Computer Engineering and Intelligent Systems
ISSN 2222-1719 (Paper) ISSN 2222-2863 (Online)
Vol.4, No.11, 2013

Proposing an Effective Retransmission Using the Relaying Nodes for Multihop Networks

Najlaa Abd Hamza AL-Mayahee
Nursing College, Baghdad University,Iraq
*E-Mail: najlaa_al_mayyahee@yahoo.com

Abstract

Coop MAC has been recently proposed as a possible implementation of cooperation protocols in the medium access control (MAC) layer of a wireless network. However, some nodes may refrain from cooperation for selfish purposes, e.g. in order to save energy, in what is called selfish behavior or misbehavior. This protocol violation worsens other nodes' performance and can be avoided if other nodes detect and punish (e.g. banning from the network) misbehaving nodes. However, fading and interference may prevent nodes from cooperating even if they are willing, therefore it is not trivial to identify misbehaving nodes. In a fading scenario where an automatic repeat request (ARQ) protocol is used, we propose a mechanism that allows detecting misbehaving nodes. Two approaches, either based on the uniformly most powerful (UMP) test or on the sequential probability ratio test (SPRT) are considered. The two techniques are characterized and compared in terms of their average detection delay and resulting network performance.

Index Terms: Ad-hoc networks, cooperative diversity, medium access control, MIMO systems, security, privacy, and authentication.

1. INTRODUCTION

The throughput of wireless networks can be significantly increased by allowing cooperation among nodes [2] and [3], which can be efficiently implemented in the medium access control (MAC) layer, as shown for example by the CoopMAC protocol [4], which provides a shorter response time and a better integration with the physical layer than traditional network layer routing [5]. However, a relay helps other nodes at the expense of its own resources (energy and time) that could otherwise be spared or used for the transmission of its own packets. This burden is compensated by the fact that a node operating as a relay in one transmission, will in turn benefit from cooperation of other nodes in another transmission. However, a node can still achieve a high throughput while further sparing energy if it does not relay other nodes' packets while still exploiting their cooperation for its own transmissions. Indeed, this selfish behavior has been extensively examined in the literature within a game theoretic framework [6], [7], concluding that the Nash equilibrium of all nodes is a non-cooperative strategy.

In order to discourage misbehavior, various approaches have been proposed. An incentive mechanism is considered in [8] [9], where each node is charged upon transmission of

its own data and reimbursed when it forwards other nodes' data. In [10] and [11] mechanisms are proposed for the detection of selfish nodes that manipulate the back off parameters of the distributed coordinating function (DCF) in the IEEE 802.11 standard, in order to gain unfair access to the channel. In reputation based systems [11], [12], cooperation is conditioned on an established level of trust. For instance, in CONFIDANT [12], nodes detect misbehavior and disseminate a report on the misbehaving nodes across the network. Eigen Trust [11] is another technique based on a reputation system, where nodes ask other nodes about the behavior of all the nodes and the process is repeated by all interrogated nodes, providing each node a global view of the network. For a review of trust and reputation based systems see also [12].

The problem of selfish and malicious nodes can be found

also in routing [11], where nodes may manipulate or drop packets, again to spare resources or intentionally disrupt the network performance (see for example [11], [12] and references therein). However, detection of selfish nodes in networks with cooperativeMAC protocols which is the focus of this paper is significantly different. First, at the MAC layer packets may not be received due to fading and collisions, and a single instance of cooperation failure is not enough to establish the selfishness of the node. Second, routing usually involves multiple hops, while MAC cooperation involves a single relay, thus allowing a simplification of the protocols used for detection of selfish nodes. Since MAC cooperation has been considered only in recent years, only few contributions have appeared on the detection of selfish nodes. In [11], coalition games are proposed to induce nodes to forward each others' packets using an amplify and forward cooperative scheme. In [10] the authors

proposed a detection mechanism based on the comparison of the probabilities of decoding the control packets. Effective mitigation of misbehavior can be attained

by using the output of the detection technique as input to a reputation mechanism [10], [11], [12] which will predict the expected future behavior of nodes by taking their past history into account. In fact, while the detection technique provides information about current state of each node, the reputation mechanism predicts future behavior combining detection output and past history of the nodes.

The main contribution of this paper is the proposal of two misbehavior detection techniques for networks using Coop- MAC and automatic repeat request (ARQ) protocols. The two approaches are based on the uniformly most powerful (UMP) test and the sequential probability ratio test (SPRT). SPRT and UMP test are well know methods for hypothesis testing (see [3] and references therein) that yield the minimum average detection delay for a bounded error rate, and the minimum miss detection probability for a bounded false alarm probability, respectively. However, they have never been applied to this case. A second contribution of the paper is the performance analysis of the proposed techniques, in terms of both detection delay and network performance, which allow to tune the parameters of the test. Moreover, a game theoretic approach is considered to examine selfish node behavior in existing cooperative protocols. A third contribution of the paper is that the proposed methods can be implemented in IEEE802.11 WLANs, only with the extension of the CoopMAC protocol [4] and without any overhead due to the detection mechanism. Numerical results are provided, comparing the two proposed techniques in a typical ad hoc wireless network setting, showing their merits in the presence of selfish nodes.

The rest of the paper is organized as follows. In Section II we introduce the Preliminary And Problem Formulation. The misbehavior detection mechanism based Cross-Layer Mac Protocol Design and Reputation Based Technique in Sections III and IV, respectively. Numerical results are presented for the various detection techniques in Section V. Lastly, conclusions are outlined in Section VI.

2. PRELIMINARY AND PROBLEM FORMULATION

A. MAC Layer Preliminaries

Consider a single-channel fully-connected wireless network supporting best effort service, where each node can be a source (S), a destination (D), or a helper (H). Here, we base our cooperative MAC on the IEEE 802.11 distributed coordination function (DCF) [3]. The legacy standard uses carrier sense multiple access with collision avoidance (CSMA/CA). Thus, only one transmission pair in the network can be active after a successful channel contention. In general, to increase network throughput, there are two viable approaches:

1) by improving the efficiency of channel access when the nodes contend with each other before data transmission (e.g., controlling the collision probability by adapting the DCF backoff parameters [4] or enabling channel-aware medium access [5]), and

2) by improving the efficiency of link utilization when an actual packet transmission takes place (i.e., by controlling the signaling overhead and increasing transmission data rate). In this work, we focus on the second approach. As the channel is reserved for a node that has won the channel contention, it is rational for the node to send its data packets at a maximum transmit power level for a maximal rate. For simplicity, we assume all nodes in the network have the same power constraint.

We define the link utilization as the effective payload transmission rate (EPTR), taking account of the MAC layer

protocol overhead. Let W, T_P, and $T_O$denote the payload length of a data packet, the times needed to transmit the payload and overhead of the packet, respectively. The EPTR is given by $W/(T_P+ T_O)$. To improve link utilization, we should decrease $T_O$and T_P, by exploring effective signaling overhead control at the MAC layer and advanced transmission techniques at the physical layer, respectively.

B. Physical Layer Preliminaries

To simplify the throughput comparison between a cooperative network and a non-cooperative network, we assume that, in each cooperation opportunity occurred in the cooperative network, the source employs the helper(s) to transmit the same information bits as those without cooperation in the noncooperative network. Further, nodes in both networks operate in half-duplex mode. Consider repetition-based1 selection cooperation [9], where a two-timeslot cooperative transmission is adopted. Focusing on the data rates in transmission, we detail the cooperation scheme as follows. In timeslot 1, the source broadcasts its packet to the optimal helper2 and the destination with a transmission rate, $R_{CI} \in R= \{r_1, r_2, ...,r_Q\}$, where R is the rate set supported by applying

adaptive modulation and coding at the physical layer, and ri<rjif i< j. In timeslot 2, the optimal helper forwards the received information bits cooperatively with the source to the destination, with a transmission rate, $R_{C2} \in R$. Cooperation built on distributed space-time coding (e.g., [6]) or interleaver (e.g., [7]) can facilitate the transmission in timeslot 2. Here, the two rates, R_{C1} and R_{C2}, are chosen such that they are the maximal rates for the optimal helper and the destination to successfully decode the data in timeslots 1 and 2, respectively. As one way to support a high data rate, the destination can collect the signal power from the source and the helper during the two timeslots, whereby according to the modulation and coding schemes a reception with packet combining at the modulation level (e.g., diversity combining [2]) or the coding level (e.g., rate-compatible punctured convolutional (RCPC) coding-based modified Chase combining [8], random binning [9]) can be facilitated.3 Notice that, if the destination only collects the signal power from the helper node, the relaying scheme is simplified to a pure multi-hop transmission.

To model a successful packet reception, given a packet length for each transmission rate in R, there is a minimum signal-to-noise-ratio (SNR) above which the packet can be decoded successfully at a receiver. In this work, we assume that, the channels among the nodes change slowly such that the channel coefficient remains constant for the whole duration of one data packet transmission, which can be justified in a low or moderate-mobility scenario.

C. Problem Formulation

We address the research problems on beneficial cooperation from a cross-layer MAC protocol design perspective. In this research, we do not consider selfish nodes. Aiming at increasing link utilization via strategically activating cooperative transmission, we consider the link utilization in a cooperative network, which is enhanced if any direct transmission in the network with a low EPTR is replaced by cooperative transmission with a higher EPTR. Furthermore, if such a replacement occurs, the helper that supports the highest EPTR is employed in the cooperation. Let R_1 (in R) denote the transmission rate of direct transmission from the source to the destination. Given a specific cooperative MAC protocol design (with known signaling overhead) and payload length W, the CR is defined as a set of rate triples, $C:=\{(R_1,R_{C1},R_{C2})\} \subseteq R_3$, such that the EPTR with cooperation is always larger than that without cooperation. Thus, for a specific payload length, a non-empty CR means beneficial cooperation exists. Utilizing the concept of CR, we can formulate the research problems on beneficial cooperation in cross-layer MAC protocol design as follows.

• When to cooperate: Find the CR C with the maximum link utilization improvement and achieve it via cooperative

MAC.

• Whom to cooperate with: Given a group of helper candidates which can support a rate in the CR, identify the optimal helper which achieves the maximum EPTR with cooperation in a distributed way.

3. CROSS-LAYER MAC PROTOCOL DESIGN

We propose a novel cross-layer cooperative MAC protocol. The study consists of three phases: 1) initial protocol setup, where we devise the signaling exchange and helper selection, and identify tunable MAC protocol parameters; 2) analysis of payload and overhead transmission times; and 3) cooperation region determination and protocol parameter setting.

A. Initial Protocol Setup

Fig. 1 depicts the signaling and data packet transmission of our proposed cooperative MAC protocol. After a random backoff, a source node establishes a communication link with its destination via the request-to-send (RTS)/clear-to-send (CTS) handshake. If the CR is empty (i.e., cooperation is not beneficial), after receiving a CTS packet and waiting for a short interframe space (SIFS), the source sends its data packet to the destination directly, according to the IEEE 802.11 DCF [3].

On the other hand, when a cooperation opportunity arises (i.e., the CR is non-empty), the source and the destination first ascertain whether there exists a helper such that a cooperative transmission is feasible. To locate such a helper, if any, we make use of a helper indication (HI) signal. If no HI signal is detected shortly after an RTS/CTS exchange, direct transmission is triggered. If an HI signal is detected, a cooperative transmission can be initiated (to be discussed). Since the helpers (rather than the source or the destination) initiate node cooperation, we refer to it as helper-initiated cooperation. Compared to a source or destination-initiated cooperation (e.g., [7]), helper-initiated cooperation is preferred in a distributed wireless system. The rationale is

that, due to the RTS/CTS exchange, any potential helper has already been aware of the channel condition between itself and the source (destination) after it overheard the RTS (CTS) packet.

To facilitate helper selection, the information on payload length and channel state of the source-destination (S-D) link (estimated by the destination) can be broadcast in the RTS and CTS packets, respectively. Therefore, every neighbor node can fully collect the channel state information (CSI) to estimate cooperative rate allocation, thereby evaluating its maximal supportable EPTR. However, to reduce overhead in helper selection, there is no information exchange among those potential helpers. That is, a potential helper has no instantaneous CSI of the channels between other potential helpers and the source (destination). Thus, a challenge of helper-initiated cooperation is how to effectively and efficiently select the optimal helper based on local CSI in a distributed way. To solve this problem, we propose the following group-based backoff mechanism.

Define a composite cooperative transmission rate (CCTR), R_h, to denote the payload transmission rate from the source to the destination. With repetition-based two-timeslot cooperation, it can be calculated as $R_h = W/(W/R_{C1}+W/R_{C2}) = R_{C1}R_{C2}/(R_{C1} + R_{C2})$. When competing for the optimal helper, the helper candidates will be organized according to their supportable CCTRs. Given payload length W and direct transmission rate R_1, let Mdenote the number of CCTRs generated from the non-empty CR (to be determined), and each of them labeled by $R_h^*(i)$, i= 1, 2, ...,M. To facilitate helper selection, we sort these M rates in descending order (i.e., $R_h^*(i) > R_h^*(j)$, if $i<j$) and partition them into Ggroups, each one with $n_g(\geq 1)$ members, where $\sum_{g=1}^{G} n_g = M$. Here, M, G, and $n_g$are protocol parameters to be optimized. Note that, reflected in the value of $R_h^*(i)$, different groups have different channel access priorities, and different members in the same group also have different channel access priorities.

To reduce overhead in helper selection, we propose both inter-group contention and intra-group contention. In the intergroup contention, a helper candidate in the gthgroup waits for a period of time, $T_{fb1}(g)$, before sending out its groupindication (GI) signal, if it overhears no GI from any higher rate group, where $T_{fb1}(g) = (g-1) \cdot t_{fb}$, $1 \leq g \leq G$, and $t_{fb}$is referred to as the backoff slot time. Thus, only the members of the highest rate group will keep contending. Then, in the intra-group contention, if a helper candidate (with groupindex $g$and member index m)

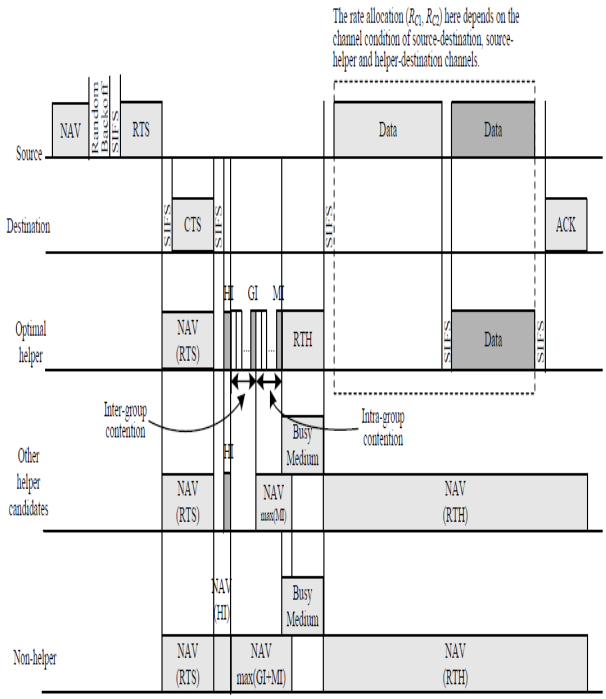

Fig. 1.An illustration of the proposed cooperative MAC protocol.

overhears no member indication (MI) signal, it sends out its MI signal after $T_{fb2}(g,m) = (m-1) \cdot t_{fb}$, $1 \leq m \leq n_g$. Thus, the helper that supports the highest $R_h$can be elected in a distributed manner, which also assures that the EPTR of the selected helper is larger than that of any other nodes failed in the helper contention. To facilitate a

distributed yet effective helper selection, on one hand, the backoff slot time should not be smaller than the duration of any indication signal (i.e., the HI, GI, and MI signals). Denote by$t_{tx}$the duration of any indication signal. It can be found that, $t_{fb} - t_{tx} \geq \max_{H}\{2\tau_{HD}\}$is a sufficient condition to assure an asynchronized yet collision-free helper contention, where $\tau_{HD}$is the propagation delay of a helper-destination (H-D) channel. On the other hand, with the proposed helper selection method, it is vital that all helper candidates share the same grouping structure with respect to the CR for the current S-D pair. We are to address the issue in determining the CR. After the contention, the optimal helper sends out a ready-tohelp (RTH) packet with rate setting to the source to initiate a cooperative transmission (see Fig. 1).

In the case of multiple optimal helpers where two or more RTH packets collide, we employ a simple strategy that lets collided helper candidates re-contend once. Given such a collision, the collided helper candidates can be aware of it by using a timer (T_d) for checking the transmission from the source. When the collision is detected, they resend their RTH packets in a randomly selected minislot from K minislots, as shown in Fig. 2. The probability of RTH packet re-collision depends on the number of minislots and the number of collided nodes. Obviously, a larger *K*gives a smaller re-collision chance, but induces more overhead in the channel time. The value of *K*should be carefully determined, to be discussed. If a re-contention fails, direct transmission is triggered immediately, taking account of signaling overhead and throughput performance.

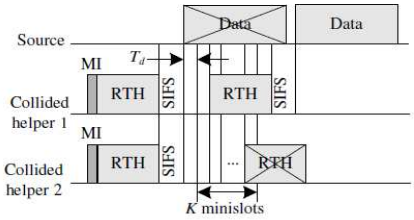

Fig. 2.Solution to RTH packet collision by contention over *K*minislots.

In summary, the proposed MAC protocol facilitates beneficial cooperation based on the CR and CSI obtained from the RTS/CTS signaling, and elects the instantaneous optimal helper in a distributed manner via the inter-group and intragroup contention. However, to maximize the link utilization in each data packet transmission and thus improve network throughput, we need to determine the CR and to optimize the protocol parameters, based on the analysis of payload and overhead transmission times.

4. REPUTATION BASED TECHNIQUE

In which there will be a single central authority maintains and updates the reputation values of all the other nodes in the network. The central authority calculates the reputation values based on two variables. They are the total number of positive feedback and the total number of negative feedback for that node. Other nodes can get this information upon request. To make reputation calculation dynamic, the central authority decays both positive and negative ratings as a function of time. The central authority weights the creditability of the agent which provides the reputation value of a node to it. The new value will be added to the existing reputation value to form an updatedreputation value.

There are various disadvantages in this approach as follows:

- The approach cannot be used in the distributed applications as it considers the central authority for reputation calculating.
- The use of decay function in reputation calculation is not sufficient approach to update the reputation value.
- In this approach the future reputation value cannot be predicted.
- There is no any pictorial representation for the reputation relationships between the nodes.
- The model for reputation is not context specific.

4.1 Punishment Based Technique:

It is one of the reputation based system in which there are four steps followed to identify the malicious nodes and remove them from the network. The first step is identifying the misbehaving nodes such as selfish or malicious nodes. In the second step, the trust manager sends alarm about the malicious nodes. In the third step, the reputation system will assign values to the nodes based on the observations made by it and by others. In the

last step, the path rather rates the path based on the values given by the reputation system and detect the path in which the malicious node present and act according to the routing request.

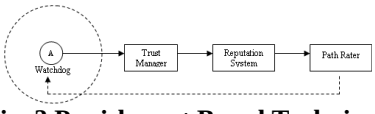

Fig. 3 Punishment Based Technique

4.2 Watch Dog Technique

Promiscuous Mode monitoring approach is one of such technique which is used to identify the malicious node. It is implemented with a routing protocol and relies on monitoring the neighbours. Each node in the transmission path monitors its successor node by overhearing the channel. Monitoring node will find the monitored node as malicious node if it drops the packets more than the threshold value. But it suffers from power control technique [11]. In fig. 3 the nodes A sends data to node B, in turn node B send data to node C and receive an acknowledgement from it. Now node A watch node B for acknowledgement from it for successful transmission.

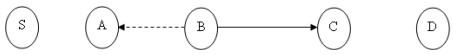

Fig. 4 Watch Dog

4.3 Two – Hop Ack Technique

Two-hop ACK [7, 10] is a technique in which the Acknowledgement travels two hops. By using this node can monitor its successor by receiving the Two-Hop ACK. In fig. 4, node A send data to node B which in turn forward the data to node C. node A now decide whether the node B malicious node or not by the acknowledgement received from node C to itself. If it receives the acknowledgement from node C then the node B is honest otherwise not.

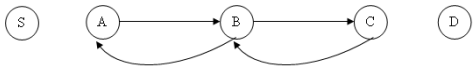

Fig. 5 Two – Hop Ack

4.4 Incentive and Eigen Trust Technique

Incentive technique [2] is one in which the node will be charged for its own transmission and reimbursed when it help for the transmission of other nodes. In this method for incentive purposes we use virtual currency also called as nuggets and the other one is priority for bandwidth. The nuggets are of two types' packet purse model and packet trade model. In packet purse model, the sender will add some nuggets in the packets which can be taken by anynode that forward its packet. In packet market model, every node will purchase the packet from its previous node byusing some nuggets and sell it to the next node for somenuggets. The technique based on the reputation, in which nodes ask all other nodes about the behaviour of all the nodes. Based on reputation the detection is done.

4.5. CoopmacWithArq

CoopMAC Protocol [2] has its implementation in the MAC layer of a wireless network. In this approach we use the CoopMAC and Automatic Repeat Request (ARQ) protocols. These two approaches are based on the Uniformly Most Powerful (UMP) and the Sequential Probability Ratio Test (SPRT). CoopMAC Protocol works as follows. Let us consider that node s wants to have cooperation transmission to node D through node C. The node S first sends a special Request to Send (RTS) packet which contains the requested rate in the link S-C and in the link C-D. Now node D sends a Clear – to – send (CTS) packet to node S and node C sends a Helper ready to send (HTS) packet to node S. On receiving both the CTS and HTS packets node S, starts the transmission. The reception of data by node C and D are acknowledged to node S through an acknowledgement (ACK). A node in

CoopMAC Protocol can behave in two cases: it can be the destination or it can be the cooperating node in transmission between some other nodes. Here the Distributed Misbehaviour Detection Technique is used in which all nodes detect the misbehaving nodes by monitoring the control packets. In centralized approach, the same technique can be applied in where patrolling nodes decodes the control packets and detect malicious activity of the nodes and spread this to all other nodes. A false alarm happens when a honest node istaken as malicious node and a miss detection happens when a malicious node is taken as a honest node.

With ARQ, the node that transmits the data keeps retransmitting the same coded data packet at each frame. The receiving node does not store the past versions of the same coded data packets. So it's assumed that the malicious node will use the same strategy to all the frames. The UMP will have large number of observations to find out the malicious nodes whereas SPRT needs a minimum number of observations to detect the malicious nodes. SPRT has minimum complexity than UMP. HARQ protocol used to detect malicious nodes must perform multiple tests. i.e., testfor each and every HARQ frame.

The shortcomings in this approach are as follows:

- In this approach, there is traffic overhead in the network by passing the control packets.
- The Expected Detection Delay is higher.
- It has to maintain a coop table, which contains the information about all the helper nodes.

5. NUMERICAL RESULTS

In order to assess the performance of the various detection techniques, we consider a wireless network in which each node has the same probability of being a destination with respect to S. For a network of νnodes, for all transmissions from S, for the generic node A, the probability of being in the scenario $A= D$is $1/\nu$. Similarly, the probability of being in the scenario $A= C$is again $1/\nu$. Since in the misbehavior detection process, node Scollects statistics of HTS and hence on potential cooperators, on average we have $E[N_f] = E[M_f]$. For each packet, the first frame has a fixed data rate $R= 1$ bit/s/Hz, normalized to the transmission bandwidth. All frames have packets of the same length. For both ARQ and HARQ we consider at most $F= 3$ frames per data packet. We assume that the decoding probability is not changing with time and is a function of d, i.e. $p_f^{(A=D)}= e^{-\psi f d \kappa}$, where $\kappa$is the path loss exponent, set here to 3.4 and $\psi_f$is a constant characteristic of the $S-A$link, depending on code, fading conditions and noise power. In the following we assume capacity achieving channel coding with coded blocks long enough so that $(1-p_f^{(A=D)})$ is the probability of outage capacity and $\psi_1 = (2^R-1)\Gamma^{-1}$, with Γ the average signal to noise ratio (SNR) at unitary distance, which we set at 20 dB. For ARQ, $\psi_f= \psi_1$, $\in \mathcal{F}$, while for HARQ node Djointly decodes multiple frames and we have $\psi_f= (2^{R/f}- 1)\Gamma^{-1}$, $f \in \mathcal{F}$,. For the nodes' placement we consider both fixed and random placement. The fixed scenario will be considered in Section V-B, while the random scenario is considered in Section V-C.

A. Node ACharacterization

For the behavior of node Awe consider two cases. In one case, node Ahas a fixed probability $\alpha$for all the frames. In the second case, node Aknows the detection technique (genie node, GN), and aims at minimizing cooperation while limiting the probability of being detected. This second case is the most challenging situation, as the misbehaving node knows the detection algorithm and aims at deceiving node S. In particular, for UMP, the selfish node updates the observation variables of the source and predicts their value at the point when the source takes a decision, assuming to behave correctly. If the GN predicts the likelihood of being detected, it cooperates; otherwise it misbehaves. Mathematically, after$\bar{M}$and$\bar{N}$ observations of frames with $A= D$and $A= C$, the predicted values are

$$Y^{(P)}(M) = Y(\bar{M}) + (1 - p^{(A=D)})(M - \bar{M}),$$
$$X^{(P)}(N) = X(\bar{N}) + (1 - p^{(A=D)})(N - \bar{N}).$$

If $Y^{(P)}(M) < \chi(X^{(P)}(N)+Y^{(P)}(M), M, N)$ then the node cooperates, otherwise it misbehaves.

For SPRT, since the detection is not performed at regular intervals, the GN mimics the detection mechanism of the source, but with a smaller indifference region, i.e. with $\lambda_{\mathrm{L}}^{(GN)} = \epsilon\lambda_{\mathrm{L}}$, with $0 \leq \epsilon \leq 1$. When the

misbehaving node computes LLR $\lambda(M)$below $\lambda_{\mathrm{L}}^{(GN)}$, it starts cooperating, otherwise it continues to misbehave. With this approach, the node attempts to reduce the probability of being detected by getting away from the detection boundary λ_{L}. The gap between λ_{L} and $\lambda_{\mathrm{L}}^{(GN)}$is related to the probability of being detected. A wider gap lowers the detection probability, but at the same time forces the node to cooperate more frequently, thus reducing misbehavior, while on the contrary a smaller gap increases the detection probability, as even a good behavior by the node may lead to detection, due to adverse channel conditions.

6. CONCLUSIONS

For the considered problem of misbehavior detection in a cooperative ad hoc network, we showed that SPRT provides the minimum average detection delay for bounded FA and MD probabilities while UMP attains the minimum MD probability for a bounded FA probability with a fixed detection delay. SPRT is best suited to environments where fastest detection of misbehaving nodes is required. On the other hand, UMP is best suited to environments that tolerate detection delay to attain the minimum MD probability.

7. REFERENCES

[1] SintayehuDehnie, Member, IEEE, and Stefano Tomasin, Member, IEEE, “Detection of Selfish Nodes in Networks Using CoopMAC Protocol with ARQ”, IEEE Transactions On Wireless Communications, Vol. 9, No. 7, July 2010.

[2] A. Sendonaris, E. Erkip, and B. Aazhang, “User cooperation diversity part I: system description,” IEEE Trans. Commun., vol. 51, no. 11, pp. 1927–1938, Nov. 2003.

[3] “User cooperation diversity part II: implementation aspects and performance analysis,” IEEE Trans. Commun., vol. 51, no. 11, pp. 1939– 1948, Nov. 2003.

[4] P. Liu, Z. Tao, S. Narayanan, T. Korakis, and S. S. Panwar, “CoopMAC: a cooperative MAC for wireless LANs,” IEEE J. Sel. Areas Commun., vol. 25, no. 2, pp. 340–354, Feb. 2007.

[5] T. K. amd Z. Tao, Y. Slutskiy, and S. Panwar, “A cooperative MAC protocol for ad hoc wireless networks,” in Proc. 5th Annual IEEE Int. Conf. on Pervasive Computing and Commun. Workshops (PerComW), 2007.

[6] F. Milan, J. J. Jaramillo, and R. Srikant, “Achieving cooperation in multihop wireless networks of selfish nodes,” in Proc. ACM Workshop on Game Theory for Communications and Networks (GAMENETS 2006), Oct. 2006.

[7] V. Srinivasan, P. Nuggehalli, F. Chiasserini, and R. R. Rao, “An analytical approach to the study of cooperation in wireless ad hoc networks,” IEEE Trans. Wireless Commun., vol. 4, no. 2, pp. 722–733, Mar. 2005.

[8] N. Shastry and R. S. Adve, “Stimulating cooperative diversity in wireless ad hoc networks through pricing,” in Proc. IEEE Int. Conf. Commun. (ICC), vol. 8, June 2006, pp. 3747–3752.

[9] O. Ileri, S.-C. Mau, and N. B. Mandayam, “Pricing for enabling forwarding in self-configuring ad hoc networks,” IEEE J. Sel. Areas Commun., vol. 23, no. 1, pp. 151–162, Jan. 2005.

[10] A. L. Toledo and X. Wang, “Robust detection of selfish misbehavior in wireless networks,” IEEE J. Sel. Areas Commun., vol. 25, no. 6, pp. 1124–1134, Aug. 2007.

[11] S. Radosavac and J. S. Baras, “Appliatication of sequential detection schemes for obtaining performance bounds of greedy users in the IEEE 802.11 MAC,” IEEE Commun. Mag., pp. 148–154, Feb. 2008.

[12] S. D. Kamvar, M. T. Schlosser, and H. Garcia-Molina, “The eigentrust algorithm for reputation management in p2p netowkrs,” in ACM International Conference on World Wide Web, 2003, pp. 640–651.