

# **DYNAMIC SEARCH IN UN-STRUCTURED PEER-TO-PEER NETWORKS**

Dr. B Sankara Babu

Professor, Department of Computer Science and Engineering, Gokaraju Rangaraju Institute of Engineering and Technology, Hyderabad, A.P, India

**ABSTRACT:** Peer-to-peer networks are becoming popular today because these networks offer sharing of resources through direct exchange. In unstructured peer-to-peer networks, each node does not have global information about the whole topology and the location of other nodes. A dynamic property of unstructured P2P networks, capturing global behaviour is also difficult. Search algorithms to locate the queried resources and to route the message to the target node. Flooding and Random walk are two typical examples of blind search algorithms by which query messages are sent to neighbours without any knowledge about the possible locations of the queried resources or any preference for the directions to send. Both algorithms are not suitable to route a message to target. Dynamic search improves the performance of searching by using the super peers which operates both as a server to a set of clients and an equal in the network of super peers. Dynamic search reduces the number of messages by using the knowledge based searching. Each node could relay query messages more efficiently to reach the target node.

**KEYWORDS:** Peer-To-Peer Networks, Flooding, Super Peer

## **I. INTRODUCTION**

Peer to peer networks are increasingly becoming popular because they offer opportunities for real-time communication, ad-hoc collaboration and information sharing in a large-scale distributed environment. Peer to peer computing discusses about the characteristics of peer to peer networks and the resource sharing and how they communicate with each other advantages of the peer to peer networks.

Peer-to-peer computing is defined as the sharing of computer resources and information through direct exchange. The most distinct characteristic of P2P computing is that there is symmetric communication between the peers, each peer has both a client and a server role. The advantages of the P2P systems are multi-dimensional, they improve scalability by enabling direct and real-time sharing of services and information, enable knowledge sharing by aggregating information and resources from nodes that are located on geographically distributed and potentially heterogeneous platforms, and provide high availability by eliminating the need for a single centralized component. The basic characteristic of the P2P network is that there is a group of nodes with the same type of interests connected over the same communication system. The P2P network is self-organized and self-administrative as the nodes autonomously discover their peers, and self-healing as the nodes automatically try to find new peers if their current peers are (temporarily or permanently) disconnected from the network. The Peer-to-Peer (P2P) architectures that are most prevalent in today's Internet are decentralized and unstructured.

### **A. UNSTRUCTURED PEER-TO-PEER NETWORKS**

Unstructured peer-to-peer (P2P) networks, each node does not have global information about the whole topology and the location of queried resources. Because of the dynamic property of unstructured P2P networks, correctly capturing global behaviour is also difficult. Search algorithms provide the capabilities to locate the queried resources and to route the message to the target node. Thus, the efficiency of search algorithms is critical to the performance of unstructured P2P networks.

# International Journal of Innovative Research in Science, Engineering and Technology

(An ISO 3297: 2007 Certified Organization)

Website: [www.ijirset.com](http://www.ijirset.com)

Vol. 6, Issue 1, January 2017

- Flooding

Flooding [11], which belongs to BFS-based methods, is the default search algorithm for Gnutella network. By this method, the query source sends its query messages to all of its neighbours. When a node receives a query message, it first checks if it has the queried resource. If yes, it sends a response back to the query source to indicate a query hit. Otherwise, it sends the query messages to all of its neighbors, except for the one the query message comes from. The drawback of flooding is the search cost.

- Random walk

On the other hand, random walk [3] (RW) is a conservative search algorithm, which belongs to DFS-based methods. By RW, the query source just sends one query message (walker) to one of its neighbours. If this neighbour does not own the queried resource, it keeps on sending the walker to one of its neighbours, except for the one the query message comes from, and thus, the search cost is reduced. The main drawback of RW is the long search time.

- Dynamic searching

The proposed is a dynamic search (DS) in unstructured peer to peer network uses knowledge based searching and performs well. DS uses knowledge base search where Knowledge-based search algorithms take advantage of the knowledge learned from previous search results and route query messages with different weights based on the knowledge. Thus, each node could relay query messages more intelligently.

## ***B. PROBLEM DEFINITION***

In unstructured peer to peer networks it is difficult to acquire the global knowledge. Which makes searching more difficult. The previous unstructured peer to peer applications used Flooding and RW are two typical examples of blind search algorithms by which query messages are sent to neighbors without any knowledge about the possible locations of the queried resources or any preference for the directions to send which causes high message cost and response delay. The proposed dynamic search in unstructured peer to peer network searches with the previous knowledge and uses the super peers in a way to reduce message traffic. A super-peer is a node in a peer-to-peer network that operates both as a server to a set of clients, and as an equal in a network of super-peers and reduces message traffic. It combines the features of client server model and pure peer to peer model which makes search more efficient. It distributes the load among the high capacity peers or super peers.

## **II. LITERATURE SURVEY**

A peer-to-peer (P2P) distributed network figure2.1 architecture is composed of participants that make a portion of their resources (such as processing power, disk storage, and network bandwidth) available directly to their peers without intermediary network hosts or servers. Peers are both suppliers and consumers of resources, in contrast to the traditional client-server model where only servers supply, and clients consume. P2P was popularized by file sharing systems like Napster. P2P networks are typically formed dynamically by ad-hoc additions of nodes. In an 'ad-hoc' network, the removal of nodes has no significant impact on the network. The distributed architecture of an application in a P2P system provides enhanced scalability and service robustness.

# International Journal of Innovative Research in Science, Engineering and Technology

(An ISO 3297: 2007 Certified Organization)

Website: [www.ijirset.com](http://www.ijirset.com)

Vol. 6, Issue 1, January 2017

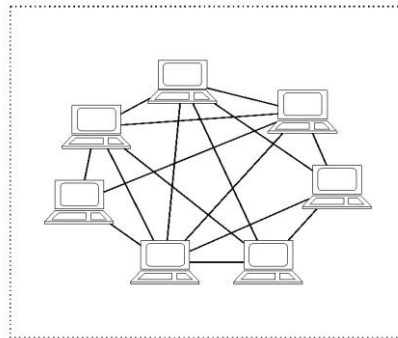


Fig. 2.1. Peer-to-Peer Network Architecture

The Fig.2.1, shows about the existing of Peer-to-Peer Network Architecture. In this architecture peers will be acting as both suppliers and consumers of resources.

## A. ARCHITECTURES OF P2P SYSTEMS

There are mainly three different architectures for P2P systems centralized, decentralized structured and decentralized unstructured (Figure 2.2). In the centralized model, such as Napster [5], central index servers are used to maintain a directory of shared files stored on peers with the intention that a peer can search for the location of a desired content from an index server. On the other hand, this design makes a single point failure and its centralized nature of the service creates systems susceptible to denial of service attacks. Decentralized P2P systems have the advantages of eliminating dependence on central servers and providing freedom for participating users to swap information and services directly between each other. In decentralized structured models, such as Chord, the shared data placement and topology characteristics of the network are strongly controlled on the basis of distributed hash functions. In decentralized unstructured P2P systems, such as Gnutella and KaZaA[5], there is neither a centralized index nor any strict control over the network topology or file placement.

### • Pure Peer-to-Peer model

Pure peer-to-peer applications will not use a central server at all (except possibly for logging onto the network). Queries for files can be flooded through the network or more intelligent mechanisms can be used. FreeNet[4] still uses this model because it offers an unprecedented anonymity, not found in any other architecture.

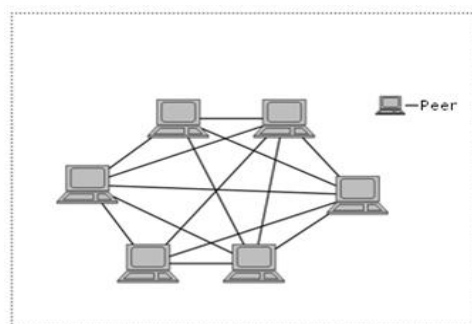


Fig. 2.2. A Purely decentralized architecture

# International Journal of Innovative Research in Science, Engineering and Technology

(An ISO 3297: 2007 Certified Organization)

Website: [www.ijirset.com](http://www.ijirset.com)

Vol. 6, Issue 1, January 2017

- Hybrid Peer-to-Peer model

Hybrid architectures are the latest development in the peer-to-peer community [1]. Their goal is to offer the best of both worlds. Through the introduction of so-called ultra peers, hybrid architectures have properties of both the mediated and the pure architectures. The ultra peer will perform the task of a server in the mediated architecture, but for only a subset of the peers. The ultra peers themselves are connected through a pure peer-to-peer network.

- Distributed Hash Tables

Distributed hash tables (DHTs) [4] are a class of decentralized distributed systems that provide a lookup service similar to a hash table: (*key, value*) pairs are stored in the DHT, and any participating node can efficiently retrieve the value associated with a given key. Responsibility for maintaining the mapping from keys to values is distributed among the nodes, in such a way that a change in the set of participants causes a minimal amount of disruption. This allows DHTs to scale to extremely large numbers of nodes and to handle continual node arrivals, departures, and failures.

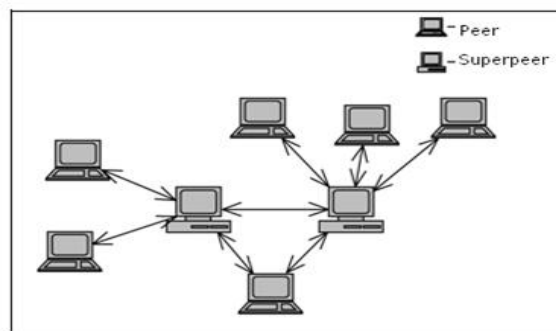


Fig. 2.3. A Partially Decentralized architecture

### III. SYSTEM DESIGN AND IMPLEMENTATION

- Procedure for Dynamic search in Unstructured Peer-to-Peer Networks

**Input:** Query (f: file name)

**Output:** Port number where the queried file is located

- Step1:- Initialize the super peers.  
Step2:- Initialize the peers.  
Step3:- Enter the port login of the peer.  
Step4:- Select the super peer to which it wants to connect. Step5:- Upload the data or files into the super peer. Step6:- Querying the super peer.  
Step7:- Super peer monitors the peers according to the query. Step8:- Updates probability table s:f if it gets reply from the peer. Step9:- Replies to the peer with the port number or ip of the peer.  
Step10:- Peer downloads the requested file or data by making a connection. Step11:- Disconnect the peer  
Step12:- end.

- System Process

Initially peer logs in by entering the ip. And it will choose one of the super peer for making a connection. After connecting it will upload the files which it wants to share. Clients or weak peers asks query to super peer only. Whenever a query sends to super peer it will search its local cache and if it finds updates the probability table and replies to peer by port number of peer where the queried file is located. Instead of sending messages to all clients it will use the previous knowledge.

# International Journal of Innovative Research in Science, Engineering and Technology

(An ISO 3297: 2007 Certified Organization)

Website: [www.ijirset.com](http://www.ijirset.com)

Vol. 6, Issue 1, January 2017

- Pseudo code for searching

1. Super peer local search(s: super peer, f: file name)
2. If any entry<f; q>exists in cache s:F then
3. Increase the priority of <f; q >s in F
4. Return q Else Return ERROR, “file f not found”
5. Super peer searches (s, f):
6. Perform a search in the Super peer network to locate a super peer t to which has any entry<f; q>in its cache
7. If succeed then Insert <f; q >into s: F
8. Return <q; t> Else
9. Return ERROR “file f not found” for s in q: s do
10. If p: S contains s then Increase priority of s in p: S
11. Else insert s into p: S

- Flow Chart for searching

The figure 3.1 is the flowchart diagram representing the searching. Initially peer queries the super peer and it searches in its cache if it found then increments the corresponding peer probability value. If it does not found then it will pass the query to the neighbouring super peer if it finds the queried resource then it will give reply to super peer and the corresponding probability will be incremented.

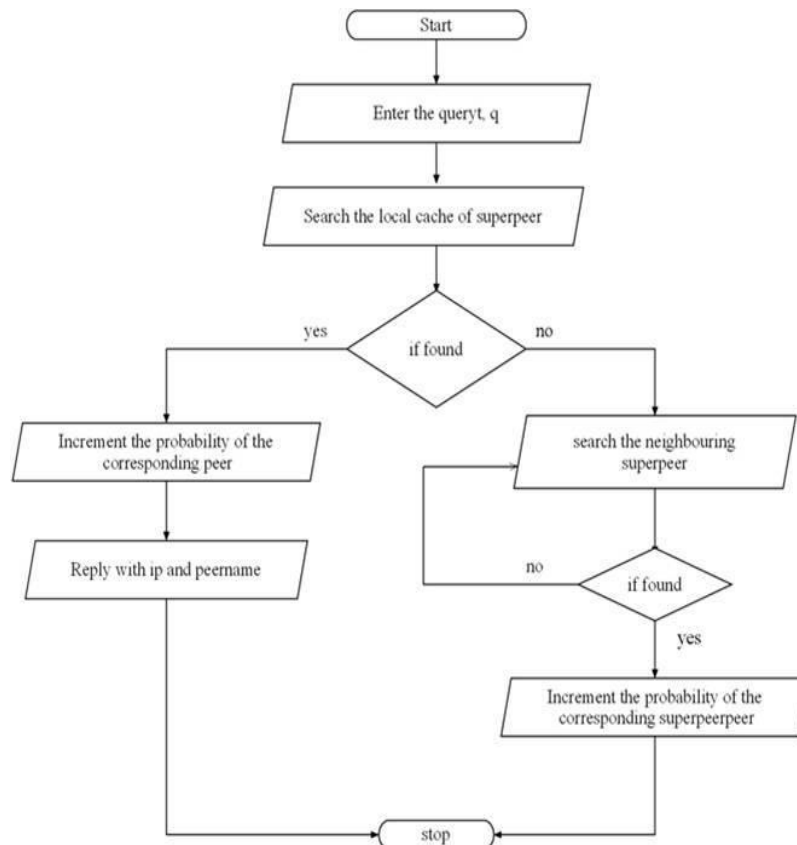


Fig. 3.1.. Flow chart for Dynamic searching

# International Journal of Innovative Research in Science, Engineering and Technology

(An ISO 3297: 2007 Certified Organization)

Website: [www.ijirset.com](http://www.ijirset.com)

Vol. 6, Issue 1, January 2017

- Modules
  1. Peer request Super Peer response
  2. Upload Super Peer updating file request
  3. Updating Probability table response

## A. IMPLEMENTATION

The figure 3.12 is the comparison of the proposed model when it uses flooding and dynamic searches the number of messages were reduced. Because in dynamic search instead of sending messages to all the neighbouring nodes it uses previous knowledge for searching and the table 3.1 values are the number of messages required for searching documents by using dynamic search and flooding.

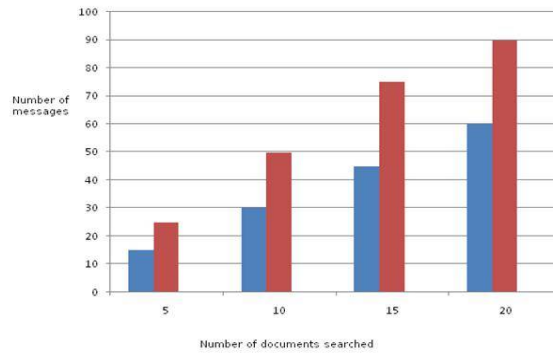


Fig. 3.2. Dynamic search in Un-Structured Peer-to-Peer Networks

Table 1. The table for finding documents using flooding and dynamic search

No of documents	No of message (DS)	No of messages (flooding)
5	15	25
10	30	50
15	45	75
20	60	90

## IV. EXPERIMENTAL RESULTS

In the figure 4.1 initially peer makes connection with one of the super peer by entering its port number as in the figure 4.2 .The peer has upload and search buttons by clicking peer can upload and search a file. The figure 4.2 making a connection to the super peer by choosing one of the super peer and clicking on the connect button.

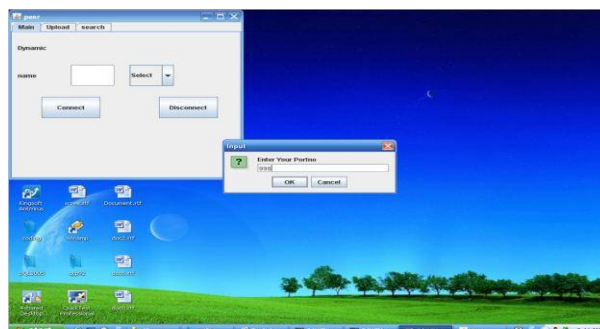


Fig. 4.1. Screenshot of starting a peer by entering the port number

# International Journal of Innovative Research in Science, Engineering and Technology

(An ISO 3297: 2007 Certified Organization)

Website: [www.ijirset.com](http://www.ijirset.com)

Vol. 6, Issue 1, January 2017

The figure 8 is after making connections peers upload the metadata about files which they want to share. The figure 9 is the probability table which is updated dynamically whenever queried source find the queried resource and whenever it does not find by updating the probability value. The figure 10 is when peer got the reply for its query from the super peer with ip and peer name. Peer makes connection with that peer and downloads the file.

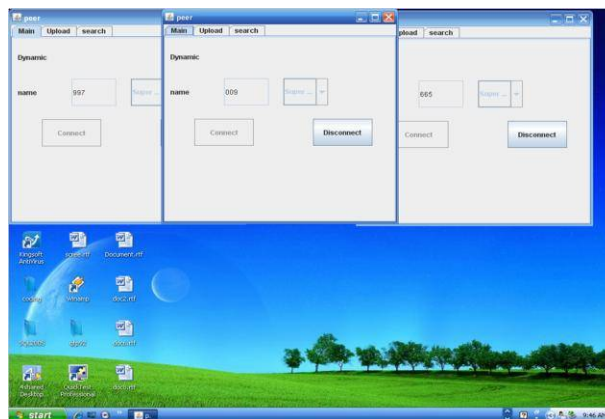


Fig. 4.2. Screen shot of peers connecting to superpeers

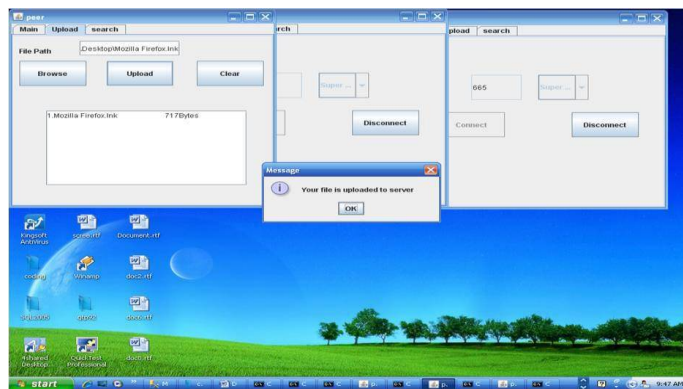


Fig. 4.3. Screenshot of uploading a file

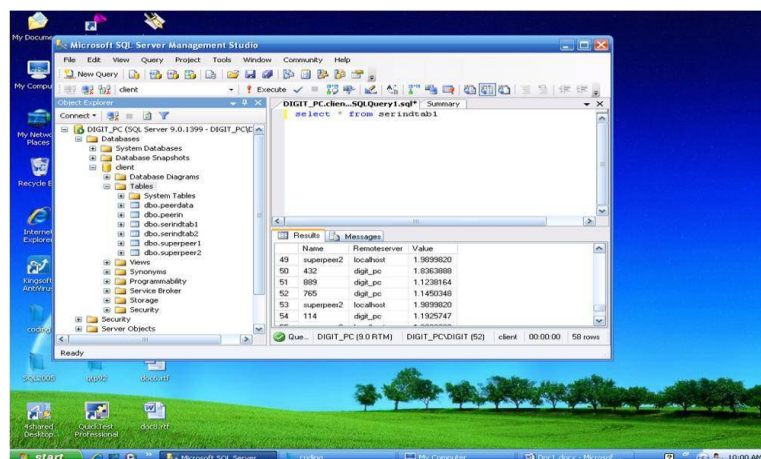


Fig. 4.4. Screenshot of the updated probability table

# International Journal of Innovative Research in Science, Engineering and Technology

(An ISO 3297: 2007 Certified Organization)

Website: [www.ijirset.com](http://www.ijirset.com)

Vol. 6, Issue 1, January 2017

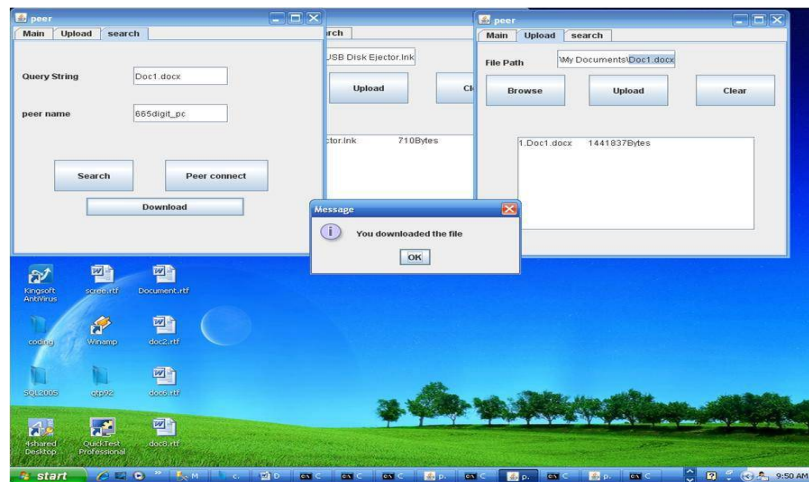


Fig. 4.5.The peer downloaded requested file

## V. CONCLUSION AND FUTURE SCOPE

The proposed dynamic search in unstructured peer to peer networks searches efficiently by using the knowledge based searching. Dynamic search improves the performance of searching by using the super peers which operates both as a server to a set of clients and an equal in the network of super peers. It reduces number of messages for searching by using the previous knowledge for locating the target node. Through the proper setting of the parameters dynamic search can obtain short search time and provide a good tradeoff between the search performance and cost. It can be further improved if clients or week peers also maintains the previous knowledge for making connection to super peer.

## REFERENCES

- [1] Anurag Singla and Christopher Rohrs, "Ultrapears: Another Step Towards Gnutella Scalability," Gnutella Developer's Forum, Nov.2002.
- [2] B.Yang and H.Garcia-Molina,"Improving search in peer-to-peer networks," ICDCS, July 2002, pp. 5–14.
- [3] Christos Gkantsidis, Milena Mihail, and Amin Saberi, "Random walks in peer-to-peer networks," in IEEE Infocom, Hong Kong, 2004
- [4] Yassin M. Y. Hasan and Lina J. Karam, "Morphological Text Extraction from Images",IEEE Transactions On Image Processing, vol. 9, No. 11, 2000
- [5] D. T Soumakos and N. Roussopoulos, "Adaptive probabilistic search for peer-to-peer networks," in Proceedings of the 3rd International Conference on Peer-to-Peer Computing (P2P'03),"September 2003, pp. 102–109."171–178.
- [6] J. Risson and T. Moors. Survey of research towards robust peer-to-peer networks: Search methods. Technical Report UNSW-EE-P2P-1-1, University of New South Wales, Sydney, Australia, September2004. <http://www.ee.unsw.edu.au/~timm/>.
- [7] Q. Lv, P. Cao, E. Cohen, K. Li, and S. Shenker, "Search and replication in unstructured peer-to-peer networks," in *Proc. ofICS'02*, June 2002, pp. 84–95.
- [8] S. Saroiu and S.D. Gribble P. Krishna Gummadi, "A measurement study of peer- to-peer file sharing systems," Proceedings of the Multimedia Computing and Networking (MMCN), January 2002.
- [9] V. Kalogeraki, D. Gunopulos, and D. Zeinalipour-Yazti, "A local search mechanism for peer-to-peer networks," in Proceedings of the 2002 ACM CIKM, November 2002, pp.300– 307.
- [10] Z. Xu and Y. Hu. Sbarc: A supernode based peer-to-peer file sharing system. In *EighthIEEE Interna-tional Symposium on Computers and Communications*, Kemer-Antalya,Turkey, jun-july 2003.
- [11] Z. Xu and Y. Hu. Sbarc: A supernode based peer-to-peer file sharing system. In *EighthIEEE Interna-tional Symposium on Computers and Communications*, Kemer-Antalya,Turkey, jun-july 2003.
- [12] P. Druschel and A. Rowstron, "Past: A large-scale, persistent peer-to-peer storage utility," May 2001.
- [13] J. Kubiatowicz, D. Bindel, P. Eaton, Y . Chen, D. Geels, R. Gummadi, S. Rhea, W. Weimer, C. Wells, H. Weatherspoon, and B. Zhao, "OceanStore: An architecture for global-scale persistent storage," in ASPLOS, Cambridge, MA, pp. 190–201, Nov. 2000.