IOSR Journal of Computer Engineering (IOSR-JCE)
e-ISSN: 2278-0661, p- ISSN: 2278-8727Volume 16, Issue 2, Ver. IV (Mar-Apr. 2014), PP 58-65
www.iosrjournals.org

Enhanced Data Sharing over Mobile Ad Hoc Network Based on non-Selfish Exposure

[1]Amrutha b k [2]Mr.I.Edwin Albert
Dept.of computer science and Engineering Jayamatha Engineering College Aralvaimozhi, Kanyakuari(dist) Tamilnadu

***Abstract**: A Mobile ad hoc network is a peer-to-peer multihop wireless network. MANETs are key to nomadic computing. Mobile units can set up spontaneous local networks and can remove the need for fixed network infrastructure, either as wireless access points or wired LAN. Each node in a MANET communicates with each other. Some nodes does not cooperate with each other nodes and termed as Selfish nodes. Selfish nodes are handled on the basis of replica allocation. Partial nodes are also handled. The reply from expected node holding a target data item may not reach its destination due to two reasons, disconnections and node selfishness. In a frequent time interval, replica allocation will perform based on the Self Centered Friendship tree of each node in order to avoid selfishness. In this paper we discuss about various detection algorithms including creating and modifying a SCF-tree using Self stabilize replica allocation. Our strategy consists of identification and detection of FRA when network disconnections occur in the network and results in data inaccessibility. A statistics will be displayed to identify the location of where the data will be residing after the reallocation. Data replacing and Malicious nodes which effect the security of the MANET are also discussed which increases data accessibility and decreases communication cost that caused by Query delay. The controller informs all other nodes regarding the failure of a particular node by providing an alarm named as FRA. The proposed system outperforms traditional cooperative replica allocation techniques in terms of data accessibility, communication cost, and average query.*

***Keywords**: MANET, credit risk, degree of selfishness, FRA, SSRA*

I. Introduction

In Mobile Adhoc NETworks (MANETs) numbers of mobile nodes are connected by wireless links and there by forming a dynamic arbitrary graph. These mobile nodes serve as both hosts and routers so they can forward packets on behalf of each other. Hence, the mobile nodes are able to communicate beyond their transmission range by supporting multi hop communication. Mobility is a vital feature of MANET. MANETs are very popular solution in the situation where network infrastructure is not available. All the nodes in MANET are mobile, power restricted, and thus, disconnection may occur frequently, causing a lot of network partitioning. Moreover many applications in this environment are time-critical and, hence, their transactions should be executed not only correctly, but also within their deadlines.

MANETs are limited by limited computing resources and due to mobile nature of node there is frequent change in network topology. These restrictions raise several new challenges for data access applications with the respects of data availability and access efficiency. This result in lower data accessibility in mobile ad-hoc networks than in wired ones. One possible solution to mitigate this problem is to employ replication techniques, which improve availability and decrease query response delay. Resource constraints such as battery and storage limitations of mobile nodes may lead to network partitioning or performance degradation. Network partitions can occur frequently, since nodes move freely in a MANET, causing some data to be often inaccessible to some of the nodes. Hence, data accessibility is often an important performance metric in a MANET .Data are usually replicated at nodes, other than the original owners, to increase data accessibility to cope with frequent network partitions. A considerable amount of research has recently been proposed for replica allocation in a MANET.

MANET are of two types of MANETs: open and closed . An open MANET comprises of different users, having different goals, sharing their resources to achieve global connectivity. This is different from closed MANETs where the nodes are all controlled by a common authority, have the same goals, and work toward the benefit of the group as a whole. MANET nodes are equipped with MANET interfaces, which have different characteristics. Dependency and decentralized of MANET allows an opponent to exploit new type of attacks that are designed to destroy the cooperative algorithms used in ad hoc networks. MANET is particularly vulnerable to several types of attacks like passive eavesdropping, active impersonation, and denial of services. The security research in MANET has focused on key management, routing protocol and intrusion detection techniques.

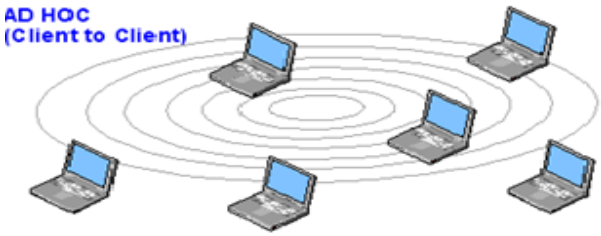


Fig 1: Nodes

MANET is a peer-to-peer network. It consists of large number of nodes connected and can communicate with each other. Mobile nodes in a MANET together have sufficient memory space to hold both the replicas (data) and the original data. A mobile node may hold a part of the frequently accessed data items locally to reduce its own query delay. Thus, the overall data accessibility would be decreased. There are some nodes within the MANET that are not communicating with the other nodes. It does not share the data contained in it. These type of nodes are termed as Selfish nodes. These nodes does not share data contained in it and there for it will affect the overall data accessibility. Each node have their own resource constraints such as memory limitations.

Due to its great features of mobility and flexibility, MANET attracts different real world application areas whereas topology changes very quickly. MANET is more vulnerable than wired network due to node's mobility, dangers from compromised nodes inside the network, limited security, dynamic topology, scalability and lack of centralized management. Because of these vulnerabilities criteria, MANET is more susceptible to malicious attacks.

In order to increase the data accessibility selfish nodes are to be handled correctly. These can be handled by replica allocation. Replica allocation is mainly done on the basis of Self centered friendship tree. SCF tree is constructed from the topology formed by the nodes in the MANET.

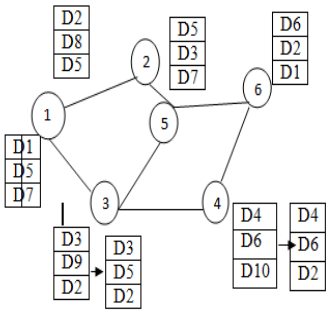


Fig 2: Data in Nodes

	NODES					
Data	N1	N2	N3	N4	N5	N6
D1	0.85	0.35	0.27	0.34	0.21	0.25
D2	0.44	0.65	0.41	0.40	0.52	0.37
D3	0.32	0.55	0.43	0.22	0.15	0.54
D4	0.42	0.74	0.28	0.45	0.31	0.24
D5	0.51	0.42	0.33	0.67	0.09	0.10
D6	0.08	0.07	0.06	0.19	0.53	0.69
D7	0.38	0.32	0.41	0.69	0.20	0.47
D8	0.18	0.77	0.37	0.40	0.58	0.32
D9	0.23	0.19	0.28	0.17	0.57	0.42
D10	0.09	0.06	0.38	0.68	0.49	0.09

Table 1: Access Frequency Table of Nodes

In this example, it consists of number of nodes connected together. Each node can communicate with each other. Each node have their own limited memory. Portion marked as grey defies the needed data for the corresponding node. And rest of the memory defines replicated data. Access Frequency table defined at the right side provides the information that are accessed frequently. In order to transfer the data without any lag ,data are stored in the limited memory space allocated to each. According to the topology the data are allocated into the neighboring nodes for the easier data accessibility based on access frequency. So for the data forwarding it can handle data from the neighboring nodes. But some nodes instead of storing the data from the neighboring nodes they store the data for their benefit only. Replicas are created based on this. This type of nodes will behave as selfish.

Here node N3 act as a selfish node. Instead of using the memory allocated to it as M3, it use of $M^{1}3$ for the benefit of it. D3, D9, D2 are the data allocated to M3.But instead of it M3 store D3, D5, D2.D9 is not accessible by other nodes. Here M3 act as a Selfish node.

A node can act as partially selfish node also.N4 stores D2 for its own access and rest of the memory are used for the benefit of other nodes for data accessibility. So N4 maintains a memory location $M^{1}4$ instead of M4.That is partial of the memory spaces are used for the benefit of the others. Partially selfish nodes are also needed to handle like selfish node for the data accessibility.

For each node topology is created for separately .The topology consists of a controller which is capable of identifying and maintaining the status of each nodes, whether the nodes are in active or inactive stage..

Neighboring nodes are identified on the basis of Self Centered Friendship tree. In the SCF tree, paths are identified for the processing of each data. Access frequency is calculated. Credit Risk scores defines the Replica to be allocated to each node. Selfish nodes are identified on the basis of replica allocation. Faulty Root Alarm are set for each in order to identify the change in the nodes in the net work. An alarm is raised based on the selfish behavior of overall nodes also identifying and handling faulty root alarm with minimum memory space utilization. The controller informs all other nodes regarding the failure of a particular node by providing FRA. As this portioning occurs the changes may occur in the proposed network, so the SCF tree has to be updated by self stabilize replica allocation technique A statistics will be displayed to identify the location of where the data will be residing after the reallocation. Detection of attacker node in the network and should be informed to all others in the network. Malicious nodes which effect the security of the MANET is also discussed, there by which increasing data accessibility and decreasing communication cost that caused by Query delay.

The technical Contribution in this paper can be recognized as:

1. Recognizing the misbehavior in data forwarding

2. Identifying the both selfish and partial selfish node

3. Setting Faulty Root Alarm.

4. Allocating replica effectively

5. Data allocation statistics

II. System Design

In this we discuss about various nodes that are present within the MANET with a different behavior. Mainly the nodes are divided into three types in the basis of memory resource constraints:

- **Type-1 node**: The nodes are non-selfish nodes. These nodes can hold replicas allocated by the other nodes for the benefit of others.

- **Type-2 node**: The nodes are fully-selfish nodes. These are the nodes which do not hold replicas allocated by other nodes, and use the memory location for the benefit of them only.

- **Type-3 node**: The nodes are partially- selfish nodes. These partially selfish nodes would use their own memory space partially for allocated replicas by other nodes and the rest for their benefit only. Their memory space may be separated logically into two parts: one is selfish area and another one is public area.

Selfish node can be detected on the basis of replica allocation. Degree of selfishness can be calculated depending upon the behavior of selfish node and partial selfish node. Each node has their own access frequency depending upon the data accessibility of the node.

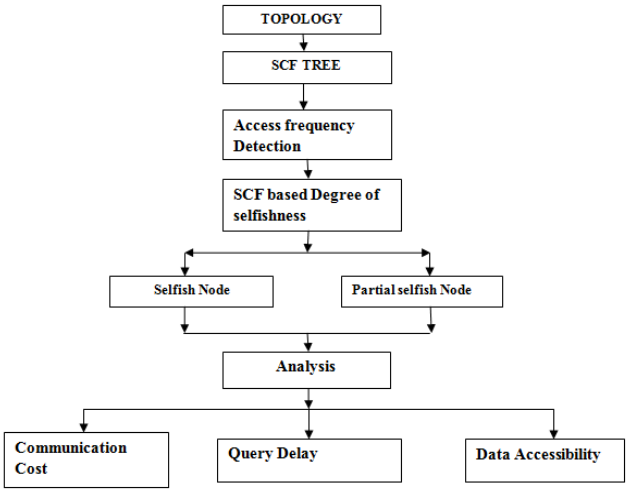


Fig 3: System Design

III. Proposed System

In this paper we mainly discuss about
A. Construction of topology
B. Creation of SCF tree
C. Allocating Replica effectively
D. Updating the SCF tree using self stabilize friendship tree
E. Setting the faulty Root alarm
F. Display the statistics of data

A. Construction Of Topology

Topology for each particular node is created manually. Each node makes its own topology graph and builds its own SCF-tree by excluding selfish nodes. The topology graph may be of partial according to the particular node. Based on the concept of SCF-tree, each node in the network allocates replica in a fully distributed manner .The CR score is updated during the query processing phase. The degree of selfishness represents relationships among nodes in a MANET, for replica allocation, termed the SCF-tree.

If a node, say N1decides for data forwarding with other nodes it makes an access request to a data item typically sending a query. The particular node will checks its own memory space first. If the requested item is present in its own local memory space then the request is successful. If it does not hold the original or replica, the request will be broadcasted to other nearby nodes which is connected to the node N1,say N2,N3...etc. The request is also successful when N1 receives any reply from at least none node connected to N1 with one hop or multiple hops of nodes, which holds the original or replica of the targeted data item. Otherwise, the request fails.

Whenever a node N1 receives a data access request, it either 1) serves the request by sending its original or replica if it holds the target data item, or 2) forward the request to its neighbors if it does not hold the target data item.

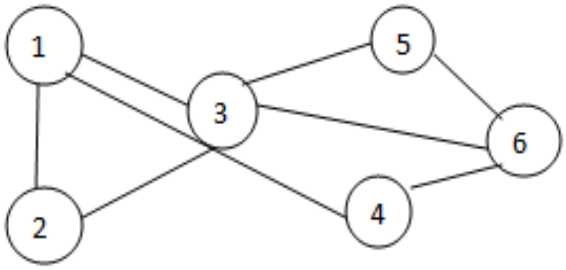


Fig 4: Topology

Using the graphics function we can create topology for the particular node separately. Corresponding to each node we can create SCF tree for each node.

B. Construction of SCF tree

Self Centered Friendship tree which resembles the human friendship tree finds the relationship between the nodes. In this for each node SCF tree is constructed separately. Consider the node Ni transfers the data to the neighboring nodes, data transferring of data are identified and Ni builds the SCF tree of its own, Ni first checks the nodes that are connected to Ni but one hop to Nis' child nodes. Then Ni checks repeatedly the child nodes of appended nodes, until the depth of the SCF tree is equal to d.SCF tree may have multiple routes for some nodes from the root node. At every relocation period each node updates its own SCF tree based on the network topology of that moment. Each depth of the node can be identified by calculating the path or by constructing the SCF tree.

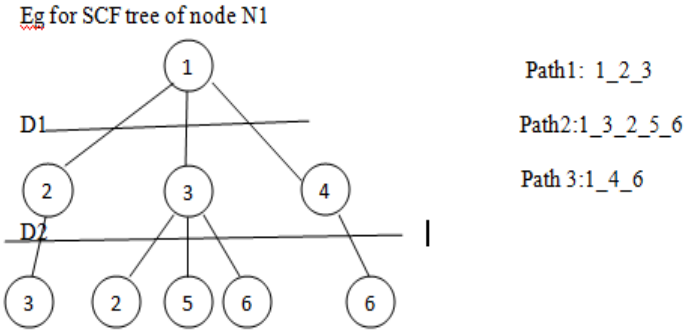


Fig 5:SCF tree

SCF tree construction is mainly done in distributed manner. Neighboring nodes are identified on the basis of distance formula when x,y coordinates of nodes are given.

Distance formula for calculating te distance formula is:

$$\sqrt{((dx)^2+(dy)^2)}$$

Where dx=difference between the two x-coordinates
dy=difference between two y-coordinates

Based on the topology graph Gi, nodes N_i builds their own SCF tree denoted as T_i^{SCF} .Each node has a parameter d, the depth of SCF Tree. When Ni builds its own SCF tree ,Ni first appends the nodes that are connected to Ni by one hop to Ni's child nodes. Then Ni check recursively the child nodes of the appended nodes, until the depth of the SCF tree is equal to d.

Pseudo code to build SCF-tree

```
constructScfTree()
 {
  append Ni to SCF-tree as the root node;
  checkChildnodes(Ni );
  return SCF-tree; }
  Procedure checkChildnodes(Nj)
{ for (each node Na Ɛ INaj) {
 if (distance between Na and the root > d ) continue;
 else if (Na is an ancestor of Nj in Ti^SCF ) continue;
 else { append Na to Ti^SCF as a child of Nj;
 checkChildnodes(Na); } } }
```

C. Allocating Replica effectively

Replication provides an attractive solution for this problem for retrieving distant data in MANET. A good replication management technique for MANETs should be efficient to deliver requested data items from the neighbouring node and capable to decide which data items can be replicated at a node. Further there should be a replica replacement algorithm to replace the old copy of data items when there is an update in the original copy of the data item. After building the SCF tree, a node allocates replica t every relocation period. Each nodes asks non selfish nodes within its SCF tree to hold replica when it cannot hold replica in its local memory space.SCF tree is done in a fully distributed manner, so each node determines replica allocation individually without any communication with other nodes.

Replication is allocated mainly by calculating Access frequency effectively. It is suitable to improve the response time of the access requests, to distribute the load of processing of these requests on several servers and to avoid the overload of the routes of communication to a unique server. Ad hoc data replication problem (ADRP) was first introduced by Hara , which was further extended to include various network connectivity related issues.

Hara proposed three replica allocation methods which differ in emphasis put on access frequency and network topology: SAF, DAFN and DCG. In SAF (Static Access Frequency) method only the access frequency to each data item is taken into account. In DAFN (Dynamic Access Frequency and Neighborhood) method the

access frequency to each data item and the neighborhood among mobile hosts are taken into account. And in DCG (Dynamic Connectivity based Grouping) method the access frequency to each data item and the whole network topology are taken into account.

The first step of this technique is to calculate the weighted access frequencies of data items based on their data and transaction types. The next step is to replicate data items with higher weighted access frequencies in servers that have the maximum remaining power. After storing the data items at appropriate servers, if there are any redundant data items among neighboring servers, they are eliminated depending upon the stability of the links connecting them and the access frequencies of the next available data items. Such a decision to replicate data items is taken every time during a certain period of time called the **relocation period**.

Access Frequency of each node can be calculated using the formula:

AF= count/period
Where Count=no of data items needed with in a time period.
Period= it is the time taken, say 10sec.

Nodes can use some portion of its memory space selfishly, we may divide memory space Mi
For replica logically into two parts: Selfish area Ms and public area Mp.Each node use its own memory space Mi freely as Ms or Mp.

At first nodes determines the priority for allocating replica .The priority is based on Breadth First Search (BFS) order of the SCF Tree. For each replica for node a table is maintained called Available
Replica Table (ART). This ART will contain the information about the replicas that are available with different node in that cluster. There will be 3 entries related to each node: node-id, item-id, space-available.

Table 2 shows the entries of ART.

NO	Entry Name	Meaning
1	node-id	denote the id of the node
2	item-id	denote the id's of the data items that are replicate with that node
3	space available	denote the free space available with that node (in KBs)

Table 2: ART table

The node will be created dynamically and the each node energy level is monitored that is, their credit risk value is calculated based upon their query request response value. Then the node with high credit score value is elected as leader by comparing all the other nodes to monitor the other node to avoid the false alarm in identifying the selfish node for novel replica allocation technique.

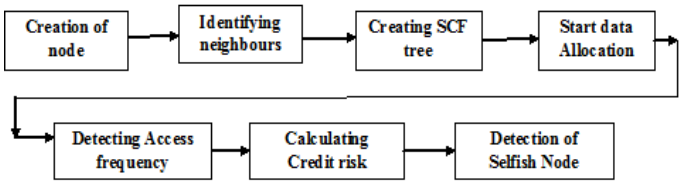


Fig 6: Proposed system

D. Updating the SCF tree using self stabilize friendship tree

In the MANET there consist of number of nodes may enter into the network. MANET is a peer to peer network. For each node it consists of different SCF tree, which defines the relationship between the neighbouring nodes. Data transferring is done through each node.
SCF tree for each node can be updated for each node. This can be done by a new technology called Self stabilize replica allocation. This technique is done by making each node stable comparing with other nodes. Selfish node can be identified using replication technique.
In this SCF tree is constructed by ignoring the selfish node and a new SCF tree is constructed.

E. Setting Faulty Root Alarm

In a MANET consists of' n' number of nodes. This nodes share their data with other nodes. There are some nodes that does not share their data with other nodes. This type of nodes is known as Selfish nodes.

In order to handle data forwarding we need to detect the selfish node. This selfish node can be handled or can be understand from the set of nodes. This is done by setting Faulty root alarm. This alarm specifies the selfish node with the resource constraints battery storage and memory.

F. Data allocation statistics

In this paper, after creating the SCF tree and replication technique data allocated to each node may vary. In the allocation statistics data are placed into various memory locations of different nodes. After creating all this nodes a data allocation statistics will be provided .Using this statistics table we can identify where all are the data placed. And each node can identify were the correspond data is placed within the node.

IV. Analysis

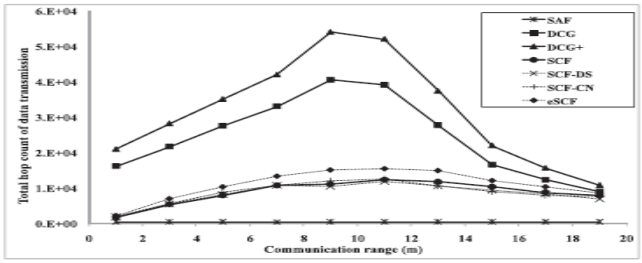
A. Communication cost

As the communication range increases, the communication cost of all techniques increases at first. When the communication range is smaller than a certain point, the communication cost increases as the communication range gets larger, since the number of nodes connected to each other increases and thus the communication cost caused by replica relocation increases. Conversely, when the communication range is larger than a certain point, the number of hops among connected nodes decreases. Therefore, the communication cost caused by replica relocation decreases. It is shown in the fig a.

Static access frequency allocates replica based only on its own access frequency, without considering or detecting selfish nodes. This allocation technique is expected to show the optimal performance in terms of communication cost.

The average query delay of all techniques degrades as the communication range increases, but it improves from a certain point, since when the communication range is larger than 9, the number of hops among connected nodes decreases. We see the data accessibility improves with the wide range of communication, since more nodes become connected. Clearly, our techniques work best. In

Credit-payment techniques, each node gives a credit to others, as a reward for data forwarding. The acquired credit is then used to send data to others.

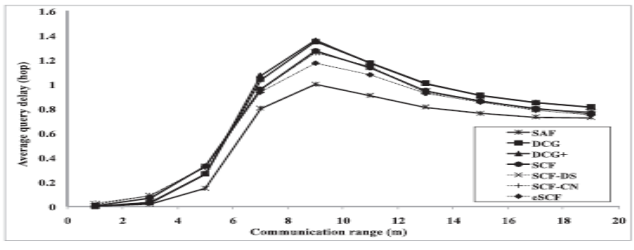


(a) Communication cost

Fig 7: Analysis of communication cost

B. Average query Delay

The SAF technique shows the best performance in terms of query delay, since most successful queries are served by local memory space. Our intuition was that query delay decreases as the size of memory space increases. This intuition is confirmed by the results in Fig. b. As the size of memory space increases, many nodes will accept replica allocation/relocation requests, since the size of public memory space increases as well. As a result, more queries are served by nearby nodes or locally. In our techniques, the number of successful queries being locally served increases slightly. This is because when the number of nodes in the SCF-tree is very small, the local public memory space may be used for data items of local interest temporarily.

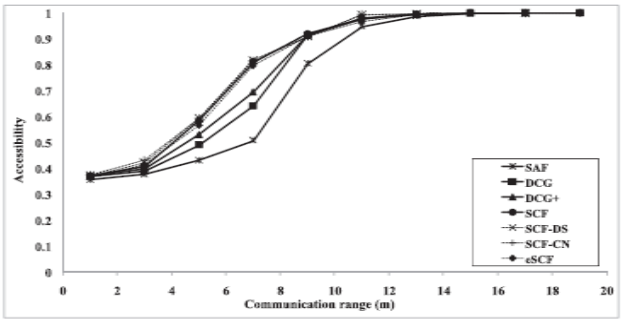


(b) Average query delay

Fig 8:Analysis of average Query delay

C. Data Accessibility

We evaluate the data accessibility of replica allocation methods under consideration. We expect that our techniques perform significantly better than other techniques in the presence of selfish nodes. The performance of our techniques improves faster than do others, since our techniques fully utilize the memory space of nodes. our techniques can detect and handle selfish nodes in replica allocation effectively and efficiently. Among our techniques, the eSCF technique shows a slightly poorer performance. Our initial intuition was that, data accessibility is stable with relocation periods. This is shown in fig c.



(c) Accessibility

Fig 9: Analysis of data accessibility

V. Conclusion

The problem addressed here are selfish nodes in the MANET. Term this problem selfish replica allocation. The proposed selfish node detection method and novel replica allocation techniques are used to handle the selfish replica Allocation appropriately. The notion of credit risk is applied to detect selfish nodes. Every node in a MANET calculates credit risk information on other connected nodes individually to measure the degree of selfishness. Every node in a MANET calculates credit risk information on other connected nodes individually to measure the degree of selfishness. Since traditional replica allocation techniques failed to consider selfish nodes, we also proposed novel replica allocation techniques. After detecting the selfish nodes each creates the data statistics for identification and data forwarding is done easily. The proposed strategies outperform existing representative cooperative replica allocation techniques in terms of data accessibility, communication cost, and query delay.

References

[1]. P. Padmanabhan, L. Gruenwald, A. Vallur, and M.Atiquzzaman, "A Survey of Data Replication Techniques for Mobile Ad Hoc Network Databases," The Int'l J. Very Large Data Bases, vol. 17, no. 5, pp. 1143-1164, 2008.

[2]. C.E. Perkins and P. Bhagwat,(1994) 'Highly Dynamic Destination- Sequenced Distance-Vector Routing (DSDV) for Mobile Computers' Proc. ACM SIGCOMM '94, pp. 234-244.

[3]. Hyun injin kim, jon M.peha, Carnegie mellon university(2008),'Detecting selfish behavior in a cooperative commons',research show,IEEE span.

[4]. Jae-Ho Choi, Kyu-Sun Shim, SangKeun Lee, and Kun-Lung Wu. (2012)'Handling Selfishness in Replica Allocation over a Mobile Ad Hoc Network in IEEE TRANSACTIONS ON MOBILE COMPUTING, VOL. 11, NO. 2

[5]. B.-G. Chun, K. Chaudhuri, H. Wee, M. Barreno, C.H. Papadimitriou, and J. Kubiatowicz, "Selfish Caching in Distributed Systems: A Game-Theoretic Analysis," Proc. ACM Symp. Principles of Distributed Computing, pp. 21-30, 2004.

[6]. G. Ding and B. Bhargava, "Peer-to-Peer File-Sharing over Mobile Ad Hoc Networks," Proc. IEEE Ann. Conf. Pervasive Computing and Comm. Workshops, pp. 104-108, 2004.

[7]. T. Hara and S.K. Madria, "Consistency Management Strategies for Data Replication in Mobile Ad Hoc Networks," IEEE Trans. MobileComputing, vol. 8, no. 7, pp. 950-967, July 2009.

[8]. S.U. Khan and I. Ahmad, "A Pure Nash Equilibrium-Based Game Theoretical Method for Data Replication across Multiple Servers,"IEEE Trans. Knowledge and Data Eng., vol. 21, no. 4, pp. 537-553,Apr. 2009.

[9]. N. Laoutaris, G. Smaragdakis, A. Bestavros, I. Matta, and I.Stavrakakis, "Distributed Selfish Caching," IEEE Trans. Parallel and Distributed Systems, vol. 18, no. 10, pp. 1361-1376, Oct. 2007.

[10]. N. Laoutaris, O. Telelis, V. Zissimopoulos, and I. Stavrakakis, "Distributed Selfish Replication," IEEE Trans. Parallel and Distributed Systems, vol. 17, no. 12, pp. 1401-1413, Dec. 2006.

[11]. H. Li and M. Singhal, "Trust Management in Distributed Systems," Computer, vol. 40, no. 2, pp. 45-53, Feb. 2007.

[12]. A. Mondal, S.K. Madria, and M. Kitsuregawa, "An Economic Incentive Model for Encouraging Peer Collaboration in Mobile-P2P Networks with Support for Constraint Queries," Peer-to-Peer Networking and Applications, vol. 2, no. 3, pp. 230-251, 2009.

[13]. V. Srinivasan, P. Nuggehalli, C. Chiasserini, and R. Rao,"Cooperation in Wireless Ad Hoc Networks," Proc. IEE INFOCOM, pp. 808-817, 2003.