MAC Layer Misbehavior Effectiveness and Collective Aggressive Reaction Approach

Vamshikrishna Reddy Giri Neeraj Jaggi
Department of Electrical Engineering and Computer Science
Wichita State University, Wichita, KS 67260
Email: {vrgiri, neeraj.jaggi}@wichita.edu

***Abstract*—Current wireless MAC protocols are designed to provide an equal share of throughput to all nodes in the network. However, the presence of misbehaving nodes (selfish nodes which deviate from standard protocol behavior in order to get higher bandwidth) poses severe threats to the fairness aspects of MAC protocols. In this paper, we investigate various types of MAC layer misbehaviors, and evaluate their effectiveness in terms of their impact on important performance aspects including throughput, and fairness to other users. We observe that the effects of misbehavior are prominent only when the network traffic is sufficiently large and the extent of misbehavior is reasonably aggressive. In addition, we find that performance gains achieved using misbehavior exhibit diminishing returns with respect to its aggressiveness, for all types of misbehaviors considered. We identify crucial common characteristics among such misbehaviors, and employ our learning to design an effective measure to react towards such misbehaviors. Employing two of the most effective misbehaviors, we study the effect of collective aggressiveness of non-selfish nodes as a possible strategy to react towards selfish misbehavior. Particularly, we demonstrate that a collective aggressive reaction approach is able to ensure fairness in the network, however at the expense of overall network throughput degradation.**

I. Introduction

MAC protocols are intended to provide a fair access to the wireless channel for all users in the wireless LAN. Such fairness serves as a backbone to the design of more sophisticated service differentiation mechanisms to provide quality of service in the network. For instance, IEEE 802.11 protocol employs Binary Exponential Backoff (BEB) scheme to introduce randomness in channel access, and avoids collision by relying upon the nodes to double their contention window (CW) upon collision [1]. However, increased programmability of network devices lately [2] has led to the possibility of individual users modifying their own protocol behavior in order to achieve higher bandwidth. Such misbehaviors [3] at MAC layer adversely affect the performance of the network in terms of overall throughput and fairness, and are quite intense when distributed coordination function (DCF) mode of 802.11 operation is in use.

There are multiple strategies which a user could employ in order to achieve the objectives. These range from a *malicious* user disrupting the normal network behavior using jamming, denial-of-service attacks [4], or non-cooperation with respect to data forwarding [5], to a *selfish* user employing mildly aggressive cheating with respect to backoff rules [3], [6], [7], or the choices of contention window [8], DIFS, SIFS in order to increase its own throughput share at the expense of other *genuine* nodes.

In this paper, we consider selfish misbehavior with respect to the choice of backoff interval chosen inappropriately via modification to the BEB algorithm. We classify five types of such misbehavior and study their effectiveness, and impact on network performance under varying traffic load scenarios. We point out the common characteristics observed for different types of misbehaviors, and comment on cheating strategies that a smart cheater may employ to avoid detection while achieving higher share of the throughput simultaneously. We also identify scenarios where it is (or is not) appropriate to trigger a reaction response. In addition, we propose a collective aggressive misbehavior response by genuine nodes as a strategy to react towards misbehavior, and demonstrate that such an approach guarantees fairness in the network.

The rest of the paper is organized as follows. Section II provides an overview of related research in this area. In section III, we classify various types of misbehaviors based upon modifications to the BEB algorithm. In section IV, we measure and analyze the *effectiveness* of different types of misbehaviors under various traffic conditions. In section V, we propose a collective aggressive reaction strategy and demonstrate that such an approach is able to ensure fairness in the network. We summarize our conclusions in Section VI.

II. Related Work

Various types of misbehaviors at MAC layer have been considered and multiple detection methodologies and reaction schemes have been proposed. Detection scheme based on observed backoff intervals chosen by other nodes and employing *Sequential Probability Ratio Test* have been proposed in [6]. Other schemes employ throughput degradation [4] or access point based adaptive mechanism [9] to detect presence of misbehavior. Authors in [3] introduce the concept of *receiver-assigned backoff*, and authors in [7] propose modifications to the BEB algorithm in order to facilitate easy detection and penalization of misbehaving sender. In this paper, however, we study the different types of misbehaviors in an attempt to understand their common characteristics that could be employed to detect and trigger a reaction response.

This work was supported in part by the Army Research Office under DEPSCoR ARO Grant W911NF-08-1-0256.

Most reaction schemes employed by genuine nodes attempt to penalize [3], [7] or isolate [5] the selfish node. The overall objective of reaction schemes is to make it disadvantageous for any user to deviate from standard protocol behavior. [5] suggests that isolation of misbehaving nodes is not the best strategy to react, as it affects the performance at network and higher layers, as more and more nodes get isolated. In this paper, we propose a reaction scheme which not only guarantees fairness, but also provides the needed disincentive to the selfish user in order to prevent misbehavior, without isolating the selfish node.

III. MAC Layer Misbehavior Classification

The behavior of the normal node following the standard BEB algorithm can be summarized as follows. A node which has data to transmit chooses a backoff interval b uniformly at random from the interval $[0 \dots CW - 1]$, where CW denotes the node's current contention window size. The node waits for b time slots before accessing the channel. However, if the channel is sensed busy during this time, the node freezes its backoff until the channel is sensed idle again and continues counting down thereafter. The initial contention window size equals CW_{min}. Upon successful transmission, the node resets its CW to CW_{min}. However, upon unsuccessful transmission (eg. due to collision), the node sets its CW to be $\min\{2 * CW, CW_{max}\}$. Standard choices for the above constants are given by $CW_{min} = 32$ and $CW_{max} = 1024$. We consider the following five types of misbehaviors associated with modifying the 802.11 BEB algorithm.

- α-misbehavior : Instead of choosing the backoff b uniformly at random from the interval $[0 \dots CW - 1]$, the selfish node chooses b uniformly at random from the interval $[0 \dots \alpha(CW - 1)]$, where $0 < \alpha < 1$. Thus the node ends up choosing a smaller backoff interval than it is supposed to, increasing its chances of accessing the channel next.
- Deterministic Backoff (db)-misbehavior: The node chooses a deterministic, constant backoff interval b irrespective of the current contention window size. For instance, the node could always choose a very small backoff (say 2), irrespective of multiple failed transmission attempts, thus trying to gain preference over other genuine nodes in terms of channel access.
- β-misbehavior: Upon unsuccessful transmission, the node instead of setting its CW to be $\min\{2 * CW, CW_{max}\}$, sets its contention window as $CW = \max\{CW_{min}, \min\{\beta * CW, CW_{max}\}\}$, where $0 < \beta < 2$. This results in the node choosing smaller backoff interval than expected. Also the node sets $CW_{min} = \min\{32, \beta * 32\}$ to appropriately distinguish between the scenarios when $\beta < 1$.
- Fixed Maximum Contention Window (CW_{max})-misbehavior: Typical value of CW_{max} employed by genuine nodes equals 1024. However, the selfish node employing CW_{max}-misbehavior sets its maximum contention window to be a value smaller than 1024. Thus the node ends up choosing smaller backoff intervals than other genuine nodes, particularly at higher traffic loads when the number of collisions in the network increase. Also the node sets $CW_{min} = \min\{32, CW_{max}\}$.
- Fixed Contention Window (CW_{fix})-misbehavior: The selfish node sets its contention window to a small, fixed size CW_{fix}, and always chooses its backoff interval uniformly at random from the interval $[0 \dots CW_{fix}]$.

Note here that in all the different types of misbehavior, the selfish node could vary the level or aggressiveness of its misbehavior by appropriate choice of the involved parameters. For instance, the smaller the value of CW_{fix}, the more aggressive the CW_{fix}-misbehavior would be. In addition, a node may employ a hybrid strategy which is a combination of two or more of the above.

IV. Misbehavior Effectiveness Characterization

In this section, we measure and analyze the impact of each type of misbehavior mentioned above on the network. In order to measure the effectiveness of a misbehavior strategy, we compute the percentage increase in the throughput of the selfish node gained via misbehaving, compared to the scenario when all the nodes are genuine. Let t_g denote the throughput of the node x when all nodes are genuine. Now, let us introduce one of the misbehaviors (say α-misbehavior with $\alpha = 0.5$) at node x, and let t_m denote the throughput of the node x, when all the other nodes are still genuine. Then, the effectiveness e of α-misbehavior for $\alpha = 0.5$ is characterized as,

$$e = \left(\frac{t_m - t_g}{t_g}\right) * 100 \tag{1}$$

Indeed, e measures the magnitude of the incentive that a selfish user has in order to misbehave.

We measure the effectiveness of various types of misbehaviors, for three different traffic load scenarios, and at different levels of aggressiveness. We simulate a 802.11 wireless LAN using OPNET with 10 nodes in a 100m x 100m area, where 9 nodes are sending traffic to one receiver and the distance between receiver and each of the senders is 30m. Out of the 9 senders, one sender is configured to misbehave according to the level and type of misbehavior desired. The size of each packet equals 512 bytes, and slot time is $20\mu s$. The heavy traffic load scenario corresponds to an exponential packet arrival rate of 100 packets per second. Similarly, the medium load scenario corresponds to 77 packets per second, and the low load scenario corresponds to 25 packets per second. The data rate of the network equals 2 Mbps. Both the high load and medium load scenarios overload the network beyond its capacity, by generating a total traffic load of 3.7 Mbps and 2.8 Mbps respectively. The low load scenario corresponds to 0.9 Mbps, which is quite less compared to the network capacity.

Figure 1 depicts the effectiveness achieved using α-misbehavior at various values of α. We observe that under low traffic load scenario, as the total LAN traffic is below the capacity, the throughput of each node remains the same at all values of α. Thus the selfish node does not gain much

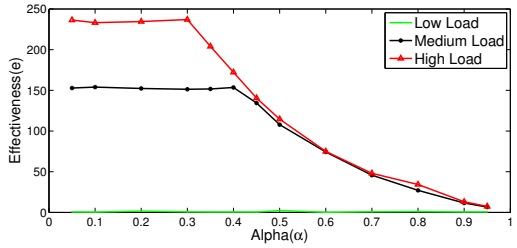

Fig. 1. Alpha(α) Misbehavior

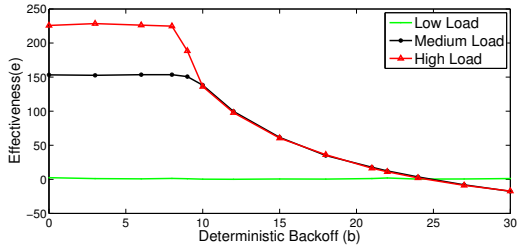

Fig. 2. Deterministic Backoff(db) Misbehavior

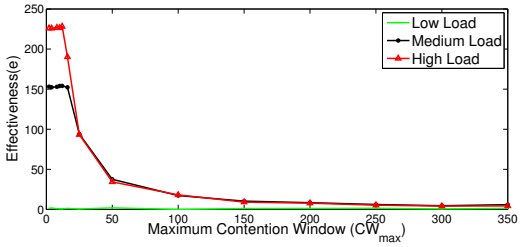

Fig. 4. Fixed Maximum Contention Window(CW_{max}) Misbehavior

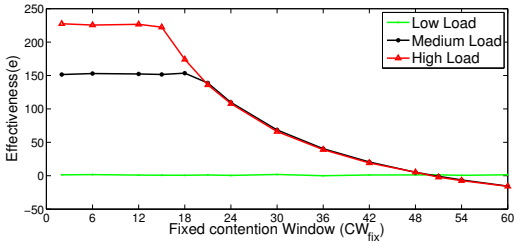

Fig. 5. Fixed Contention Window(CW_{fix}) Misbehavior

by misbehaving in this scenario. We also observe that under medium and high loads, there is a non-linear increase in misbehavior effectiveness with a decrease in the value of α below 1. However, this gain saturates after a while (which corresponds to selfish node being able to get all its data across successfully), and further increasing the level of misbehavior does not lead to substantial throughput gains for the selfish node. Therefore, a selfish node may choose to operate close to $\alpha = 0.4$ for medium traffic, and near $\alpha = 0.3$ for high traffic, in order to reap substantial throughput gains while hoping to avoid detection. Note that the selfish node could easily estimate the best value of α by noticing the diminishing returns as the level of misbehavior is increased further. All these observation hold true for other misbehavior types as well.

Figure 2 depicts the effectiveness achieved using db-misbehavior at various values of deterministic backoff (b) chosen by the selfish node. We observe results similar to the case of α-misbehavior. We also observe that under saturated traffic scenarios, and in the absence of misbehavior, the mean value of the backoff interval chosen by a genuine node over time ≈ 22. Therefore, we notice a decrease in throughput for the selfish user for values of b significantly greater than 22. From the above two figures, we also observe that the maximum throughput gain or effectiveness is limited to around 150% under medium traffic and around 230% under high traffic.

Figure 3 depicts that for β-misbehavior, the increase in effectiveness is close to linear with a decrease in β for $1 < \beta < 2$. As β is decreased below 1, the misbehavior effectiveness increases non-linearly initially, particularly because the value of CW_{min} also gets decreased. The effectiveness saturates however, as is the case with other misbehavior types. Figures 4 and 5 depict the effectiveness with CW_{max} and CW_{fix}-misbehaviors respectively. Under saturated traffic conditions with no misbehavior, the average value of contention window used by a genuine node ≈ 50. This leads to negative effectiveness for the selfish node when $CW_{fix} > 50$.

Next, we evaluate the effectiveness achieved using a hybrid strategy. Figure 6 and 7 correspond to α-misbehavior along with CW_{max} and β respectively. Note that saturation is reached at higher values of α compared with just α-misbehavior, however the maximum achievable effectiveness remains unchanged. Thus a selfish node does not have any additional advantage to apply a hybrid strategy. In all of the misbehaviors types considered, mild misbehaviors or low

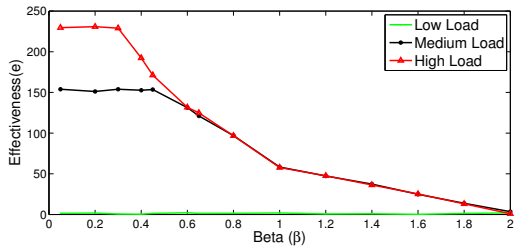

Fig. 3. Beta(β) Misbehavior

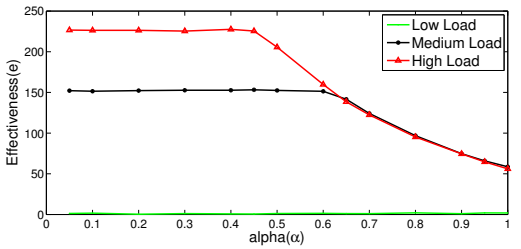

Fig. 6. Hybrid Misbehavior ($CW_{max} = 64$)

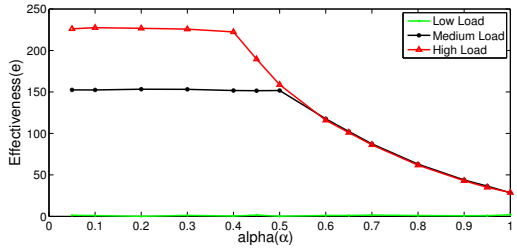

Fig. 7. Hybrid Misbehavior ($\beta = 1.5$)

traffic load do not lead to large improvements in the throughput of the selfish node. As the level of misbehavior is increased, from mild to aggresive, there is generally a non-linear increase in the effectiveness. However the effectivess saturates around the region where the node misbehavior is quite aggresive. Thus a detection and reaction strategy should be employed only when the traffic load is considerably high and the level of misbehavior is reasonably aggressive.

A. *Effect on fairness*

We study the effect on fairness in the throughputs achieved by genuine nodes, in the presence of misbehavior. In the high traffic load scenario with 9 senders, the throughput of a genuine node, in the absence of any misbehavior, is around 145Kbps. We consider 8 genuine senders and one misbehaving sender employing α-misbehavior with $\alpha = 0.05$. The throughput of the selfish node increases by 225% to 471Kbps, also suggested by Figure 1.[1] Table I shows the throughputs of the 8 genuine nodes in the presence of misbehavior. We observe that the average throughput of the genuine nodes is around 106.6Kbps, corresponding to a decrease of 26.5% each. Note that cumulative decrease in throughput of the genuine nodes $= 26.5 * 8 = 212\%$ is slightly less than the increase in throughput for the selfish node. Thus, presence of misbehavior causes the overall LAN throughput to increase slightly.

We also observe that the fairness in throughput of the genuine nodes is not effected due to the presence of the misbehavior. Using Jain's fairness index [10], the fairness in the throughput of the 9 genuine nodes in the absence of misbehavior is computed to be 0.9999 (A value of 1 corresponds to the maximum possible fairness index.) In the presence of misbehavior, the fairness decreases to 0.622 due to the drastic increase in the throughput share of the selfish node. However, the fairness in the throughputs of the 8 genuine nodes in the presence of misbehavior equals 0.9998. This suggests that the structure of the CSMA/CA protocol, along with the RTS/CTS mechanism and randomly chosen backoff, is able to guarantee fairness among the genuine nodes even in the presence of a misbehaving node. This is the intuition behind our proposed reaction strategy, wherein the genuine nodes, upon detecting the presence of misbehavior in the network, respond by collectively applying the same level of misbehavior in order to guarantee a fair share of throughput for all nodes in the network.

[1]Note that OPNET throughput computations include a header of size 78 bytes for each packet, regardless of the packet size.

Node	Throughput in Kbps
1	105.343
2	107.134
3	105.232
4	107.298
5	108.243
6	107.298
7	104.205
8	108.121

TABLE I
THROUGHPUT OF GENUINE NODES WITH α-MISBEHAVIOR ($\alpha = 0.05$)

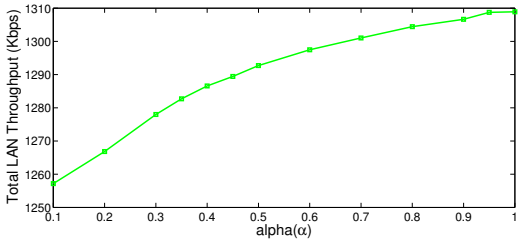

Fig. 8. Overall LAN throughput with collective α-misbehavior reaction response

The comparison of effectiveness yields that most of the misbehaviors are quite effective in increasing the selfish node's throughput share. Next, we consider the α and CW_{fix} misbehaviors to study the effect of collective aggressive reaction strategy on throughput and fairness achieved in the network.

V. PROPOSED REACTION APPROACH

The primary goal of a reaction strategy is to provide sufficient disincentive for the selfish node so that it does not try employing any misbehavior strategy. This could be achieved by triggering a reaction response by all genuine nodes such that the selfish node's throughput becomes less than what it would have been in the absence of any misbehavior. One approach to achieve this goal would be for the genuine nodes to accurately estimate the level of misbehavior of the selfish node, and try to replicate that misbehavior as a reaction response. We show that such a reaction response not only

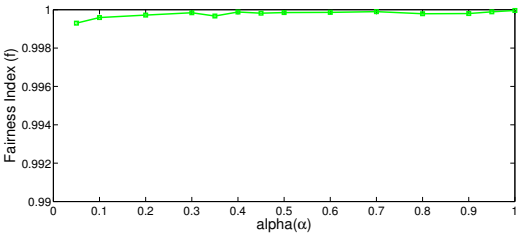

Fig. 9. Fairness index with collective α-misbehavior reaction response

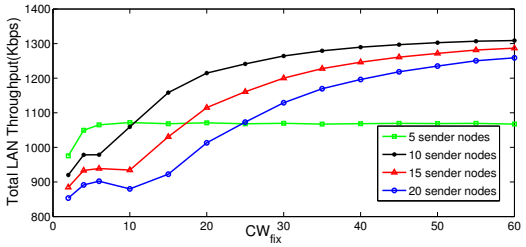

Fig. 10. Overall LAN throughput with collective CW_{fix}-misbehavior reaction response

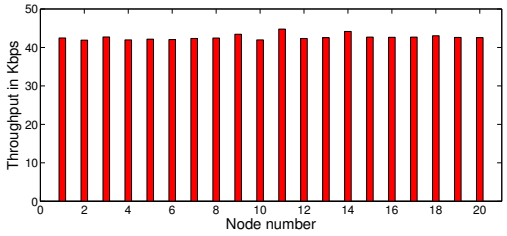

Fig. 11. Throughput share of all nodes with $CW_{fix} = 2$

causes the selfish node's throughput effectiveness to decrease below 0 (thus providing the necessary disincentive), but also ensures fairness in the throughput of all nodes in the network. This result is inline with our earlier findings in Section IV which suggest that multiple nodes choosing backoff in a similar manner achieve almost equal share of the throughput.

Assuming that the genuine nodes are able to detect the level of misbehavior in the network, we analyze the impact of the proposed reaction response on throughput and fairness achieved in the network. For the high load scenario with 9 senders, Figure 8 depicts the overall LAN throughput achieved when all the nodes collectively apply α-misbehavior at various values of α. We observe that such a reaction response could degrade the overall throughput, particularly at higher values of α. However, the selfish node's throughput also reduces to the levels available to other genuine nodes. Figure 9 depicts that the fairness index is close to optimal for all values of α.

Figure 10 depicts the total LAN throughput when all the N nodes in the network collectively apply CW_{fix}-misbehavior for various values of N and CW_{fix}. The packet size is 512 bytes and the traffic at each node equals 45 packets per second. When $N = 5$, this corresponds to a low traffic load scenario (similar to Section IV). However, when $N = 20$, it corresponds to the high load scenario. $N = 15$ is close to the medium load scenario. In the case of low traffic ($N = 5$), the degradation in overall throughput with increase in level of misbehavior is quite less, as expected. However, the impact of level of misbehavior on throughput degradation is higher as traffic load (or N) increases. For a particular value of CW_{fix}, and under saturated traffic conditions, the overall throughput decreases with an increase in load. And this difference is more prominent for reasonably aggressive choices of CW_{fix} (12 - 25). In all these scenarios, the nodes in the network are able to achieve a fair share of the throughput. Figure 11 depicts the share of throughput for the high load scenario corresponding to the most aggressive reaction response, $CW_{fix} = 2$. The fairness index in this case equals 0.9997.

A. Comparison with known detection and reaction schemes

The proposed aggressive reaction response is similar to the meaningful Nash equilibrium outlined in [8]. However, if the level of misbehavior is the most agressive, the reaction response converges towards the equilibrium corresponding to network collapse. The detection and estimation scheme could be based upon the backoff values chosen [6] (or the throughputs observed [4]) of all nodes in the network over time. In future, we intend to design a distributed algorithm which allows the genuine nodes to detect the presence as well as estimate the level of misbehavior in the network. Then, the genuine nodes could replicate the misbehavior aggressiveness in order to achieve fairness in the network. The algorithm should also be adaptive such that if the misbehaving node, upon realizing that its throughput is decreasing, chooses to return back to normal behavior, then the genuine nodes should be able to detect the same and return to normal BEB algorithm.

VI. Conclusions and Open Issues

We classified various types of MAC layer misbehaviors and studied their impact on throughput and fairness under various traffic load scenarios. We also proposed a collective aggressive reaction response which is able to ensure fairness in the network. The estimation of misbehavior type and its level of aggressiveness, is an interesting problem of future research.

References

[1] *Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) Specification*, IEEE Std. 802.11, 1997.

[2] L. Guang, C. Assi, and A. Benslimane, "Mac layer misbehavior in wireless networks: Challenges and solutions," *IEEE Wireless Communications*, vol. 15, no. 4, pp. 6–14, Aug. 2008.

[3] P. Kyasanur and N. H. Vaidya, "Selfish mac layer misbehavior in wireless networks," *IEEE Transactions on Mobile Computing*, vol. 4, no. 5, pp. 502–516, Sep. 2005.

[4] Y. Rong, S.-K. Lee, and H.-A. Choi, "Detecting stations cheating on backoff rules in 802.11 networks using sequential analysis," in *Proc. 25th IEEE International Conference on Computer Communications (INFOCOM)*, Barcelona, Spain, Apr. 2006, pp. 1–13.

[5] L. Guang, C. Assi, and Y. Ye, "Dream: A system for detection and reaction against mac layer misbehavior in ad hoc networks," *Elsevier Computer Communications*, vol. 30, no. 8, pp. 1841–1853, Jun. 2007.

[6] S. Radosavac, J. S. Baras, and I. Koutsopoulos, "A framework for mac protocol misbehavior detection in wireless networks," in *Proc. 4th ACM workshop on Wireless security*.

[7] L. Guang, C. Assi, and A. Benslimane, "Modeling and analysis of predictable random backoff in selfish environments," in *Proc. 9th ACM international symposium on Modeling analysis and simulation of wireless and mobile systems*, Terromolinos, Spain, 2006, pp. 86–90.

[8] M. Cagalj, S. Ganeriwal, I. Aad, and J.-P. Hubaux, "On selfish behavior in csma/ca networks," in *Proc. 24th IEEE International Conference on Computer Communications (INFOCOM)*, Miami, USA, Mar. 2005, pp. 2513–2524.

[9] M. Raya, I. Aad, J.-P. Hubaux, and A. E. Fawal, "Domino: Detecting mac layer greedy behavior in ieee 802.11 hotspots," *IEEE Transactions on Mobile Computing*, vol. 5, no. 12, pp. 1691–1705, Dec. 2006.

[10] D.-M. Chiu and R. Jain, "Analysis of the increase and decrease algorithms for congestion avoidance in computer networks," *Computer Networks and ISDN Systems*, vol. 17, no. 1, pp. 1–14, Jun. 1989.