Cross-layer Design to Solve Cooperation Problem in Ad Hoc Network

Li Li, Liu Wei, Chen Hai-yan, Zhou Guo-zhu
Department of Telecommunication
Beijing Polytechic
Beijing, People's Republic of China
buedc@163.com,bkylw@126.com

***Abstract*—A wireless ad hoc network is a self-configuring infrastructurelessh network of mobile devices connected by wireless. In order to support the basal networking functions like routing and packet forwarding, the nodes in wireless ad hoc network should be cooperate with each other due to the limited range of wireless transmission. However, nodes are unwilling to relay packets for others considering the poor energy and bandwidth especially in the open community environment. Some approaches to stimulate nodes cooperation have been introduced recently. In this paper, we provide a novel method based on cross-layer design to solve the cooperation issues both in Mac and network layer. The method with the help of reputation mechanism and utility function based on game theoretical approach can promote nodes to cooperate and make the whole network to reach Nash equilibrium.**

Keywords-Ad Hoc; Cooperation; Cross-layer; Routing

I. INTRODUCTION

The application of wireless ad hoc network for the support of open communities has emerged in recent years. In such environment, different users, which have the different goals, share the resources of their devices, ensuring global connectivity. There is no good reason to believe that all nodes will eventually cooperate, since network operation consumes energy and a particularly scarce resource for battery power and bandwidth in ad hoc network.

The lack of cooperation between the nodes has become the important problem in wireless ad hoc network. The nodes that behave uncooperatively are called selfishness. Generally speaking, the selfish node does not deliberately to damage other nodes by causing network partitioning or disrupting routing information but it simply does not adhere to the basic network functioning, saving battery life for its own communications. Simulation based studies [1, 2] show that evens a small percentage (10-40%) of non-cooperating nodes can bring the network throughput considerably drop down (16-32% degradation).

Several papers have addressed the issues of cooperation in wireless ad hoc network and some approaches are proposed to stimulate the nodes of wireless ad hoc network to forward packets for others. We have paid more attention to the current researches and come to conclusion that if a node is selfish and wants to maximize its own utilities in a self-interested way, its selfishness characteristic will show in each layer not only in network layer. Cooperation issues should be taking into account for each layer of the protocol stack, with different aims and ways of acting. In Mac layer, for example, the selfish node may fail to adhere to the back off algorithm of IEEE 802.11, with the intent of obtaining more than its fair shares of the channel bandwidth by selecting backoff values from a smaller range than [0, CW]. Such greedy behavior may seriously degrade the throughput of well-behaving nodes [11].

Motivated by [8, 9], we use a cross-layer design to solve the cooperation issues both in Mac and network layer. The method with the help of reputation mechanism and utility function based on game theoretical approach can promote nodes to cooperate and make the whole network to arrive Nash equilibrium.

The remainder of the paper is organized as follows. Section II reviews the related work. The next section outlines our approach and describes each component in detail. Thereafter in section IV gives some simulation and results. We end with conclusion and further work in section V.

II. RELATED WORK

The general solutions to stimulate cooperation at network layer can be classified in two categories: 1) Pricing based approach; and 2) Reputation based approach.

Pricing based approach**:** The key idea is that nodes providing a service should be rewarded, while nodes receiving a service should be charged. In other words, stimulate packet forwarding by means of virtual monetary. Based on this concept, [3] proposes a tamper-resistant security module which maintains a nuglet counter. The proposed protocol can be used two billing models. In the packet purse model, the sender pays to every intermediary node for the message transmission, while in the packet trade model is the receiver that is charged. Hence, if a node wants to send its own packets, it must forward packets for the benefit of others. Besides the drawback of requiring a tamper resistant security module, this solution does not face the problem of nodes running out of nuglets, because of their position on the network borders.

Reputation based approach: The key idea is that prevent misbehavior from estimating node's reputation and punishing nodes with bad behavior. The starting step for this class of approaches is given by [4]. It is based on two modules: The watchdog module, which detects malicious behavior of nodes, and the path rater module, which rates each route. With the help of these two modules, a node is able to avoid sending packets through malicious nodes in the routes.

CORE [5] is also the reputation-based mechanism to

Published by Atlantis Press, Paris, France.

enforce cooperation in wireless ad hoc networks. Like Pathrater/ Watchdog, it uses a watchdog mechanism that is placed on each node. Core uses direct observations and indirect reputations. The system will be able to combine different reputation values. Only positive rating factors of other nodes are used to prevent the distribution of false information. To stimulate cooperation in wireless ad hoc network, any request or data packet of a misbehaving node is rejected.

Researchers have also used game theory to analyze of cooperation of nodes in wireless ad hoc networks [6]. Game theory is a powerful tool for modeling interactions between self-interested users and predicting their choice of strategy All these papers define the game as an algorithm that dictates the behavior of a node at a given point in time as a response to the behavior of other nodes. The authors of [7] provide the Generous TIT-FOR-TAT (GTFT) algorithm. GTFT is a distributed and scalable method, which decides whether to accept or reject a packet-forwarding request. The authors use elementary game theory to prove that there is an optimal tradeoff between throughput and lifetime. The results of GTFT algorithm can reach Nash Equilibrium. Their mainly work proposes a mathematical framework only, but does not present any implemented system.

III. Proposed Method

The method we proposed has two components. One is the modified backoff algorithm of IEEE 802.11 DCF in Mac layer similar to [8] and the other is use reputation scheme to enforce all the nodes to cooperate and to reach Nash equilibrium. The node reputation value is decided not only by its cooperative characteristic in network layer but also by its selfish behavior in Mac layer. Our method takes into account for both Mac layer and network layer, which is superior to other schemes.

A. Modifications to the backoff algorithm

In order to detect node misbehavior in MAC layer, we enable a receiver to identify sender misbehavior within a small observation interval. Instead of the sender choosing random backoff values to initialize the backoff time, the receiver assigns a random backoff value though the CTS (Clear to Send) and ACK packets to sender. The sender uses this assigned backoff value in the next transmission to the receiver. Whether the sender performs the assigned backoff time or not can be observed by the receiver. If this observed number of idle slots is less than the assigned backoff time, the sender will be detected misbehavior with high probability. Details are explained below.

We firstly consider the simple condition without retransmission and collision. At first, a sender A sends a packet to a receiver B. When B receives a RTS (Ready to Send) from A, as show in Fig. 1, B will choose the backoff value T_d=t1 and assign it to A in the CTS packet as well as the subsequent ACK packet. A is requested to use this backoff value t1 for sending the next packet to B. If A is a selfish node and wants to obtain more than its fair share of the channel, it will be backoff for a smaller duration (T_s) than T_d. The receiver can find shorten of the backoff value on the channel through the interval between the sending of an ACK by B and the reception of the next RTS from A. The sender is designated as selfish node if the observed number of idle slots T_s is less than a specified fraction αof the assigned value T_d.

$$T_s < \alpha \times T_d \quad 0 < \alpha < 1 \tag{1}$$

The parameter αin equation (1) can be suitably chosen, based on the channel conditions, to reduce the incidence of false deviations [8]. In the simulation, we choose αto be 0.85.

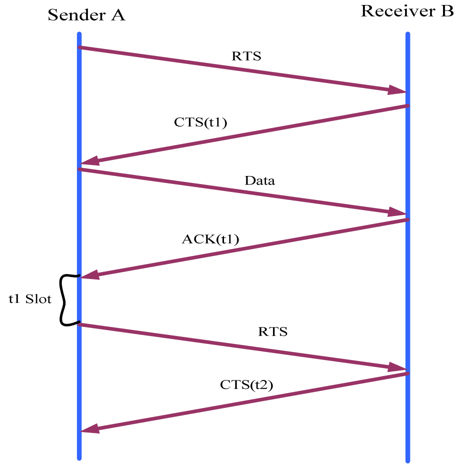

Figure 1. Receiver-Sender interaction

Next, we take into account the second situation where a RTS is unsuccessful to send to receiver or the collision happens. The sender will select a new backoff using a deterministic function that is known by the receiver:

$$Newbackoff = F(Backoff, nodeID, n) \times cw \tag{2}$$

As show in equation (2), the parameter backoff is previously assigned by the receiver, n stands for the number of retransmission attempt and nodeID is the sender's identifier. The function F used by the sender for computing backoff values for retransmission attempt is given by:

$$F = (c_1 x + c_2) \bmod (cw_{\min} + 1) \tag{3}$$

Where $x = (backoff + nodeID) \bmod (cw_{\min} + 1)$ and $c_1 = 5, c_2 = 2n + 1$.The effect of equation (3) is to generate a uniform random number between $[0, cw_{\min}]$ and divide the required number between 0 and 1 by $cw_{\min}$. The function can ensure that the colliding senders will select the different backoff value with high probability after collisions happen.

When the sender uses a deterministic function F to compute the new backoff value t from a larger range after a collision, the receiver on reception of a packet from the sender can also calculate this backoff value t by applying the same deterministic function and easily to know whether the sender is well-behaved node. According to the attempt number in received RTS, the receiver will estimate the total time, T_d, for which the sender was expected to backoff, using function F as:

$$T_d = backoff + \sum_{i=2}^{n} F(backoff, nodeID, i) \times cw_i \tag{4}$$

where $cw_i = \min((cw_{\min}+1)\times 2^{i-1}-1, cw_{\max})$ is the contention window for the ith transmission attempt. The estimated backoff T_d is used to check for the difference compare with the actual backoff values T_s, by using equation (1) as explained before.

With these modifications to backoff algorithm, a receiver can know more about the behavior characteristic of a sender. According to the number idle slots of observed deviations over a small history of received packets, the sender will be considered as selfish node and its reputation value will be dropped by the receiver as punishment.

B. Reputation Scheme

In the initialization of network, every node has same reputation value R and stores in reputation table. The value of node reputation will decrease or increase according to its behavior during communication process. The changes of reputation value are mainly decided by node's behavior in MAC layer as describe above. Receiver monitoring is used to collect information about the change of reputation value. We use two parameters M and T to define the change of node's reputation. The receiver maintains a moving window containing the information about the last M packets received from each sender. When a new packet is received, the difference T_d-T_s is stored in the moving window. A positive (negative) difference indicates that the sender has waited for less (more) than the backoff duration expected by the receiver. If the sum of these differences in the previous M packets from the sender is greater than a threshold T, then the receiver will reduce the sender's reputation value. On the country, if the sum of T_d-T_s is below T and the reputation value will increase. Different reputation values can reflect different behaviors. In order to make the whole network share the reputation information of other nodes, the receiver nodes will broadcast the update messages of reputation value and its neighbor nodes that received this message will response for reputation propagation. When other node in the network receives the broadcast message, it will update its own reputation table.

When a node's reputation falls down a predefined threshold, service provision to the selfish node is interrupted. For example, during the build up of the route path, the source node will check for the reputation value table and avoid the nodes with lower reputation value to participate in routing. On the other hand, if the source node is the lower reputation node, its neighbor nodes won't accept its routing request, while the higher reputation value node will have priority to transmit data. The only method for lower reputation value nodes is to change their selfish behaviors.

In order to enforce all nodes to participant the packet forwarding function, we build a utility function based on game theoretical approach [10]. We assume that every node is the player of game theory, which is rational to maximize its own utilities in a self-interested way. The utility function, which stands for available network resources, is used to model the selfishness problem, which considers the relationship between the energy that a node dedicates for its own communications and the energy that the node contributes to participate in the routing protocol and forward data packets on behalf of other nodes. Node behavior is represented as the energy distributed for different purpose.

$$U_i(S_i) = E_{self}\times S_i - (1-S_i)\times(E_r + E_{pf})\times\frac{1}{R_i}$$

(5)

Where $S_i \in (0,1)$ corresponds to the strategy adopted by node i . E_{self} is the energy spend on own communications. E_r is the energy consumed for participating to the routing protocol and E_{pf} is the one spend on relaying packets for neighbor nodes.

If node i is the pure selfish node, it will spend all available energy on its own communications and choose $S_i = 1$. In order to prevent this kind of selfish behavior from happening, we assign the reputation value R_i to node i as an effective tool to change its strategy and restrict its selfish behaviors asymptotically.

From the equation (5), we can know that the relationship between U_i and R_i is in direct ratio. If a node wants to maximize its utility function and acquire more available network resource, the efficient method is to cooptation with other nodes in order to improve its reputation value. A selfish node may have high utility in short time, but as its selfish behavior is detected and the reputation drops down to the predefined value, it won't get any network resources until its reputation value increases. With the incorporate the advantages of utility function and reputation scheme, the selfish behaviors in MAC layer and network layer are restricted. When all of nodes adopt cooperation strategies, the best operating point from the perspective of network is to reach Nash equilibrium.

IV. Simulation And Results

In order to demonstrate the effectiveness of the proposed scheme, we use simulator tools for our simulations. There are eight nodes to compose of a small ad hoc network and the topology is fixed during simulation. The nodes randomly distribute in the area of 350m*350m.We use free space model as physical layer model and AODV as the routing protocol. The traffic from the sender to the receiver is a CBR (Constant Bit Rate) flow with rate 2 Mbps and size of CBR packets is 512 bytes. The two parameters M and T are set to five packets and 25 slots respectively. We define the coefficient C represents the amount of selfish behavior. When C equals to w means that the node uses a fixed contention window equal to $(1-w)\cdot cw_{\min}$ and chooses its backoff from this contend window. Thus, w=0 means that no selfish behavior, and w=1 shows that the node transmits without any backoff. The value of C depends on the node's strategy S. As describe in section III,every node is a rational player and tries to change its strategy for the purpose of efficient utility function during communication process. When the system arrives at Nash equilibrium, the value of C and S will not change much. In order to observe the changes of node's strategy, we randomly select two nodes to be purely selfish. The simulation time is 320 seconds. The results are averaged over 5 runs of the simulation.

We mainly analyze the performance of both MAC and network layer. As show in Fig. 2, the modified backoff algorithm can effectively identify node misbehavior and improve the network throughput compare with IEEE 802.11.It restricts the selfish node to a fair share, thereby ensuring that the throughput of other nodes are not affected, especially when the selfish coefficient C is below.

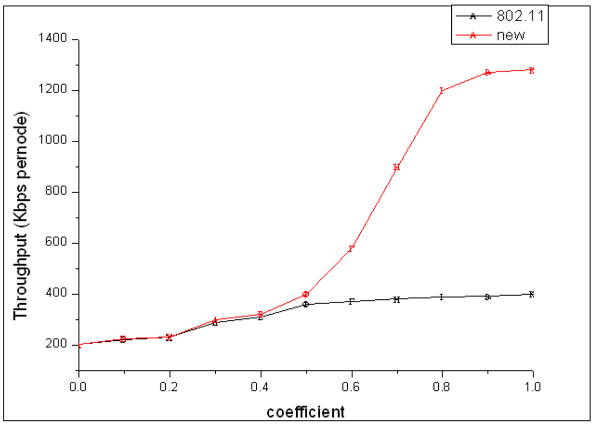

Figure 2. Throughput comparisons

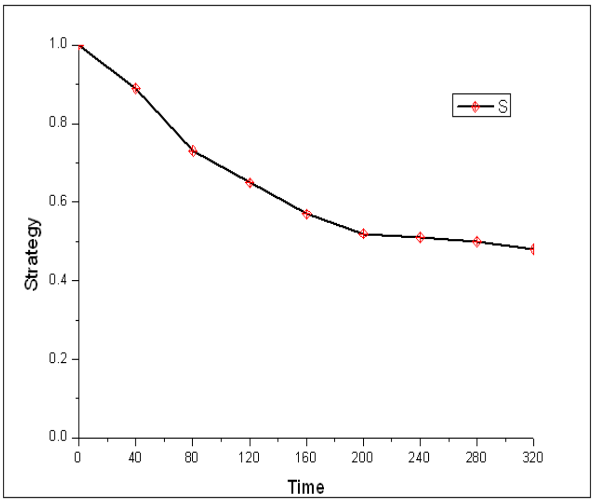

Figure 3. Node behavior tendencies

V. Conclusion

Based on reputation mechanism and game theoretical approach, we provide an effective method that combines the MAC and network layer together to detect the selfish behavior and stimulate nodes to cooperation with each other. Simulation results have indicated that our scheme can solve cooperation problem in wireless ad hoc network well.

How to optimize this method is our future work and evaluate its performance in the scenario with variety of network topologies.

References

[1] S. Marti, T. Giuli, K. Lai, and M. Baker, “Mitigating routing misbehavior in mobile ad hoc networks”, Proc. ACM MOBICOM, pp.255-265, 2000.

[2] P. Michiardi and R. Molva, “Simulation-based analysis of security exposures in mobile ad hoc networks” , Proc. European Wireless Conference, 2002.

[3] L. Buttyan and J.P. Hubaux, “Stimulating cooperation in self-organizing mobile ad hoc networks”, ACM/Kluwer Mobile Networks and Applications (MONET), Vol. 8, pp.579-592, October 2003.

[4] S. Zhong, J. Chen and Y.R. Yang, “Sprite: a simple,cheat proof, credit-based system for mobile ad-hoc networks”, Proc. IEEE INFOCOM’03, San Francisco,pp.1987-1997, March 2003.

[5] J. Crowcroft, R. Gibbens, F. Kelly, and S. Ostrring, “Modelling incentives for collaboration in mobile ad hoc networks”, Proc. WiOpt’03, Sophia-Antipolis, pp.427-439, March 2003.

[6] S. Buchegger and T.Y.Le Boudec, “Performance analysis of the CONFIDANT protocol” in Proc ACM International Symposium on Mobile Ad Hoc Networking and Computing, Lausanne, pp.226-236, June 2002.

[7] P. Michiaridi and R. Molva, “CORE: a collaborative reputation mechanism to enforce node cooperation in mobile ad hoc networks” in Pro.of Communication and Multimedia Security 2002 Conference, pp.107-121, September 2002.

[8] P. Kyasanur and N. Vaidya, “Detection and handling of MAC layer misbehavior in wireless networks”, Dependable Systems and Networks, pp.173-182, June 2003.

[9] M. Raya, J.P. Hubaux and I. Aad, “DOMINO: a system to detect greedy behavior in IEEE 802.11 hotspots”, Proc. MobiSYS’04, pp.84-97, June 6-9, 2004.

[10] P. Michiardi, and R. Molva, “A game theoretical approach to evaluate cooperation enforcement mechanisms in mobile ad hoc networks”, Proc.WiOpt 2003: Modeling and Optimization in Mobile, Ad Hoc and Wireless Networks, France, March 3-5, 2003.

[11] M. Conti, E. Gregori and G. Maselli, “Cooperation issues in mobile ad hoc networks” , Proc. 24th International Conference on Distributed Computing Systems Workshops, 2004.

Published by Atlantis Press, Paris, France.