http://ansinet.com/itj

ITJ

ISSN 1812-5638

INFORMATION TECHNOLOGY JOURNAL

ANSInet
Asian Network for Scientific Information
308 Lasani Town, Sargodha Road, Faisalabad - Pakistan

Information Technology Journal 8 (8): 1197-1204, 2009
ISSN 1812-5638

Distributed Index based on Geographic Hashing Table for Mobile Ad Hoc Networks

Yongsheng Fu, Xinyu Wang and Shanping Li
Department of Computer Science and Information Technology,
Zhejiang University, No. 38, Zheda Road, Hangzhou, Zhejiang, 310027, China

Abstract: Distributed Hash Table (DHT) has proven to be an efficient platform for building a variety of scalable and robust distributed applications like content sharing and location in the internet. However, the adaptation of DHT technology to Mobile Ad-hoc NETwork (MANET) is not straightforward. Network scalability and routing as well as information distribution are major problems for nodes in a MANET, who are only aware of their immediate neighborhood. Several algorithms implement DHT using geographic information in MANET, but they can not adapt well in large-scale network without an efficient localization mechanism. This study propose a new DHT implementation named Distributed Index based on Geographic Hash Table (DI-GHT) in MANET. In DI-GHT, using hashing function, the shared resource location information (index) is mapped to nodes in a geographic area rather than a geographic position. The network is partitioned into domains and DI-GHT distributes resource index in all domains. The requestor finds the index information in the nearest domains using the hash function and then retrieves the resource. The simulation results and analysis show that DI-GHT outperforms original Geographic Hash Table (GHT) in terms of query success rate and message cost.

Key words: Distributed hashing table, index perimeter, localization mechanism, multiple domains

INTRODUCTION

No existing infrastructure or central administration can be used to organize the wireless nodes within a Mobile Ad hoc NETwork (MANET). The nodes communicate in multi-hop peer-to-peer mode. Resource discovery is the technology of enabling a node to find resources matching its needs and has become an integral part of any modern network systems. In general, a resource can be any type of service or capability, such as nodes with high energy, processing power or storage, multiple interfaces, printing capability, or sensing capability.

To search for a required resource in a MANET, a node sends out a request packet which will be forwarded by others. When receiving request packets, every node with matched resources responds with a reply packet, which will be forwarded reversely to the source of the corresponding request packet. There are two types of algorithms to implement resource discovery in MANETs (Wu, 2006): unstructured and structured methods.

The unstructured methods are usually based on some kind of flooding mechanisms. Gnutella (Ripeanu and Foster, 2002) is a classical case for pure peer-to-peer searching. However, pure flooding consumes network resources quickly and cannot meet user requirements. Local index (Yang and Garcia-Molina, 2002), dominating set (Yang and Li, 2005) and clustering (Oliveira *et al*., 2008) based techniques were proposed to reduce flooding scope and locate the target resources more efficiently. They still perform not well in environments with high mobility.

On the other hand, structured methods depend on structured networks (Abdullah *et al*., 2009). Geographical routing protocols (Seada and Helmy, 2005) provide the chance for structuring MANETs. Distributed Hash Tables (DHTs) (Frodigh *et al*., 2000), such as Chord (Stoica *et al*., 2003), have proven to be an efficient platform for building a variety of scalable and robust distributed applications like content sharing and location in the Internet. We argue that the DHT extension to MANETs could similarly provide an efficient way of constructing distributed applications and services, in low mobility. For example, service sharing can benefit from the distributed insert/lookup convergence provided by DHTs.

However, the adaptation of DHT technology to MANETs faces many challenges (Pucha *et al*., 2004a), such as no center servers provide the Domain Name Service (DNS) and the mobility. Furthermore, in many cases wireless networks may scale up to thousands of nodes rendering the discovery problem even more challenging. In (Gao *et al*., 2006), comparative analysis and simulation studies were performed on several existing service discovery protocols for MANETs. But these protocols need additional cost for nodes grouping and very weak in large-scale network.

Corresponding Author: Yongsheng Fu, Department of Computer Science and Information Technology, Zhejiang University, No. 38, Zheda Road, Hangzhou, Zhejiang, 310027, China Tel: +86-13575483578 Fax: +86-0571 28111775

As geographical routing performs better in large-scale networks (Das *et al*., 2005), it is a nature way to use geographical information for resource discovery. Many similar works in scalable resource discovery in MANET for geographical routing were published. The Grid Location Servise (GLS) (Li *et al*., 2000) is a scalable location service that performs the mapping of a node identifier to its location. GridlLocation servise is designed to be combined with geographic forwarding to implement unicast. The implementation of GLS effectively provides a DHT interface; it routes a message with a nodeId Y to a node whose nodeId is closest to Y.

GHT (Geographic Hash Table) designed to work in sensor networks for datacentric storage (Ratnasamy *et al*., 2002), but the challenge of applying GHT directly to MANET is that GHT is designed and evaluated in sensor networks, a static and dense environment. Liu and Ruonan (2005) modified the GHT algorithm to make it suitable for MANET. Recently, Meshkova *et al*. (2008) did a massive of work on analysis of resource discovery frameworks and found that fairly few solutions targeted the local scale (localization priority), especially in the resource constrained and wireless large scale networks. In GHT, all data with the same general name (e.g., elephant sightings) will be stored at the same node (not necessarily the one that originally gathered the data). Pucha *et al*. (2004b) gives tow ways to construct DHT in MANET above DSR. However, these solutions can not adapt well in large-scale MANETs without localization mechanism.

In this study, we present a new DHT implementation named Distributed Index on GHT (DI-GHT), which approaches resource discovery in MANETs. There are two roles in our implementation: owner node who shares resources and the requestor who requires the shared resources. The shared resources are replicated in the neighborhood of owner node. The shared resource location information (index) is mapped to nodes in a geographic area rather, thereby supporting a hash-table-like interface. Resource discovery is carried out first by finding the resource location using hashing functions and then retrieving the resource. The network is partitioned into domains and DI-GHT distributes resource index in all domains. Therefore, DI-GHT provides a unified solution to the problems of the discovery of resources and is targeted for large-scale networks with local priority. Distributing resource location information instead of resources it can save network bandwidth and energy. Both theoretical analysis and simulation results show that DI-GHT outperforms GHT in large-scale MANETs.

MOTIVATION

There are two main obstacles for DHT to work in MANETs, an environment with higher mobility:

- Different from Sensor networks, the most critical problem is the movement of nodes not the failure of nodes in MANETs. The method, which is used by GHT (Ratnasamy *et al*., 2002) to ensure the availability of data stored, may not works well, as the refresh protocol generates too many messages periodically even no node failure occurs
- In Sensor Networks, DHT assumes an external data gatherer or a base station exits and as a result, the cost for querying is approximately constant. However, in DHT for MANET, the data consumers are also in the same network boundary as the nodes that puts the data. When two nodes are close to each other, they also need to travel a long distance to exchange data. Such case may consume a lot of network bandwidths and node energy. Space localization can reduce the message and bandwidth cost

In this study, DI-GHT is proposed to handle these two obstacles. To handle the first obstacle, DI-GHT maps the index of shared resources to a geographic area instead of a geographic position according to the key of the index and then uses GPSR to disseminate the index to the nodes in the area. The refresh messages are generated when the node moves beyond the area, which reduces the unnecessary communications.

For the second obstacle, MANET is partitioned into multiple domains in DI-GHT. The shared resource index is mapped to all domains by the key value using hashing function and available for the requestor in its domain. By partition, DI-GHT enhances the network scalability in GHT.

DISTRIBUTED INDEX ON GEOGRAPHIC HASHING TABLE

DI-GHT is built on Greedy Perimeter Stateless Routing (GPSR) (Karp and Kung, 2000) system for multi-hop wireless networks. Greedy perimeter stateless routing provides two interfaces similar with DHT:

- Put (k, v) stores v according to the key k, the name of the data
- Get (k) retrieves the value stored associated with key k

Index perimeter: Different from GHT (Ratnasamy *et al*., 2002), the core step in DI-GHT is the hashing of a key k into geographic square area, instead of a coordinate. A key-value pair is stored at nodes in the square area to which its key hashes. Choosing nodes consistently is central to building a DI-GHT system, which means that the resource sharing and querying request for the same k to be routed to the same nodes in a static network.

Under DI-GHT, Put()or Get() packet does not know the identifier of the node that is the eventual destination of the packet. We assume the hashing function H (k) defines the mapping from a key k to geographic square area S_k:

$$\begin{cases} H(k) = (x_k, y_k) \\ S_k = \{(x, y) \bullet | x - x_k | \le r, | y - y_k | \le r\} \end{cases} \quad (1)$$

where, (x_k, y_k) is the center coordinate and r is the radio transmission range in MANET. An example is shown in Fig. 1, where x is the mapping coordinator and f is selected to be the home node.

The hash function is ignorant of the placement of individual nodes in the topology; it merely spreads the different key names evenly across the geographic region where the network is deployed. Thus, the possibility exists sometimes that there is no nodes at the square area hash function H (k) produces, as shown in Fig. 2.

We define the home node for a DI-GHT packet to be the node geographically nearest the center coordinate (x_k, y_k) of the packet and the index perimeter for the geographic square area S_k named k index perimeter. The shared resource index will be distributed on nodes in the index perimeter or home node if no nodes in the index perimeter.

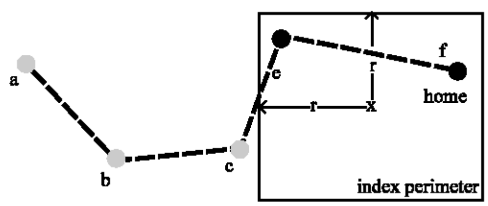

Fig. 1: DI-GHT axample: Center coordinate x, home node f, node e in index perimeter

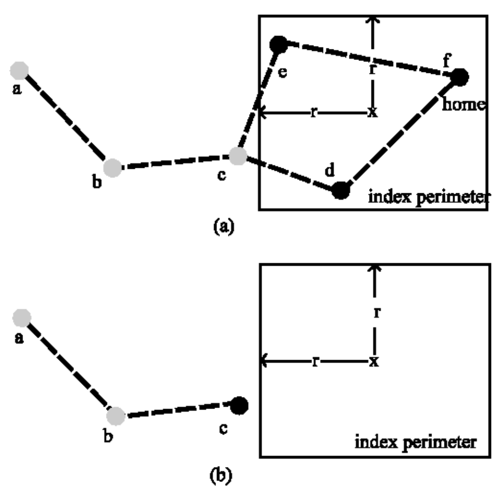

Fig. 2: (a) Node d joins the index perimeter; (b) the nearest node c is the home node when no nodes in index perimeter

Because a DI-GHT packet is not addressed to a specific node, but rather only to a specific location, it is treated by GPSR as a packet bound for a disconnected destination: no receiver ever sees the packet addressed to its own identifier. GPSR will route such a packet to the appropriate home node and index perimeter in the perimeter mode. Under DI-GHT, the packet enters perimeter mode at the home node, as no neighbor of the home node can be closer to the center coordinate. We configure the GPSR algorithm to make the packet traverse the index perimeter that encloses the center coordinate, before returning to the home node (Karp and Kung, 2000). Meanwhile, the home node knows to consume the packet when it returns after this tour of the index perimeter.

With only the home node binding mechanism mentioned above, the mapping nodes consistently by hash function is guaranteed in a static network. On the other hand, two actions works for the topology changes in MANET:

- Nodes join in the index perimeter

Assume index u is stored on node u. If u detects that a new node p stopped in the α index perimeter, u will transfer a copy of index α to p . On the other hand, if u is the home node which does not locate in α index perimeter, u transfers a copy of index α to p , marks p to be the new home node. As shown in Fig. 2a, node d joins the index perimeter, the home node f transfers the index data stored in this index perimeter to d.

- Last node leaves the index perimeter

Assume node u is the only one in the α index perimeter. If u is moving out, u is responsible to find a nearest home node u′ for α , mark u′ to be the new home node and transfers a copy of index α to u′. Figure 2b shwos the case. When node e , f leave the index perimeter, there are no nodes left, so the node c will be the new home node.

In fact, for any snapshot of the network topology, there exists a home node and index perimeter for every location in the network. However, no action is necessary if there is no index stored on the nodes in an index perimeter.

Localization: multiple domains: Hierarchy strategies are good candidates to achieve the DHT localization. One simple way is like Domain Name System (DNS), which is a hierarchical naming system for computers, services, or any resource participating in the Internet. Different sub-nets comprise the internet in a geographical perspective and one sub-net connects to another one by the address parsing using DNS servers.

But the infrastructure for DNS functionalities is not available in MANETs. In DI-GHT, by geographical dividing, a MANET is partitioned into sub-MANETs which are named domains. Nodes in one domain connect to another domain by geographical location instead of DNS.

- Horizontal partition

The horizontal line is divided into l parts and the set C_h of center coordinates for each domain is:

$$C_h = \left\{((2x+1)\times\frac{m}{l}, n) \mid x \in Z,\ 0 \le x < l\right\} \tag{2}$$

The hashing function H (k) for resource index k in Eq. 1 is changed to map the index into all domains. For the domain a where $0 \le a \le 1$:

$$\begin{cases} x_k(a) = (2a+1)\times\dfrac{x_k}{l} \\ H(k,a) = (x_k(a), y_k) \end{cases} \tag{3}$$

- Vertical partition

The vertical line is divided into l parts and the set C_v of center coordinates for each domain is:

$$C_v = \left\{(m, (2y+1)\times\frac{n}{l}) \mid y \in Z,\ 0 \le y < l\right\} \tag{4}$$

For the domain a where $0 \le a \le 1$, like horizontal partition, the hashing value is changed:

$$\begin{cases} y_k(a) = (2a+1)\times\dfrac{y_k}{l} \\ H(k,a) = (x_k, y_k(a)) \end{cases} \tag{5}$$

Any other partition can be treated as a combination of horizontal and vertical partitions. In a horizontally partitioned MANET, the resource discovery process for index k is as follows (the requestor node u (x, y)):
Calculate the domain of node u:

$$d = [(y \times \frac{1}{2n}) - l] \tag{6}$$

Calculate the hashing value H (k, d) = ($x_.$, y_k (d)) in the domain d according to Eq. 5.

Retrieve the value stored in index perimeter in domain. If found, return the value and success. Otherwise, search other domains. If finally non-found, return failure. Figure 3 shows a network partitioned horizontally into 2 domains. The resource index is stored on two index

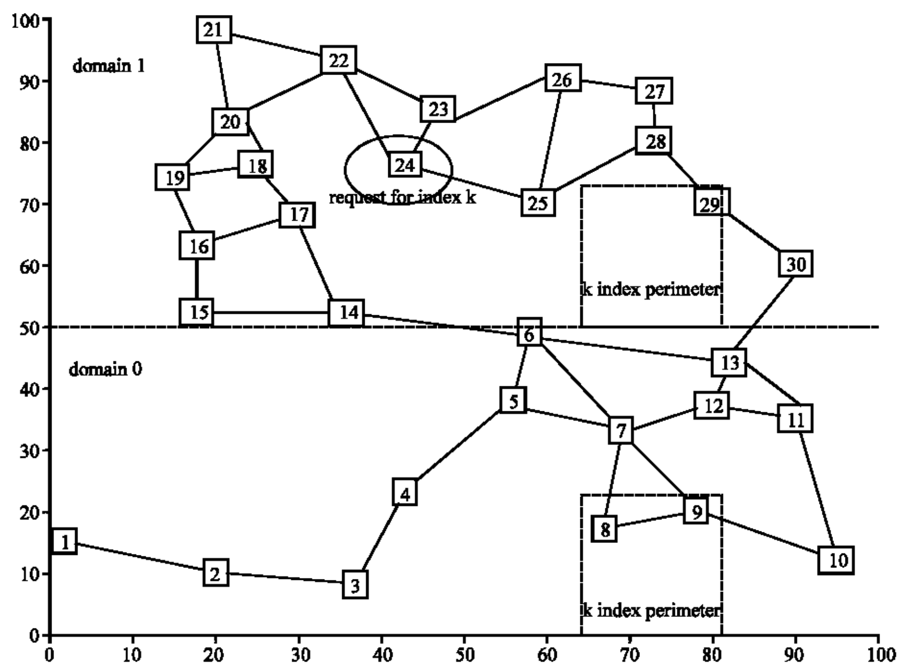

Fig. 3: An example for multiple domains: v is 100×100, l = 2 and the k index perimeter for key k in two domains

perimeters in domain 0 and domain 1. The requestor node 24 first calculates its domain 1 and retrieves the value at node 29 in domain 1, though the value for is stored on node 8 and 9 in domain 0 too.

SIMULATION AND ANALYSIS

Since, the simulation need reflect the situations in the real world, the simulation model must be carefully selected. In this study, we used Random Way Point (RWP) model (Camp *et al*., 2002), an entity mobility model for a more mobile environment in MANETs.

In RWP model, each node begins by staying in one location for a certain period of time, chooses a random destination in the simulation area and a speed that is uniformly distributed between [minspeed, maxspeed] and then travels toward the newly chosen destination at the selected speed. Upon arrival, the node pauses for a specified time period before starting the process again (Liu and Ruonan, 2005).

We implemented DI-GHT in ns-2 (McCanne and Floyd, 2008), which supports detailed simulation of mobile wireless networks using IEEE 802.11 radios. The performance comparison between DI-GHT and GHT (Liu and Ruonan, 2005) is simulated under various node density and mobility conditions (different velocity). In this section, the simulation was conducted in both small-scale (100 nodes) networks and large-scale (>1000 nodes) networks.

Small networks: The simulation for small-scale networks was based on ns-2 tool. The values of parameters of DI-GHT, GHT and GPSR we used in our experiments are given in Table 1.

The metrics we used to evaluate DI-GHT and GHT are:

- The number of index distribution message (No. of Put) and the number of search the resource index (No. of Get). These two metrics reflect the traffic pattern
- The number of index refresh message, generated in the refresh interval (No. of Refresh). This metrics reflects the cost of data dissemination and consistence maintain
- The success rate of data search (Success rate). This metrics is the most important one that reflects the overall performance of DI-GHT and GHT in MANET

As shown in Table 2, the performances in mobile situations is low, especially Success Rate. The higher the speed of nodes move, the more frequently the topology changes and the higher possibility the fraction of the GHT home perimeter. So, the maintenance message increases and the number of refresh messages raises. As mentioned in (Liu and Ruonan, 2005), the reason for success rate decreasing is due to tow factor: network partition and long home perimeter. However, in DI-GHT, the index perimeter is fixed (Transmission Range 100 m) and two domains partition the network. The larger network has high possibility of partition, so the success rate can be kept in a high level in DI-GHT.

Table 1: Simulation parameters in ns-2

Parameters	Value
GPSR beacon interval	1 sec
GPSR beacon expiration	4.5 sec
DI-GHT domains	2
DI-GHT index perimeter range	100 m
Planarization	GG
Number of nodes	100
Simulation time	300 sec
GHT refresh interval	10 sec
Simulation area	1000*1000
Number of items in data set	20

Table 2: Performance with different speed

Algorithm	Speed (m sec^{-1})	Success rate (%)	No. of put	No. of get	No. of refresh
GHT	0.5	98.1	500	500	1.28
	1.0	90.5	490	492	12.39
	5.0	81.2	476	479	19.03
DI-GHT	0.5	100.0	500	500	1.02
	1.0	97.2	482	482	5.22
	5.0	91.5	477	489	13.51

Large networks: The ns-2 tool is used only for small-scale networks, as it limits to system sizes on the order of no more than 1000 nodes. For the large-scale network simulation, using Java SDK, we built a special-purpose simulator that assumes that nodes are stable and stationary and that packet delivery to neighboring nodes is instantaneous and error-free.

Our simulation is in static networks, the number of resource query is fixed at 100 and two domains partition the network in DI-GHT. We use two metrics to evaluate the performance: the query success rate and the total number of packets generated.

As shown in Fig. 4, the success rate is decreasing with the network expanding for both GHT and DI-GHT. The main reason is that the searching length is too long, the hashing value is too distributed in the whole network and more failures happened due to the TTL (time to live) of query. On the other hand, the possibility of no nodes in home perimeter increases in large networks. GHT has a lower success rate than DI-GHT in large networks, because the requestor can get correct response in its domain more quickly and the other domain backs up the query processing.

Figure 5 shows the linear trend of the generated messages increase in large networks. Fewer messages are

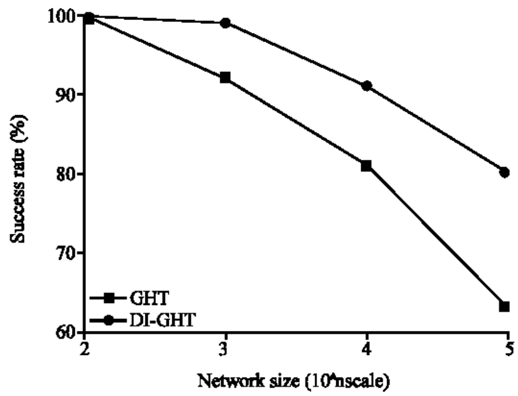

Fig. 4: Query success rate in different network scale

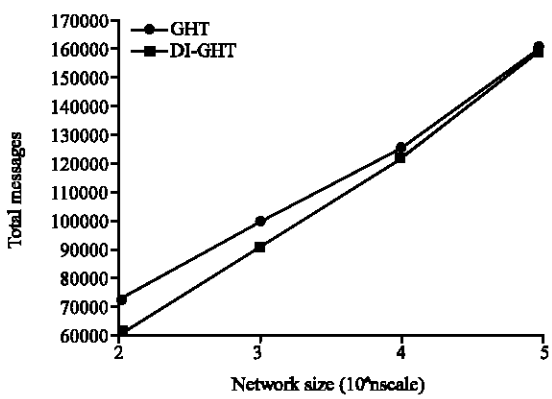

Fig. 5: Total number of messages generated in different network scale

generated in DI-GHT for two domains communications. However, the difference is reduced when the network scale is 106. The possible reason is that the partition produces two large domains and the ratio of the messages generated for 100 querys (resource index put or get) is too small comparing with the total maintenance messages. If many domains are produces by the partition, the messages cost may be increased, which need more evaluations.

DISCUSSION

Because of mobility and decentralized management, resource sharing is vulnerable in MANETs. Traditional algorithms tried to control the scope of flooding by unstructured methods. However, they perform not well in large networks and flood the whole network in worst cases.

Geographic routing protocols provide a chance to structure MANETs for DHT implementation. In this study, our proposed DI-GHT mainly focuses on two problems for adapting DHT technique to MANETs: node mobility and network localization. By creating index perimeter, the indexes of shared resources are mapped to a geographic area instead of a geographic position according to the key of the index. The communication cost is reduced since the refresh messages are generated when the node moves beyond the area, especially in large networks, as shown in Fig. 5 and Table 2. Meanwhile, domain partitioning builds the localization mechanism in MANETs. DI-GHT distributes resource index in all domains. The requestor finds the index information in the nearest domains using the hash function and then retrieves the resources, which shortens the response time and improve the success rate, as shown in Fig. 4.

Although, achieving good performance in the simulation, our solution still needs further evaluation on other scenarios:

- In mobile scenarios, DI-GHT achieve high success rate than GHT, because the index perimeter is fixed where the stored index is not affected by the node movement. However, the index perimeter is limited by the transmission range in our simulation. Increasing the index perimeter in DI-GHT might improve the performance in mobile environment
- Another aspect is the localization by partitioning the network into domains. We observed the high success rate in large network for two domains. However, more domains might increase the maintenance cost. To handle this problem, one solution is to increase the index perimeter parameters to reduce the refresh be generated when a node moves beyond the index perimeter. Another solution is using mirroring method: like the structured replication in (Liu and Ruonan, 2005), the key stored in home node can be divided among multiple mirrors. More simulations are required to evaluate the impact of increasing domains

CONCLUSION

This study presented the design and evaluation of a DHT implementation for MANETs built on geographic routing, named DI-GHT. The shared resource location information (index) is mapped to nodes in a geographic area. The network is partitioned into domains and DI-GHT distributes resource index in all domains. The requestor finds the index information in the nearest domains using the hash function and then retrieves the resource. The results show that DI-GHT performs better than GHT in both mobile and large networks. The simulation proves that the index perimeter concept and network partitioning into multiple domains are effective and efficient.

The initial goal of our research is to adapt DHT design into MANETs. Since the network topology is decided by nodes geographic positions in geographical routing protocols, geography based approach is advanced than other approaches which depends on network local topology. However, the data update and consistent are not covered in this study. How accurately DI-GHT can provide the capacity of data maintenance is worth for future evaluation.

ACKNOWLEDGMENT

This research was financial supported by National Natural Science Foundation of China with No. 60873022 to Dr. Shanping Li.

REFERENCES

Abdullah, T., L.O. Alima, V. Sokolov, D. Calomme and K. Bertels, 2009. Hybrid resource discovery mechanism in Ad Hoc grid using structured overlay. Proceedings of the 22nd international Conference on Architecture of Computing Systems, Mar. 10-13, Delft, The Netherlands, pp: 108-119.

Camp, T., J. Boleng and V. Davies, 2002. A survey of mobility models for ad hoc network research. Wireless Commun. Mobile Comput., 2: 483-502.

Das, S.M., H. Pucha and Y.C. Hu, 2005. Performance comparison of scalable location services for geographic ad hoc routing. Proceedings of the 24th Annual Joint Conference of the IEEE Computer and Communications Societies, Mar. 13-17, IEEE Xplore, London, pp: 1228-1239.

Frodigh, M., P. Johansson and P. Larsson, 2000. Wireless ad hoc networking-the art of networking without a network. Ericsson Rev., 4: 248-263.

Gao, Z., Y. Yang, J. Zhao, J. Cui and X. Li, 2006. Service discovery protocols for MANETs: A survey, Proceedings of the 2nd International Conference, MSN 2006, Dec. 13-15, Hong Kong, Chona, pp: 232-243.

Karp, B. and H.T. Kung, 2000. GPSR: Greedy perimeter stateless routing for wireless networks. Proceedings of the 6th Annual International Conference on Mobile Computing and Networking (Mobicom), Aug. 6-11, Boston, Massachusetts, USA., pp: 243-254.

Li, J., J. Jannotti, D.S.J.D. Couto, D.R. Karger and R. Morris, 2000. A scalable location service for geographic Ad hoc routing. Proceedings of the 6th Annual International Conference on Mobile Computing and Networking, Aug. 06-11, Boston, Massachusetts, United States, pp: 120-130.

Liu, F. and R. Ruonan, 2005. Analysis of GHT in mobile Ad Hoc networks. Proceedings of the ISPA 2005 International Workshops, AEPP, ASTD, BIOS, GCIC, IADS, MASN, SGCA and WISA, Nov. 2-5, Nanjing, China, pp: 35-44.

McCanne, S. and S. Floyd, 2008. The network simulator-ns-2. http://www.isi.edu/nsnam/ns/.

Meshkova, E., J. Riihijarvi, M. Petrova and P. Mahanen, 2008. A survey on resource discovery mechanisms, peer-to-peer and service discovery frameworks. Comput. Networks, 52: 2097-2128.

Oliveira, R., L. Bernardo and P. Pinto, 2008. Searching for Resources in MANETs-A Cluster Based Flooding Approach. In: E-business and Telecommunication Networks, Filipe, J. and M.S. Obaidat (Eds.). Springer Verlag, Berlin Heidelberg, pp: 236-245.

Pucha, H., S.M. Das and Y.C. Hu, 2004a. Ekta: An efficient DHT substrate for distributed applications in mobile ad hoc networks. Proceedings of the 6th IEEE Workshop on Mobile Computing Systems and Applications, Dec. 2-3, English Lake District, UK., pp: 163-173.

Pucha, H., S.M. Das and Y.C. Hu, 2004b. How to implement dhts in mobile ad hoc networks? Proceediungs of the 10th ACM International Conference on Mobile Computing and Network (MobiCom 2004), Sept. 26-Oct. 1, Philadelphia, PA, pp: 1-2.

Ratnasamy, S., B. Karp, L. Yin, F. Yu, D. Estrin, R. Govindan and S. Shenker, 2002. GHT: A geographic hash table for data-centric storage. Proceedings of the International Workshop on Wireless Sensor Networks and Applications, Sept. 28-28, Atlanta, Georgia, USA., pp: 78-87.

Ripeanu, M. and I.T. Foster, 2002. Mapping the Gnutella Network: Macroscopic Properties of Large-Scale Peer-to-Peer Systems. In: Peer-To-Peer Systems, Druschel, P., M. F. Kaashoek and A.I. Rowstron (Eds.). Springer Verlag, Berlin Heidelberg, pp: 85-93.

Seada, K. and A. Helmy, 2005. An overview of geographic protocols in ad hoc and sensor networks. Proceedings of the ACS/IEEE 2005 International Conference on Computer Systems and Applications, Jan. 03-06, Washington, DC., USA., pp: 62-1.

Stoica, I., R. Morris, D. Liben-Nowell, D.R. Karger, M.F. Kaashoek, F. Dabek and H. Balakrishnan, 2003. Chord: A scalable peer-to-peer lookup protocol for internet applications. IEEE/ACM Trans. Network., 11: 17-32.

Wu, J., 2006. Handbook on Theoretical and Algorithmic Aspects of Sensor, Ad Hoc Wireless and Peer-To-Peer Networks. Auerbach Publications, Boca Raton, Fl., pp: 874.

Yang, B. and H. Garcia-Molina, 2002. Improving search in peer-to-peer networks. Proceeding of the 22nd International Conference on Distributed Computer System, Jul. 2-5, IEEE Xplore, London, pp: 5-14.

Yang, C. and X. Li, 2005. Dominating-set-based searching in peer-to-peer networks. Int. J. High Performnnce Comput. Network., 3: 205-210.