Available Online at www.ijcsmc.com

International Journal of Computer Science and Mobile Computing

A Monthly Journal of Computer Science and Information Technology

ISSN 2320–088X

IJCSMC, Vol. 2, Issue. 3, March 2013, pg.112 – 119

RESEARCH ARTICLE

A Collaborative Selfish Replica with Watchdog and Pathrater in MANETS

D.Anitha[1], Dr.M.Punithavalli[2]

[1]Research scholar, Dept of Computer Science, Karpagam University, India

anithasuresh2003@gmail.com

[2]Director-MCA, Department of Computer Application, Sri Ramakrishna Engineering College, India

Abstract— Mobile Ad-Hoc Networks are autonomous and decentralized wireless systems. MANETs consist of mobile nodes that are free in moving in and out in the network. Nodes are the systems or devices i.e. mobile phone, laptop, personal digital assistance, MP3 player and personal computer that are participating in the network and are mobile. These nodes can act as host/router or both at the same time. They can form arbitrary topologies depending on their connectivity with each other in the network. These nodes have the ability to configure themselves and because of their self-configuration ability, they can be deployed urgently without the need of any infrastructure. Internet Engineering Task Force (IETF) has MANET working group (WG) that is devoted for developing IP routing protocols. Routing protocols is one of the challenging and interesting research areas. Many routing protocols have been developed for MANETS, i.e. AODV, OLSR, DSR etc.

Key Terms: - MANETS, AODV, Selfish nodes, Credit Risk, Watch Dog, Pathrater.

I. INTRODUCTION

An Adhoc wireless network is a collection of two or more devices equipped with wireless communications and networking capability. Such devices can communicate with another node that is immediately within their radio range or one that is outside their radio range. The challenges that are faced in Adhoc mobile networks are as follows. They are media access, routing, multicasting, energy efficiency, TCP performance, service location, provision and access and security & privacy.

In general, replication can simultaneously improve data accessibility and reduce query delay, if the mobile nodes in a MANET together have sufficient memory space to hold both the replicas 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. Hence, to maximize data accessibility, a node should not hold the same replica that is also held by many other nodes. A node may act selfishly, i.e., using its limited resource only for its own benefit, since each node in a MANET has resource constraints, such as battery and storage limitations [3]. A node would like to enjoy the benefits provided by the resources of other nodes, but it may not make its own resource available to help others. Such selfish behavior can potentially lead to a wide range of problems for a MANET. For example, selfish nodes may not transmit data to others to conserve their own batteries [1] [5].

The rest of the paper is organized as follows. In Section II, The Related work with regards to our work is being discussed. In Section III we introduced the Routing protocol which we will use AODV in our simulation. In Section IV, We introduce and present our new system to detect selfish nodes. In Section V, We review our system for simulation environment. In section VI, We conclude the work.

II. RELATED WORK

Several systems have been proposed to detect misbehaving nodes in mobile ad hoc network. This system can be classified into three categories:

A. CREDIT-BASED SYSTEM

Credit based systems [6], [9] are designed to provide incentives for forwarding packets in the form of virtual money (specifically called as Credit). Nodes earn Credit by providing forwarding services to others and have to pay to get services from other nodes. However, to protect the Credit value from attacks and modification, some costly security modules independent of nodes have to be used. In addition, colluding nodes can agree to forward their own flows to accumulate credits while dropping all other flows. Moreover, a well behaved node that is not asked to route enough packets could not earn credits and will be unable to send its own packet.

B. REPUTATION BASED SYSTEM

Reputation based systems on the other hand rely on building a reputation metric for each node according to its behavioral pattern. A monitoring method used by most systems in this category is called a watchdog. Watchdog was proposed by Marti et al. [8] to detect data packet non-forwarding by overhearing the transmission of the next node. [11], [12], [13] us similar monitoring scheme but then propagate collected information to nearby nodes and are susceptible to false praise and false accusation attacks. Mr. Bansal and Mr. Baker proposed a system called OCEAN [16] where the reputation of a neighbor is evaluated using only locally available information and thus avoid sophisticated and potentially vulnerable techniques of reputation propagation throughout the network. It is reported that even with direct observations of the neighbor; OCEAN performs almost as we ll and sometimes even better compared to schemes that share second hand reputation information.

C. ACKNOWLEDGEMENT BASED SYSTEM

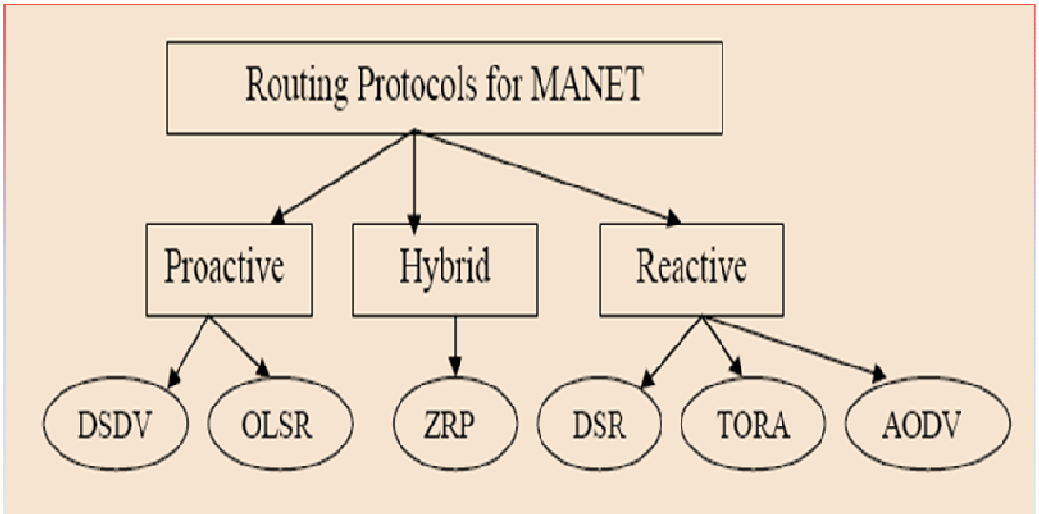
The last category is acknowledgment based systems which rely on the reception of an acknowledgment to verify that a packet has been forwarded. Liu et al. [17] proposed the 2ACK system where nodes explicitly send acknowledgment two hops upstream to verify cooperation. This system is susceptible to collusion of two or more consecutive nodes. Furthermore, colluding nodes can frame honest ones by claiming not to receive the acknowledgment. All of the mechanisms mentioned above are designed to detect and handle misleading nodes. There are a few systems that have been proposed to detect selfish nodes in a MANET. One example is Context Aware Scheme [18] introduced by Mr. Paul and Mr. Westhoff. This system uses unkeyed hash chains and a promiscuous mode to detect the misbehavior during route discovery phase. The observers of misbehavior independently communicate their accusation to the source. To convict a culprit, more than three accusations are needed. If there is only one accusing node, the accusing node itself will be considered to be an attacker. The drawback of this system is that it is more beneficial for a node not to send the alarm message to avoid the risk being the only accuser and regarded as attacker. In [19], Djenouri et al. propose two different techniques to detect two different types of control packet droppers. They suggest the use of two hop ACK approach for monitoring directed packets (RREP, RRER) and promiscuous based overhearing technique for monitoring broadcast packets (RREQ). Huang et al. [17] suggest that the monitoring node simply compares the ratio of relay RREQ number between its neighbor and itself. If the ratio is smaller than a threshold, the neighbor is regarded as selfish and its packet is dropped as the punishment.

III. CLASSIFICATION OF ROUTING PROTOCOLS IN MANETS

Routing protocols in MANETs are classified into three different categories according to their functionality.

- Reactive protocols
- Proactive protocols
- Hybrid protocols

The hierarchy of these protocols is shown below in the figure1.



Ad-Hoc On Demand Distance Vector Protocol (AODV)

AODV is described in RFC 3561 **[GFAQ, 3561].** It's reactive protocol, when a node wishes to start transmission with another node in the network to which it has no route; AODV will provide topology information for the node. AODV use control messages to find a route to the destination node in the network. There are three types of control messages in AODV which are discussed below.

- **Route Request Message (RREQ):**

Source node that needs to communicate with another node in the network transmits RREQ message. AODV floods RREQ message, using expanding ring technique. There is a time to live (TTL) value in every RREQ message, the value of TTL states the number of hops the RREQ should be transmitted.

- **Route Reply Message (RREP):**

A node having a requested identity or any intermediate node that has a route to the requested node generates a route reply RREP message back to the originator node.

- **Route Error Message (RERR):**

Every node in the network keeps monitoring the link status to its neighboring nodes during active routes. When the node detects a link crack in an active route, (RERR) message is generated by the node in order to notify other nodes that the link is down.

IV. SYSTEM MODEL

In this paper, we presume that every node has partial local memory space and acts as a data provider of quite a few data items and a data consumer. Each node holds replicas of data items, and maintains the replicas in local memory space. The replicas are relocated in an explicit time. There are tn nodes, N1,N2, . . .,Ntn. Constructing a representation for MANET is an undirected graph G= (FS, CL) that consists of a finite set of nodes, FS, and a finite set of communication links, CL, where every element is a tuple (Nj,Nk) of nodes in the network. The following assumptions are ready and it is similar to those in [6]. A piece node in a MANET has an exclusive identifier.

All nodes with the intention of are placed in a MANET are denoted by N =(N1,N2, . . .Ntn) where tn is the total number of nodes and the set of all data items is denoted by DI =(DI1,DI2, . . .,DIn), where n is the total number of data items.

Each node Ni (1 < i< m) has restricted memory space for replica and original data items. The size of the memory space is Smi. Each node can hold only C, where (1 < C < n), replica in its memory space.

Each node Ni (1< i< m) has its own access frequency to data item Dj D (1< j< n), AFi .The access frequency does not change.

For that reason, the three types of behavioral states for nodes from the viewpoint of selfish replica allocation are described.

1. Type-1 node: The nodes are non-selfish nodes. The nodes hold replicas owed by further nodes controlled by the limits of their memory space.

2. Type-2 node: The nodes are fully selfish nodes. The nodes do not hold replicas owed by other nodes, but allocate replicas to other nodes for their convenience.

3. Type-3 node: The nodes are to some extent selfish nodes. The nodes utilize their memory space to some extent for allocated replicas by other nodes. Their memory space may be divided logically into two parts: selfish and public area. These nodes allocate replicas to other nodes for their convenience.

The nodes in the MANET calculate the credit risk in the connected nodes individually to measure the degree of selfishness. The selfish node is detecting by the self-replica allocation. It is based on the concept of a self-centered friendship tree SCF tree.

Detecting Selfish Node:
The concept of credit risk can be described by the subsequent equation:
Credit Risk =expected risk / expected value

N_i	N_k					
	N_1	N_2	N_3	N_4	N_5	N_6
N_1	·	0.30	0.85	0.80	0.45	0.22
N_2	0.40	·	0.80	0.90	0.30	0.50
N_3	0.25	0.35	·	0.75	0.65	0.75
N_4	0.45	0.44	0.51	·	0.23	0.37
N_5	0.30	0.60	0.85	0.40	·	0.21
N_6	0.40	0.50	0.90	0.52	0.30	·

Fig 1 Credit Risk Table

Selfish features are alienated into two category Node specific and query processing specific

1) Node specific query
This is specified as the expected value of a node. For instance when node N_i observes that node N_k shares large SS_k^i, ND_k^i, node N_k may be treated as a valuable node by node N_i. In our approach the size of Nk's shared memory space is denoted as SS_k^i and the number of N_k's shared data item, denoted as $ND^k{}_i$, observed by a node N_i, are used as node-specific features. The fig.1 gives a sample of credit risk table.

2) Query Processing specific
This feature is specified as the expected risk of a node. We utilize the ratio of selfishness alarm of N_k On N_i , denoted by p_k^i , which is the ratio of of N_i's data request being not served by the expected node N_k due to N_k selfishness in its memory space.

$$nCR = \frac{p_k^i}{\alpha * SS_k^i / S + (1-\alpha) * ND_k^i / N^i}$$

Building SCF-Tree
The SCF-tree based replica allocation techniques are stirred by human friendship management in the real world, where every person makes his/her own friends forming a web and manages friendship by himself/herself. He/she does not have to confer these with others to preserve the friendship. The main intention of the novel replica allocation techniques is to diminish traffic overhead, while achieving high data accessibility. Before constructing the SCF-tree, each node makes its own partial topology graph Gi = (INi, ILi), which is a component of the graph G. Gi consists of a finite set of the nodes connected to Ni and a finite set of the links, where Ni € FSi, FSi _ FS, and CLi _ CL. Based on Gi ns , Ni builds its own SCF-tree, denoted as TiSCF . Algorithm 1 describes how to construct the SCF-tree. 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 checks recursively the child nodes of the appended nodes, until the depth of the SCF-tree is equal to d. Fig. 2 illustrates the network topology and some SCF-trees of N1 in Fig. 1.

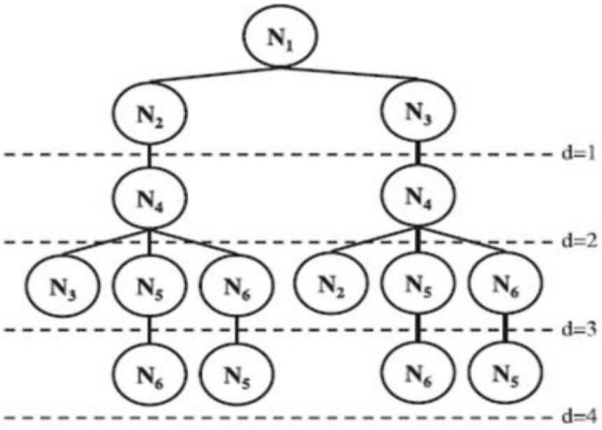


Fig 2. SCF Tree for N1

Algorithm 1

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 € INaj)
 {
 if (distance between Na and the root > d )
 continue;
else if (Na is an ancestor of Nj in TiSCF )
 continue;
else { append Na to TiSCF as a child of Nj; checkChildnodes(Na); }
 } }
```

Watch dog and Pathrater

Watchdogs are used to detect selfish nodes in computer networks these are initiated by Replica server. Pathrater is used to delete the nodes and to create a new path.

The system introduces two extensions to the AODV algorithm to moderate the things of routing misconduct: the Watchdog, to perceive the misbehaving nodes and the Pathrater, to counter to the intrusion by isolating the selfish node from the network operation.

Watchdog run on every one node. When a node frontwards a packet, the node's watchdog component verify that the next node in the path also forwards the packet. The Watchdog does this by listening in promiscuous mode to the next node's transmissions. If the next node does not forward the packet, then it is measured to be mischievous and is reported. This is done by distribution an alarm message to the other nodes on its friends list. When individuals' nodes accept the alarm message, they assess it and change the status of the accused node only if the alarm source is fully trusted or the same node was accused by a number of moderately trusted nodes. The watch dog concept will maintain a counter value in the network. The previous status of the node is also maintained. This table is termed as status table. The structure contains Server node ID, destination node ID, hop count and drop packet.

After the Watchdog component detects the malevolent node, the Pathrater component then deletes the analogous route from the route cache and tries to resolve if there is another route accessible to the destination by looking in its cache table. If not, Pathrater will broadcast a Route Request to get a new route to the destination. Then the server will send the signal to the nodes. It will refresh counter that is it will clear the counter value so that the selfishness can be removed.

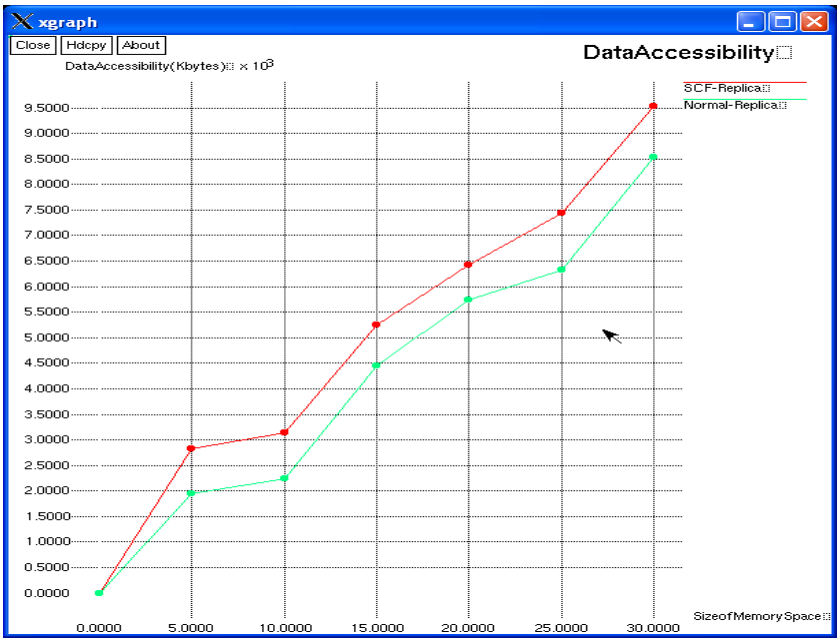
V. RESULTS ANALYSIS

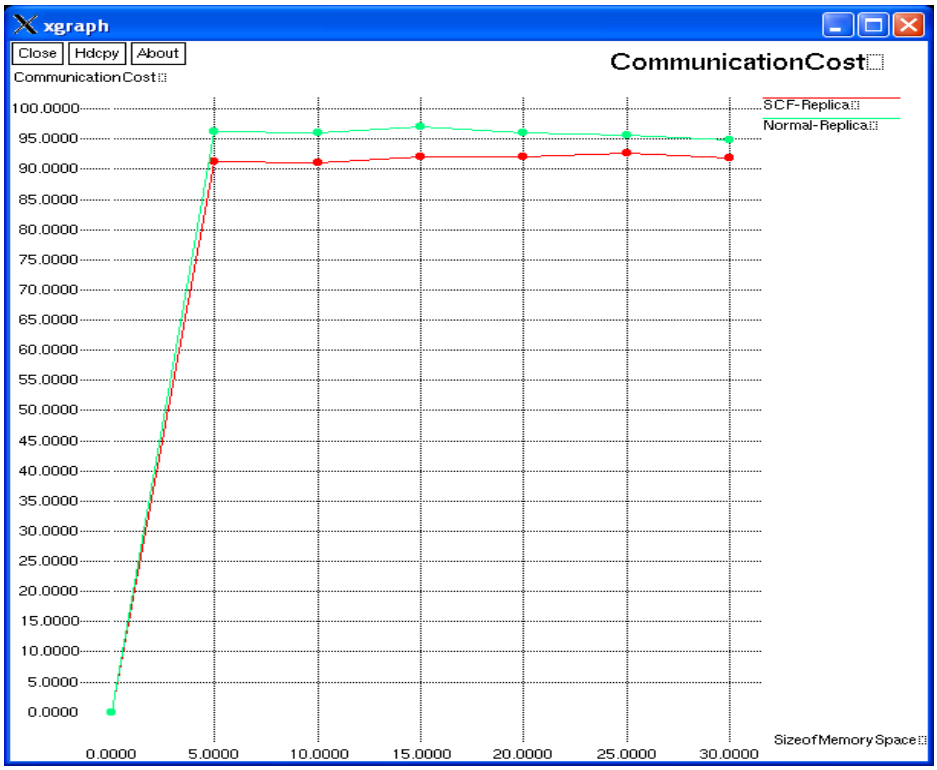
Simulation Parameters:

In this section, we use simulations to evaluate the performance of CR with WD&P with different overlaying applications.CR with WD&P is implemented within the ns-2 simulator [12]. The radio model is based on the existing commercial hardware with a wireless transmission range of 270 meters and channel capacity of 3Mbps. Each simulation runs for 270 seconds and the results are compared with normal replica. The applications of interest include, FTP-driven TCP traffic, CBR-driven (constant bit rate) UDP traffic, audio-driven UDP traffic and video-driven UDP traffic. For the FTP, CBR, and audio traffic, the modules in the ns-2 distribution are used. For the video traffic, we use the actual traces to drive the ns-2 simulations. All packets are set to be of 512 bytes, except that video traffic has varying packet sizes based on the actual traces. AODV is used for routing. The data accessibility rate has been increased in our SCF model. The communication cost has been reduced to some extent in our proposed model. The simulation parameters of our work are as follows.

Table 1: Simulation parameters

Parameters	Values
Examined protocol	AODV
Traffic type	Constant Bit Rate
Packet size	1024 bytes
Date rate	100kb/s
Pause time	10s
Minimum speed	1 m/s
Simulation time	900s
Area	100x100m





VI. CONCLUSION

In contrast to the MANET viewpoint, this paper has addressed the problem of selfish nodes from the replica allocation perspective. We have proposed a selfish node detection method and method to solve selfishness to handle the selfish replica allocation appropriately. By using Proxy replica server the selfishness of MANET nodes can be removed. Watch dog concept helps to rectify the selfishness identified. The status table is updated and is informed to the server about the status of every node. Pathrater concept is used to delete the route and create a new route. Both of these methods are combined and selfishness is fully evaluated. This proposed

system is capable of handling selfishness in small size network. Based on the server's capacity the selfishness can be handled by the server. We plan to identify and handle false alarms in selfish replica allocation. The proposed strategies improve the data accessibility, identify the selfish node and also reduce communication cost.

REFERENCES

[1] 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.

[2] K. Paul and D. Westhoff, "Context Aware Detection of Selfish Nodes in DSR Based Ad-Hoc Networks," Proc. IEEE Global Telecomm. Conf., pp. 178-182, 2002. [11]. Y. Yoo and D.P. Agrawal, "Why Does It Pay to be Selfish in a MANET," IEEE Wireless Comm., vol. 13, no. 6, pp. 87-97, Dec. 2006.

[3] 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. [13] L.J. Mester, "What's the Point of Credit Scoring?" Business Rev.pp. 3-16, Sept. 1997.

[4] L. Buttyan and J. Hubaux, "Stimulating cooperation in self-organizing mobile ad hoc networks," in Mobile Networks and Applications, vol. 8, no. 5, October 2003, pp. 579 – 592.

[5] G. Xiapeng and C. Wei, "A novel gray hole attack detection scheme for mobile ad - hoc networks," in IFIP International Conference on Network and Parallel Computing, September 2007, pp. 209 –214.

[6] A. Babakhouya, Y. Challal, and A. Buouabdallah, "A simulation analysis of routing misbehavior in mobile ad hoc networks," in The Second International Conference on Next Generation Mobile Applications, Services and Technologies NGMAST'08, 2008, pp. 592 – 597.

[7] S. Zhong, J. Chen, and Y. R. Yang, "Sprite: A simple, cheat - proof, credit - based system for mobile ad -hoc networks," in INFOCOM 2003, 2003.

[8] S. Marti, T. Giuli, K. Lai, and M. Bakar , "Mitigating routing misbehavior in mobile ad hoc networks," in Proc. 6th Annual Int. Conf. on Mobile Computing and Networking (MobiCom'00), August 2000, pp. 255 – 265.

[9] Q. He, D. Wu, and P. Khosla, "Sori: A secure and objective reputation based incentive scheme for ad- hoc networks," in WCNC 2004, 2004.

[10] S. Buchegger and J. L. Boudec, "Performance analysis of the confidant protocol: (cooperative of nodes - fairness in dynamic ad hoc networks)," in Proc. IEEE/ACM Workshop on Mobile Ad Hoc Networking and Computing (MobiHoc'02), June 2002, pp. 226.

[11] Khairul Azmi Abu Bakar and James Irvine "A Scheme for Detecting Selfish Nodes in MANETs using OMNET++"2010, Sixth International Conference on Wireless and Mobile Communications.

[12] J. Broch, D. B. Johnson, and D. A. Maltz, "The dynamic source routing protocol for mobile ad hoc network," in IETF, February 2003, internet Draft Version 08.

[13] C. E. Perkins, E. M. Royer, and S. R. Das, "Ad hoc on-demand distance vector (AODV) routing (rfc3561)," in The Internet Society, 2003, memo RFC 3561.

[14] E. M. Royer and C. K. Toh, "A review of current routing protocols for ad- hoc mobile wireless network" IEEE Personal Communications, vol. 6, pp. 46–55, Apr 1999.

[15] H. Yang, H. Y. Luo, F. Ye, S. W. Lu, and L. Zhang, "Security in mobile ad hoc networks: Challenges and solutions," IEEE Wireless Communications, vol. 11, pp. 38–47, 2004.

[16] Bansal S, Baker M. (2003). "Observation-based cooperation enforcement in ad hoc networks", in Technical Paper on Network and Internet Architecture (cs.NI / 0307012).

[17] L. Huang, L. Li, L. Liu, H. Zhang, and L. Tang, "Stimulating cooperation in route discovery of ad hoc networks ",in Proceedings of the 3rd ACM Workshop on (Q2SWinet'07), October 2007.

[18] K. Paul and D. Westhoff, "Context aware detection of selfish nodes in dsr based ad-hoc networks," in proceedings of IEEE Vehicular Technology Conference 02, 2002.

[19] D. Djenouri, O. Mahmoudi, M. Bouamama, D. Llewellyn-Jones, and M. Merabti, "On securing manet routing protocol against control packet dropping," in The 4th IEEE (ICPS'2007), Istanbul, July 2007, pp. 100–108.

