

## Research Article

# Improved Degree Search Algorithms in Unstructured P2P Networks

**Guole Liu,<sup>1,2,3</sup> Haipeng Peng,<sup>1,2,3</sup> Lixiang Li,<sup>1,2,3</sup>  
Yixian Yang,<sup>1,2,3</sup> and Qun Luo<sup>1,2,3</sup>**

<sup>1</sup> Information Security Center, Beijing University of Posts and Telecommunications,  
P.O. Box 145, Beijing 100876, China

<sup>2</sup> Research Center on Fictitious Economy and Data Science, Chinese Academy of Sciences,  
Beijing 100190, China

<sup>3</sup> National Engineering Laboratory for Disaster Backup and Recovery,  
Beijing University of Posts and Telecommunications, Beijing 100876, China

Correspondence should be addressed to Haipeng Peng, penghaipeng2003@yahoo.com.cn

Received 1 November 2011; Revised 21 February 2012; Accepted 14 April 2012

Academic Editor: Yi-Chung Hu

Copyright © 2012 Guole Liu et al. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

Searching and retrieving the demanded correct information is one important problem in networks; especially, designing an efficient search algorithm is a key challenge in unstructured peer-to-peer (P2P) networks. Breadth-first search (BFS) and depth-first search (DFS) are the current two typical search methods. BFS-based algorithms show the perfect performance in the aspect of search success rate of network resources, while bringing the huge search messages. On the contrary, DFS-based algorithms reduce the search message quantity and also cause the dropping of search success ratio. To address the problem that only one of performances is excellent, we propose two memory function degree search algorithms: memory function maximum degree algorithm (MD) and memory function preference degree algorithm (PD). We study their performance including the search success rate and the search message quantity in different networks, which are scale-free networks, random graph networks, and small-world networks. Simulations show that the two performances are both excellent at the same time, and the performances are improved at least 10 times.

## 1. Introduction

Searching and retrieving the demanded correct information is becoming more and more important with the emergence of the huge amounts of information and the growth in the size of computer networks [1]. Especially, in unstructured P2P networks, the node's joining and failure are both random and dynamic [2], and in this case, it is unfeasible and unpractical that each node of the network has known and stored the global information about the whole

network topology and the location of queried resources. Thus, designing efficient search algorithms according to the local network information is critical to the performance of unstructured P2P networks.

Considerable amount of work has been done in this field, so far, a number of search algorithms have been proposed, including BFS algorithm [1], modified BFS algorithm [1, 3, 4], local search algorithm [5, 6], rumor broadcasting algorithm [7–10], the betweenness [11, 12], shortest path algorithm [13], iterative deepening algorithm [1, 14, 15], update propagation algorithm [16], and random walks search [17–30]. These search algorithms can be classified into two categories: BFS-based method and DFS-based method. Although these algorithms achieve relatively satisfying effects, these two types of search algorithms tend to be inefficient, either generating too much load on the networks [1, 2] or not meeting the search success rate of network resources. On the one hand, BFS-based algorithm shows the perfect performance in the aspect of search success rate of network resources, but at the same time, it brings the huge search messages. The number of search messages will grow exponentially with the hop counts in the search process [17, 18]. On the other hand, DFS-based algorithm generates the search message quantity far smaller compared with BFS-based algorithm, and the search message quantity will grow linearly with the hop counts [20]. The main drawback is to drop the search success ratio of network resources.

The references [3, 4, 7–10, 17–30] adopt the BFS-based method or DFS-based method to achieve their goal, respectively, but do not overcome the problem that only one of performances of the network loads, and the search success rate is excellent. To address this problem, in this paper, according to the degree of nodes [2], we propose memory function maximum degree algorithm (MD) and memory function preference degree algorithm (PD). These two algorithms can combine the advantages of the BFS-based algorithm and DFS-based algorithm, which can be efficiently used to search random graph [31, 32] networks and power-law networks such as scale-free networks and small-world networks [33, 34]. We have studied their performances in the search success rate of network resources and the search message quantity. Simulations illustrate their validity and feasibility. The results show that MD algorithm is better than PD algorithm in the search success rate. The search success rate of MD algorithm is average 14 times better than the standard random walks algorithm; the search message quantity is the same order of magnitude with it. Compared with modified BFS algorithm, the search success rate of MD algorithm is higher than it, and the search message quantity averagely reduces by over 18 times. Although PD algorithm can reduce the huge search message quantity, the search success rate of it is inferior to the modified BFS algorithm.

## 2. The Improved Degree Algorithms Methods

The degree of a node in a network (sometimes referred to as the connectivity of a graph) is the number of connections or edges the node has to other nodes. The degree of a node is an important index for some problems, which is used to measure the importance of the node. For the aspect of information transmission speed, the more edges connected to the node, the faster information dissemination by the node. Namely, the node is more important. For the aspect of the shortest path viewpoint, the greater betweenness centrality, the more importance of the node. Meanwhile, the degree of the node may be very small. Considering these, adopting the idea of assigning unique ID to each node in unstructured P2P networks, we, respectively, propose the memory function maximum degree algorithm (MD) and the memory function preference degree algorithm (PD). In the context, “Memory function” has two aspects contents. The first is that one node needs to store its neighbors’ ID and degree

information. The second is that one node has to remember the return node's ID according to its memory information. This requires the nodes of networks to save their neighbors' ID and, at the same time, save their related degree information. It is the advantage of these two algorithms from the point of view reducing the unnecessary search messages, and it is the shortcoming from the point of view occupying the storage space. Compared with the MB algorithm and the random walks algorithms, they double the storage space. In the context, the degree of a node is the number of connections or edges the node has to other nodes, including the traversed nodes.

### **2.1. The Memory Function Maximum Degree Algorithm**

In this strategy, when a node starts the resource search procedure, it firstly traverses all its adjacent nodes according to the BFS method to determine whether they contain the resources or not. If all the neighbors do not contain the resources, it changes the flag of its neighbors to denote that these nodes have been traversed, then broadcasts the search messages along directions of nodes with the highest degree according to DFS method, and updates the flags to denote that message has passed these nodes. (When the procedure is over, all the flags recover zero.) If it does not find the resource along directions of nodes with the highest degree, the search message will return the precursor node and broadcast along its neighbor node with the second highest degree. This procedure will stop until the age counter is increased to threshold or all the nodes in the network have been traversed. In extreme cases, when the neighbor's degrees are all the same, the algorithm degenerates into the standard BFS algorithm. When all the nodes' degrees are different, the algorithm degenerates into the standard DFS algorithm.

In the context, assuming that the red node is the source node and the blue node is the resource node, the green nodes are the intermediate nodes passed by search messages in the process. The orange nodes are the labeled nodes that can be found, which do not need to send the search messages. The digits on the top of arrows are the age values. The solid line arrows represent the spread direction of messages, and the dotted line arrows denote the direction of response messages.

Figure 1 shows the search process that MD algorithm search the resource node located in the path composed by the highest-degree nodes. Figure 1(a) is the first step; unlike the maximum degree algorithm [11], node 1 firstly traverses all its neighbors and ascertains whether the neighbors have the resource or not. All its neighbors do not contain the resource, and node 1 changes the flags of its neighbors according to BFS method; then according to DFS method it broadcasts the search message to node 3 whose degree is the highest among the neighbors. Figures 1(b) and 1(c) repeat the search process. In Figure 1(d), the search message finds the resource node and responds to the required message.

This algorithm has two aspects of difference compared with the maximum degree algorithm. On the one hand, in maximum degree algorithm the search message broadcasts along the nodes with highest degree. Although, a node can send several messages to its neighbor's nodes at the same time (it has the neighbors with the same highest degree), the algorithm does not search resources according to BFS method in essence. Thus, the maximum degree algorithm can only find the resource nodes located in the path composed by the highest degree nodes. By querying neighbors first before sending out search messages, MD algorithm provides higher success rate. So the search success ratio of maximum degree algorithm is inferior to that of MD algorithm. Figure 2 shows the search process of the maximum degree algorithm and the difference compared with the MD algorithm. Figures 2(a)–2(d) are the search process of the maximum degree algorithm. Figure 2(a) is the first step; the degree

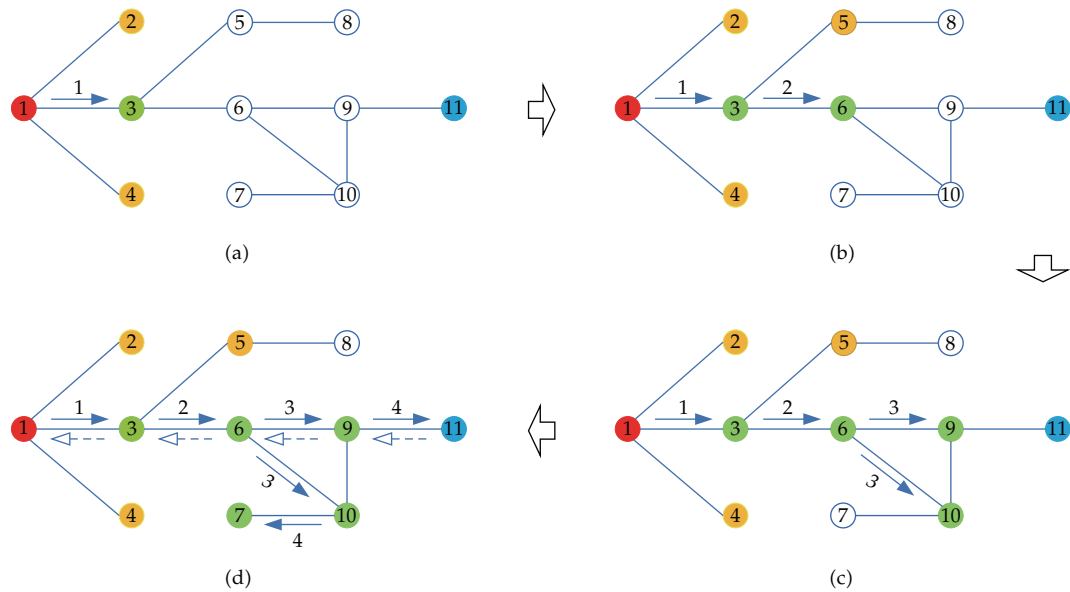


Figure 1: Search process of the MD algorithm.

of node 3 is the largest, so the search message is sent to node 3. Figure 2(b) is the second step; node 3 sends the search message to node 6. Figure 2(c) is the third step; the neighbor's nodes of node 6 have the same degree, and it sends two search messages to the neighbors. Figure 2(d) is the fourth step; the algorithm returns failed. If the resource is located in the path composed by the highest-degree nodes, such as node 3, 6, 7, 9, 10, and 11, the maximum degree algorithm can easily find these nodes. Figures 2(e) and 2(f) are the difference with MD algorithm. Obviously, the MD algorithm easily finds the resource node. The maximum degree algorithm searches the resource in failure, while MD algorithm searches the same resource in success.

On the other hand, MD algorithm can search more resources according to the memory of the ID and degree information. Figure 3 shows how to search the resource node with memory function using MD algorithm, assuming that the age value is large enough. In Figure 3(a), the source node 1 traverses all its neighbors and labels the neighbors according to BFS method; then according to DFS method it broadcasts the search message to node 3 whose degree is the highest among the neighbors. Figures 3(b)–3(d) repeat the search process. In Figure 3(e), node 7 and node 11 are the terminal nodes; the search messages return node 9 and node 10 according to the memory information which are the ID information of their precursors. Node 9 and node 10 remember the returned node's ID, which the search messages do not broadcast along the direction of these nodes. In Figure 3(f), the neighbors of node 9 and node 10 are all traversed, so the search messages continue to return. In Figure 3(g), the search message returns to node 3. In Figure 3(h), the search message will broadcast along the node with the second highest degree, because it does not find the resource along the direction of nodes with the highest degree. In Figure 3(i), the search message finds the resource node, then node 8 responds to the required messages.

In summary, MD algorithm can search both categories resource nodes: highest degree nodes and nonhighest degree nodes. It labels the nodes with different flags and uses the nodes' ID to reduce messages and, at the same time, to improve the search success ratio.

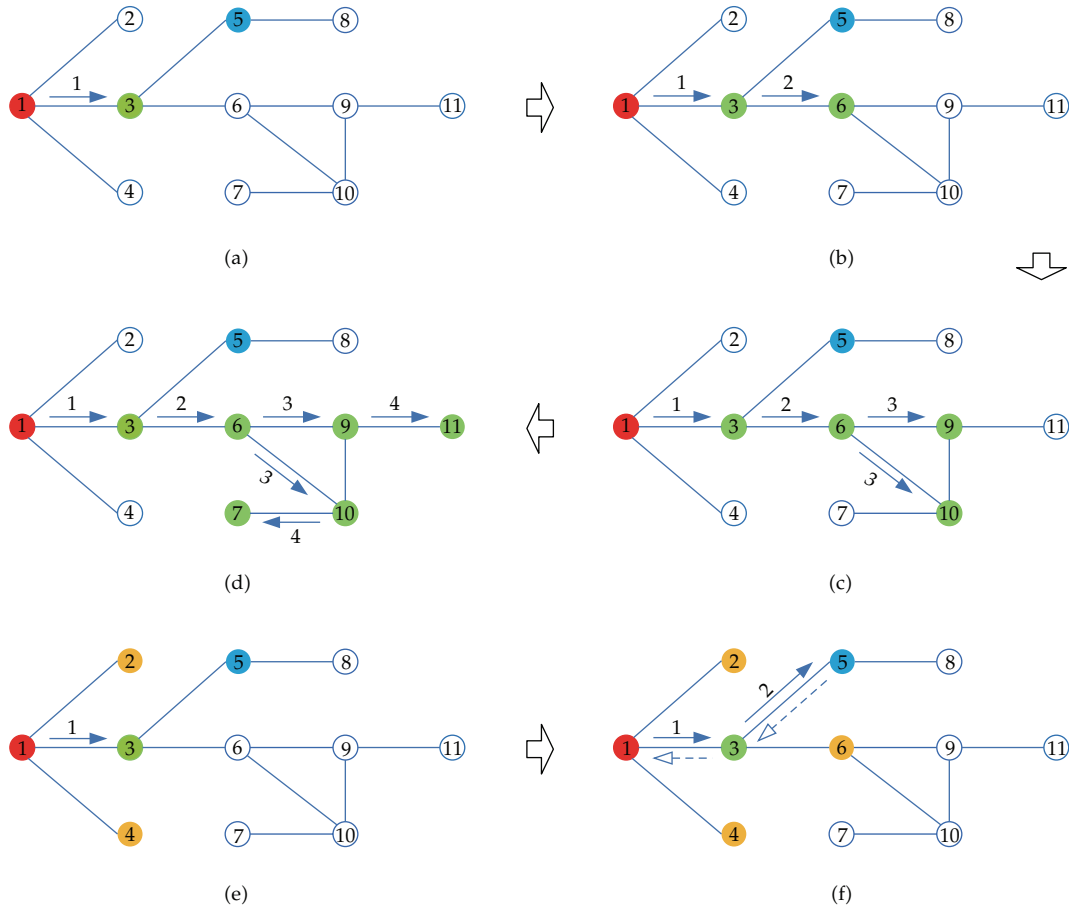


Figure 2: Search process of the maximum degree algorithm and the difference with the MD algorithm.

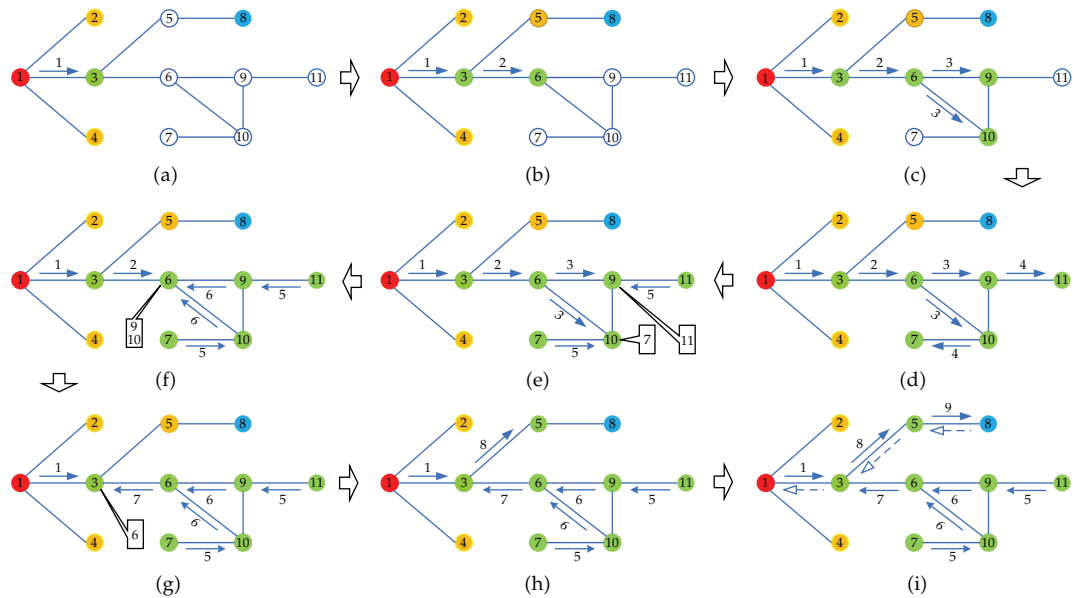
### 2.2. The Memory Function Preference Degree Algorithm

Compared to MD algorithm, the difference of PD algorithm is the search process. In the strategy of PD algorithm, when a node finds all the neighbors do not contain the resources, it randomly chooses the corresponding node with the preference probability  $\Pi$ . Assuming that the neighbors of the node  $A$  are  $A_1, A_2, \dots, A_n$ , and the degree of these neighbors are  $d_1, d_2, \dots, d_n$ , these degrees meet  $d_1 \leq d_2 \leq \dots \leq d_n$ . How to choose the preference node  $A_i$  is to compute as follows:

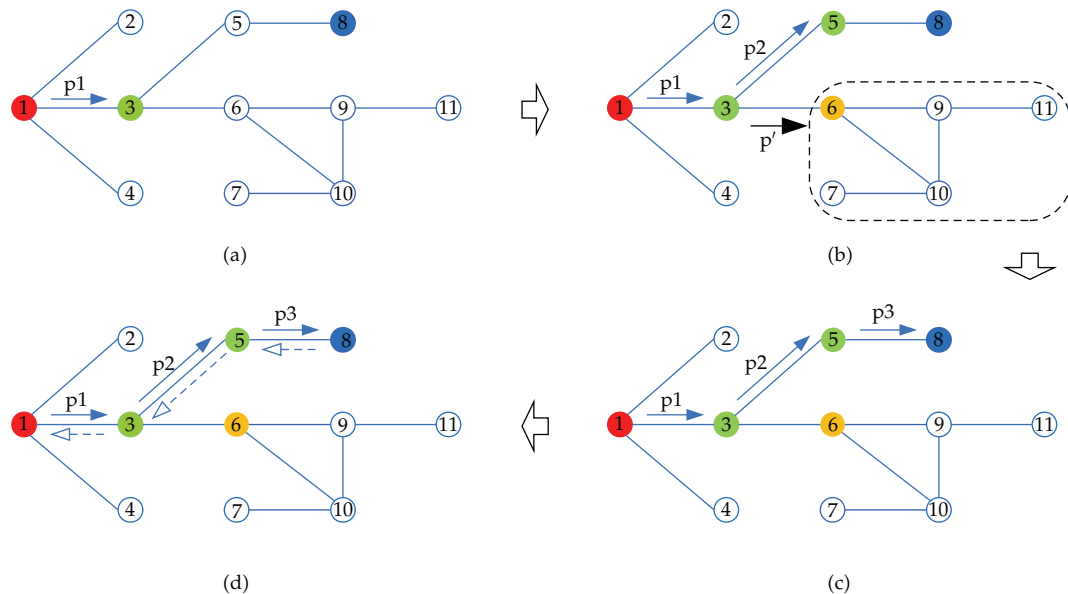
$$d = \sum_{i=1}^n d_i, \quad i = 1, 2, \dots, n, \tag{2.1}$$

$$\Pi_i = \frac{d_i}{d}, \quad i \rightarrow A_i.$$

In the worst conditions, the PD algorithm degenerates into the standard DFS algorithm for either same degree case or different-degree case. The search process of this algorithm is



**Figure 3:** Process of the MD algorithm searching the resource node with memory function.



**Figure 4:** Search process of the PD algorithm.

random, so the success rates are stochastic. The search process of this algorithm is shown in Figure 4. Unlike the MD algorithm, it randomly generates a preference probability  $p$  according to the neighbor nodes' degree. Figures 4(a)–4(d) are the specific process in the ideal conditions. In Figure 4(b), if the node 3 chooses the neighbor node 6 according to the stochastic preference, this algorithm will search resources in failure. It is shown as the dotted line box in Figure 4(b).

ID   Degree   Neighbor ID   Neighbor degree   Traverse flag   Returned ID   Resource key
--

**Figure 5:** List of message type.

### 3. Simulations and Discussions

Search message quantity, network resource search success rate, and search response time are the key parameters in unstructured P2P networks. Random graphs [31, 32] are widely used models, which help the study of the network. Because the degree distribution of the Internet nodes presents a power-law distribution, scale-free network and small-world network [33, 34] are the typical network structure to study the Internet. In this section, scale-free BA network, Watts-Strogatz (WS) small-world network, and ER random graph network are taken as the examples of network models, and numerical simulations are given to study these parameters. All the parameters are contrasted under the condition of search success. The maximum age is chosen 100 in the simulations. The complete list of message type is shown in Figure 5. The “traverse\_flag” of the list denotes the status of this node. When its value is 0, the node is not traversed. When its value is 1, the node is traversed and the search message does not pass the node. When its value is 2, the node is traversed and the search message passes the node.

#### 3.1. Search Messages Quantity

In modified BFS algorithm (MB), each node instead of forwarding a search message to all its neighbors, it randomly selects a subset of its neighbors to propagate the search request messages. The fraction of neighbors that are selected is a parameter to the mechanism. Given a P2P network, a node can search the others of the network more efficiently with a smaller number of messages compared with the standard BFS algorithm. So the total number of search messages is average to

$$\phi = \sum_{\text{age}=1}^T (d \cdot q)^{\text{age}}, \quad (3.1)$$

where  $T$  is the threshold,  $d$  is the average degree of a node,  $q$  in the chosen ratio and is general not more than 0.5. Here it is 0.25 in our simulations.

When considering the random walks case, the requesting node sends out one search message to a randomly chosen neighbor, that is, standard random walks algorithm. This search message is seen as a walker. Then the walker directly keeps in touch with the source node in the process of walking and asks whether to proceed to the next step. If the requestor is agreed to continue walking (the termination conditions have not been satisfied), it randomly chooses a neighbor to forward the walker. Otherwise, the algorithm terminates the walking process. The search message quantity of this algorithm is related to the age value; thus, it reduces the network loads and achieves a message reduction by over an order of magnitude compared to the standard BFS algorithm (it is also called flooding search algorithm in some literatures [1, 2]). In order to improve the search success rate, the requesting node sends out  $k$  search messages to its  $k$  neighbors, that is,  $k$  random walks algorithm, assuming that



the number of search messages for each hop keeps fixed as  $k$ , that is, the number of walkers. Therefore, the total number of search messages for random walks algorithm is

$$\phi = \sum_{\text{age}=1}^T k \cdot \text{age}, \quad k = 1, 2, \dots, N, \quad (3.2)$$

where  $T$  is the threshold of age. When the walker  $k$  meets  $k = 1$ , it is standard random walks algorithm (R1), and when  $k$  meets  $k > 1$ , it is  $k$  random walks algorithm (RK).  $k$  is general not more than 16 [22], and here it is 4 in our simulations.

In degree search algorithms, the query messages spread to these neighbor's nodes with the characteristics (the preference degree or the maximum degree) every step in the search process. In the context, degree search algorithms include the MD and PD. The characteristics include the maximum degree and preference degree. So the total number of search messages for degree search algorithms is

$$\phi = \sum_{\text{age}=1}^T m \cdot \text{age}, \quad (3.3)$$

where  $m$  is the number of nodes with the characteristics. For instance, in Figures 1(a) and 1(b),  $m$  value is 1 because the number of nodes with the maximum degree is 1. In Figure 1(c),  $m$  value is 2 because node 9 and node 10 have the same maximum degree.

Figure 6 shows the search message quantity ( $\phi$ ) generated by the various algorithms in different topology networks, where the average degree ( $D$ ) of the networks is chosen from 2 to 10. Figure 6(a) is the scale-free BA network, Figure 6(b) is the ER random graph network, and Figure 6(c) is the WS small-world network. The number of nodes of the networks is 5,000.

Simulations show that in the three cases, the search message quantity is increasing with the growth of the average degree of the networks. The search message quantity of MB algorithm confirms the exponential growth, and random walk algorithms reduce the message quantity into linear growth. The search message quantity of MD algorithm and PD algorithm is slightly less than the standard random walks algorithm, but, in general, they are the same order of magnitude according to the simulations. Due to memory function of MD algorithm and PD algorithm proposed in this paper, they reduce unnecessary search messages in the search process. Compared with  $k$  random walks algorithm, the search message quantity of MD algorithm and PD algorithm is less. These two algorithms can decrease the message quantity about 18 times than MB algorithm.

### 3.2. Search Success Rate

Search success is at least one request message sent from the requesting node seeks out the requested resources. Assume that the queried resources are uniformly distributed in the network with a replication ratio  $r$ . We calculate the search success rate ( $\Psi$ ) according to the following formula:

$$\Psi = \frac{C \cdot r}{N} \quad (3.4)$$



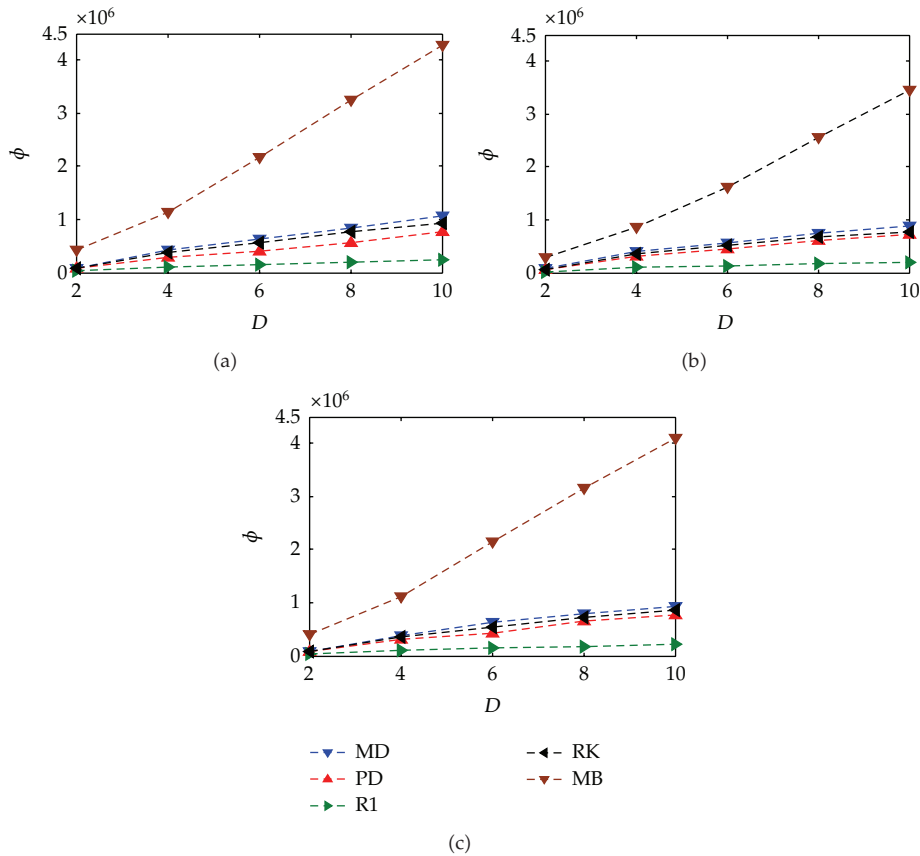
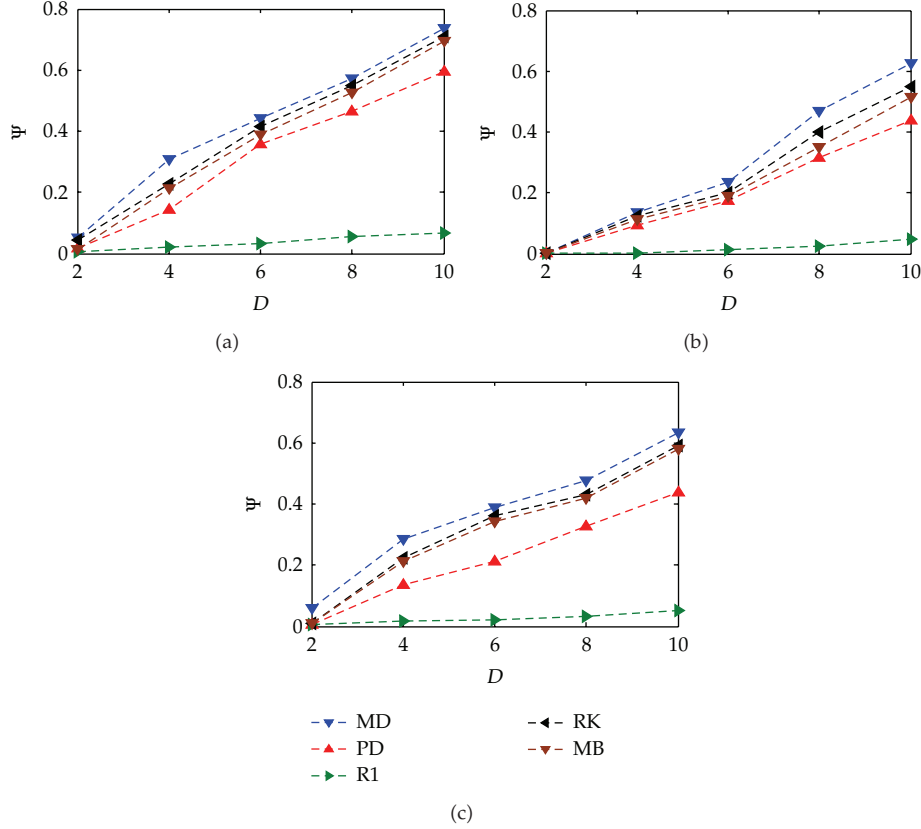


Figure 6: Message quantity ( $\phi$ ) generated by the various algorithms in different topology networks.

where  $r$  is the replication ratio,  $C$  is the number of nodes covered by the algorithm, and  $N$  is the total number of nodes of the network. This formula shows that the search success rate highly depends on the coverage of the search algorithms. The age value determines the coverage of the search algorithms. MB algorithm, random walks algorithm, and PD algorithm have random factors. Thus, their search success rates vary greatly depending on network topology and the random choices which have been made.

According to the results of our simulations, the age value of the search algorithms is not too large except the standard random walks algorithm. The maximum age value is chosen 100 in our simulations. In random walks algorithms, the walker  $k$  is chosen 4.

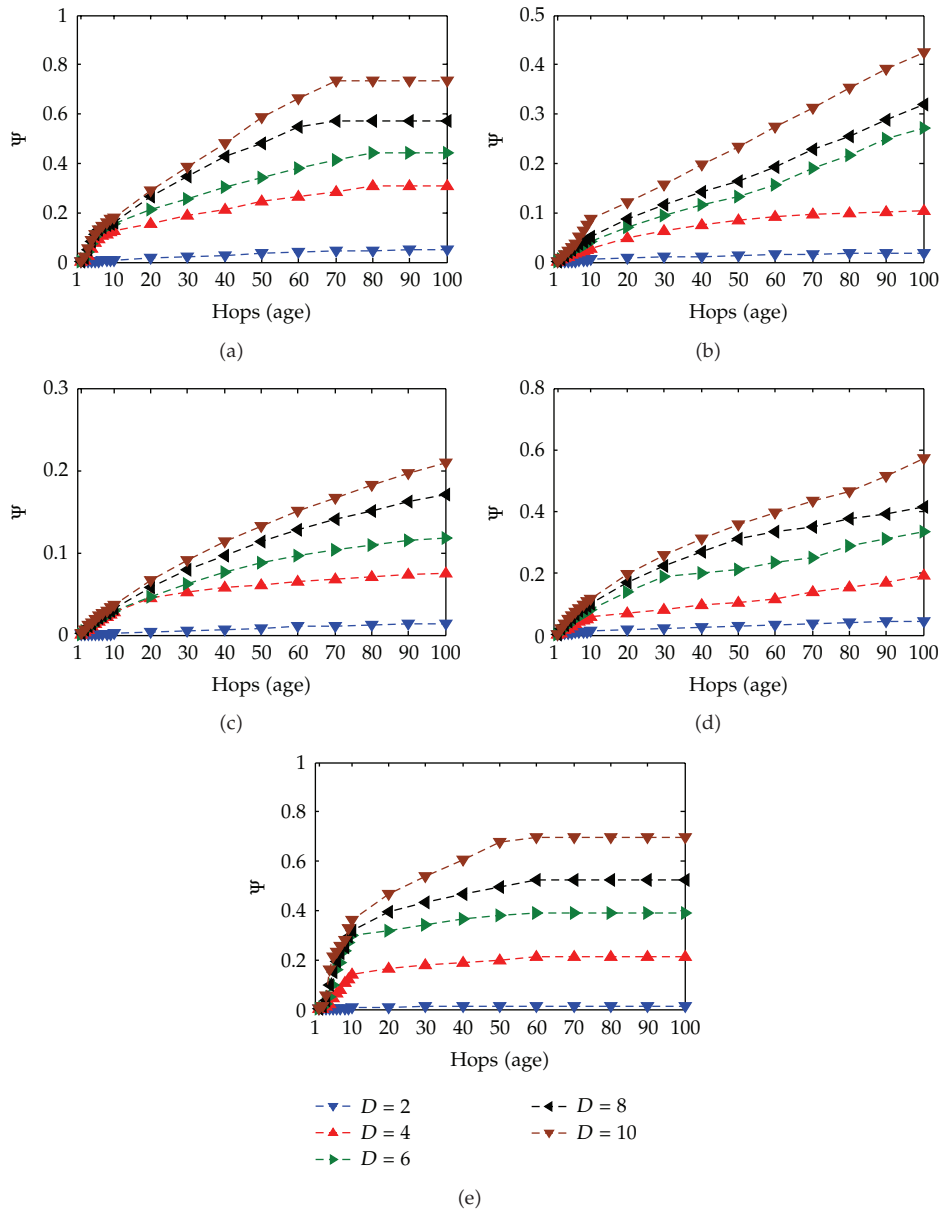
Figure 7 shows the search success ratio ( $\Psi$ ) of the various algorithms in different topology networks, where the average degree ( $D$ ) of the networks is chosen from 2 to 10. Figure 7(a) is the scale-free BA network, Figure 7(b) is the ER random graph network, and Figure 7(c) is the WS small-world network. Simulations show that in the three cases, the search success rates of the various search algorithms are increasing with the growth of the average degree of the networks. The search success rate of MD algorithm is the highest; the search success rate of RK algorithm is slightly higher than that of MB algorithm. The search success rate of R1 algorithm is the least. The search success rate of PD algorithm is better than the R1 algorithm and is inferior to the MB algorithm. The messages of MD algorithm can



**Figure 7:** Search success ratio ( $\Psi$ ) of the various algorithms in different topology networks.

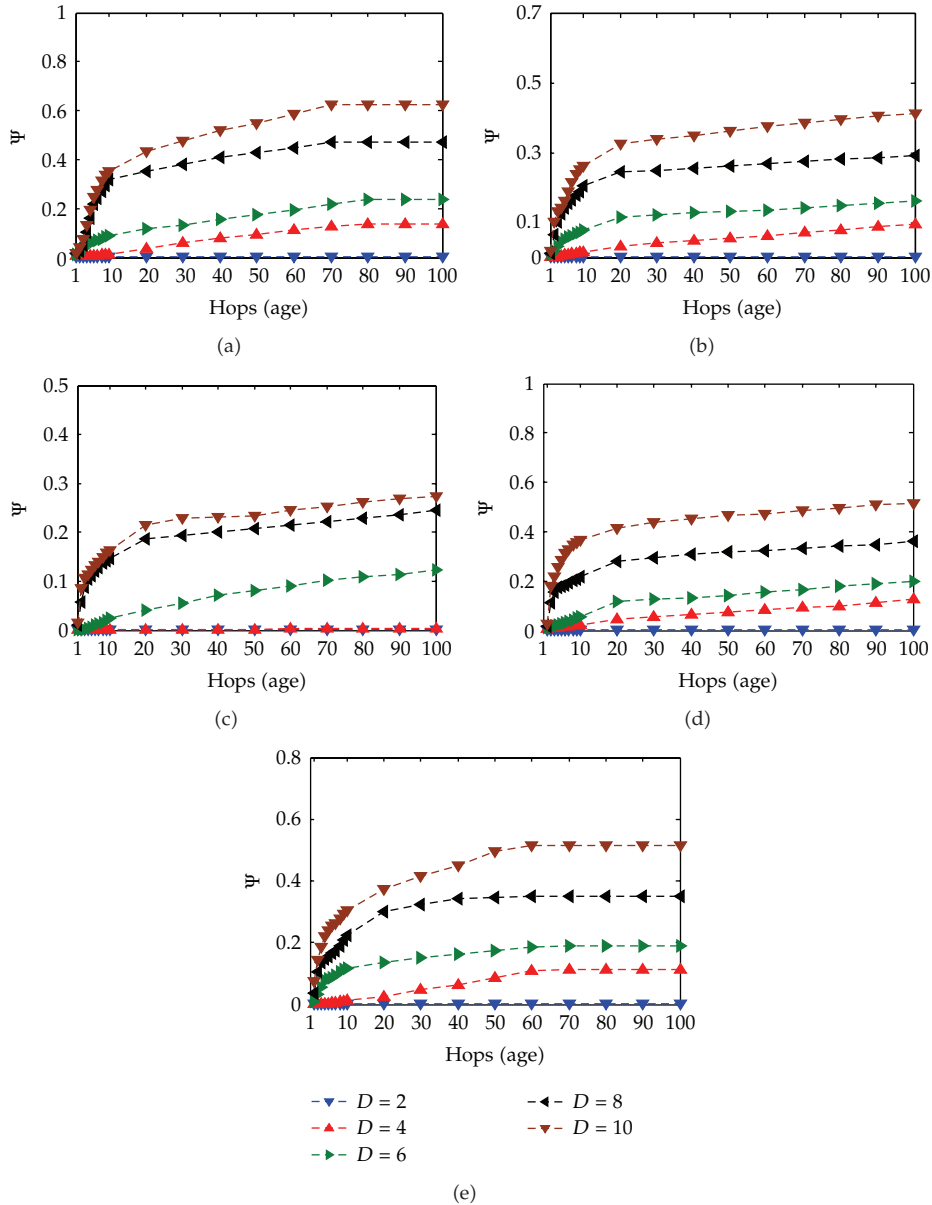
return the precursor, so it has more search scope. In RK algorithm, there are  $k$  random walkers to search the resources at the same time; the success rate is higher than that of the R1 algorithm. The MB algorithm, a random algorithm, can find the resources quickly if these resources locate in the chosen search paths. And it will generate the massive redundancy messages if the chosen search paths do not exit the resources. In our simulation, we calculate the mean value; the chosen proportion of MB algorithm is 0.25 and small, so the success rate is slightly inferior to that of MD and RK algorithm.

Figure 8 shows the search success ratio ( $\Psi$ ) of the various algorithms in scale-free BA networks, where the age value is changed from 1 to 100. Figure 8(a) is the MD algorithm, Figure 8(b) is the PD algorithm, Figure 8(c) is the R1 algorithm, Figure 8(d) is the RK algorithm, and Figure 8(e) is the MB algorithm. We can see that all the search success rates are increasing with the growth of the average degree. With the constraint of search ages, only the search success rate of MB algorithm is not affected. In particular, in the same scale of network and the same average degree of network, the search success rate of MD algorithm is the highest among the five algorithms, that of R1 algorithm is the least, and the search success rate of PD algorithm is between the R1 algorithm and the MB algorithm. To the search ages, the age value of MB algorithm is the least when the search success rate reaches the maximum, and the age value of R1 algorithm is the largest. The age values of MD algorithm and PD algorithm are between the R1 algorithm and the MB algorithm.



**Figure 8:** Search success ratio ( $\Psi$ ) of the various algorithms in scale-free BA networks.

Figure 9 shows the search success ratio ( $\Psi$ ) of the various algorithms in ER random graph networks, where the age value is changed from 1 to 100. Figure 9(a) is the MD algorithm, Figure 9(b) is the PD algorithm, Figure 9(c) is the R1 algorithm, Figure 9(d) is the RK algorithm, and Figure 9(e) is the MB algorithm. The simulations show that the search success rate is increasing with the growth of the average degree. In particular, in the same scale of network and the same average degree of network, the search success rate of MD algorithm is still the highest among the five algorithms, and that of R1 algorithm is the least. The search success rate of PD algorithm is between the R1 algorithm and the MB algorithm. To the search



**Figure 9:** Search success ratio ( $\Psi$ ) of the various algorithms in ER random graph networks.

hops, the age value of MB algorithm is the least when the search success rate reaches the maximum, and that of R1 algorithm is the largest. The age values of MD algorithm and PD algorithm are between the R1 algorithm and the MB algorithm.

Figure 10 shows the search success ratio ( $\Psi$ ) of the various algorithms in WS small-world networks, where the age value is changed from 1 to 100. Figure 10(a) is the MD algorithm, Figure 10(b) is the PD algorithm, Figure 10(c) is the R1 algorithm, Figure 10(d) is the RK algorithm, and Figure 10(e) is the MB algorithm. We can see that the search success rate is increasing with the growth of the average degree. In particular, in the same scale of network

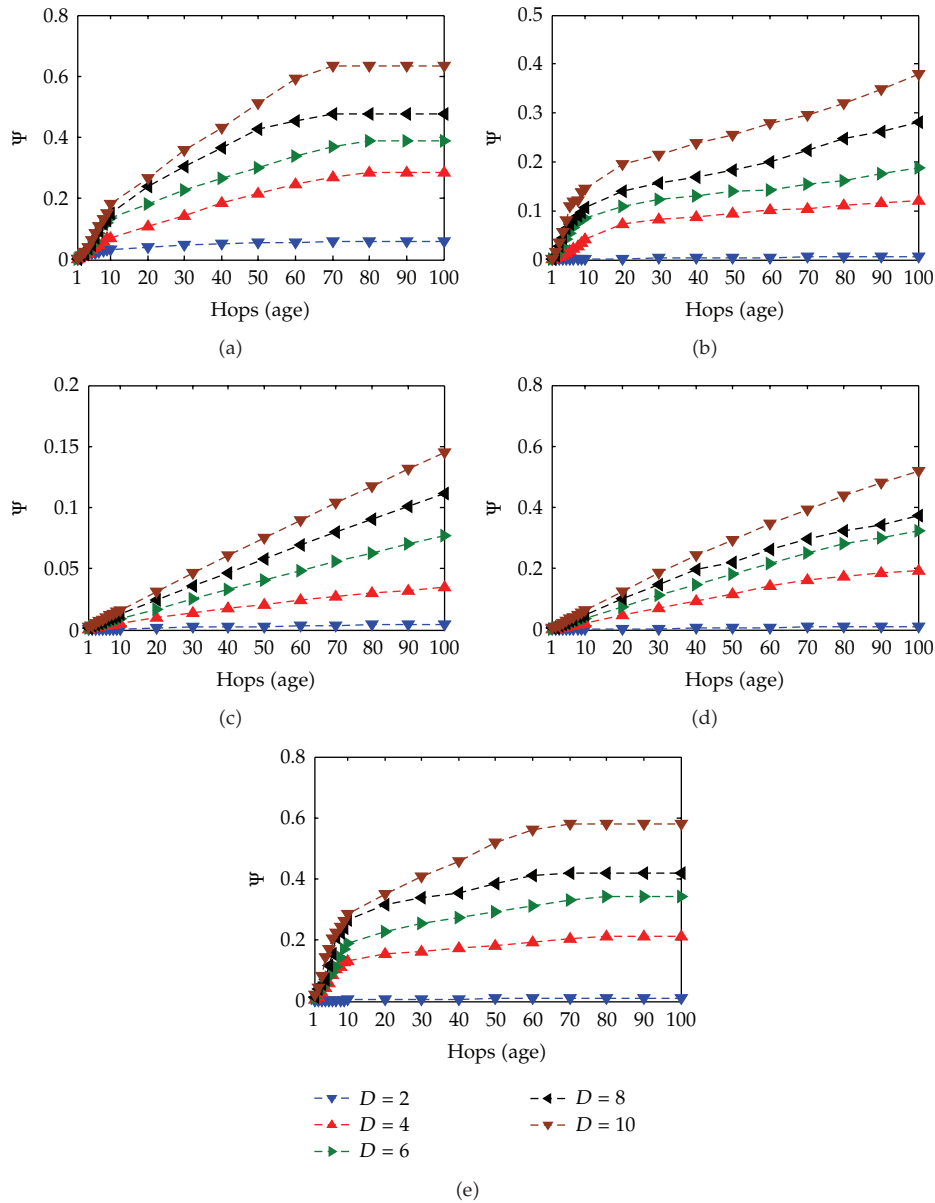


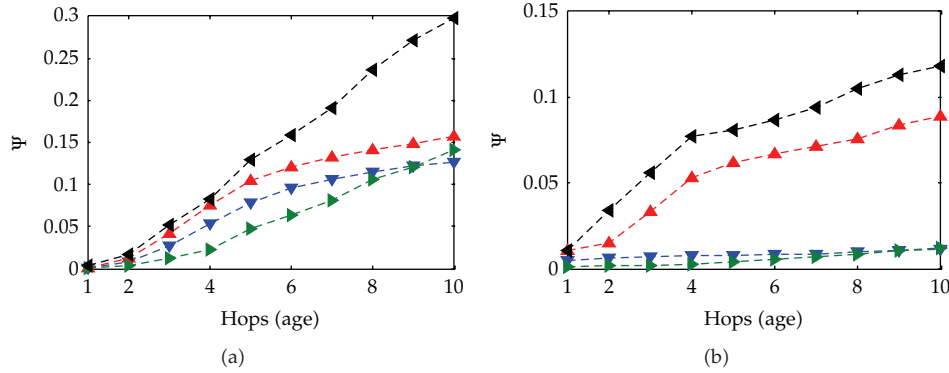
Figure 10: Search success ratio ( $\Psi$ ) of the various algorithms in WS small-world networks.

and the same average degree of network, the search success rate of MD algorithm is the highest among the five algorithms, that of R1 algorithm is the least, and the search success rate of PD algorithm is between the R1 algorithm and the RK algorithm. To the search hops, the age value of MB algorithm is the least when the search success rate reaches the maximum, and that of R1 algorithm is the largest. The age values of MD algorithm and PD algorithm are between the R1 algorithm and the MB algorithm.

In general, the search success rate of MD algorithm excels the MB algorithm and the random walks algorithm in the same conditions. But considering the smaller value of age,

**Table 1:** The message quantity of the MD and MB algorithms in different topology networks.

$D$	Algorithm	Network topology	Message quantity
$D = 4$	MD	BA	$6.1986 \times 10^4$
		ER	$5.3581 \times 10^4$
		WS	$5.9506 \times 10^4$
	MB	BA	$2.8833 \times 10^5$
		ER	$2.4758 \times 10^5$
		WS	$2.6752 \times 10^5$
$D = 6$	MD	BA	$6.8913 \times 10^4$
		ER	$6.1551 \times 10^4$
		WS	$6.5726 \times 10^4$
	MB	BA	$1.1347 \times 10^6$
		ER	$7.4835 \times 10^5$
		WS	$9.3142 \times 10^5$

**Figure 11:** Search success ratio ( $\Psi$ ) of the MD and MB algorithms in different topology networks (BA, ER, and WS).

the success rate of MB is slightly higher than or equal to that of MD (e.g., the success rate of MB when the age value is 10 shown in Figures 8, 9, and 10). This is because the MD algorithm does not have enough hops to return the nodes with second highest degree when it cannot find the resource along the direction of nodes with highest degree. Thus, it returns search failure. With the age increasing, the success rate of MD algorithm gradually transcends that of MB algorithm. To PD algorithm, the search success rate is better than R1 algorithm but is inferior to RK algorithm and MB algorithm.

It is obvious that the search success rate highly depends on the coverage of the search algorithms defined by the age value. However, a large value of age will incur an essential large response time for a requester to obtain search results. A smaller value of age is more appropriate. Thus, we have given the MD and MB algorithm success rate and message quantity in the case of small age value. The response time of these two algorithms is very short and almost the same. The results are shown in Figure 11 and Table 1. From Figure 11, we can see that the success rate of MD algorithm is slightly better than that of MB algorithm when the average degree is small. Instead, the success rate of MB is higher when the average degree is large. And at the same time, MB algorithm generates a mass of messages as shown

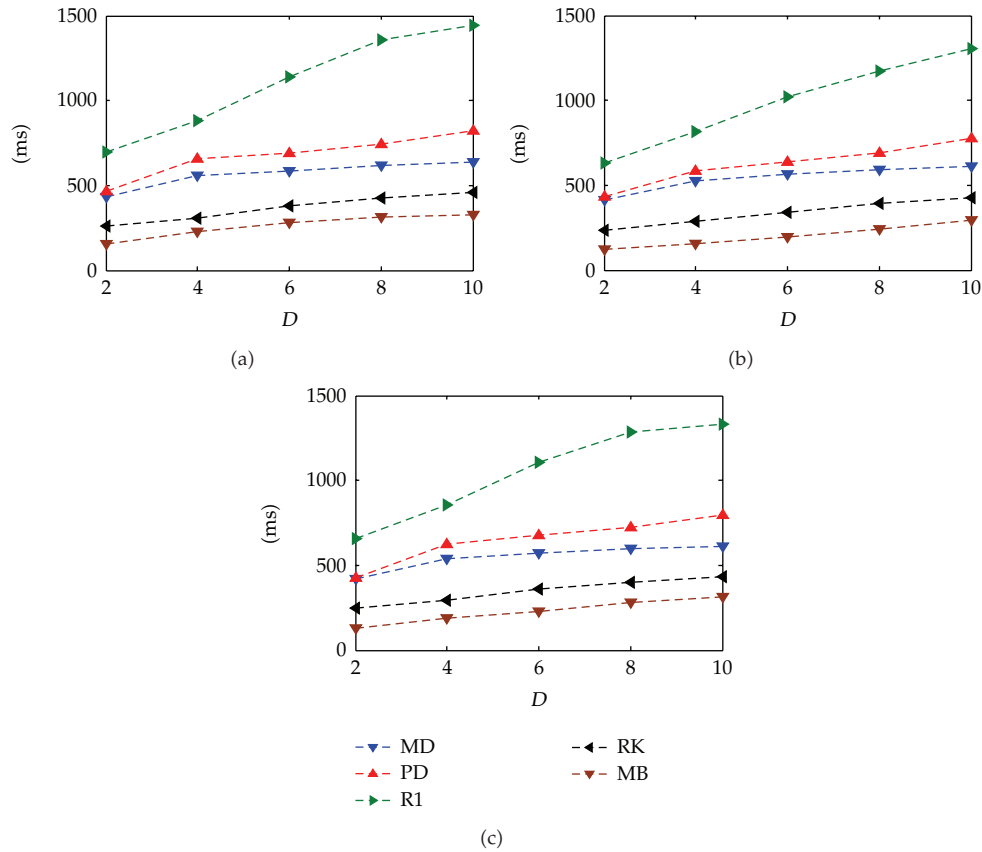


Figure 12: Average response time generated by the various algorithms in different topology networks.

in Table 1; it is about 15 times higher than that of MD algorithm. Therefore, the comprehensive performances of MD algorithm are slightly higher than MB algorithm.

### 3.3. Search Response Time

An unstructured P2P network is a highly dynamic network, and the nodes of the network can join and leave freely. Thus, the search response time is a critical metric for measuring the performance. The search time should be short enough to make sure that the search result is update to date. We define the search response time of a query as the time period when the query is issued until when the source peer receives a response result from the first responder. We calculate the average response time in the condition of search success.

Figure 12 shows the average response time of the various algorithms in different topology networks, where the average degree ( $D$ ) of the networks is chosen from 2 to 10. Figure 12(a) is the scale-free BA network, Figure 12(b) is the ER random graph network, and Figure 12(c) is the WS small-world network. Simulations show that in the three cases, the average response time is increasing with the growth of the average degree of the networks. The search response time of MB algorithm is the least, and the R1 algorithm is the most time



consuming. The average response time of MD and PD algorithm is about twice as much as that of MB algorithm.

Although the average response time of MD algorithm is about twice higher than that of MB as shown in Figure 12, the message quantity of MB algorithm is about 18 times higher than that of MD as shown in Figure 6. When the age value is small as shown in Figure 11, the response time of MD and MB algorithm is almost the same, and the search success rate of these two algorithms is also close under the same values of  $D$ . However, the message quantity of MB is about 15 times higher than that of MD. The algorithm does not exit whose all performance indexes are perfect. Thus, in view of the comprehensive indexes, MD algorithm outperforms MB algorithm.

#### 4. Conclusion

This paper presents the design and evaluation of two memory function search algorithms over the unstructured P2P networks, which, respectively, built on the top of scale-free BA networks, ER random graph networks, and WS small-world networks. The performance of these two algorithms has been compared with the current algorithms used in existing unstructured P2P networks. The search success rate of MD algorithm is averagely 14 times better than the standard random walks algorithm, while the search message quantity is the same order of magnitude with it. Compared with modified BFS algorithm, the search success rate of MD algorithm is higher than that, while the search message quantity averagely reduces by over 18 times. Although PD algorithm can reduce the huge search message quantity, the search success rate of it is inferior to the modified BFS algorithm.

#### Acknowledgments

This paper is supported by the National Natural Science Foundation of China (Grant nos. 61170269, 61121061), the Foundation for the Author of National Excellent Doctoral Dissertation of PR China (Grant no. 200951), the Asia Foresight Program under NSFC (Grant no. 61161140320), the National Science Foundation of China Innovative (Grant no. 70921061), the CAS/SAFEA International Partnership Program for Creative Research Teams and the Program for New Century Excellent Talents in University of the Ministry of Education of China (Grant no. NCET-10-0239).

#### References

- [1] E. Meshkova, J. Riihijarvi, M. Petrova, and P. Mahonen, "A survey on resource discovery mechanisms, peer-to-peer and service discovery frameworks," *Computer Networks*, vol. 52, no. 11, pp. 2097–2128, 2008.
- [2] A. Iamnitchi, M. Ripeanu, I. Foster et al., "Small-world file-sharing communities," in *Proceedings of the IEEE INFOCOM*, vol. 2, pp. 952–963, 2004.
- [3] V. Kalogeraki, D. Gunopulos, and D. Zeinalipour-Yazti, "A local search mechanism for peer-to-peer networks," in *Proceedings of the 11th International Conference on Information and Knowledge Management (CIKM '02)*, pp. 300–307, 2002.
- [4] B. F. Cooper and H. Garcia-Molina, "Ad Hoc, self-supervising peer-to-peer search networks," *ACM Transactions on Information Systems*, vol. 23, no. 2, pp. 169–200, 2005.
- [5] J. Risson and T. Moors, "Survey of research towards robust peer-to-peer networks: search methods," *Computer Networks*, vol. 50, no. 17, pp. 3485–3521, 2006.

- [6] J. Jeong and P. Berman, "Low-cost search in scale-free networks," *Physical Review E*, vol. 75, no. 3, Article ID 036104, 4 pages, 2007.
- [7] V. Isham, J. Kaczmarska, and M. Nekovee, "Spread of information and infection on finite random networks," *Physical Review E*, vol. 83, no. 4, Article ID 046128, 12 pages, 2011.
- [8] D. Trpevski, W. K. S. Tang, and L. Kocarev, "Model for rumor spreading over networks," *Physical Review E*, vol. 81, no. 5, Article ID 056102, 11 pages, 2010.
- [9] D. Shah and T. Zaman, "Rumors in a network: who's the culprit?" *Institute of Electrical and Electronics Engineers. Transactions on Information Theory*, vol. 57, no. 8, pp. 5163–5181, 2011.
- [10] S. Boyd, A. Ghosh, B. Prabhakar, and D. Shah, "Randomized gossip algorithms," *Institute of Electrical and Electronics Engineers. Transactions on Information Theory*, vol. 52, no. 6, pp. 2508–2530, 2006.
- [11] L. A. Adamic, R. M. Lukose, A. R. Puniyani, and B. A. Huberman, "Search in powerlaw networks," *Physical Review E*, vol. 64, no. 4, Article ID 046135, 8 pages, 2001.
- [12] A. Tizghadam and A. Leon-Garcia, "On traffic-aware betweenness and network criticality," in *Proceedings of the INFOCOM IEEE Conference on Computer Communications Workshops*, pp. 1–6, 2010.
- [13] S. Misra and B. J. Oommen, "An efficient dynamic algorithm for maintaining all-pairs shortest paths in stochastic networks," *IEEE Transactions on Computers*, vol. 55, no. 6, pp. 686–702, 2006.
- [14] A. Reinefeld and T. Marsland, "Enhanced iterative-deepening search," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 16, no. 7, pp. 701–710, 1994.
- [15] Y. Yao and J.-C. Yao, "On modified iterative method for nonexpansive mappings and monotone mappings," *Applied Mathematics and Computation*, vol. 186, no. 2, pp. 1551–1558, 2007.
- [16] Z. Wang, S. K. Das, M. Kumar, and H. Shen, "An efficient update propagation algorithm for P2P systems," *Computer Communications*, vol. 30, no. 5, pp. 1106–1115, 2007.
- [17] Q. Lv, P. Cao, E. Cohen, K. Li, and S. Shenker, "Search and replication in unstructured peer-to-peer networks," in *Proceedings of the 16th international conference on Supercomputing (ICS '02)*, pp. 84–95, 2002.
- [18] C. Gkantsidis, M. Mihail, and A. Saberi, "Random walks in peer-to-peer networks," in *Proceedings of the IEEE INFOCOM*, vol. 1, pp. 1–12, 2004.
- [19] A. Kumar, J. Xu, and E. W. Zegura, "Efficient and scalable query routing for unstructured peer-to-peer networks," in *Proceedings of the IEEE INFOCOM*, vol. 2, pp. 1162–1173, 2005.
- [20] C. Gkantsidis, M. Mihail, A. Saberi et al., "Random walks in peer-to-peer networks: algorithms and evaluation," *Performance Evaluation*, vol. 63, no. 3, pp. 241–263, 2006.
- [21] M. Zhong, K. Shen, and J. Seiferas, "The convergence-guaranteed random walk and its applications in peer-to-peer networks," *Institute of Electrical and Electronics Engineers. Transactions on Computers*, vol. 57, no. 5, pp. 619–633, 2008.
- [22] T. Lin, P. Lin, H. Wang, and C. Chen, "Dynamic search algorithm in unstructured peer-to-peer networks," *IEEE Transactions on Parallel and Distributed Systems*, vol. 20, no. 5, pp. 654–666, 2009.
- [23] X. Ming, Z. Shuigeng, G. Jihong, and H. Xiaohua, "A path-traceable query routing mechanism for search in unstructured peer-to-peer networks," *Journal of Network and Computer Applications*, vol. 33, no. 2, pp. 115–127, 2010.
- [24] K. Oikonomou, D. Kogias, I. Stavrakakis et al., "Probabilistic flooding for efficient information dissemination in random graph topologies," *Computer Networks*, vol. 54, no. 10, pp. 1615–1629, 2010.
- [25] R. Beraldi, L. Querzoni, and R. Baldoni, "Low hitting time random walks in wireless networks," *Wireless Communications and Mobile Computing*, vol. 9, no. 5, pp. 719–732, 2009.
- [26] F. Guofu, Z. Jincheng, L. Sanglu, C. Guihai, and C. Daoxu, "RER: a replica efficiency based replication strategy," *Chinese Journal of Electronics*, vol. 17, no. 4, pp. 589–594, 2008.
- [27] G. Huiwei and I. H. S. Horace, "A study of parallel data mining in a peer-to-peer network," *Concurrent Engineering-Research and Applications*, vol. 15, no. 3, pp. 281–289, 2007.
- [28] N. Bisnik and A. A. Alhussein, "Optimizing random walk search algorithms in P2P networks," *Computer Networks*, vol. 51, no. 6, pp. 1499–1514, 2007.
- [29] Y. Liu, J. Han, J. Wang et al., "Rumor riding: anonymizing unstructured peer-to-peer systems," *IEEE Transactions on Parallel and Distributed Systems*, vol. 22, no. 3, pp. 464–475, 2011.
- [30] C.-K. Chau and P. Basu, "Analysis of latency of stateless opportunistic forwarding in intermittently connected networks," *IEEE/ACM Transactions on Networking*, vol. 19, no. 4, pp. 1111–1124, 2011.
- [31] M. E. J. Newman, "Scientific collaboration networks. I. Network construction and fundamental results," *Physical Review E*, vol. 64, no. 1, Article ID 016131, 8 pages, 2001.
- [32] A. Fronczak, P. Fronczak, and J. A. Holyst, "Average path length in random networks," *Physical Review E*, vol. 70, no. 5, Article ID 056110, 7 pages, 2004.

- [33] L. A. Adamic and B. A. Huberman, "Growth dynamics of the world wide web," *Nature*, vol. 401, p. 131, 1999.
- [34] M. Ostili, A. L. Ferreira, and J. F. F. Mendes, "Critical behavior and correlations on scale-free small-world networks: application to network design," *Physical Review E*, vol. 83, no. 6, Article ID 061149, 21 pages, 2011.



# Hindawi

Submit your manuscripts at  
<http://www.hindawi.com>

