



# An Enhanced Searching Algorithm over Unstructured Mobile P2P Overlay Networks

Babar Shah and Ki-II Kim\*, *Member, KIICE*

Department of Informatics, Engineering Research Institute, Gyeongsang National University, Jinju 660-701, Korea

## Abstract

To discover objects of interest in unstructured peer-to-peer networks, the peers rely on flooding query messages which create incredible network traffic. This article evaluates the performance of an unstructured Gnutella-like protocol over mobile ad-hoc networks and proposes modifications to improve its performance. This paper offers an enhanced mechanism for an unstructured Gnutella-like network with improved peer features to better meet the mobility requirement of ad-hoc networks. The proposed system introduces a novel caching optimization technique and enhanced ultrapeer selection scheme to make communication more efficient between peers and ultrapeers. The paper also describes an enhanced query mechanism for efficient searching by applying multiple walker random walks with a jump and replication technique. According to the simulation results, the proposed system yields better performance than Gnutella, XL-Gnutella, and random walk in terms of the query success rate, query response time, network load, and overhead.

**Index Terms:** Gnutella, Peer-to-peer, Random walk, Ultrapeer

## I. INTRODUCTION

Peer-to-peer (P2P) networks consist of higher layer logical connections between peers to form an overlay network independent of the underlying physical network. The access to information anytime and anywhere is the main objective of modern mobile computing. However, due to many limitations, the client-server architecture does not satisfy these demands. Thus, P2P applications build a virtual network over the physical network to address problems that occur in the client-server architecture [1].

In unstructured P2P systems, there is no predefined structure among peers, and a search query is flooded or randomly distributed in the network. The drawback of such searching mechanisms is that a query message may be sent to a node multiple times. The current techniques cannot

remember the nodes that have already been contacted in unstructured P2P systems. As a result of these limitations in current search mechanisms, the efficiency level is very low in unstructured P2P systems [2]. Many researchers have attempted to solve the message duplication problem, but it still exists and affects the efficiency of unstructured P2P systems.

Because of the similarities among P2P and mobile ad-hoc networks (MANETs), a P2P overlay easily runs on supportive MANETs. Mobile P2P networks can be deployed in application scenarios without a reliable and existing infrastructure to mitigate task overload by distributing them among other peers [3, 4]. Differences also exist between the two systems, and therefore simply adopting a P2P overlay over MANETs is undesirable due to the deficiency of the infrastructure, node mobility, and energy issues. The limited

Received 21 January 2013, Revised 20 February 2013, Accepted 04 March 2013

\*Corresponding Author Ki-II Kim (E-mail: [kikim@gnu.ac.kr](mailto:kikim@gnu.ac.kr), Tel: +82-55-772-1373)

Department of Informatics, Engineering Research Institute, Gyeongsang National University, 501 Jinju-daero, Jinju 660-701, Korea.

**Open Access** <http://dx.doi.org/10.6109/jicce.2013.11.3.173>

print ISSN: 2234-8255 online ISSN: 2234-8883

© This is an Open Access article distributed under the terms of the Creative Commons Attribution Non-Commercial License (<http://creativecommons.org/licenses/by-nc/3.0/>) which permits unrestricted non-commercial use, distribution, and reproduction in any medium, provided the original work is properly cited.

Copyright © The Korea Institute of Information and Communication Engineering

features of an ad-hoc network decrease the query success rate and connectivity in a P2P system. Continuous changes will occur in the underlying topology as the nodes move in MANETs [5, 6].

Earlier research evaluated the performance of different content searching techniques in P2P networks over wired and fixed scenarios like the internet. Their performance outcomes are not directly applicable to MANETs, since the wireless and mobile medium is much more dynamic. To solve these limitations in unstructured Gnutella-like networks over MANET, we limit our work in this paper to two contributions. First, we consider enhanced techniques for peer discovery and their connection to the existing P2P overlay. In this section, we also propose new techniques for selection of ultrapeers and different pong caches used in leaf peers and ultrapeers. Our second contribution is the enhancement of unstructured searching techniques over MANET models richer than those of previous works. Our analysis for P2P over MANETs considers the searching overhead, number of peers, search latency, and query success rate to provide a more convincing depiction of the proposed work. Thus, the proposed work will reduce the average searching overhead and delay, and will enhance query success rate by the searching mechanism and in unstructured P2P networks.

The rest of the paper is organized as follows: Section II is the algorithm description of the proposed protocol. The subsections of Section II discuss the pong cache, ultrapeer, peer discovery, and searching mechanism of the algorithm in detail. Section III discusses and analyzes the experimental results. Finally, Section IV concludes with a critical summary and some directions for future work.

## II. ALGORITHM DESCRIPTION

In our evaluation, we represent an unstructured Gnutella-like P2P network with enhanced techniques applied mainly in new peer discovery, connection, and searching using multiple walker random walks. Gnutella v.6 with an open source code [7], is used as a starting point for the enhanced Gnutella-like P2P system over MANETs.

We applied the desired changes to the file containing the bootstrap mechanism for newly arrived peers to keep information about neighboring peers in the P2P network in order to start quick communication. We also introduced some major changes to the uniform resource locators to update the traditional pong cache mechanism so that the leaf peers would not handle discovery request *HELLO* messages and forward them to their ultrapeer. The ultrapeer and its pong cache works as a local base station to provide solutions locally and reduce P2P network resources. We enhanced the caching mechanism to provide efficient searching

and sharing of information. Such a technique will also reduce the load on the overall P2P network. The query search message of the traditional Gnutella protocol was also enhanced by adding new features in the relevant classes. The proposed approach uses multiple walker random walks with jump functionality and contacting the query generator peer before message replication. To balance the workload and provide efficient access to all of the peers in the P2P network, a new approach was added to help the newly arriving peers. Query search messages are handled in the same way as peer discovery. As in our algorithm, the ultrapeer plays the role of a local base station and tries to handle query messages locally.

The propose for the protocol to use random walk with  $m$  multiple walkers is to avoid flooding. The source peer creates and forwards a query message to its ultrapeer. The ultrapeer creates  $m$  walkers of the same query message with fixed a peer-to-leave (PTL) value of 4 and forwards them to randomly selected  $n$  neighboring ultrapeer nodes. The priority in the random selection of the neighbors is based on a defined metric value. This information is also stored in the pong cache of the ultrapeer for future reference to avoid duplicate queries. When the neighbor ultrapeer receives a walker, it will examine itself for the requested object. If it finds the object, then the walker is removed from the network and the source peer is contacted accordingly. If the information is not found locally and the PTL file of the walker has not reached zero, then the walker is forwarded and its PTL is decremented by one unit. Ultrapeers and leaf peers use ping/pong messages to control the availability of their neighbors.

### A. Pong Cache and Ultrapeer Mechanism

A pong cache technique was introduced in Gnutella v.6 to reduce network overhead and bandwidth [8]. A pong cache mechanism has a direct connection with the response message of the ping and routing of the control messages at the application level. Most of the studied schemes assume a static environment and do not consider the characteristics of mobile environments. The authors in [8] discussed a mobile environment in their work and proposed a caching scheme for peers. The caching scheme in their work maintains a separate cached data structure for the neighbor peers. The present study aims to overcome these problems in the proposed caching technique. Our caching technique will use a separate pong cache for leaf peers and ultrapeers in order to have flexibility regarding the pong information and improve the overhead occurred by exchange of ping/pong messages.

The current Gnutella v.6 defines peers in two different formats, that is, leaf and ultrapeers. In the Gnutella P2P overlay, ultrapeers are considered more powerful nodes

compare to leaf peers on the basis of their resources, such as storage, processing power, and network bandwidth. Due to these rich resources, ultrapeers have been assigned the role of local head to perform different operations, such as searching, downloading, and processing [10]. An application is required to provide an entrance to the unstructured P2P network and a connection to already existing peers. Thus new peers establish an initial connection with one of the existing ultrapeers. The connected ultrapeers then forward information to other neighbor ultrapeers in the network. To forward query walkers, the priority in selecting random neighbors is given to ultrapeers that satisfy the defined metric. This type of useful information is also stored in the pong cache of the ultrapeer for future use.

## B. Peer Discovery and Connection

In this proposed P2P system, a reactive routing protocol is applied for the peer discovery process. The new peer establishes an initial connection with the existing peer in a P2P overlay by broadcasting a *HELLO* message to any peer reachable in its transmission range within one hop. If the receiver peer is a leaf peer, then it will reply with a *HELLO* response and forward this newly arrived peer information to his ultrapeer. If the ultrapeer receives the request directly or through a leaf peer, it will reply with a *WELCOME* message and forward detailed information about itself and all connected in-range ultrapeers. Upon receiving such information the requesting peer will decide on the outgoing connection.

The ultrapeer will share the pong cache table and information regarding the number of slots available with each ultrapeer and whether it is mobile or stationary to the requesting peer. The proposed unstructured Gnutella-like P2P protocol offers the ability to monitor the maximum number of incoming and outgoing connections for each peer and ultrapeer. Each ultrapeer will use the maximum outgoing connections, but for maximum incoming connections, it will keep one slot empty all the time to accept incoming connection requests. This means that the state is set to full and will not accept any new connection but can accept any newly arrived peer *HELLO* message and can share information with it by a *WELCOME* message.

For a better P2P overlay performance, our peer connection process considers the high capacity of an ultrapeer and the number of both leaf peers and neighbor ultrapeers connected to a selected ultrapeer to make the joining decision. Once a newly arrived peer discovers its closest best ultrapeer, it begins the Gnutella-like handshaking process to establish a connection. This enhanced connection establishment technique will try to connect the peer to the best ultrapeer having the required data object. The procedure for this algorithm is presented in Table 1.

**Table 1.** Algorithm description for discovery

Peer Discovery and Pong Caching Algorithm
<pre> procedure Ping_Message (new_ping, conn)   if new_ping = HELLO then     if rec_peer = leaf_peer then       reply welcome       forward info to ultrapeer     else       if rec_peer = ultrapeer then         reply welcome         forward pong_cache details       end if     end if   else     if new_ping.ptl &gt; 0 then       update pong_cache and maintain connection       reply Pong(conn, pong)     else       discard new_ping message     end if   end if end procedure  procedure Pong (conn, pong)   if pong_cache.status != full then     if pong = new_pong then       pong_cache &lt;- new_pong(credentials)       forward Pong (conn, pong) to neighbor     else       update pong_cache &lt;- new_pong       forward Pong (conn, pong) to neighbor     end if   else     remove pong_cache &lt;- old_pong (credentials)     pong_cache &lt;- new_pong(credentials)     forward Pong (conn, pong) to neighbor   end if end procedure                     </pre>

Suppose  $\mu_i$ ,  $\tau_i$ , and  $\omega_i$  represent the capacity of ultrapeer  $i$ , and the number of leaf peers and neighbor ultrapeers connected to an ultrapeer  $i$ , respectively. The probability  $\rho_i$  of ultrapeer  $i$  for newly arrived peer  $e$  is directly proportional to the current number of connected neighbor ultrapeers  $\omega_i$  and its capacity, where  $\rho_i$  is inversely proportional to its currently connected number of leaf peers  $\tau_i$ .

$$\rho_i = \frac{\mu_i \times \omega_i}{\tau_i}, i \in AU(t), \quad (1)$$

where  $AU(t)$  is the set of active ultrapeers at time  $t$ . To avoid a workload on a specific ultrapeer and delay on the overall P2P overlay network, we define three variables  $\alpha$ ,  $\beta$ , and  $\gamma$  to control the probability of joining new peers.

$$\rho_i = \frac{(\mu_i)^\alpha \times (\omega_i)^\beta}{(\tau_i)^\gamma}, \quad (2)$$

where  $\alpha > 0$ ,  $\beta > 0$ , and  $\gamma > 0$ .

We further extend the above equation to define the overall probability  $\rho_i$  for the maximum number of leaf peer and ultrapeer connections.

$$\rho_i = \frac{(\mu_i)^\alpha \times (\varpi_i)^\beta}{(\tau_i)^\gamma} \cdot \sum_{j \in AU(t)} \frac{(\mu_j)^\alpha \times (\varpi_j)^\beta}{(\tau_j)^\gamma} \quad (3)$$

By applying the above in our algorithm, each ultrapeer has to forward its capacity  $\mu$  number of leaf peers attached ( $\tau$ ) and number of ultrapeers connected ( $\varpi$ ) to the newly arrived peer to make a connection decision. This information is also exchanged between the neighbor ultrapeers to assign a transition probability to their neighbor ultrapeer edges.

### C. Searching Mechanism

The search mechanism in the proposed protocol is biased on multiple walker random walks with a jump and replication technique instead of blindly flooding query messages as in the typical Gnutella protocol. For a given query, the ultrapeer selects neighbor ultrapeers randomly based on defined criteria and forwards walkers to them. The proposed approach will continue this search process until the query is solved or the walkers reach its end.

Suppose the querying ultrapeer  $Q$  initiates  $m$  random walkers to search object  $O$  in the P2P overlay of  $n$  ultrapeers. Because this work discusses the mobile P2P environment, we are considering only  $n$  ultrapeers and not dealing with the total number of leaf peers in the network. Each walker can jump to a randomly selected 2-hop neighbor ultrapeer. The randomly selected 2-hop neighbor ultrapeer will be the one that has not been previously visited or selected for that walker unless this is the only neighbor ultrapeer. Each ultrapeer shares its pong cache with its directly connected or one-hop neighbor.

The proposed mechanism allows a peer to randomly select the neighbor peers from available online active peers in the P2P network. Because of its dynamic nature, the P2P network needs an advanced topology control scheme for maintaining an up-to-date list of neighbors. To overcome this problem, all of the peers in the Gnutella protocol periodically send ping messages to their neighbors and wait for a pong message in order to check whether their neighbors are still active. When no answer is received from a neighbor, it is replaced by a peer randomly chosen from the set of online peers. Table 2 shows the detail procedure for mentioned algorithm.

**Table 2.** Algorithm description for searching

Search Algorithm
<pre> procedure MWRSW(o, ptl)   if found object (o) or query = duplicate then     ignore query   else     if ptl = 0 then       if query = replicate then         reply unsuccessful       else         contact query initiator       end if     else       walker &lt;- jump_selected_random_peer()       call jump.MWRSW(o, ptl-1)     end if   end if end procedure  procedure m_MWRES(m, o, ptl)   for all success[i] = NULL   for i ∈ (1 to m) do     success[i] &lt;- MWRSW(o, ptl)   end for   reply found_object(success) end procedure </pre>

### III. PERFORMANCE EVALUATION

In order to evaluate our proposed unstructured P2P Gnutella-like protocol over mobile ad-hoc networks, we used Network Simulator (NS2 v.2.34). The analysis and measurement of the protocol were based on three primary metrics: search queries, query success rate, and average network overhead and queries solved by number of time and hops. We choose three different existing protocols, Gnutella (flooding), XL-Gnutella, and Random Walk, for performance comparison with our proposed protocol. In order to evaluate performance, we created a scenario based on network mobility and analyzed the interaction of the protocol with the underlying reactive routing protocols.

Search overhead is further reduced to an acceptable value by efficient selection of a neighbor to forward the query message. Thus, the requesting ultrapeer sends query walkers to a subset of neighbor ultrapeers, which are the best choice to return responses of good quality. The proposed new P2P protocol uses the capacity of the neighbor ultrapeer and the degree consists of the number of leaf peers and neighbor ultrapeers attached as a criterion for a high-capacity neighbor. The following Fig. 1 shows the comparison of the average search overhead between the new and existing unstructured P2P protocols. The Gnutella protocol overhead is very high because of flooding and duplication of messages. The overhead of the proposed protocol is very low compared to the other three. The reason is the selection criteria, efficient forwarding of walkers, and keeping check

on message duplication.

Fig. 2 shows the average of successful queries during simulation based on the number of peers. Other unstructured protocol results are satisfactory when the number of peers in the overlay is increased. At the beginning, when the number of peers is around 40, the query success rate is very low. In the proposed protocol, the rate is quite good and acceptable even from the start.

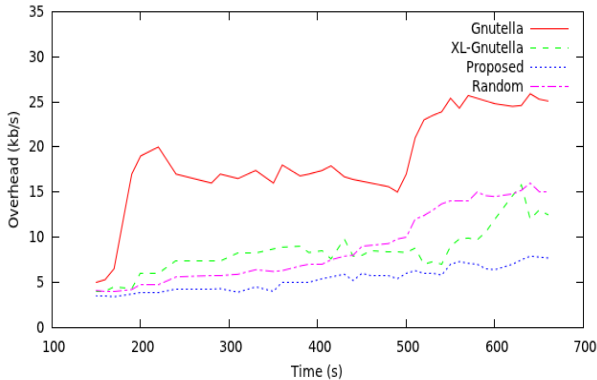


Fig. 1. Average search overhead.

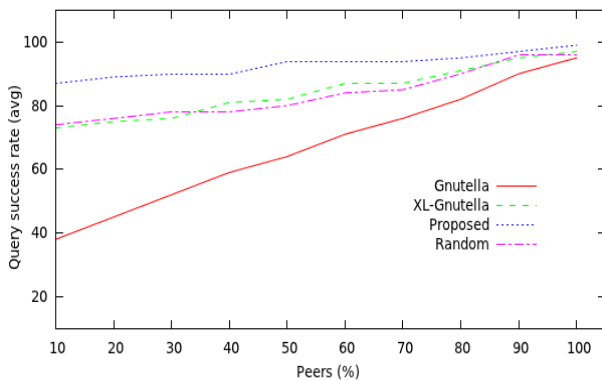


Fig. 2. Average query success rate.

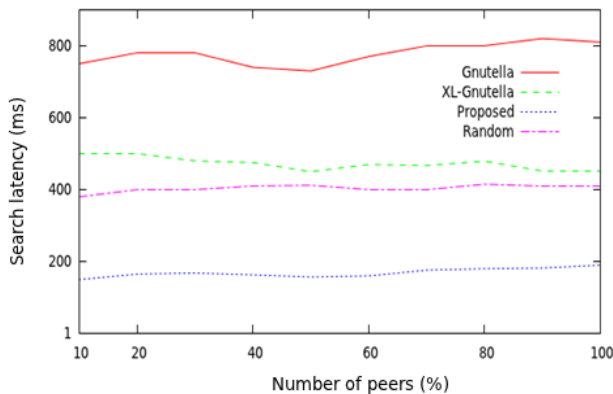


Fig. 3. Average time-based search latency.

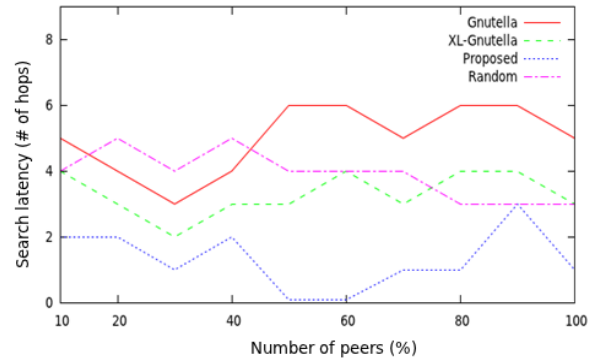


Fig. 4. Average hops-based search latency.

In Figs. 3 and 4, search latency is used as a performance metric to evaluate the search efficiency of the proposed P2P protocol. The search latency is measured by the shortest time the protocol takes for a query search to reach a peer that has a requested object.

Figs. 3 and 4 show the average search latency for a requested object measured by the time in milliseconds and the number of hops, respectively. The average search latency of the proposed protocol in both of the figures is lower than the other unstructured P2P overlays. The search performance of the proposed unstructured P2P protocol is better than the others because of the metric used for selection of the neighbor ultrapeer to forward walkers and the jump mechanism. Similarly, another reason for efficient searching is that every ultrapeer contains the sharing objects of directly connected ultrapeers and the address of the one-hop-away ultrapeers. The proposed approach aimed to eliminate duplication of query messages. Thus, every hop forwarding discovered a new peer visited for the first time for a requested object; therefore, a search message can quickly cover a large area of peers within less time and fewer hops.

#### IV. CONCLUSION

In this paper, the information distribution and efficient search in a P2P network under multiple walker random walks with jump and replication mechanisms over a MANET has been studied and analyzed. The novel approach of new peer discovery and connectivity is also discussed in detail. The enhanced pong caching technique and multiple walker random walks mechanism was studied, analyzed, and simulated from a novel perspective. We reduced the searching and discovery time by considering the capacity and degree of the ultrapeers. The selection of efficient neighbor ultrapeers in a P2P network will result in a reduction in the searching time, overhead, and traffic through new defined metric based on the ultrapeer capacity

and degree for such selection.

Further study on the replication mechanism, pong cache, discovery process, and its numerous parameters with detailed simulations will be considered in future work in the area.

## ACKNOWLEDGMENTS

This research was supported by the Ministry of Science, ICT & Future Planning, Korea, under the "SW master's course of a hiring contract" support program (NIPA-2013-HB301-13-1004) supervised by the National IT Industry Promotion Agency.

## REFERENCES

- [ 1 ] R. Schollmeier, I. Gruber, and F. Niethammer, "Protocol for peer-to-peer networking in mobile environments," in *Proceedings of the 12th International Conference on Computer Communications and Networks*, Dallas, TX, pp. 121-127, 2003.
- [ 2 ] M. Yang and Z. Fei, "A novel approach to improving search efficiency in unstructured peer-to-peer networks," *Journal of Parallel and Distributed Computing*, vol. 69, no. 11, pp. 877-884, 2009.
- [ 3 ] C. Gottron, A. Konig, and R. Steinmetz, "A cross-layer approach towards robustness of mobile peer-to-peer networks," in *Proceedings of the IEEE 8th International Conference on Mobile Ad-Hoc and Sensor Systems*, Valencia, Spain, pp. 703-708, 2011.
- [ 4 ] D. Talia and P. Trunfio, "Toward a synergy between P2P and grids," *IEEE Internet Computing*, vol. 7, no. 4, pp. 94-96, 2003.
- [ 5 ] A. Mawji, H. Hassanein, and X. Zhang, "Peer-to-peer overlay topology control for mobile ad hoc networks," *Journal of Pervasive and Mobile Computing*, vol. 7, no. 4, pp. 467-478, 2011.
- [ 6 ] H. D. Choi, H. H. Park, and M. Woo, "An enhanced Gnutella for Ad-Hoc networks," in *Proceedings of IEEE International Conference on Systems and Networks Communication*, Tahiti, French Polynesia, 2006.
- [ 7 ] T. Klingberg and R. Manfredi, "Gnutella 0.6," Network Working Group, 2002 [Internet], Available [http://rfc-gnutella.sourceforge.net/src/rfc-0\\_6-draft.html](http://rfc-gnutella.sourceforge.net/src/rfc-0_6-draft.html).
- [ 8 ] H. Shen, M. S. Joseph, M. Kumar, and S. K. Das, "PRECinCt: a scheme for cooperative caching in mobile peer-to-peer system," in *Proceedings of the 19th IEEE International Parallel and Distributed Processing Symposium*, Denver, CO, pp. 57-66, 2005.
- [ 9 ] A. Boukerche, A. Zarrad, and R. Araujo, "A novel optimized caching technique for mobile Gnutella based network to support large-scale collaborative virtual environment," in *Proceedings of the 41st Annual Simulation Symposium*, Ottawa, Canada, pp. 289-297, 2008.
- [ 10 ] A. Boukerche, A. Zarrad, and R. Araujo, "A smart Gnutella overlay formation for collaborative virtual environments over mobile ad-hoc networks," in *Proceedings of the 10th International Symposium on Distributed Simulation and Real-Time Applications*, Torremolinos, Spain, pp. 143-156, 2006.



**Babar Shah**

received an M.Sc. degree from the University of Peshawar, an M.S. from the University of Derby, UK, and is enrolled as a Ph.D. researcher at Gyeongsang National University, WMN Lab, South Korea. His research interests include wireless and mobile networking, routing protocols, wireless sensor networks, P2P, and cloud computing.



**Ki-II Kim**

received M.S. and Ph.D. degrees in computer science from the ChungNam National University, Daejeon, Korea, in 2002 and 2005, respectively. He is currently with the Department of Informatics at Gyeongsang National University. His research interests include routing for MANETs, QoS in wireless networks, multicast, and sensor networks.