A Novel Incentive Based Scheme to Contain Selective Forwarding in Wireless Sensor Network

Saswati Mukherjee, Matangini Chattopadhyay, Samiran Chattopadhyay, Debarshi Kumar Sanyal, Roshni Neogy, Samanwita Pal

A Novel Incentive Based Scheme to Contain Selective Forwarding in Wireless Sensor Network

Saswati Mukherjee[1], Matangini Chattopadhyay[1], Samiran Chattopadhyay[2], Debarshi Kumar Sanyal[3], Roshni Neogy[2], Samanwita Pal[2]

[1]School of Education Technology, Jadavpur University, India
[2]Department of Information Technology, Jadavpur University, India
[3]Xilinx India Technology Services Pvt. Ltd., Hyderabad
sash_cal@rediffmail.com, matanginic@gmail.com,
samirancju@gmail.com,debarsh@xilinx.com

Abstract. Selective forwarding or dropping of packets is a serious threat to multi hop communication in a Wireless Sensor Network (WSN). There are various schemes to induce cooperation in a WSN to overcome this problem. In this paper, we have introduced a novel adversary model and have proposed an incentive based scheme to inspire cooperation among nodes in a WSN. The scheme has been formally analyzed. The efficacy of the scheme is also established through various simulation experiments.

Keywords: Wireless Sensor Network, Selective Forwarding, Random Graphs, Virtual Currency, Throughput, Network Security

1 Introduction

Nodes in mobile ad-hoc networks are arbitrarily deployed without relying on any fixed network infrastructure. In a multi-hop wireless network, many pairs of nodes cannot communicate directly and must forward data to each other via one or more intermediate forwarding nodes.

Multi-hop communication is not an issue where nodes faithfully forward packets according to a global algorithm. Selfish nodes may like to send their own packets but may not be ready to relay packets for others since relaying packets for others consumes bandwidth and energy. This, in turn, decreases both individual and system throughput and might even lead to loss of connectivity in a network.

Hence, cooperation among the nodes needs to be enforced. The basic aim of any such mechanism is to force nodes to forward packets sent to it by other nodes. There are many proposed solutions [1-2] which use game theoretic and graph theoretic notions to examine whether cooperation can exist in multi-hop communication while many solutions are proposed based on providing incentives. Incentives can be positive or negative. That is, a node can be made to cooperate within a network either by providing some incentive or by taking punitive actions against a node when its rate of packet forwarding falls below a particular value. Marti et al. [4] have discussed

K. Saeed et al. (Eds.): CISIM 2013. LNCS 8104, p. 294-305, 2013.

schemes to identify misbehaving nodes (non-forwarders) and deflect traffic around them. Michiardi and Molva [5] have devised reputation mechanisms where nodes observe the behavior of others and prepare reputation reports. Zhong et al. [6] proposed the use of currencies to enforce cooperation. Buttyan and Hubaux [7, 8] devised a scheme based on a virtual currency called a *nuglet* that a node pays to send its own packets but receives, if it forwards other's packets.

In all these papers, nodes are classified in two categories: trusted those who forward packets and malicious, those who do not like to forward others' packets. Moreover, malicious nodes, according to these papers are content by dropping packets to conserve their resources. In this paper, we have introduced two further dimensions to this misbehavior model. First, we introduced a 'rational adversary' category of nodes. 'Rational adversary nodes' do not mind dropping packets if they are not penalized for that. Second, we have incorporated an idea by which 'malicious' nodes inspire their neighboring nodes to drop packets.

In this paper, we have also proposed design of a point based credit scheme in a fairly dense sensor network that encourages nodes to cooperate in packet forwarding and thereby contains selective forwarding and restores throughput of the network. In the proposed scheme, nodes that forward packets get incentives in the form of credit points. Nodes that drop packets to conserve resources are penalized by deducting credit points from them. Malicious nodes may send 'bribe' packets to neighbouring nodes to inspire them to drop packets. If a 'bribe' packet reaches a trusted node, the sender node is stripped of a good number of credit points. Credit point reserve of any node may not go below a threshold limit. The network is considered as an undirected graph with flat/unstructured topology. In flat topology each sensor node performs similar functions like packet forwarding data sensing with some exceptions in their behaviour as defined by the adversary model.

We have shown that if the graph is sufficiently dense then the likelihood of detecting malicious nodes is very high. We have also proposed an algorithm which helps to use the proposed scheme when the network is less dense. If the proposed scheme is allowed to run then simulation results suggest that the throughput of the network remains considerably high as long as the numbers of rational adversary nodes are less in proportion. Message overhead is low with small number of trusted and rational adversary node despite having large number of malicious nodes. The proposed scheme achieves scalability even if the number of nodes gets increased. Although the basic approach assumes a dense network, we have also presented a mechanism by which our scheme continues to work in less dense networks.

The remainder of this paper is organized as follows. Section 2 presents the related research work. In Section 3, the model definitions of the proposed work are described. Section 4 presents algorithmic view of the proposed scheme. In this section, we have also analyzed the scheme analytically. In Section 5, experimental results are presented to measure the performance of our proposed method. Finally, we draw our conclusion.

2 Related Work

The work to enforce cooperation in mobile and wireless ad hoc network is based on providing incentives while other works are reputation and trust based. Some others are formalized in game theoretic framework. In this section, we review some important reputation based and incentive based schemes.

CORE [5], CONFIDANT [5], OCEAN [9] are examples of reputation and trust based systems. In CORE and CONFIDANT systems, non-cooperative nodes are detected by using certain reputation measures. In CONFIDANT, the reputation of non-cooperative nodes is propagated throughout the network for punishment. In CORE, nodes with bad reputation are gradually removed from the network. CORE consists of two basic components: a watchdog mechanism and a reputation table. The watchdog mechanism is used to detect misbehavior nodes but the disadvantage is that it not only creates a performance bottleneck by increasing network congestion, transmission overhead etc. but also diminishes the network scalability. OCEAN, another trust based method, has five components to detect non-cooperative nodes and mitigate the risk of selective forwarding. The overhead to calculate trust or reputation values is a common severe problem of these methods.

Buttyan and Hubaux [7-8] proposed the concept of providing incentives to the nodes in a static wireless ad hoc network so that they faithfully forward packets. According to this scheme, nodes get paid for sending packets to other nodes. The 'money' used in this scheme is termed as *Nuglets*. The scheme is implemented using two models: The *Packet Purse Model* and *Packet Trade Model*. In *Packet Purse Model*, the originator of the packet pays for the packet forwarding service. The originator loads it with the number of *nuglets* sufficient to reach the destination. Each forwarding node acquires one or several *nuglets* from the packet and thus, increases its stock of *nuglets*. If the packet does not have enough *nuglets* to be forwarded, the packet is discarded. The problem with this model is that estimating the number of *nuglets* to be loaded with the packet is difficult. In *Packet Trade Model*, the packet is traded for *nuglets* by intermediate nodes. Each intermediary buys it from previous one for some *nuglets* and sells it to the next one for more *nuglets*. A basic disadvantage of this model is that packet flooding is possible. This scheme requires tamper proof hardware for reliably calculating the *nuglet* distribution.

Sheng Zhong et al. [6] proposed a scheme called *Sprite,* a simple cheat-proof credit based system. In *Sprite,* a Credit Clearance Service (CCS) is introduced to determine the charge and credit to each node involved in message transmission. When a node receives a message, the node keeps a receipt of the message and later reports it to the CCS. This scheme requires a central server to determine the charge and credit to each node involved in message transmission.

Salem et al. [10] proposed another incentive mechanism based on charging and rewarding scheme which forces selfish nodes to rationally opt for forwarding packets. Their proposal provides a set of protocols that rely on symmetric cryptography techniques.

In [11], Jackobson and others have proposed a micro-payment scheme for multi-hop cellular networks that encourages collaboration in packet forwarding. In this paper, an asymmetric communication model is assumed.

The scheme presented in this paper uses the basic concept of incentives as discussed in [6-8]. But our paper differs from them in many ways. First, our paper introduces a new adversary model in a fairly dense static wireless sensor network which is richer than the one proposed in all these papers. Second, our scheme is distributed with minimal overhead. Third, simulation results are provided to show that the scheme is scalable and effective.

3 Model Definitions

In this section, we give a clear definition of the proposed models and the reasons why we have chosen this model. Since we are studying cooperation in packet forwarding, we assume that the main reason for packet losses in the network is the non-cooperative behavior of the nodes.

3.1 Adversarial Model

We have identified primarily two types of non-cooperative nodes: faulty or malicious or misbehaved and selfish or rational adversary. Faulty/malicious activity refers to that class of misbehavior where nodes try to attack the system. They threaten the entire network by dropping packets in order to conserve their resources and by inspiring neighbors not to forward packets. As our scheme is based on credit point system where points are to be acquired by every node, the sanctity of the credit points in each node is essential. The misbehaved nodes are assumed to be capable of manipulating the credits available in the non-trusted neighbors around them. This is modeled by the malicious nodes being able to send “bribe” packets to their neighbors. A bribe packet offers certain points to its receivers. By receiving a bribe packet, a selfish node may drop some packets without jeopardizing its stock of points.

The adversary is rational, in the sense that it will only attempt to cheat if the expected benefit of doing so is greater than the expected benefit of acting honestly in network related operations. Naturally, if a rational adversary node is offered a “bribe” packet to increase its credit point, it will actually accept it as long as there is no harm in accepting it.

The third category of nodes is termed trustworthy or trusted. They refer to the class of well-behaved nodes that functions reliably and honestly throughout the network operations.

3.2 Network Model

We have considered a wireless network containing *N* nodes and the nodes are divided in three categories as explained in the Adversary model. The topology of the wireless sensor network is basically an undirected graph where an edge between two nodes denotes that they can communicate with each other. The topology is flat as these nodes are homogeneous except in their capability defined by the adversary model.

The network is modeled as an “Uncorrelated random graph (Erdős–Rényi)” [12] where *N* nodes are connected through *n* edges which are chosen randomly from the *N(N-1)/2* possible configurations. The probability of selecting an edge is *p*.

3.3 Point Based Credit Scheme

A Point Based credit mechanism of charging/rewarding the service (which is forwarding a packet in this case) is presented to stimulate node cooperation in wireless ad-hoc networks. In this section, we give a formal description of proposed Point Based credit scheme.

- All the nodes in the network are initialized with some credit points.
- Nodes earn credit points as rewards if they forward packets.
- Nodes lose credit point as penalty if they do not forward packets.
- Credit points assigned to all the nodes must not fall below a specified threshold value.
- Trustworthy nodes always forward packets and earn points and hence they always help in increasing the network throughput.
- Bribe packets are offered by the malicious/misbehaving nodes to random neighboring nodes inspiring them to drop packets in order to pull down the network efficiency. Every bribe packet contains a certain number of points. These points are deducted from the stock of points of the malicious nodes.
- The rational adversary nodes on receiving the bribe packets check if its accumulated points are above the threshold value. If so, then they do not forward packets so long as points accumulated by accepting bribe packets are more than points lost due to not forwarding packets. Otherwise, they forward packets and earn credit points as rewards.
- A malicious/misbehaving node forwards packet if it’s accumulated credit point falls below the threshold.
- Trustworthy nodes on receiving bribe packets from malicious nodes penalize them by deducting points.

The execution of the proposed charging/rewarding scheme requires tamper proof hardware [7-8] to monitor the addition or deduction of points assigned to the nodes or requires a central server to assess the charge involved or credit points of each node for message transmission. Thus, we do not assume any MAC layer misbehavior.

4 Algorithmic Description of the Proposed Scheme

The proposed scheme has two phases: Network Deployment and Charging/Rewarding Service.

```
  Network Deployment Phase
Step 1:   Input the number of nodes for each class of nodes.
Step 2:   Assign equal credit points E to all the nodes.
Step 3:  Set threshold limit T of credit point to all the nodes.
```

```
Algorithm PACKET_FORWARDING_SCHEME
For (;;) {
    Step A.     The MALICIOUS nodes send out a number of "bribe" packets depending on
                the amount of its stock of points to randomly selected neighbouring nodes.
   On receiving a packet
    Step B.1:     If the node is malicious,
    Step B.1a:        Check its credit points.
    Step B.1b:        If credit point of malicious node is greater than threshold limit,
    Step B.1c:              Then drop the packet.
    Step B.1d:              Else forward the packet and earn C points.
    Step B.2:     Else If the node is trusted
    Step B.2a:        If the type of the packet is "bribe packet"
    Step B.2b:              Then deduct P points from the sender malicious node by send-
                            ing punishment packet
    Step B.2c:        Else forward the packet and earn C points.
    Step B.3:     Else If the node is rational adversary
    Step B.3a:        If the type of the packet is "bribe packet" with B point
    Step B.3b:              Then Accumulated_Bribe += B
    Step B.3c         If C <= Accumulated_Bribe_Points
    Step B.3c:              Then drop the packet; Accumulated_Bribe -= C
    Step B.3d:              Else, forward the packet and earn C points.
}
```

In the *for* loop, Step A and Step B run in parallel.

4.1 Analysis of Algorithm

For a random graph with n number of nodes, suppose that the probability of choosing an edge is *p*. That is, *p*, fraction of the total number of edges is selected randomly in such a graph. It is known that in such a graph, as n tends to infinity, the probability that a graph of *n* vertices with edge probability $p = 2ln\ (n)/n$ is connected, tends to 1.

It is also known that the average degree of a node in such random graph is *np* with variance npq where $q = 1- p$.

In our case, $n = (n1+n2\ +n3)$ where *n1*, *n2*, *n3* denote the number of malicious nodes, rational adversary nodes and trustworthy nodes respectively. Thus, the average number of edges from any node to a trusted node is n_3p. However, this is the mean value. The connectivity of any node to a trusted node may decrease from the mean value by the square root of the variance. Thus, the following inequality must hold to maintain connectivity to a trusted node from any given node.

$$n_3 p > \sqrt{npq} \quad (1)$$

Solving the inequality, we get,

$$p/(1-p) > n/n_3^2 \quad (2)$$

Substituting n/n_3^2 with k, we get,

$$p > k/(1+k) \tag{3}$$

Therefore, if the fraction of edges in the network is greater than $k/(1+k)$ then it is likely that a malicious node is always connected to some trusted node. In such a scenario, the bribe packet randomly sent out by a malicious node would reach the trusted node some time and the malicious node will surely be penalized.

4.2 Finding positions of additional trusted nodes

The proposed scheme heavily depends on a misbehaved node being connected with at least one trusted node in its neighborhood. So, the network becomes dense. In a given deployment, it may so happen that each malicious node does not have a trusted node as its neighbor. In such situations, we can adopt the following scheme to deploy additional trusted nodes if necessary so that every malicious node is ensured to be connected to a trusted node. Our approach is based on Minimum Connected Dominating Set of the network.

We have made some additional assumptions to handle the case where each malicious node is not connected directly to a trusted node. These assumptions are as follows. The trusted nodes are assumed to be deployed with a key for secured communication. All nodes are assumed to be equipped with location information. Location information of a node denotes information about its spatial coordinates in a given area.

The following algorithm determines the locations where additional trusted nodes should be deployed so that direct edge connectivity between a malicious node and a trusted node is ensured.

If the graph contains n nodes then Step 1, Step 2 can be performed in O(n) [13] time. If the approximate MCDS of G contains h number of nodes then Step 3a, Step 4a can be performed in O(h) time [14]. Step 3b.and Step 4b takes $O(h^2)$ time. Step 5 can be performed in O(h) time.

```
Algorithm FindPositionsOfAdditionalTrustedNodes
Input: The Graph G describing the topology of the wireless sensor network
Output: Locations where additional trusted nodes should be deployed
Step 1:     Let T = set of trusted nodes. Let u be a designated trusted node.
            // These nodes have a key for secured communication.
Step 2:     Find an approximate Minimum Connected Dominating Set (MCDS) C of
            G starting with T using a standard greedy algorithm in the literature.
Step 3a:    u broadcasts a message asking for the location information from each
            node in C.
Step 3b:    u receives the location information from all nodes.
Step 4:     u broadcasts a challenge message to all nodes in C
            u receives responses only from the trusted nodes.
Step 5:     u finds out the locations of the nodes in C which are not trusted and writes
            them in P
Step 6:     return P
```

5 Experimental Analysis

A simulation experiment is performed with a by considering a number of random graphs having 100 wireless nodes and 1000 edges. In a given topology, there would be many different possibilities of mixes of trustworthy nodes, misbehaved nodes and rational adversary nodes. In order to model these varieties, we have chosen ten sets of different ratios of these three classes of wireless nodes as shown in Figure 1.

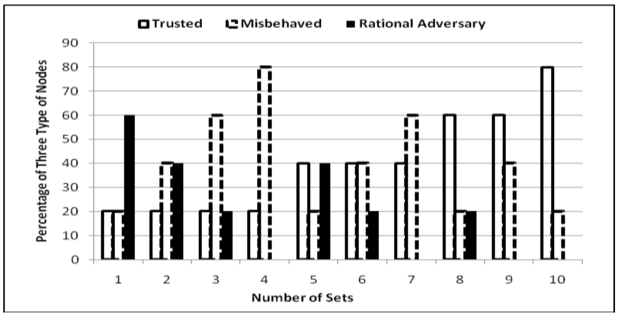

Fig. 1. Different sets of nodes with various proportions of trusted, misbehaved and rational adversary nodes in the Network.

The rational behind the choice of different sets is as follows. Suppose we fix a small number of trusted nodes, say 20, in the network. Keeping this number fixed, we can choose several numbers of malicious nodes. Suppose that, the number of malicious nodes is chosen to be 20, 40, 60 and 80. Once the number of trusted nodes and malicious nodes are fixed, the number of rational adversary nodes can be computed. Next, we increase the number of trusted nodes to 40 and continue the same process. For every set, a number of random graphs are generated in our experiment.

The PACKET_FORWARDING_SCHEME algorithm is executed in terms of rounds where each round signifies time duration. There are several parameters in the algorithm. In our experiments, parameters of the algorithm are assigned with the following values.

- Each node is initialized with 15 credit points.
- Each node earns 1 credit point as reward if it forwards a packet.
- Each node is penalized by 1.5 credit point if it drops a packet.
- *Misbehaved* nodes are penalized with 2 points by the trusted node, if the trusted node receives a bribe packet from the misbehaved node.
- Offer in the “BRIBE” is set at 0.1 percent of the points owned by the malicious nodes sending the bribe packet
- Threshold credit point of a node is 7.5.

In the experiments *throughput* is defined as the number of packets dropped to number of packets generated. Detection ratio is defined as the ratio of the number of

MALICIOUS nodes detected to the actual number of MALICIOUS nodes in the network.

- Throughput measurement in the face of selective forwarding

In Figure 2, throughput is plotted against different proportion of *trusted*, *misbehaved* and *rational adversary* nodes. Recall that ten different compositions of trusted, rational adversary and misbehaved nodes are considered as shown in Figure 1.

We can see that the throughput drops for Set 1 and Set 5. We also note that this is because in both these sets, the number of *rational adversary* nodes is considerable. As the number of *rational adversary* nodes is significant, *misbehaved* nodes can offer many “bribe” packets to them. These bribe packets induces *rational adversary* nodes not to forward packets resulting in a drop in throughput.

In the presence of less number of rational adversary nodes, throughput is improved even if there are a small number of *trusted nodes* compared to the number of *malicious nodes*. This is observed for Set 4, Set 7, Set 9, and Set 10.

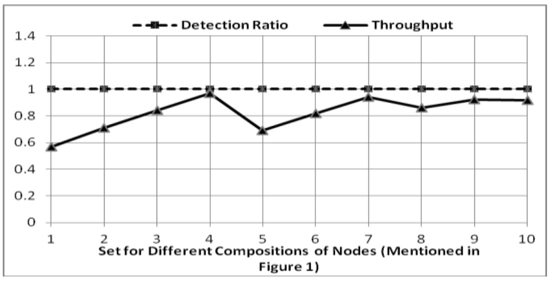

Fig. 2. Throughput of 10 sets of sensor nodes having different proportions of trusted, rational adversary, and malicious nodes

Figure 3 demonstrate that throughput remains almost unchanged once all malicious nodes are detected and they are routinely penalized. In Figure 3, we have chosen Set 5 and plotted throghput with respect to the number of rounds. It can be seen that after 10 rounds all malicious nodes are detected and throughput also remains unchanged at around 0.7 thereafter.

Similar trends have also been observed for Set 4 and Set 7.

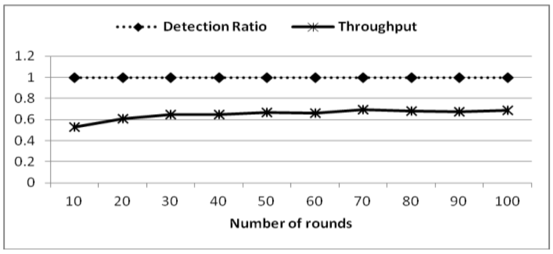

Fig. 3. Throughput after increasing number of rounds for Set 5

– Communication Overhead and 0.Scalability

Figure 4 depicts the communication overhead incurred in the algorithm. Overhead is measured by number of “bribe” packets and “penalty” packets exchanged. The number of these messages is averaged over 100 rounds.

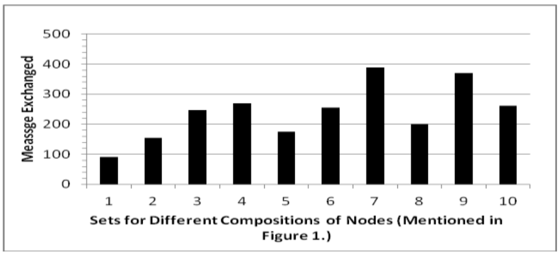

Fig. 4. Message Exchange overhead for all sets of combination of 3 types of node

Figure 5 shows the relationship between the number of rounds with the number of nodes to detect all malicious nodes. We have chosen three combinations of three types of nodes as defined by Set 1, Set 2 and Set 3. It can be seen that the number of rounds to detect all malicious nodes remains almost the same even if the number of nodes is increased.

Figure 6 plots average number of messages exchanged per node when number of nodes increase for various compositions of 3 types of nodes. The average numbers of message exchanges are not growing rapidly with respect to the number of nodes. Thus, it can be claimed that the scheme is scalable.

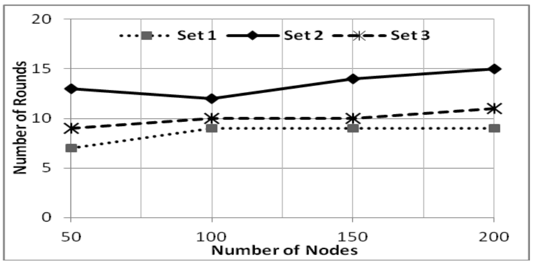

Fig. 5. Number of rounds required for 100% detection of misbehaved nodes with various number of nodes

– Effect of rational adversary nodes

In Figure 7, we have tried to capture the effect of the number of rational adversary nodes keeping the number of *trusted* nodes to 20. The number of misbehaved nodes is changed. Accordingly, the number of rational adversary nodes increase as the total numbser of nodes is kept at 100. It is observed that throughput steadily increases as the number of rational adversary nodes decrease (and the number of *misbehaved* nodes increase).

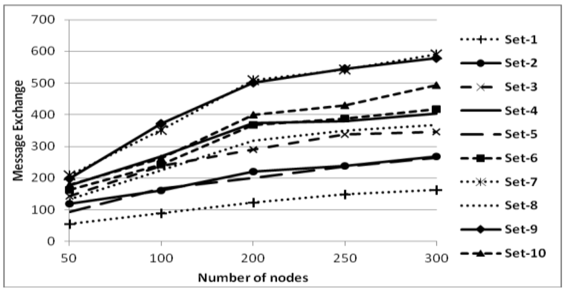

Fig. 6. Average number of messages exchanged for all sets having different compositions of nodes (as mentioned in Figure 1)

We can also verify the same from Figure 2. You may recall that the proportion of rational adversary nodes is more in the Sets 1, 2, 5, 3, 6, 8 in a non-increasing order. Throughput values for these sets in the graph reflect that the presence of rational adversary nodes bring throughput down.

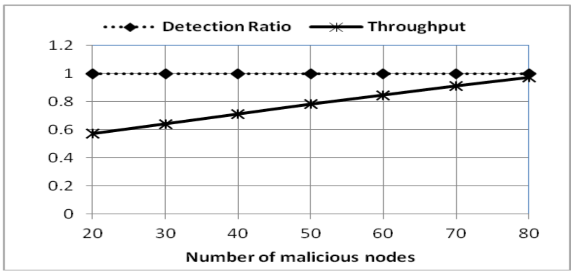

Fig. 7. Effect of rational adversaries with respect to throughput

6 Conclusion

In the paper, we have introduced a new adversary model for Wireless Sensor Networks. In this model, the idea of rational adversary nodes is floated in addition to trusted and malicious nodes. An incentive based scheme is presented to induce cooperation among nodes to solve the problem of selective forwarding. The scheme is analyzed and studied through simulation experiments.

The incorporation of rational adversary nodes introduces new twists to the problem. As a future work, we need to study further the effect of such nodes in throughput. We also plan to invent a trace algorithm which can detect malicious nodes even when it is not directly connected to some trusted node.

Acknowledgements. The work proposed is supported in part by UGC-UPE Phase II program of Jadavpur University.

References

1. Felegyhazi, M., Hubaux, J.P., Buttyan, L.: Nash equilibria of packet forwarding strategies in wireless ad hoc networks. In: IEEE Transactions on Mobile Computing. 5(5), 463-476 (2006)
2. Mukherjee, S., Dey, S., Mukherjee, R., Chattopadhyay, M., Chattopadhyay, S., Sanyal, D K.: Addressing Forwarder's Dilemma: A Game-Theoretic Approach to Induce Cooperation in a Multi-Hop Wireless Network. In: Das, Vinu V., Stephen, J. (eds.) LNICST, vol. 108, pp. 93 – 98. Springer, Heidelberg (2012)
3. Marti, S., Guili, T.J., Lai, K., Baker, M.: Mitigating Routing Misbehavior in Mobile Ad Hoc Networks. In: 6th annual international conference on Mobile computing and networking, pp. 255-265. ACM, New York (2000)
4. Michiardi, P., Molva, R.: CORE: A COllaborative REputation mechanism to enforce node cooperation in mobile ad hoc networks. In: IFIP TC6/TC11 Sixth Joint Working Conference on Communications and Multimedia Security, pp. 107-121. ACM, New York (2002)
5. Zhong, S., Yang, Y. R., J. Chen.: Sprite: A simple, cheat-proof, credit-based system for mobile ad hoc networks. In: Twenty-Second Annual Joint Conference of the IEEE Computer and Communications, INFOCOM, IEEE Press, New York (2003)
6. Buttyan, L., Hubaux, J. P.: Enforcing service availability in mobile ad-hoc WANs. In: MobiHOC, pp. 87-96. IEEE Press, New York (2000)
7. Buttyaan, L., Hubaux, J. P.: Nuglets: a virtual currency to stimulate cooperation in self-organized mobile ad hoc networks. Technical Report, DSC/2001/001, Department of Communication Systems, Swiss Federal Institute of Technology (2001)
8. Bansal, S., Baker, M.: Observation-based Cooperation Enforcement in Ad Hoc Net-works. 2003, http://arxiv.org/pdf/cs/0307012.pdf
9. Salem, N.B., Buttyan, L., Hubaux, J.P., Jakobsson, M.: A charging and rewarding scheme for packet forwarding in multi-hop cellular networks. In: 4th ACM International Symposium on Mobile ad hoc networking & computing, pp. 13-24. IEEE Press, New York (2003)
10. Jakobsson, M., Hubaux, J. P., Buttyan, L.: A micropayment scheme encouraging collaboration in multi-hop cellular networks. In: Wright, Rebecca. N. (ed.) LNCS, vol. 2742, pp. 15 – 33. Springer, Heidelberg (2003)
11. Erdős, P., Rényi, A.: On the Evolution of Random Graphs. In: Publication of the Mathematical Institute of The Hungarian Academy Of Sciences, pp. 17-61 (1960)
12. Das, B., Bharghavan, V.: Routing in Ad-hoc Networks Using Minimum Connected Dominating Sets. In. IEEE International Conference on Communications, pp. 376-380. IEEE Press, New York (1997)
13. Sinha, P., Shivkumar, R., Bharghavan, V.: MCEDER: Multicast Core-Extraction Distributed Ad-hoc Routing. In: IEEE Wireless Communications and Networking Conference, pp. 1313-1317. IEEE Press, New York (1999)
14. Pandana, C., Han, Z., Liu, K. J. R.: Cooperation enforcement and learning for optimizing packet forwarding in autonomous wireless networks. Proceedings of the IEEE Transactions on Wireless Communications. 7(8), 3150-3163 (2008)
15. Crosby, G.V., Pissinou, N.: Evolution of Cooperation in Multi-Class Wireless Sensor Networks. In: 32nd IEEE Conference on Local Computer Networks, pp.489-495. IEEE Press, New York (2007)
16. Michiardi, P., Molva, R.: Prevention of Denial of Service attacks and Selfishness in Mobile Adhoc Networks. In: Mobile Ad Hoc Networks, Institute Eurecom Research Report (2002)