Model of Complex Searching Over Structured P2P Overlay under Dynamic Environment

Saiful Khan and Abdullah Gani

***Abstract*—This paper aims to present a model to achieve complex searching (e.g. Wild card searching, blind searching, full text searching, metadata searching, etc.) over Structured P2P Overlay under dynamic environment. Lack of complex searching over Structured Overlay network and poor resilience due to peer dynamics makes it unsuitable for implementing the P2P systems and thus makes it very unpopular. In implementing the complex searching technique in structured overlay, the query must be processed by each node (peer) locally. This requires query to be efficiently broadcasted to all the nodes. However, broadcasting based algorithms tend to perform badly under highly dynamic environment and also create huge amount of maintenance traffic. This paper proposes a model which combines broadcasting based algorithm with efficient maintenance algorithm, which will reduce the maintenance traffic and increase the search efficiency.**

***Index Terms*—Structured overlay, distributed hash table (dht), churn, P2P and broadcast.**

I. Introduction

P2P system is a distributed network architecture, where interconnected nodes able to self-organize into network topologies with the purpose of resource sharing (e.g. processing power, storage capacity, network link capacity, printers). P2P system support wide area routing, efficient search of data, massive scalability, fault tolerance and redundant storage. P2P system does not depend on the central server for storage, lookup and to meditate between end systems. In P2P nodes behave as client as well as server. On the other hand high-end desktop computing is emerging and P2P system exploits the effective utilization of resources by increasing the utilization of resources at the edge of the internet. So the popularity of P2P system is increasing rapidly. However, the current P2P system is still partially centralized due to the deficiency of the Structured and Unstructured Overlay network. Structured Overlay implements decentralized system with lots of benefits, except its inability to perform complex searching and poor performance in dynamic environment. This paper proposes a conceptual model to achieve complex searching over Structured Overlay under dynamic environment.

II. Motivation

P2P is emerging as most powerful distributed system, which provide high degree of decentralization, self-organization, low cost deployment, organic growth, resilience to faults and attacks and abundance of diverse resources [1]. P2P system has been successfully deployed in sharing and distributing files, streaming media, telephony, application layer multicasting, volunteer computing, etc. Current P2P architecture is based on centralized and hybrid architecture, which is more or less dependent on central server, also the problem lies due to the huge amount of traffic induced that renders the solution un-scalable[2]. The problem of implementing a scalable P2P overlay with no central server has became a challenging problem, and the efforts to solve this problem emerged as Structured Overlay, also called as Distributed Hash Tables (DHTs). Structured P2P Overlay offers fully decentralized system. It provides efficient look-up operations via DHT with complexity of $O(\log N)$, where N is total number of nodes [3]. Structured Overlay use Consistent Hashing [4] to distribute resources among the nodes across the Overlay Network and achieve efficient load balancing. Structured Overlay only supports exact keyword based simple search, it is not possible to perform complex queries over Structured Overlay [2, 5]. The ease of manageability of P2P system under Dynamic environment is a open question [1]. Structured Overlay network has highly transient nodes which are not well supported by DHT [2] so the searching over Structured Overlay within dynamic environment (Churn) is not efficient [2]. These problems make Structured Overlay very unpopular.

III. Statement of Problem

A. Evolution of P2P System

Over the years many popular P2P application has been developed based on several architecture. The evolution of P2P architecture according to the degree of centralization can be divided in to three categories (a) Centralized (b) Hybrid and (c) Decentralized [3].

Centralized and Hybrid system do not offer scalability and resilience due to the dependency on server. Therefore, to address the problems of centralized and hybrid systems, decentralized systems have been introduced. The main challenge in decentralized system is searching or lookup of resources. Advanced searching mechanism allows users to effectively locate desired data in a resource efficient manner. Searching in decentralized system is classified in two broad categories- Unstructured Overlay and Structured Overlay [3].

Unstructured Overlay does not impose any well defined network structure and build a random graph, so this kind of system work without any prior knowledge of the topology

Manuscript received January 10, 2012; revised February 26, 2012.
Saiful Khan and Abdullah Gani are with the University of Malaya, Kuala Lumpur, Malaysia (e-mail: saifulkhan@siswa.um.edu.my, abdullah@um.edu.my).

and Overlay structure. Searching in Unstructured Overlay is based on flooding or random walk algorithms, which is effective on locating highly replicated items and resilient to peer dynamics. Flooding based algorithms are not efficient for locating unpopular files and not suitable for large scale system [4]. In response to the scaling problem in Unstructured Overlay, Structured Overlay is proposed. Structured Overlay organizes the resource and nodes in well defined graph structure. In Structured Overlay nodes collaborate among them self by virtue of well defined topology and protocol. Structured Overlay protocols such as Chord [6], Pastry [7], Kademlia [8], Tapestry [9] is implemented using DHT functionality. It provides efficient look-up operations in the order of $O(\log N)$, where N is the total number of nodes [4]. It maintains distributed state by using Distributed Hash Table (DHT) [2] depicted in Fig. 1. But Structured Overlay does not support Complex searching due to the limitation of DHT functionality [3].

B. Graph Theory Perspective of Overlay Network

The network behavior of P2P is described as an Overlay network because P2P forms a logical network or virtual network on top of existing physical network also called Underlay network, shown in Fig. 2. Overlay network is built in the application layer of the protocol stack. An Overlay network can be visualized as a directed graph $O = (N,\ L)$, where a node (vertex) $n \epsilon N$ and logical links (edge) between the nodes is two tuple relations $L \subset N \times N$.

C. Graph under Dynamic Environment

In dynamic environment nodes joins and leave the network randomly, this is called Churn. Presence of network dynamics dynamically changes the graph $O\ (N,L)$ to graph $O_i(N_i,L_i)$, $O_{i+1}(N_{i+1},L_{i+1})$, $O_{i+2}(N_{i+2},L_{i+2})$, ... over time. Joining of a new node n_{i+1} in the graph $O_i(N_i,L_i)$, changes the graph to $O_{i+1}(N_{i+1},L_{i+1})$. Where,

$$N_{i+1} = N_i \cup \{n_{i+1}\} \quad (1)$$

$$L_{i+1} = \forall_{x,y}\ L_i \cup \{(n_{i+1},x)\} \cup \{(y,n_{i+1})\} \quad (2)$$

The operation adds one node to the graph $O_i(N_i,L_i)$ and at least one incoming and one outgoing link between the new node and rest of the graph. When a node n_{j+1} leaves the network $O_j(N_j,L_j)$ the new graph $O_{j+1}(N_{j+1},L_{j+1})$ is formed, where,

$$N_{j+1} = N_j - \{n_{j+1}\} \quad (3)$$

$$L_{j+1} = \forall_{x,y}\ L_j - \{(n_{j+1},x)\} - \{(y,n_{j+1})\} \quad (4)$$

D. Routing table of Overlay Nodes

Routing table or Finger table of a node keeps the partial or full information of the rest of the Overlay Graph. Routing table of a node   of Overlay Graph $O\ (N,L)$ can be defined as $R_m(N_m,L_m)$. R_m consists of all the nodes adjacent to node m. where

$$N_m \subseteq N\ ,\ L_m \subseteq L \quad (5)$$

$$L_m \equiv \forall_{(m,n)} \in L \implies (m,n) \in L_m \quad (6)$$

$$N_m \equiv \forall_{(m,n)} \in L \implies (n \in N_m) \wedge (\sim\exists_n \in N_m) \wedge ((m,n) \in L_m) \quad (7)$$

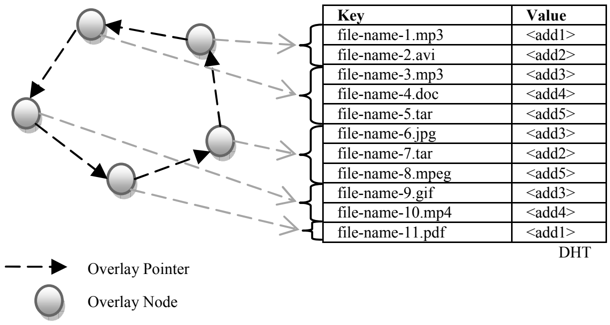

Fig. 1. Example of lookup("file-name-3.mp3") at any node should return address of the node belong it, which is here <add3>.

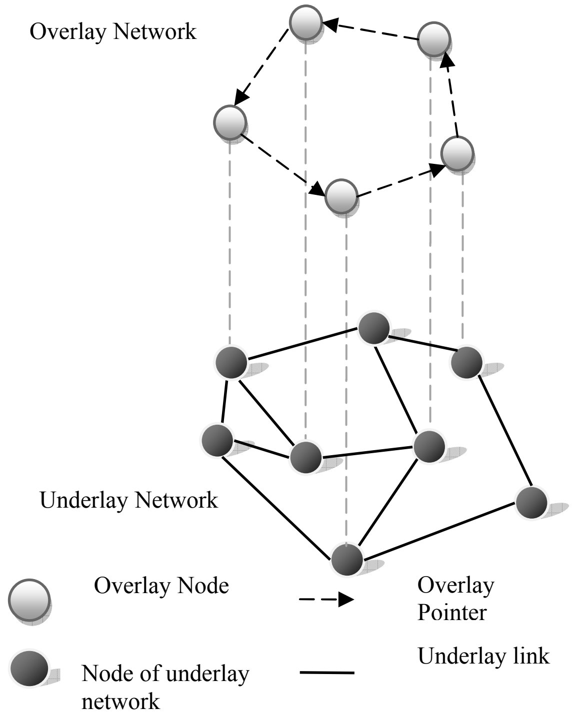

Fig. 2. Overlay is formed logically on top the Underlay network.

E. Distributed Hash Table (DHT)

DHT is a decentralized distributed system that provides a lookup service similar to a hash table; (key, value) pairs are stored in a DHT. DHT infrastructure distributes an ordinary hash table onto a set of cooperating nodes. In DHT each resource and node is assigned a unique identifier Id = H(key), generated by a hash function H. The key associated to the resource and node is uniquely chosen, which allows any node to find the resource associated with a given key. Each node in the system is responsible for storing certain range of keys. DHT just implements one operation *lookup (key)* that returns the identity (e.g., IP address) of the node storing the object with that key. DHT provides more efficient and controlled searching over Structured Overlay.

F. Main Drawbacks of DHT

Hash table search expect exact key, so DHT only support exact string or keyword based search. It does not support complex searching, because it completely relies on hashing technique. In dynamic environment nodes are transient that changes the structure of the Overlay, thus pollutes the routing table maintained by each node. This causes most of the lookup failure. Therefore, it requires stabilizing the overlay periodically.

G. Broadcast Based Algorithm for Complex Searching Over Structured Overlay: Related Work

Several methods have been proposed to broadcast query efficiently over Structured Overlay. Paper [10] describes a method to efficiently broadcast over a CAN network, and paper [11] describes a method to efficiently broadcast over a Pastry network, these broadcasting methods can be used to implement complex queries over Structured Overlay. Efficient Broadcast algorithm [12] proposed a technique to broadcast complex queries over Chord network efficiently. In DKS [13] efficient broadcast algorithm is extended and Self-correcting Broadcast [14], [15] is proposed to handle out-dated routing tables. Pseudo-reliable Broadcast [14] describes an algorithm to detect and handle node failures during broadcasting. Recursive Partitioning Search (RPS) [16]-[18] introduced TTL value in broadcasting also optimized other parameters to decrease traffic without affecting the success rate. Partition-based Broadcast Algorithm [19] modifies the Efficient Broadcast algorithm [12] to perform binary search, this results a slower but more balanced search. Dynamic Querying over DHT [20]-[22] technique uses an iteratively increasing TTL value and decrease the query traffic without affecting the success rate. Efficient Broadcast in P2P Grids [23] proposes an algorithm which is mixture of efficient broadcast and epidemic communication to achieve a better success rate under dynamic environment. In paper [24] Efficient Broadcast Algorithm under dynamic environment is proposed to detect and handle failed nodes.

H. Structured Overlay Maintenance Algorithm: Related work

Due to the presence of Churn in the network, the routing tables become inaccurate. This increases the system latency due to the query forwarded to unavailable nodes, leads to timeout. Increase in churn intensifies this problem. Since lookup query traffic and maintenance traffic both adds to the utilization of underlay bandwidth and resources, efficient maintenance algorithm is required to stabilize the overlay periodically. Techniques for Maintaining Structured Overlay networks [25] are described in this section.

Periodic Stabilization- is a maintenance algorithm which keeps running separate routines periodically for correcting the routing table of the nodes. Structured Overlay protocols Chord [26], CAN [27] and Pastry [28] use periodic stabilization. In Periodic stabilization each node periodically checks its neighbors and detect changes occurred in the vicinity of the checking node. Chord run stabilize and *fix_finger* methods. This technique can detect the change in Overlay structure faster, but at the cost of high bandwidth consumption.

Adaptive Stabilization- consumes high bandwidth even when the nodes are less transient and system is under low dynamism. To address the problem of periodic stabilization, adaptive stabilization algorithm is proposed, in which the stabilization frequency is tuned according to some observed parameters, suggested in [29]. Adaptive stabilization reduces unnecessary bandwidth consumption by estimating the size of the system and the failure rate. The main drawback is to identify the system parameters to be observed and the methods of observing these parameters. For large scale system under highly dynamic environment it is difficult to estimate these values.

Correction-on-use- method takes use the overlay network in order to self-organize in face of changes. A newly joined node in DKS [13] first receives an approximate routing information and makes it accurate over time. Correction-on-use uses the following mechanisms:

1. When a node receives a message from another node, the receiving node adapts itself to account the presence of the sender.

2. When a node X sends message to another node Y, node X embeds information about its current and accurate "local view" of the network. Then the receiving node Y precisely determines whether the sender X had a correct view at the sending time. If not, a bad-pointer notification is sent back to node X with the identifier of a candidate node for correction. Upon receipt of such a notification, the sender node X corrects itself. If the ratio of the traffic injected into the system is high due to dynamism, the overlay network converges to a legitimate configuration. The main advantage of this algorithm is that it completely eliminates unnecessary bandwidth consumption. Each node pays for what it usage. However, if the ratio of the traffic injected into the system due to dynamism is not high, the convergence of the overlay network to legitimate configuration is slowed down.

Correction-on-change- technique, whenever a change is detected, all nodes that depend on the node where the change occurred are corrected [30].

IV. Proposed Model

A. Algorithm for Broadcast over Structured Overlay

The main strategy to perform complex searching over structured overlay is to get the query processed by each node locally. The peer-to-peer application running at each node has implemented complex searching algorithms. Upon receiving a lookup query the node locally search matches with its shared contents and returns back the result to the sender node. Therefore, the query messages need to be sent to all the nodes of the network which is achieved by broadcasting the lookup messages to all the nodes of the network. The model proposed in this chapter is to efficiently broadcast the query over D1HT structured overlay. Fig.4 presents a flowchart illustrating a high label strategy to broadcast over structured overlay. For every message sent, an Init thread is created which sends the message/query to a particular node and waits for the acknowledgement. Therefore, in a single broadcast multiple *Init* threads are created. If a thread does not receive any acknowledgement within timeout, it assumes the node has failed, therefore generates an event. Each node creates a listener thread which receives messages and acknowledges the receipt of a message to the sender. The listener thread also checks the received message; if the message is maintenance event then it updates the routing table and forwards the event further. If the message is complex query, it evaluates the query against its local content by running complex searching algorithms. If the content is found, sends the result back to the sender, else it broadcasts the message further to all its nodes from the routing table. The broadcast algorithm uses the structure of the overlay to broadcast the message efficiently.

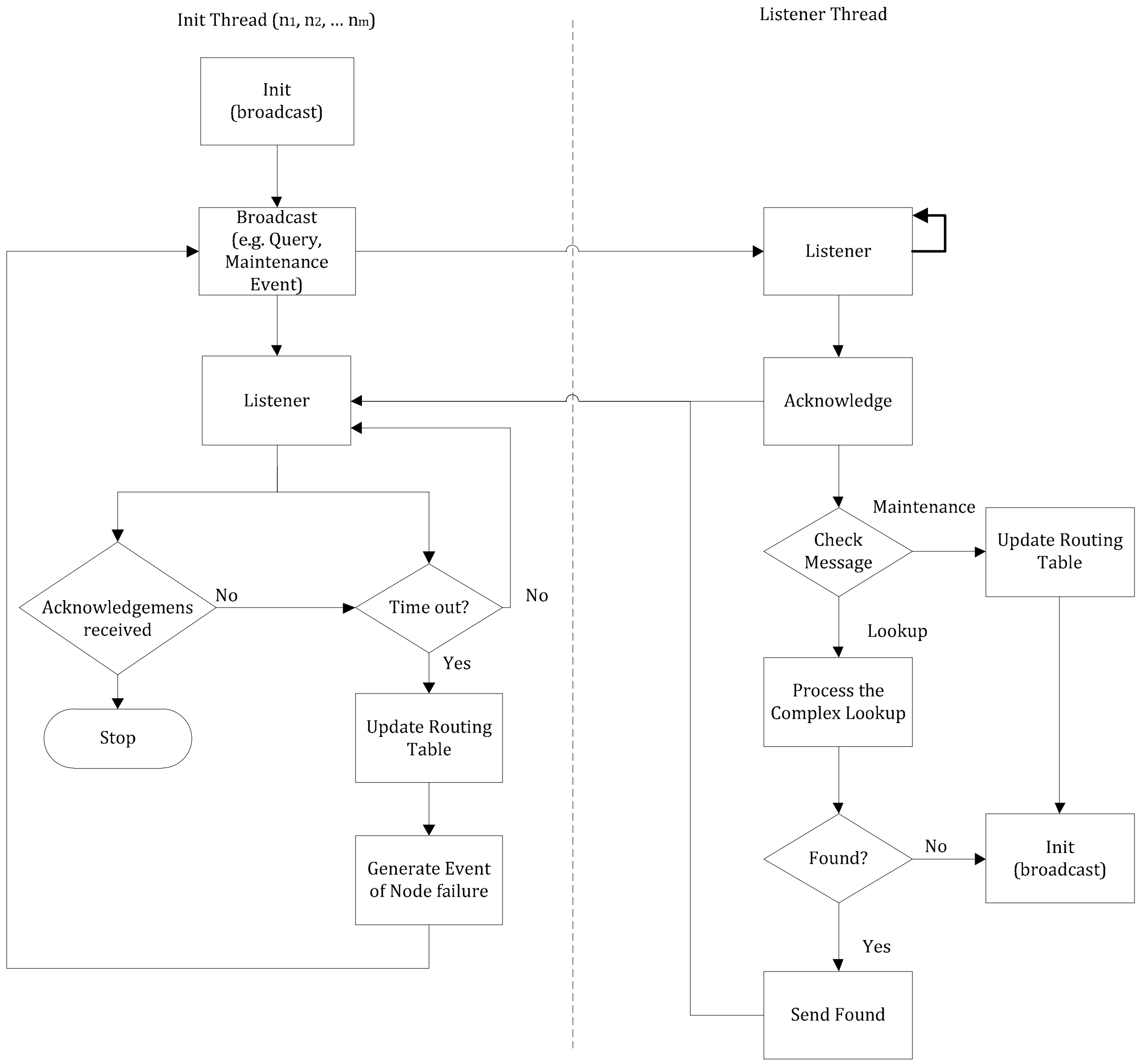

Fig. 3. Proposed model (flowchart) of Broadcast based scheme

B. Maintenance Algorithm

The broadcast based algorithms tend to perform badly under high churn rate; it requires employing some efficient maintenance algorithm. Structured Overlay D1HT [31] employ efficient maintenance algorithms called Event Detection and Routing Algorithm (EDRA) to inform nodes of a node joining or leaving the network. D1HT assure that large number of lookups are performed with one hop with 99% success rate, low bandwidth overhead, good balance of the maintenance traffic between the nodes and adapt in the system dynamics. In this model one such maintenance algorithm is proposed to use, to reduce the maintenance traffic and increase the stability of the Overlay Structure.

V. Conclusion

This paper proposes a model for complex searching technique over Structured Overlay under dynamic environment. This is achieved when all the queries are processed by each node locally, and the queries need to be efficiently broadcasted to all the nodes. On the other hand due to the presence of Churn in the network pollutes the routing table, thus increases the latency due to the query forwarded to unavailable nodes and leads to timeout. Increase in churn intensifies this problem. Since lookup query traffic and maintenance traffic both adds to the utilization of underlay bandwidth and resources. Efficient maintenance algorithm is required to stabilize the overlay periodically, so an efficient maintenance algorithm is proposed in this paper.

References

[1] P. D. Rodrigo Rodrigues, "Peer-to-Peer Systems," *Communications of the ACM* vol. 53 p. 11, 2010.

[2] J. Risson and T. Moors, "Survey of research towards robust peer-to-peer networks: Search methods," *Computer Networks,* vol. 50, pp. 3485-3521, 2006.

[3] E. Lua, *et al*, "A Survey and Comparison of Peer-To-Peer Overlay Network Schemes," 2005.

[4] D. Karger, *et al*, "Consistent hashing and random trees: Distributed caching protocols for relieving hot spots on the World Wide Web," 1997, pp. 654-663.

[5] M. Castro, *et al*, "Should we build Gnutella on a structured overlay," *ACM SIGCOMM Computer Communication Review,* vol. 34, p. 136, 2004.

[6] I. Stoica, *et al*, "Chord: A scalable peer-to-peer lookup service for Internet applications," *Computer Communication Review,* vol. 31, pp. 149-160, Oct 2001.
[7] P. Druschel and A. Rowstron, "Pastry: Scalable, decentralized object location, and routing for large-scale peer-to-peer systems," *Lecture* Notes *in Computer Science,* vol. 2218, 2001.
[8] P. Maymounkov and D. Mazieres, "Kademlia: A peer-to-peer information system based on the XOR metric," *Peer-to-Peer Systems,* vol. 2429, pp. 53-65, 2002.
[9] B. Zhao, *et al*, "Tapestry: A resilient global-scale overlay for service deployment," *IEEE Journal on selected areas in* communications, vol. 22, pp. 41-53, 2004.
[10] S. Ratnasamy, *et al*, "Application-level multicast using content-addressable networks," *Networked Group Communication,* pp. 14-29, 2001.
[11] M. Castro, *et al*, "An evaluation of scalable application-level multicast built using peer-to-peer overlays," 2003, pp. 1510-1520.
[12] S. El-Ansary, *et al*, "Efficient broadcast in structured P2P networks," *Peer-to-Peer Systems II,* pp. 304-314, 2003.
[13] L. Alima, *et al*, "DKS (N, k, f): a family of low communication, scalable and fault-tolerant infrastructures for P2P applications," 2003.
[14] A. Ghodsi, "Distributed k-ary system: Algorithms for distributed hash tables," *KTH Stockholm, Diss,* 2006.
[15] A. Ghodsi, *et al*, "Self-correcting broadcast in distributed hash tables," 2003, pp. 93-98.
[16] V. Vishnevsky, *et al*, "Scalable Blind Search and Broadcasting in Peer-to-Peer Networks," 2006.
[17] V. Vishnevsky, *et al*, "Scalable blind search and broadcasting over Distributed Hash Tables," *Computer Communications,* vol. 31, pp. 292-303, 2008.
[18] V. Vishnevsky, *et al*, "Tag routing for efficient blind search in peer-to-peer networks," 2006, pp. 409-416.
[19] K. Huang and D. Zhang, "A partition-based broadcast algorithm over DHT for large-scale computing infrastructures," *Advances in Grid and Pervasive Computing,* pp. 422-433, 2009.
[20] D. Talia and P. Trunfio, "Dynamic Querying in Structured Peer-to-Peer Networks," *Managing Large-Scale Service* Deployment, pp. 28-41, 2008.
[21] A. A. Fisk. (2003, 10-08-2010). Gnutella dynamic query protocol v0.1[Online]. Available: http://www.ic.unicamp.br/~celio/peer2peer/gnutella-related/gnutella-dynamic-protocol.htm
[22] P. Trunfio, *et al*, "Implementing dynamic querying search in k-ary DHT-based overlays," 2008, pp. 275-286.
[23] P. Merz and K. Gorunova, "Efficient broadcast in P2P grids," 2005, pp. 237-242.
[24] W. Li, *et al*, "An Efficient Broadcast Algorithm in Distributed Hash Table Under Churn," 2007, pp. 1929-1932.
[25] L. Alima, *et al*, "A framework for structured peer-to-peer overlay networks," 2005, pp. 223-249.
[26] I. Stoica, *et al*, "Chord: A scalable peer-to-peer lookup service for internet applications," 2001, p. 160.
[27] S. Ratnasamy, *et al*, "A scalable content-addressable network," 2001, p. 172.
[28] A. Rowstron and P. Druschel, "Pastry: Scalable, decentralized object location, and routing for large-scale peer-to-peer systems," 2001, p. 329.
[29] R. Mahajan, *et al*, "Controlling the cost of reliability in peer-to-peer overlays," *Peer-to-Peer Systems II,* pp. 21-32, 2003.
[30] A. Ghodsi, *et al.*, "Increasing Robustness while Minimizing Bandwidth Consumption in Structured Overlay Networks," Tech. Report ISRN KTH/IMIT/LECS.
[31] L. Monnerat and C. Amorim, "D1HT: a distributed one hop hash table," 2006, p. 10.

Saiful Khan received the B.Tech in Electronics and Telecommunication Engineering from National Institute of Technology, Silchar, India in 2005 and MS (by research) degree from Department of Computer System and Technology, University of Malaya, Kuala Lumpur in 2011. He was a Development Engineer at ABB from 2007 to 2009, Member Technical Staff at Oracle from 2007 to 2009 and Senior System Software Engineer at HP. His research interest includes Distributed Computing, P2P Network, Visualization, Decision making techniques, and Parallel Computing.

Abdullah Gani is an Associate Professor of Faculty of Computer Science & Information Technology, University of Malaya, Malaysia. He received his PhD from University of Sheffield, UK and University of Hull, UK for his B.Phil, and M.Sc. His research interest includes Switched Networks (QoS, Traffic Engineering), Soft Computing (Reinforcement learning, Intelligent Agent), Routers Technology (Resource Control), Middleware Technology (Intelligent Middleware), Mobile Computing (Resource control), Self-organized Systems. He has published 85 journal and proceeding papers locally and internationally.