitutes its  $IP(V_j)$  information with  $IP(V_jL)$ ; or if failed,  $V_j$  sends "null", and the parent node substitutes with "null".

The most complexity of this algorithm is the case of  $D(i) = R(i) - 1$  and  $V_i$  hasn't established standby nodes on resource  $s$ . Firstly, it needs to search in the list,

List\_children, which has  $m$  records at most. So the searching step needs to compare  $m$  times at worst. The second step is to sort out the resource that has the most connections. It is a internal sorting problem and needs to compare  $m$  times at worst too. The next step is to search in the network, and its time complexity  $T$  rests with the P2P network searching algorithm. In *Case three*, there is no node whose remainder capacity of supporting more connections is not zero, the previous three steps should repeat  $m$  times at worst. The last step is to choose out two nodes from  $k$  nodes as the standby nodes and the comparison times are  $2k-1$ . Therefore, the total time complexity of this algorithm is  $m*(m+m+T)+2k-1$ . From the meanings of  $m$  and  $k$ , we could know that they are all numerable. So the time complexity should be  $O(T)$ , which just rests with the P2P searching algorithm.

V. SIMULATION RESULTS AND DISCUSSIONS

Our simulation is based on Gnutella protocol [27] and Flooding searching algorithm. We use the generalized BA model [28] to construct the topology and simulate the operation of P2P system on machine. We assume that there are 3000 nodes and 1000 files distributed on these 3000 nodes randomly and every node shares 100 different files. Our experiment simulates 3000 times resource requests in the simulation network with our control model and no control separately based on Gnutella protocol and Flooding searching algorithm. The comparison result is presented in Fig.5.

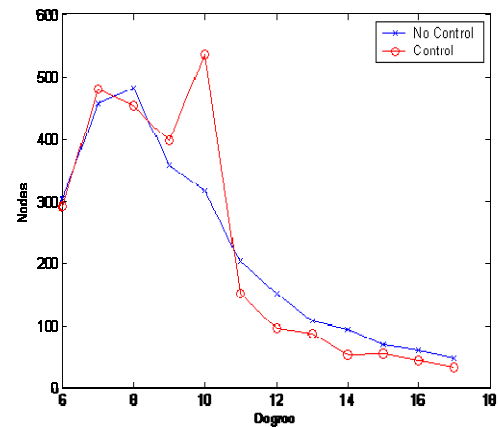
Fig.5(a) shows the comparison result of two methods in the segment  $D < 18$ . We could see that in the region of some low degrees, such as 9~10, the number of nodes in the experiment with our control model is larger than that with no control, but in the region after it, 11~17, is smaller.

Fig.5(b) shows the comparison result of two methods in the segment  $D \geq 18$ . The number of higher degree nodes and the value of the highest degree in the experiment with our control model are all smaller than that with no control.

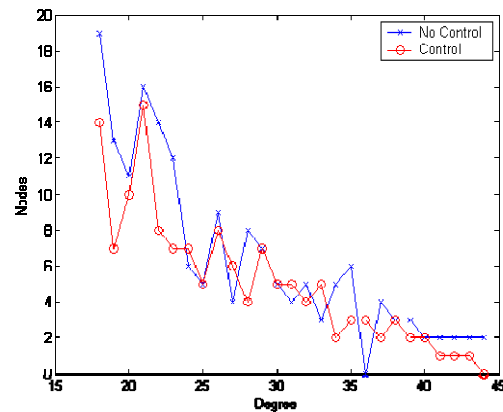
This is because our control model transmits the connection request for quasi-hub to the standby node which has lower degree, i.e. equalizes the nodes' degree to avoid generating the hubs. The impact on the graph is just that the nodes' number will grow in the lower degree region and reduce in the higher degree region and the value of the highest degree will also reduce.

Integrating the results in Fig.5(a) and (b), we can get the conclusion that our model can equalize the connections of nodes in P2P network and reduce the number of nodes that possess of exorbitant connections, at the same time, the model also reduces the value of the highest degree in the network. That is to say, the number of hubs in P2P network could be reduced and the phenomenon of hubs could be controlled effectively by our model. Besides, the reduction of the hubs must boost up the network competence to defend against coordinated attacks. Consequently, the network robustness is promoted, and the vulnerability induced by hubs is improved. Wherefore, the simulation result demonstrates

that our control model is effective.



(a)



(b)

Figure 5. Simulation results: (a)  $D < 18$  Region, (b)  $D \geq 18$  Region.

VI. CONCLUSIONS

The existence of the hubs in the P2P network could greatly reduce the network competence to defend against coordinated attacks, increase the network's vulnerability and do harm to the performance and the quality of the whole P2P system. This paper presents a new method to avoid generating the hubs in the network by controlling the logical topology structure of P2P network. It firstly introduces the controlling ideas about hierarchizing the hubs, then discloses and interprets the controlling model. Next, it gives out the particular methods to implement the controlling algorithm and analyses the time complexity of the algorithm. Finally, it uses the simulation to simulate the process of resource connection requests in the network and compares the simulation result using our control model with the result using no control method. The comparison result demonstrates that our control model can control the hubs and avoid generating the hubs in the P2P network effectively, which means that our model can improve the network competence to defend against coordinated attacks, promote the network robustness and ensure that the network would develop continually and healthily. Besides, this method will not

induce the negative effect that the number of users may reduce, so it has great practical application value.

#### ACKNOWLEDGMENTS

This work is supported by the Natural Science Foundation of China (Grant No. 60673163).

#### REFERENCES

- [1] A. T. Stephanos and S. Diomidis, "A Survey of Peer-to-Peer Content Distribution Technologies," *ACM Computing Surveys*, December 2004, 36, (4), pp.335-371
- [2] A. Klemm, C. Lindemann, M. Vernon, and O. Waldhorst, "Characterizing the Query Behavior in Peer-to-Peer File Sharing Systems," Proceedings of the 4th ACM Internet Measurement Conference (IMC), ACM, Taormina, Italy, Oct 2004, pp. 55-67
- [3] W. Wang, H. Chang, A. Zeitoun, et al., "Characterizing Guarded Hosts in Peer-to-Peer File Sharing Systems," Proceedings of IEEE Global Communications Conference (Globecom 2004), Dallas, USA, 2004, pp. 1539-1543
- [4] E. Adar and B. Huberman, "Free Riding on Gnutella," *First Monday*, 5, (10), October 2000, pp.134-139
- [5] M. Ripeanu, A. Iamnitchi, I. Foster, "Mapping the Gnutella Network: Properties of Large-Scale Peer-to-Peer Systems and Implications for System Design," *IEEE Internet Computing*, 2002, 6, (1), pp.50-57
- [6] S. Sen, J. Wang, "Analyzing Peer-to-Peer traffic across large networks," *IEEE/ACM Transactions on Networking*, Apr. 2004, 12, (2), pp.219-232
- [7] D. Hughes, G. Coulson, and J. Walkerdine, "Free Riding on Gnutella Revisited: The Bell Tolls?" *IEEE Distributed Systems Online*, 6, (6), 2005, pp. 1-18
- [8] R. Krishnan, M. D Smith, Z. Tang, et al., "The Impact of Free-Riding on Peer-to-Peer Networks," Proceedings of the 36th Annual Hawaii International Conference on System Sciences (HICSS-37 2004), IEEE Computer Society, Big Island, HI, USA, January 2004, pp. 199-208
- [9] Karakaya M., Korpeoglu I., Ulusoy O., "A distributed and measurement based framework against free riding in Peer-to-Peer networks," Proceedings of the 4th International Conference on Peer-to-Peer Computing, Zurich, 2004, pp.276-277
- [10] S. Meng, C. Shi, D. Han, X. Zhu, and Y. Yu, "A Statistical Study of Today's Gnutella," Proceedings of the 8th Asia-Pacific Web Conference (APWeb'2006), pp.189~200
- [11] Y. Yu, H. Jin, "A Survey on Overcoming Free Riding in Peer-to-Peer Networks," *Chinese Journal of Computers*, Jan. 2008, 31, (1), pp. 1-15
- [12] L. Ramaswamy and L. Ling, "Free Riding: A New Challenge to Peer-to-Peer File Sharing Systems," Proceedings of the 36th Annual Hawaii International Conference on System Sciences (HICSS-36 2003), IEEE Computer Society, Big Island, HI, USA, January 2003, pp.220-229
- [13] A. Habib and J. Chuang, "An Incentive Mechanism for Peer-to-Peer Media Streaming," Proceedings of International Workshop on Quality of Service (IWQoS), Montreal, Canada, June 8, 2004, pp. 171-180
- [14] E. Anceaume, M. Gradinariu, A. Ravoajia, "Incentive for P2P fair resource sharing," Proceedings of the 5th IEEE International Conference on Peer-to-Peer Computing, Konstanz, 2005, pp.253-260
- [15] M. Karakaya, I. Korpeoglu, O. Ulusoy, "Counteracting free riding in Peer-to-Peer networks," *COMPUTER NETWORKS*, Feb. 2008, 52, (3), pp.675-694
- [16] R. Gupta and A. K. Somani, "Game Theory as a Tool to Strategize as well as Predict Nodes' Behavior in Peer-to-Peer Networks," Proceedings of the 11th International Conference on Parallel and Distributed Systems (ICPADS'05), IEEE, Fukuoka, Japan, 2005, pp.244-249
- [17] RTB. Ma, SCM. Lee, JCS. Lui, et al., "Incentive and service differentiation in P2P networks: A game theoretic approach," *IEEE/ACM TRANSACTIONS ON NETWORKING*, Oct. 2006, 14, (5), pp.978-991
- [18] S. M. Lui, K. R. Lang, S. H. Kwok, "Participation incentive mechanisms in Peer-to-Peer subscription systems," Proceedings of the 35th Annual Hawaii International Conference on System Sciences, Hawaii, 2002, pp.3925-3931
- [19] D. Figueiredo, J. Shapiro, D. Towsley, "Incentives to promote availability in Peer-to-Peer anonymity systems," Proceedings of the 13th IEEE International Conference on Network Protocols, Boston, 2005, pp.110-121
- [20] K. Bohm, E. Buchmann, "Free riding-aware forwarding in content-addressable networks," *VLDB JOURNAL*, Oct. 2007, 16, (4), pp. 463-482
- [21] M. Zghaibeh, F.C. Harmantzis, "A Lottery-based pricing scheme for peer-to-peer networks," *TELECOMMUNICATION SYSTEMS*, Apr. 2008, 37, (4), pp. 217-230
- [22] Y. Liu, C. Yang, K. Xu, Hongcai Chen, "Rank Based Incentive Mechanism for P2P Network," Proceedings of the 4th International Conference on Natural Computation (ICNC-FSKD 2008), IEEE, Jinan, China, 18-20 October 2008, pp.252-256.
- [23] Y. Liu, C. Yang, K. Xu, Hongcai Chen, "Controlling Mechanism for the Hubs in Peer-to-Peer Networks," Accepted to appear in the Proceedings of the 2008 International Conference on Computational Intelligence and Security (CIS 2008), IEEE Computer Society, Suzhou, China, 13-17 December 2008.
- [24] R. Krishnan, M. D. Smith, Z. L. Tang, et al, "The impact of free-riding on Peer-to-Peer networks," Proceedings of the 37th Hawaii International Conference on System Sciences, Hawaii, 2004, pp.199-208
- [25] S. Tao, Y. Liu, K. Xu and D. Tan, "The Strategies against Vulnerability of Hubs in Complex Networks," *Computer Engineering and Applications*, 43, (2), 2007, pp. 151-153
- [26] Policies for IPv4 Address Space Management in the Asia Pacific Region  
<http://www.apnic.net/trans/cns/add-manage-policy.pdf>
- [27] The Gnutella Protocol Spec. v0.4 1, document revision 1.2.  
<http://www.clip2.com>, accessed March 2008
- [28] Y. Liu, C. Yang, L. Tan, S. Tao, "A Generalized B-A Model for the Recognition of Intermediate Vertex Effect," *Dynamics of Continuous, Discrete and Impulsive Systems (DCDIS), Series B*, December 2007, 14(S7), pp.23-27

**Yuhua Liu** received the PHD degree in College of Computer Science and Technology from Huazhong University of Science and Technology, in Computer Architecture. Now Dr. Liu is a professor, MS advisor in the Department of Computer Science, Huazhong Normal University. Her research interests are in the field of Computer Network and Communication Technology, including Network Communication, Network Optimization,



Wireless Networks and Sensor Networks, Intelligent Vehicle Transportation, Multi-Net, Complex Networks. At present, she is in charge of the Natural Science Foundation Project of China. In recent years, she also has performed many other developments of application topics. Until now, Dr. Liu has published more than 70 papers on authority magazine and IEEE International Conference.

Email: yhliu@mail.ccnu.edu.cn



**Chun Yang** is a graduate student in the Department of Computer Science of Huazhong Normal University, China. His research interests are in complex network and P2P network. Email: c\_yang@yeah.net



**Kaihua Xu** is a Profession in the College of Physical Science and Technology of Huazhong Normal University, China. His research interests include computer network and wireless communications technology, integration technology of the system.

Email: xukh@phy.ccnu.edu.cn



**Naixue Xiong** is a research scientist in Department of Computer Science, Georgia State University, USA. He has obtained two PhD Degrees on different research fields in Wuhan University and Japan Advanced Institute of Science and Technology, respectively. Both PhDs are on Information Science. His research interests include Communication Protocols, Network Architecture and Design, Network

Technologies, and Distributed and parallel Systems. Until now, Dr. Xiong published about 100 research articles (including about 36 international journal articles). Some of his works were published or submitted in IEEE or ACM transactions, JSAC and IEEE INFOCOM. He has been a Program Chair, General Chair, Publicity Chair, PC member and OC member of about 57 international conferences, and was invited to serve as a reviewer for about 36 international journals. Now, he is serving as an Associate Editor, Editorial Board Member, and Guest Editor for about 9 international journals. He is a member of IEEE. E-mail: nxiong@cs.gsu.edu.