

# Protecting MANETs from Spurious CTS Attacks with Randomized Carrier Sensing

Jing Deng<sup>†</sup>, Zhiguo Zhang<sup>†</sup>, Sreekanth Pagadala<sup>†</sup>, and Pramod K. Varshney<sup>‡</sup>

<sup>†</sup>Dept. of Computer Science, University of New Orleans, New Orleans, LA 70148

<sup>‡</sup>Dept. of Electrical Engineering and Computer Science, Syracuse University, Syracuse, NY 13244

**Abstract** — The Request-To-Send and Clear-To-Send (RTS/CTS) exchange has been used in Mobile Ad-hoc Networks (MANETs) to alleviate the hidden and exposed terminal problems. In such an exchange, the so-called Network Allocation Vector (NAV) indicates the current and future state of the channel. Unfortunately, this technique may suffer from virtual jamming by malicious nodes in the network. For example, malicious nodes may send periodic Spurious CTS (SCTS) packets with the sole purpose of forcing other nodes to update their NAV values and preventing them from using the channel. In this paper, we investigate the effect of such SCTS attacks and propose a solution, termed Carrier Sensing based Discarding (CSD). The CSD scheme serves as an add-on to the original RTS/CTS-based Medium Access Control schemes such as IEEE 802.11 DCF MAC. We further demonstrate the performance of our proposed scheme through analysis and the ns2 simulator.

## I. INTRODUCTION

IN multi-hop wireless networks such as Mobile Ad-hoc Networks (MANETs), the exposed/hidden terminal problems may significantly reduce the throughput of the shared channel. Many Multiple Access Control (MAC) schemes have been designed to solve these problems. In particular, the IEEE 802.11 DCF MAC scheme employs the Carrier Sense Multiple Access with Collision Avoidance (CSMA/CA) technique, in which the so-called Request-To-Send and Clear-To-Send (RTS/CTS) exchange is used [1]. The RTS/CTS exchange works as follows: A client wishing to transmit a message sends an RTS packet, which includes source address, destination address, and duration for the transmission. The receiver responds with a CTS packet if the channel is free, with duration information in it. After receiving the CTS packet, the sender responds with data packets and the receiver sends an acknowledgment to inform the sender that the transmission has completed. All the exposed nodes (which are within the transmission range of the sender but out of that of the receiver) overhearing the RTS packet and all the hidden nodes (which are within the transmission range of the receiver but out of that of the sender) overhearing the CTS packet keep silent during the transmission duration which guarantees the successful transmission and reception of the message.

In IEEE 802.11 DCF, a Network Allocation Vector (NAV) is implemented for channel reservation. The NAV is a timer that indicates the duration for which the medium has been reserved. The sender sets the Duration/ID field of its RTS

packet equal to the time for which it expects to use the medium, including the transmission time of all the packets in the sequence. Every exposed node updates its NAV accordingly after overhearing the RTS packet (the NAV value is changed only when the new value is greater than the current NAV). CTS packets have the same field to be used by overhearing nodes in a similar manner.

Nodes set up a timer to count down the NAV. When the NAV is greater than zero, the so-called virtual carrier-sense function indicates that the medium is busy. Nodes can only start transmission when both the physical carrier-sense function and the virtual carrier-sense function indicate an idle medium. So in this way, the medium is reserved for a sender/receiver pair until the end of the transmission.

However, there is no mechanism to verify the NAV values of RTS/CTS packets. When a node overhears an RTS or CTS packet, it does not know whether the corresponding NAV value is legitimate or not. This makes spurious packet transmission an attractive approach for malicious nodes to disrupt communications. For example, if a malicious node sends a spurious CTS packet, in which it intentionally sets a long NAV timer, other nodes within the transmission range will set up their NAVs equal to this value without suspicion. These nodes will defer all transmissions for the entire NAV period while the channel is idle. This vulnerability may be exploited by attackers to block neighboring nodes from accessing the shared medium for an extended period of time [2][3]. Such attacks reduce the throughput especially when the node density is high.

In this paper, we investigate the effect of SCTS attacks and propose a solution to address this problem. Our solution is termed Carrier Sensing based Discarding mechanism (CSD). The main idea of the CSD scheme is to ask nodes overhearing CTS packets to look for the expected data packet transmission. If no transmission carrier is sensed, the CTS packet is treated as spurious and the NAV value on the CTS packet is discarded. Such detections of carrier on the channel may be performed multiple times before discarding the NAV in order to overcome potential missed detections.

Our paper is structured as follows: in Section II, we introduce related works. In Section III, we explain the details of the SCTS attack. In Section IV, we propose the CSD mechanism as a solution to avoid SCTS attacks. In Section V, we analyze the performance of CSD. Numerical results and simulation results are presented in Section VI. Section VII provides some concluding remarks.

## II. RELATED WORK

The unique characteristics of MANETs, which include stringent resource constraints and highly dynamic network topology, present a new set of challenges to security design [1][4][5][6]. We focus on denial-of-service attacks [7][8][9] and user selfishness [10][11][12] in this work.

During Spurious RTS/CTS attacks, malicious nodes access an unfair share of the channel by manipulating the duration value in their control packets, i.e., setting the NAV falsely high. In IEEE 802.11, to reduce the risk of Denial-of-Service via the use of fake RTS packets, a node is permitted to reset its NAV if no PHY-RXSTART [1] indication is detected from the Physical layer (PHY) some time after receiving the RTS packet. Parker et al. [11] simulated such an RTS attack and proposed a scheme to accurately diagnose malicious attacks in ad hoc networks by combining the input from all layers of the network stack. Acharya et al. [13] investigated fake RTS attacks in IEEE 802.11b networks, using a single fake RTS jammer and proposed the CTSR (CTS Reservation) protocol based on assessment of the channel status and resetting NAV value if the channel is idle. These papers focused Spurious RTS packets, but our paper investigates the attacks with Spurious CTS packets.

Chen et al. [2] investigated Spurious RTS/CTS attacks and NAV attacks. A solution for NAV attacks was proposed and a protocol modification was recommended that IEEE 802.11 should have a provision to reset the NAV value after a fixed period of time if the channel is found idle. Ray et al. [3] investigated the channel blocking problem and proposed a solution for RTS induced Congestion due to virtual blocking. They used an RTS validation technique to solve the so-called “false blocking” problem.

## III. THE SPURIOUS CTS ATTACK

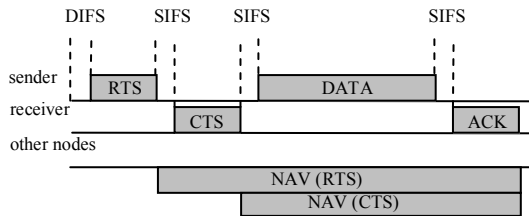


Figure 1 The usage of NAV in the RTS/CTS exchange. DIFS and SIFS represent DCF Inter-Frame Spacing and Short Inter-Frame Space, respectively.

We explain and illustrate the mechanism of spurious CTS attacks in this section. In the IEEE 802.11 MAC protocol, both virtual carrier-sense and physical carrier-sense functions are used to reduce the probability of collisions on the shared wireless channel. A node can only send packets when both of these two functions indicate that the medium is idle. Network Allocation Vector (NAV) serves as the key for the virtual carrier-sense function (see Figure 1).

The RTS/CTS packet contains a duration field, which is used to set the NAV of the nodes overhearing the RTS/CTS

packet. Every node overhearing an RTS or CTS packet will set its NAV accordingly. The NAV specifies the earliest time at which the node is permitted to attempt transmission. For example, the neighboring nodes of the RTS sender will set their NAVs according to the overheard RTS packet. Similarly, the neighboring nodes of the intended receiver will set their NAVs according to the overheard CTS packet. Note that these two sets of nodes may be different due to their different physical locations. This mechanism protects the transmission from the sender to the receiver from any transmission by these neighboring nodes.

The vulnerability of the NAV scheme is that stations do not verify the NAV values. This is because they do not know if the transmission is actually happening during the expected duration. Since a node must defer its transmission when it overhears an RTS/CTS packet, it will be effectively blocked if the corresponding transmission does not take place and the shared medium is left idle. An attacking node may launch Denial of Service (DoS) attacks by sending out spurious CTS packets periodically. The worst situation is that the attacked nodes are completely blocked from using the channel by a sequence of Spurious CTS (SCTS) packets.

## IV. CARRIER SENSING BASED DISCARDING (CSD) MECHANISM

As described above, when nodes are blocked by SCTS attacks, the shared medium remains unused and wasted. In this section, we propose a solution termed Carrier Sensing based Discarding (CSD). The mechanism of the CSD scheme can be explained as follows: a node overhearing a CTS packet will assess the status of the channel at the time when the corresponding data packet transmission should start. If the medium within the carrier sensing range is idle, indicating no transmission on the channel, the CTS packet is treated

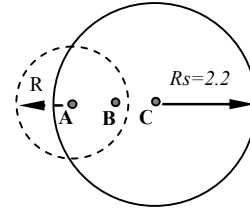


Figure 2 The transmission range and the carrier sensing range. In this figure, we show the transmission range of node A, the sender. The carrier sensing range of node C is shown (the solid circle) as  $2.2R$ .

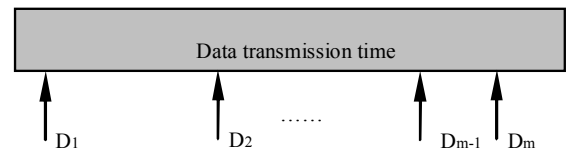


Figure 3 The random placements of the  $m$  detection points in the CSD scheme.

spurious and the corresponding NAV value is discarded. On the other hand, if the channel is sensed busy, the CTS packet is treated as normal. Note that the correctness of such SCTS detection scheme depends on the long carrier sensing range.

In the IEEE 802.11 standard, the carrier sensing range is  $2.2R$ , where  $R$  is the wireless transmission range [1]. This makes sure that the signal of data transmission from the sender can be sensed by any node that is within a distance of  $2.2R$  from the sender (cf. Figure 2).

An intelligent attacker may send bursts of signals to avoid being detected by the CSD scheme. In order to protect the network from such intelligent attackers, several carrier sensing points of CSD are randomly chosen among the entire expected data packet transmission time. We design the CSD scheme to sense the carrier on the shared channel  $m$  times. These  $m$  detection points work in the following way: if any of these  $m$  detection points reveals an idle channel, the CTS

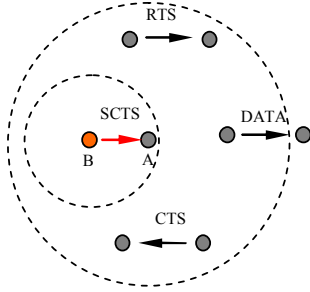


Figure 4 Illustration of missed detection. Node B sends an SCTS packet in node A's neighborhood. If one or more nodes within  $2.2R$  distance of node A send packets at each of the  $m$  detection points, CSD fails to detect the SCTS packet.

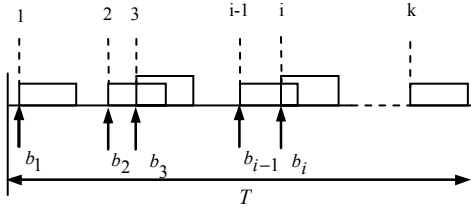


Figure 5 Illustrations of packet overlap during the expected DATA packet transmission time. Packets 2 and 3 overlap. So do packets  $(i-1)$  and  $i$ . These overlap times affect the overall busy time,  $T_{busy}$ , within  $T$ .

packet in question is declared as SCTS and the NAV is reset. This process is also known as the OR fusion rule in distributed detection [15]. Other rules such as  $k$  out of  $m$  rules may be possible, but we leave those to our future work. An illustration of the CSD scheme is shown in Figure 3.

## V. ANALYSIS OF THE CSD MECHANISM

The correctness of the CSD scheme is based on the long carrier sensing range (of  $2.2R$ ). However, we should consider the case in which other nodes within the sensing range send packets, affecting the carrier sensing results. This is because a CSD failure occurs when there are transmissions on all detection points (cf. Figure 4). We compute the probability of missed detection in the CSD scheme, i.e., CSD failure, in the following.

We first introduce our assumptions for analysis: The number of packet transmissions taking place during one unit time within the sensing range of a node is assumed to be

Poisson distributed with average packet arrival rate of  $G$  per unit time. We also assume that these packets have an average duration of  $\tau$ , the CSD mechanism has  $m$  randomly selected detection points, and the period of each NAV of SCTS is  $T$ . The duration of busy period due to packet transmissions in  $T$ ,  $T_{busy}$ , can be determined by summing the durations of packets transmitted during  $T$ . However, there may be packet overlaps as illustrated in Figure 5, where packets 2 and 3 overlap and packets  $(i-1)$  and  $i$  overlap. Therefore, when the time between the beginning of two consecutive packets is less than  $\tau$ , they overlap.

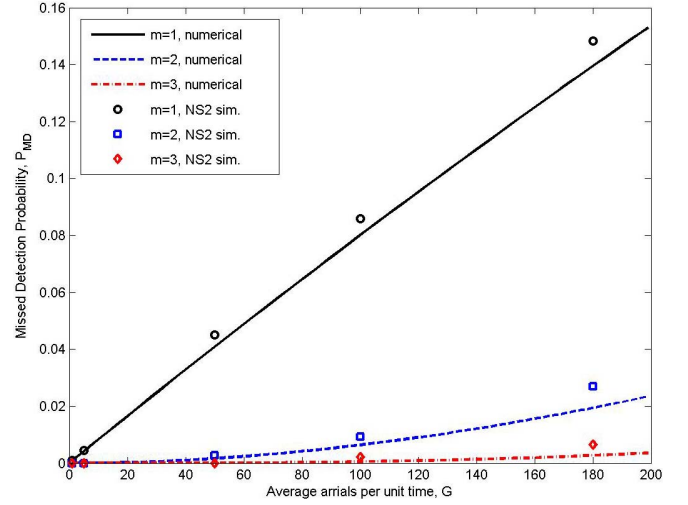


Figure 6 Comparison between simulation results and numerical results of missed detection probability.

Denote  $t_i = b_i - b_{i-1} < \tau$ , so the overlap time is  $\tau - t_i$  between packets  $(i-1)$  and  $i$ . Based on the Poisson packet arrivals, the expected overlap time between a packet and the next packet can be expressed as:

$$\varepsilon = \int_0^{\tau} (\tau - t) \cdot G \cdot e^{-Gt} dt. \quad (1)$$

Note that the calculation of  $\varepsilon$  is an approximation, which ignores the possibility of more than two packets overlapping. We argue that this is a good approximation when  $G$  is relatively small.

When there are  $k$  packets arriving in  $T$ , the total busy period is:

$$T_{busy}(k) = k \cdot \tau - (k-1) \cdot \varepsilon. \quad (2)$$

Therefore, the average value of the busy period is:

$$T_{busy} = G \cdot T \cdot \tau - (G \cdot T - 1) \cdot \varepsilon, \quad (3)$$

where we have assumed an average of  $G \cdot T$  packets during  $T$  units of time.

We assume that an SCTS packet has been sent. The probability of one detection point detecting busy status, which fails to detect SCTS, is:

$$q = \frac{T_{busy}}{T}. \quad (4)$$

When there are  $m$  detection points, a missed detection (of SCTS packets) occurs if all detection points sense the channel to be busy. The probability of a missed detection,  $P_{MD}$ , is then given by:

$$P_{MD} = q^m = \left[ \frac{G \cdot T \cdot \tau - (G \cdot T - 1) \cdot \varepsilon}{T} \right]^m, \quad (5)$$

where  $\varepsilon$  is given by (1).

The analysis of false alarms (or false positives) of SCTS detection in the CSD scheme is trivial. This is because any detection point will sense the busy channel due to the actual data transmission taking place assuming that the sensing mechanism is perfect. Therefore, the probability of false alarm of the CSD scheme is 0.

## VI. PERFORMANCE EVALUATION

### A. Accuracy of $P_{MD}$ Estimation

In order to demonstrate the effects of the number of detection points,  $m$ , on missed detection probability,  $P_{MD}$ , we present Figure 6. The data packet transmission time is 603 msec and the period of each NAV of SCTS,  $T$ , is 45.5 msec. Based on Figure 6,  $P_{MD}$  increases with  $G$ . When  $m$  increases,  $P_{MD}$  reduces because of higher chances of detecting SCTS packets. Our simulation results match well with numerical results except in very large  $G$  regions, where the assumption of at most two packets colliding with each other becomes invalid. That is why simulation values are slightly larger than numerical values and difference increases with  $G$ .

### B. Performance of CSD

We used ns2 [14] to simulate several IEEE-802.11b-based MANETs and to investigate the effect of SCTS attacks and that of our CSD scheme. In our simulation, we assumed that all SCTS packets tried to reserve the use of the channel for the maximum possible time, which we identified as 45.5 msec. Another important parameter for the malicious SCTS packet senders is the average interval between two consecutive SCTS packets. We term it ASI (Average duration of SCTS Intervals). Unless specified otherwise, ASI is 65.3 msec.

In order to see the effects of ASI on network throughput, we study a simple fixed 3-node wireless network with fixed CBR (Constant Bit Rate). In this network, a sender and a receiver communicate in the presence of an SCTS attacker (all nodes are within range of each other). In Figure 7, we show the relative throughput compared with that of a network without SCTS attackers. We investigate four cases, where the packet generation rates of CBR are 0.4 Mbps, 0.6 Mbps, 0.8 Mbps and 1.0 Mbps, respectively. The SCTS packets cause more negative effect with higher traffic load. As ASI increases, throughput improves with smaller number of SCTS

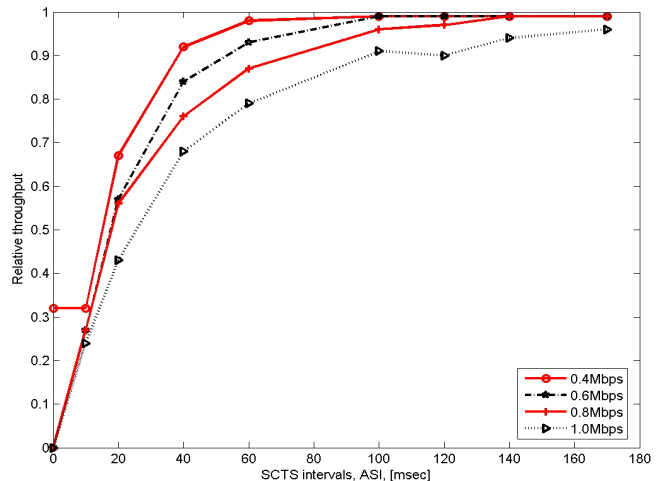


Figure 7 Effect of ASI on the throughput in a 3-node network. There are SCTS attackers but no CSD protection.

packets in sent to the network.

We compared the throughput ( $S$ ) of a normal network (without SCTS), one with SCTS, and one with SCTS and CSD under different traffic loads in the fixed three-node wireless network in Figure 8. SCTS attackers cause more damage at higher traffic-load settings ( $L$ ), because higher traffic load means higher probability for attackers competing for more unfair use of the shared medium. And with the CSD scheme, the network throughput can be restored effectively.

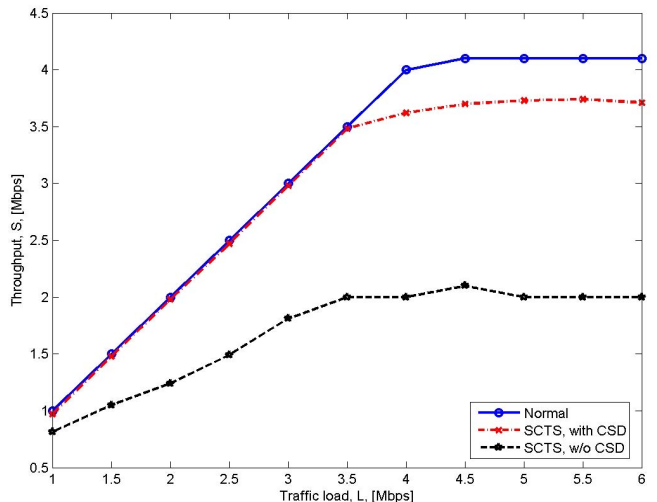


Figure 8 Throughput comparisons between the SCTS and the CSD for a fixed 3-node model.

We also set up a 100-node wireless network over a  $1000 \times 1000 \text{ m}^2$  region. These nodes are distributed randomly. There are 10 random sender/receiver pairs. We investigated the network throughput obtained under SCTS attacks and compared the effect of SCTS and CSD mechanism by increasing the number of spurious nodes. The results are shown in Figure 9. Obviously, more SCTS attackers cause more severe degradation of network throughput. With CSD, the network throughput is restored to 85% level of a similar network without SCTS attackers.

The effect of the SCTS and the CSD approach with

different number of detection points is investigated in Figure 10. As  $m$  increases, the performance of the CSD scheme improves and approaches the performance of a similar network without SCTS attackers. The cost of a larger  $m$  in the CSD scheme is the increased memory usage and CPU resources at the detecting nodes.

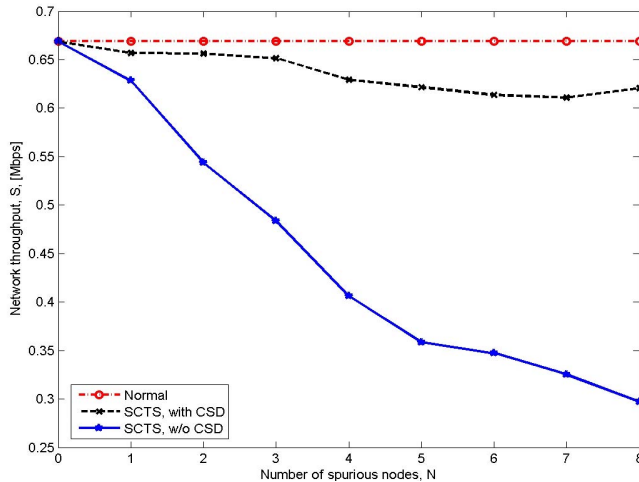


Figure 9 Throughput comparisons of the SCTS and the CSD mechanisms in a 100-node network.

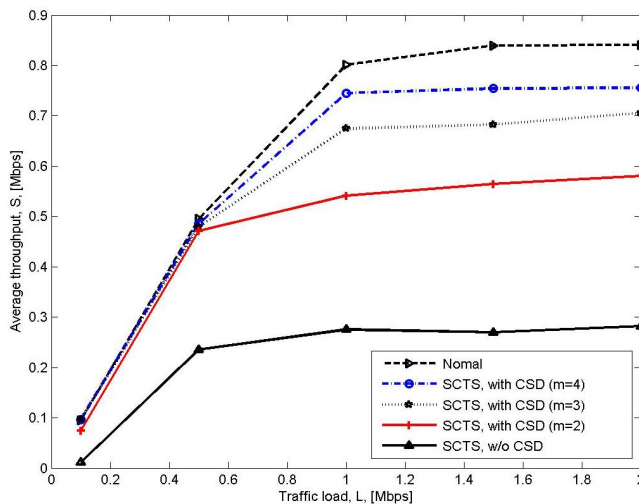


Figure 10 Throughput comparisons between SCTS and CSD with different  $m$  for a 100-node network.

## VII. CONCLUSION AND FUTURE WORKS

The RTS/CTS mechanism combined with the NAV scheme is currently used to avoid packet collisions caused by hidden nodes in many ad-hoc network MAC scheme such as IEEE 802.11 DCF. Unfortunately, this leads to the vulnerability of the virtual carrier-sense function: misbehaving or malicious users may send spurious packets especially spurious CTS packets to block other users from accessing the channel. Due to the inherently vulnerable design of the IEEE 802.11 DCF scheme, the attackers are able to block such channels with only small number of packet transmissions.

In this work, we have proposed the Carrier Sensing based

Discarding (CSD) scheme to mitigate such adverse effects of the spurious CTS packets. Instead of allowing each node overhearing a CTS packet to update its NAV value, the CSD scheme requires a node to verify the on-going data communication during the entire data packet transmission period. Such carrier sensing is possible thanks to the larger carrier sensing range in IEEE 802.11 DCF (2.2R). We have presented the technical details of the CSD scheme and our analysis. Simulation results show that the CSD scheme recovers most of the channel throughput in networks under spurious CTS attacks as compared to regular networks.

In future work, we plan to investigate CSD with different fusion rules [15], to implement CSD in IEEE 802.11 hardware and test its usability under more practical settings.

## REFERENCES

- [1] IEEE Computer Society, "Wireless LAN medium access control (MAC) and physical layer (PHY) specifications," IEEE standard 802.11, 1999.
- [2] D. Chen, J. Deng, and P. K. Varshney, "Protecting Wireless Networks against a Denial of Service Attack Based on Virtual Jamming," ACM MobiCom '03, Poster, San Diego, CA, USA, September 14-19, 2003.
- [3] S. Ray, J. B. Carruthers, and D. Starobinski, "RTS/CTS-Induced Congestion in Ad Hoc Wireless LANs," in *Proc. of IEEE WCNC '03*, New Orleans, LA, USA, Mar. 16 - 20, 2003.
- [4] H. Yang, H. Luo, F. Ye, S. Lu, and L. Zhang, "Security in mobile ad hoc networks: challenges and solutions," *IEEE Wireless Communication*, vol. 11, no. 1, pp. 38–47, Feb. 2004.
- [5] P. Albers, O. Camp, J. Percher, B. Jouga, L. Me, and R. Puttini, "Security in ad hoc networks: a general intrusion detection architecture enhancing trust based approaches," in *Proc. 1st International Workshop on Wireless Information Systems '02*, pp. 1–12, Apr. 2002.
- [6] S. Buchegger and B. Le, "Nodes bearing grudges: Towards routing security, fairness, and robustness in mobile ad hoc networks," in *Proc. of Parallel, Distributed and Network-based Processing '02*, pp. 403–410, 2002.
- [7] J. Bellardo and S. Savage, "802.11 Denial-of-Service Attacks: Real Vulnerabilities and Practical Solutions," in *Proc. of the USENIX Security Symposium*, Washington, DC, USA, Aug. 2003.
- [8] V. Gupta, S. V. Krishnamurthy, and M. Faloutsos, "Denial of Service Attacks at the MAC Layer in Wireless Ad Hoc Networks," in *Proc. of IEEE MILCOM '02*, Anaheim, CA, USA, 2002.
- [9] S.-R. Ye, Y.-C. Wang and Y.-C. Tseng, "A Jamming-Based MAC Protocol to Improve the Performance of Wireless Multihop Ad Hoc Networks," *Wireless Communications and Mobile Computing*, Vol. 4, No. 1, pp. 75-84, Feb. 2004.
- [10] P. Kyasanur and N. Vaidya, "Selfish MAC Layer Misbehavior in Wireless Networks," *IEEE Transactions on Mobile Computing*, vol. 4, no. 5, Sep. 2005.
- [11] J. Parker, A. Patwardhan, and A. Joshi, "Detecting wireless misbehavior through cross-layer analysis," in *Proc. of IEEE Consumer Communications and Networking Conference Special Sessions*, January 2006.
- [12] S. Radosavac, J.S. Baras and I. Koutsopoulos, "A framework for MAC layer misbehavior detection in wireless networks," in *Proc. of ACM Workshop on Wireless Security (WiSe) '05*, Cologne, Germany.
- [13] M. Acharya and D. Thuento, "Intelligent Jamming Attacks, Counterattacks and (Counter) Attacks in 802.11b Wireless Networks," in *Proc. of OPNETWORK-2005 Conference*, Washington DC, USA, Aug. 2005.
- [14] The ns Manual -- A collaboration between researchers at UC Berkeley, LBL, USC/ISI, and Xerox PARC. K. Fall and K. Varadhan Eds.
- [15] P. K. Varshney, *Distributed Detection and Data Fusion*, Springer-Verlag, 1997.