Hindawi Publishing Corporation
EURASIP Journal on Wireless Communications and Networking
Volume 2007, Article ID 28315, 12 pages
doi:10.1155/2007/28315

Research Article

Design and Implementation of an Enhanced 802.11 MAC Architecture for Single-Hop Wireless Networks

Ralph Bernasconi,[1] Silvia Giordano,[1] Alessandro Puiatti,[1] Raffaele Bruno,[2] and Enrico Gregori[2]

[1] *Department of Innovative Technologies, The University of Applied Sciences of Southern Switzerland (SUPSI), Via Cantonale, Gallera 2, 6928 Manno, Switzerland*
[2] *Institute for Information Technology (IIT), National Research Council (CNR), Via G. Moruzzi 1, 56124 Pisa, Italy*

Received 29 June 2006; Revised 25 September 2006; Accepted 27 November 2006

Recommended by Marco Conti

Due to its extreme simplicity and flexibility, the IEEE 802.11 standard is the dominant technology to implement both infrastructure-based WLANs and single-hop ad hoc networks. In spite of its popularity, there is a vast literature demonstrating the shortcomings of using the 802.11 technology in such environments, such as dramatic degradation of network capacity as contention increases and vulnerability to external interferences. Therefore, the design of enhancements and optimizations for the original 802.11 MAC protocol has been a very active research area in the last years. However, all these modifications to the 802.11 MAC protocol were validated only through simulations and/or analytical investigations. In this paper, we present a very unique work as we have designed a flexible hardware/software platform, fully compatible with current implementations of the IEEE 802.11 technology, which we have used to concretely implement and test an enhanced 802.11 backoff algorithm. Our experimental results clearly show that the enhanced mechanism outperforms the standard 802.11 MAC protocol in real scenarios.

1. INTRODUCTION

In the last decade, we have witnessed an exceptional growth of the wireless local area network (WLAN) industry, with a substantial increase in the number of wireless users and applications. This growth was due, in large part, to the availability of inexpensive and highly interoperable networking solutions based on the IEEE 802.11 standards [1], and to the growing trend of providing built-in wireless network cards into mobile computing platforms. Due to its extreme simplicity and flexibility, the IEEE 802.11 standard is a good platform to implement both infrastructure-based WLANs and single-hop ad hoc networks. In addition, the 802.11 technology has been successfully employed to deploy multihop wireless networks in which self-organized groups of devices communicate via multihop wireless paths. Recently, the Wi-Fi market is experiencing a renewed growth as new standardization efforts are carried out [2, 3] and new market opportunities are explored with the deployment of *metro-scale* 802.11-based mesh networks, which are metropolitan areas with 802.11 coverage providing a cellular-like connectivity experience [4].

The WLANs, either in single-hop or multihop configurations, inherit the classical problems of wireless communications and wireless networking. In particular, the wireless medium has neither absolute nor readily observable boundaries outside of which stations are known to be unable to receive correct frames. In addition, the channel is unprotected from external signals. For these reasons, the wireless medium is significantly less reliable than wired media, it is characterized by time-varying interference levels and asymmetric propagation properties, and it is affected by complex phenomena such as the hidden-terminal and the exposed-terminal problems (see [5, 6] for an in-depth discussion on these issues). Note that the hidden-terminal phenomenon may occur both in infrastructure-based and ad hoc networks. However, it may be more relevant in ad hoc networks where almost no coordination exists among the stations. Other potential inefficiencies for the IEEE 802.11 technology come from the fact that this standard adopts a CSMA/CA-based MAC protocol with no collision detection capabilities. This design is mainly due to the limitations of the wireless technology, which usually employs just one antenna for both sending and receiving. In addition, the fast attenuation of the radio signal causes an asymmetric perception of the medium state at the receiver and transmitter. Therefore, acknowledgment packets (ACK) are sent, from the receiver to the sender, to confirm that packets have been correctly received. As no collision detection mechanism is present, colliding stations always complete their transmissions, severely reducing channel utilization [7]. To mitigate the occurrence of collision events, the channel access scheme is regulated by the exponential backoff: nodes failing to obtain the channel have to backoff a random time before trying again. It is widely recognized that, depending on the network configuration, the standard IEEE 802.11 protocol can operate very far from the theoretical limit of the wireless network, as well as unfairly allocate channel resources to each node. While this unfairness is somehow controlled in the infrastructure-based configurations, it can dramatically grow in distributed ones. Furthermore, both unfairness and low channel utilization impact upper layer protocols, especially transport layer if TCP is used. These phenomena have been shown through simulations [8–10], and appeared even worse when tested in real experiments [11, 12].

In recent years a variety of extensions to the random access 802.11 MAC protocol have been investigated such as to cope with the aforementioned issues. Concerning the MAC protocol efficiency, it is now well consolidated that an appropriate tuning of the IEEE 802.11 backoff algorithm can significantly increase the protocol capacity [7, 13–16]. The basic idea is that the random backoff duration should be dynamically tuned by choosing the contention window size as a function of the network congestion level. The major shortcoming of this prior work is that it lacks experimental evidences gained from practical prototypes of the proposed enhanced 802.11 MAC protocols. It is evident that both simulations and theoretical analysis are fundamental to elaborate a clear understanding of the system behaviors and to rapidly evaluate the effectiveness of innovative strategies and techniques. However, practical experiences on trial platforms are also essential to demonstrate the feasibility of proposed mechanisms and to confirm the analytical/simulative predictions. For these reasons, recently the development of hardware/software platforms implementing new MAC protocols has gathered a lot of attention in the research community. In this paper, we will present the activities carried out in the framework of the MobileMAN project, which have led to the architectural design and implementation of an enhanced 802.11 MAC protocol more suitable for ad hoc environments.

The MobileMAN project is an initiative funded by the European FET FP5 Programme with the primary technical objective of investigating the potentialities of the mobile ad hoc network (MANET) paradigm, both in single-hop or more complex multihop configurations. As one of the major aims of the MobileMAN project was to perform experiments in real scenarios, we decided to redesign the MAC architecture and to realize a prototype implementing the new MAC protocol specified for the MobileMAN network. The building block of the enhanced MAC protocol we implemented in software is the asymptotically optimal backoff (AOB) mechanism [16], which dynamically adapts the backoff window size to the current network contention level and guarantees that an IEEE 802.11 WLAN asymptotically achieves its optimal channel utilization. The AOB protocol has been selected as the reference MAC protocol for the MobileMAN network because it relies only on topology-blind estimates of the network status based on the standard physical carrier sensing activity. Hence, it appears as a suitable and robust solution for both single-hop and multihop configurations. Several extensions for the AOB protocol have been proposed in the framework of the MobileMAN project such as to make it more efficient and fair when used in traditional WLANs and ad hoc environments. In this paper, we do not go into details of the various proposed mechanisms, but we specifically focus on describing the architecture of our enhanced IEEE 802.11 wireless network card and on showing experimental results proving the effectiveness of the implemented solutions [17]. Note that our medium access platform has been designed to be a versatile architecture that could be used for implementing and testing: (1) backoff algorithms more adequate to multihop operations; (2) dynamic channel switching schemes to exploit channel quality diversity; (3) efficient layer-2 packet-forwarding; and (4) cross-layering optimizations through the exploitations of topology information provided by the routing layer. In this paper, we present our activity concerning point (1) above. Specifically, we present our card architecture and we describe how the AOB protocol has been implemented in our MAC platform. Moreover, we describe the implementation of a *credit-based strategy* which extends the contention control algorithm adopted by the AOB protocol, such as to improve its efficiency. This scheme has been proposed and evaluated via simulations in a prior work [17]. In this paper we show experimental results obtained by comparing our enhanced MAC card with traditional IEEE 802.11 wireless cards, which demonstrate the significant per-station throughput improvement ensured by our enhanced MAC protocol. Furthermore, the experimental outcomes open promising directions to investigate additional enhancements, as discussed in Section 5.

The rest of this paper is organized as follows. In Section 2 we briefly outline the strategies proposed in literature to increase the 802.11 MAC protocol efficiency. Section 3 describes the algorithms that have been implemented in the network card. In Section 4 we present the measurement environment and we report the results of our real experiments, discussing the most relevant points. Section 5 concludes this chapter with some further discussion and detailed description of the ongoing and future work. A final appendix describes the architecture of our network card platform and discusses the main hardware and firmware design choices.

2. INCREASING THE 802.11 MAC PROTOCOL EFFICIENCY

As discussed above, the 802.11 frame transmissions can be subject to collision events because the random access MAC protocol cannot schedule perfectly the channel accesses. As a consequence, the strategies adopted to mitigate the probability of colliding and to coordinate the frame retransmissions in case of collision are essential in determining the MAC protocol efficiency. The standard 802.11 MAC protocol employs a truncated binary exponential backoff algorithm to schedule retransmissions after a collision. Specifically, each retransmission is delayed by an amount of time depending on the number of collisions that frame has been involved in. However, the retransmission timeout cannot increase indefinitely but when it reaches a ceiling it does not increase any further.

Several analytical studies of the 802.11 MAC protocol efficiency have pointed out that the legacy backoff algorithm can lead to very inefficient utilization of the channel resources. In particular, two major drawbacks can be identified. First, in high contention situations the average backoff delay introduced by the 802.11 algorithm is not sufficient to mitigate the collision probability that rapidly increases. Second, the legacy 802.11 backoff algorithm estimates the contention level in the network using only the number of consecutive retransmissions. However, this information does not provide a precise and complete measure of the network contention level. Previous proposals made to improve the 802.11 MAC protocol efficiency have attempted to resolve the aforementioned issues. In particular, a considerable amount of research efforts has been dedicated to derive the backoff value that maximizes the network capacity by optimally spreading the channel accesses [7, 13, 18]. In addition, a variety of techniques have been investigated to measure the network contention level in a more precise manner than simply monitoring the number of retransmissions. It is quite intuitive that the most straightforward approach would be to estimate the number of competing terminals in the networks and to compute the optimal backoff window for this network population size [7, 13]. The main limitation of this approach is that precisely computing the number of backlogged stations in a wireless network is difficult and error-prone. A more sophisticated measure of the contention level is obtained by monitoring the average duration of idle periods and collisions. In [15, 18] a mathematical relationship between the optimal backoff window value and the ratio between idle periods and collision lengths is derived. Although this theoretical result allows gaining a more in-depth understanding of the MAC protocol dynamics and it leads to the design of a simple and effective optimization of the backoff algorithm, it is not easily extendible to ad hoc environments. A third different approach is proposed in [14, 16], in which the utilization rate of the slots (*slot utilization SU*) is used as an estimate of the current network contention level. The slot utilization can be computed as the ratio between the number of slots in the backoff interval in which one or more stations start a transmission attempt, that is, busy slots, and the total number of backoff slots available for transmission in the backoff interval, that is, the sum of idle slots and busy slots.[1] In particular, in [16] the optimal slot-utilization level that ensures to maximize the channel utilization given a certain network contention level is derived. This optimal slot utilization is called *asymptotic contention limit* (ACL(q)), which depends mainly on the average size, say q, of the frames that are transmitted on the common wireless channel, whereas it is negligibly affected by the number of stations in the network [16]. To exploit the knowledge of the ACL(q) value, the AOB mechanism introduces a *probability of transmission* P_T according to the following formula:

$$P_T = 1 - \left[\min\left(1, \frac{\mathrm{SU}}{\mathrm{ACL}(q)}\right)\right]^{N\_A}, \qquad (1)$$

where N_A is the number of unsuccessful transmission attempts already performed by the station for the transmission of the current frame. When the standard 802.11 MAC protocol assigns a transmission opportunity to a station (i.e., that station has backoff timer equal to zero and sense the channel idle), the station will perform a real transmission with probability P_T; otherwise (i.e., with probability $1 - P_T$) the station deems the transmission opportunity as a *virtual collision*, and the frame transmission is rescheduled as in the case of a real collision, that is, after selecting a new backoff interval using a doubled contention window. By using the P_T defined in formula (1), the AOB mechanism guarantees that asymptotically the slot utilization of the channel never reaches the value ACL(q), namely, the channel utilization is maximal in networks with a large number of stations.

In our prototyping network interface card (NIC) platform we decided to adopt the AOB solution as baseline because, differently from other proposals, it relies only on topology-blind estimates of the network status based on the standard physical carrier sensing activity. Hence, in addition to being easily employed in traditional WLANs it also appears as a suitable and robust solution for ad hoc environments. However, the AOB scheme has some drawbacks. First of all, unless the slot utilization is null, the P_T value is always lower than one. As a consequence, even in lightly loaded networks stations will sometimes refrain to transmit reducing the protocol efficiency. In addition, the AOB algorithm assumes a homogenous wireless network formed of collaborative devices. However, for backward compatibility it is necessary to design specific provisions to permit AOB-enabled devices to interact with legacy 802.11-enabled devices without being disadvantaged. Finally, the AOB protocol should be extended to cope with the unfair allocation of channel resources that occurs in multihop configurations. Previous papers have considered these important aspects and possible solutions have been proposed and evaluated via simulations [17, 19]. In this work we do not aim at proposing novel solutions to the limitations of the original AOB protocol. On the contrary, this paper describes the architectural design and the implementation of a NIC card based on the AOB protocol and the extensions defined in [17]. This card is used to conduct experiments in real scenarios such as to prove the effectiveness of the implemented solutions in a prototype system.

---

[1] It is useful to recall that, for efficiency reasons, the IEEE 802.11 MAC protocol employs a discrete-time backoff scale. That is to say, the backoff time is slotted, and a station is allowed to transmit only at the beginning of each slot time.

3. MAC PROTOCOL IMPLEMENTATION

In this section, we present the various modules that have been developed in the MobileMAN NIC card to implement the AOB protocol as defined in [16] and the credit-based enhancements as specified in [17]. The description of the NIC hardware platform is reported in our prior paper [20] and in the appendix.

The first component that has been developed in our card is the one needed for the run-time estimation of the slot utilization values. However, in our implementation we do not estimate the aggregate slot utilization, as done in [16], but we split it into two contributions: the internal slot utilization ($\mathrm{SU}_{\mathrm{int}}$) and the external slot utilization ($\mathrm{SU}_{\mathrm{ext}}$), such as to differentiate between the contribution to the channel occupation due to the node's transmissions and to its neighbors' transmissions. This differentiation is motivated by the need to keep our implementation as much flexible as possible, such as to allow future modifications as the one described in [19]. Another variation with respect to the original AOB is the time interval over which we compute the slot utilization. In fact, the original AOB computes the slot utilization after each backoff interval, while in our implementation we used a constant observation period T of 100 ms. This choice is motivated by the need to avoid frequent slot utilization computations, which could interfere with the time constraints of the atomic MAC operations (e.g., RTS/CTS exchange). Each station monitors the channel status during the time window T to compute the slot utilization values. In particular, the computing node can observe on the channel three types of events.

(i) *Busy periods*, that is, time intervals during which the radio receivers perceive on the channel a signal power above the *receiving threshold*. Note that a busy period can be due to channel occupations caused by collided frames, frames corrupted by channel noise, successful transmissions carried out by computing node's neighbors, or external interferences. Let n_{rx} be the number of busy periods during the time window T. Note that two channel occupations should be considered separated busy periods only when they are separated by an idle period longer than the *DIFS* interval. This guarantees that the MAC ACK frames are not counted as channel occupations different from the data frames they acknowledge.

(ii) *Frame transmissions* performed by the node itself. Let n_{tx} be the number of frames transmitted by the computing node.

(iii) *Idle periods*, that is, time intervals longer than a *SIFS* interval during which there is no channel activity. Let n_{idle} be the duration of an idle period, normalized in terms of time slots. Note that an idle period is not composed only of backoff slots, but we count also the time intervals during which the *DIFS* and *EIFS* timers are active. This is in contrast with the original definition of the slot utilization as introduced in [14]. However, we preferred this novel formulation because it is more general and it provides a more robust estimation of the utilization rate of slots in multihop configurations (the reader is referred to [17] for a more in-depth discussion of these aspects).

From these measurements of the n_{rx}, n_{tx}, and n_{idle} quantities, the two slot utilization values are computed as follows:

$$\mathrm{SU}_{\mathrm{int}} = \frac{n_{tx}}{n_{\mathrm{idle}} + n_{tx} + n_{rx}}, \tag{2a}$$

$$\mathrm{SU}_{\mathrm{ext}} = \frac{n_{rx}}{n_{\mathrm{idle}} + n_{tx} + n_{rx}}. \tag{2b}$$

It is easy to recognize that the original SU value as defined in [16] can be computed as the sum of $\mathrm{SU}_{\mathrm{int}}$ and $\mathrm{SU}_{\mathrm{int}}$ values. Thus, our implementation and the original AOB scheme are equivalent.

Using formulas (2a) and (2b) we compute a single sample of the slot utilization. However, to avoid sharp fluctuations in the slot utilization estimates we should average these single measures. To solve this problem we apply a moving average-window filter to the slot utilization measures. Specifically, assume that the station is observing the channel during the ith observation period. Then, it follows that

$$\overline{\mathrm{SU}}_{\mathrm{int}}^{(i)} = \alpha_1 \cdot \overline{\mathrm{SU}}_{\mathrm{int}}^{(i-1)} (1 - \alpha_1) \cdot \mathrm{SU}_{\mathrm{int}}^{(i)}, \tag{3a}$$

$$\overline{\mathrm{SU}}_{\mathrm{ext}}^{(i)} = \alpha_1 \cdot \overline{\mathrm{SU}}_{\mathrm{ext}}^{(i-1)} + (1 - \alpha_1) \cdot \mathrm{SU}_{\mathrm{ext}}^{(i)}, \tag{3b}$$

where α_1 is the smoothing factor, $\overline{\mathrm{SU}}_{\mathrm{int}}^{(i)}$ ($\overline{\mathrm{SU}}_{\mathrm{ext}}^{(i)}$) is the average internal (external) slot utilization estimated at the end of the ith observation period, and $\mathrm{SU}_{\mathrm{int}}^{(i)}$ ($\mathrm{SU}_{\mathrm{ext}}^{(i)}$) is the internal (external) slot utilization measured during the ith observation period using formula (2a) (2b).

Exploiting the $\overline{\mathrm{SU}}_{\mathrm{int}}$ and $\overline{\mathrm{SU}}_{\mathrm{ext}}$ estimates we can easily compute the probability P_T of executing a transmission attempt granted by the standard backoff process by implementing the classical formula proposed in [16]:

$$P_T = 1 - \left[\min\left(1, \frac{\overline{\mathrm{SU}}_{\mathrm{int}} + \overline{\mathrm{SU}}_{\mathrm{ext}}}{\mathrm{ACL}(q)}\right)\right]^{N\_A}. \tag{4}$$

Since the $\mathrm{ACL}(q)$ value depends almost only on the average frame size q and it does not depend on the number of stations in the network, as proved in [16], the $\mathrm{ACL}(q)$ values for different frame sizes can be stored a priori inside the radio interface card. Similarly to the slot utilization computation, we prefer to use an average P_T value, which is obtained by applying a smoothing function to the outcomes of expression (4). In particular, let us assume that the jth backoff interval is terminated (i.e., the backoff counter is zero). Then, it follows that

$$\overline{P_T}^{(j)} = \alpha_2 \cdot \overline{P_T}^{(j-1)} + (1 - \alpha_2) \cdot P_T^{(j)}, \tag{5}$$

where α_2 is the smoothing factor, $\overline{P_T}^{(j)}$ is the average probability of transmission to use when deciding whether performing the transmission attempt or not, and $P_T^{(j)}$ is the probability of transmission computed according to formula (5). It is worth noting that it should be $\alpha_2 > \alpha_1$ because the

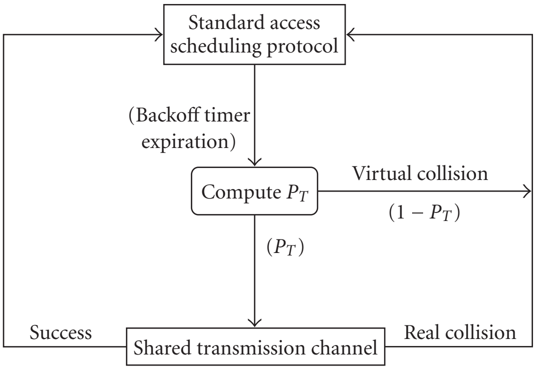


FIGURE 1: Block diagram of the implemented AOB protocol.

P_T value is updated after each backoff interval, therefore significantly more often than the SU, which is updated only after each observation interval T (in our implementation, we employed $\alpha_1 = 0.9$ and $\alpha_2 = 0.95$).

Figure 1 depicts the flow diagram outlining the different components that have been defined to implement the AOB MAC protocol, and the relationships between the blocks.

As illustrated in Figure 1, the AOB implementation requires to compute the P_T value according to formulas (4) and (5), and to keep updating the $\overline{\text{SU}}_{\text{int}}$ and $\overline{\text{SU}}_{\text{ext}}$ estimates using formulas (3a) and (3b). However, to implement the extensions to the AOB protocol designed in the MobileMAN project, we have to develop additional modules capable of collecting credits. As described in [17], each station should earn credits when it releases a transmission opportunity granted by the standard basic access mechanism. These credits, in turn, are spent to perform additional high-priority transmission attempts. More precisely, let us assume that the jth backoff interval is terminated (i.e., the backoff counter is zero) and that the backoff timer was uniformly selected in the range $[0, \dots, CW(k) - 1]$, where $CW(k) = \min(2^{k-1}, 2^{k_{\text{MAX}}}) \cdot CW_{\text{MIN}}$. If, according to the probability of transmission P_T, the station releases its transmission opportunity granted by the standard backoff procedure, the new contention window used to reschedule the frame transmission will be $CW(k+1) = \min(2^k, 2^{k_{\text{MAX}}}) \cdot CW_{\text{MIN}}$. Thus, after the virtual collision the number of credits CR collected by that station will be

$$CR = CR_{\text{old}} + \min(2^k, 2^{k_{\text{MAX}}}), \tag{6}$$

where CR_{old} is the number of credits owned by the station before the virtual collision.

Each station should use the collected credits to perform consecutive transmission attempts separated by *SIFS* intervals. The analytical and simulative studies conducted in [17, 19] have demonstrated that the use of multiple consecutive transmissions regulated by considering the credits owned by each station is an effective technique to mitigate some of the fairness problems arising when the AOB protocol is used in multihop networks or heterogeneous WLANs. In addition, using frame bursting is also beneficial to improve the efficiency of the 802.11 MAC protocol and to increase the throughput performances. Indeed, frame bursting is one of the new features that the IEEE standardization bodies are considering to be added in the next generation of 802.11 products (see the IEEE TGn and its draft specifications [3]). Note that implementing all the logic required to support and to manage the frame bursting operations has been one of the most difficult challenges to address during the card development.

As explained in [17], the number of credits needed to perform consecutive transmissions should depend on the average backoff interval. More precisely, each station estimates the average backoff interval that the standard backoff scheme would use in the case that no filtering of the channel access is implemented. To accomplish this estimation, it is useful to recall that the collisions suffered from stations using the AOB protocol can be either virtual collisions, when a station voluntarily defers a transmission attempt, or real collisions, when a station performs the transmission attempt but it does not receive the MAC ACK frame. Let us assume that the total number of transmission opportunities assigned to a station before the successful transmission is K, and that K_{rc} have been the real collisions occurred. Hence, $K - K_{rc}$ have been the virtual collisions, that is, the released transmission opportunities. Denoting with $\overline{CW}^{(j)}_{\text{enh}}$ the average contention window estimated after the jth successful transmission, and with $\overline{CW}^{(j)}_{\text{std}}$, average contention window of the equivalent standard MAC protocol estimated after the jth successful transmission, we have that

$$\overline{CW}^{(j)}_{\text{enh}} = \alpha_2 \cdot \overline{CW}^{(j-1)}_{\text{enh}} + (1 - \alpha_2) \cdot \frac{\sum_{k=1}^{K} CW(k)}{K}, \tag{7a}$$

$$\overline{CW}^{(j)}_{\text{std}} = \alpha_2 \cdot \overline{CW}^{(j-1)}_{\text{std}} + (1 - \alpha_2) \cdot \frac{\sum_{k=1}^{K_{rc}} CW(k)}{K_{rc}}. \tag{7b}$$

Note that the rightmost term in formula (7a) is the simple average of the contention windows used during the jth successful transmission. An exponential moving average filter is then employed to smooth the fluctuations of the average contention window adopted during the network operations. The $\overline{CW}^{(j)}_{\text{std}}$ value will be used as threshold to decide if the station has enough credits to perform a transmission attempt. We denote with AOB-CR the standard AOB protocol enhanced with the capabilities of collecting credit and using these credits to regulate the duration of frame transmission bursts. Figure 2 depicts the flow diagram outlining the different components that have been defined to implement the AOB-CR MAC protocol and the relationships between the blocks.

As shown in Figure 2, when the station performs a successful transmission attempt, it should compare the available credits against the CW_{std} threshold, computed according to formula (7b). If $CR > CW_{\text{std}}$, the station should transmit a burst of frames rather than a single frame. Two consecutive transmission attempts within the same burst are separated by a *SIFS* interval such as to guarantee that these additional frame transmissions have higher priority than other node's transmission attempts. It is intuitive to observe that

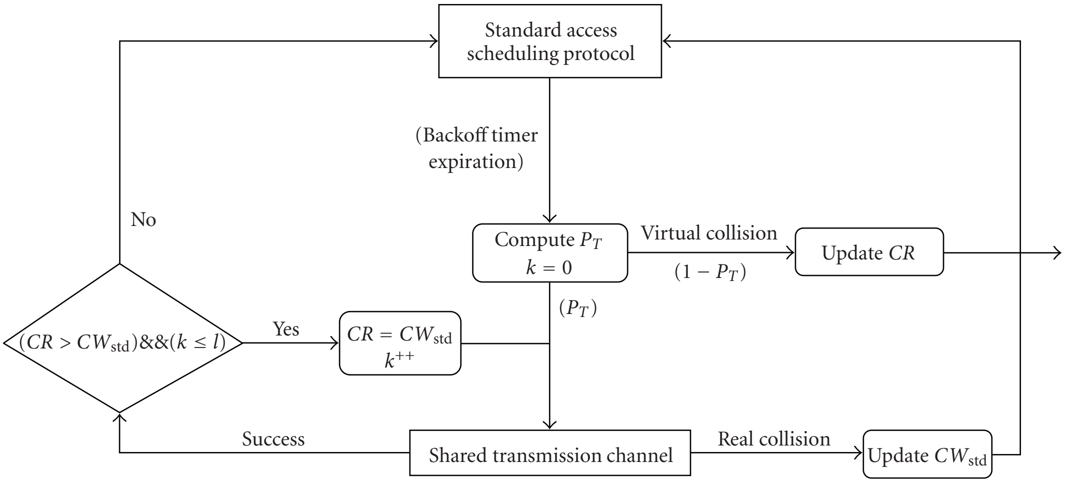


Figure 2: AOB-CR protocol with credit collection and frame bursting.

transmission bursts can induce short-term unfairness in the network. To mitigate this shortcoming we establish a maximum burst size of l frames. In other words, no more than l consecutive frames can be transmitted before the standard backoff procedure is applied again. It is out of the scope of this work to define optimal and adaptive strategies to set the threshold l. For this reason in our implementation we adopted the simplest approach, namely, we set a fixed threshold of five frames. It is worth pointing out that transmitting a burst of frames should not affect the computation of the slot utilization. This implies that the entire burst is counted once in the computation of the n_{tx} value. Similarly, all the other stations consider the entire burst as a single channel occupation and they increment the n_{rx} value only once.

4. EXPERIMENTAL RESULTS

To validate our enhanced MAC architecture we carried out comparative tests of the performance achieved by the legacy IEEE 802.11 backoff mechanism and the enhanced ones, that is, the AOB protocol and the AOB-CR protocol. In all the experiments we use our NIC implementation both for the AOB-based solutions and for the standard 802.11 protocol. We decide to implement the original IEEE 802.11 standard at 2 Mbps and not the newer versions at higher speed (for instance 802.11b and 802.11g) due to hardware limitations, and in particular the unavailability of inexpensive and extendable modems implementing more sophisticated physical layers. All the tests are performed in a laboratory environment and we consider ad hoc networks in single-hop configurations. Nodes are communicating in ad hoc mode and the traffic is artificially generated. In our scenarios we have a maximum of four stations, due to hardware limitations. However, this is not a problem, because we are able to demonstrate the performance of our solution and the coherence with simulations conducted in previous work.

As discussed in Section 2, the average backoff value that maximizes the channel utilization is almost independent of the network configuration (number of competing stations), but depends only on the average packet sizes [16]. Therefore, the ACL(q) value for the frames size used in our experiments can be precomputed and loaded in the MAC firmware. The implementation in software of the algorithm used to compute the ACL(q) value such as to evaluate it at run time is an ongoing activity.

The network scenarios used during the experiments consist of 2, 3, and 4 stations. The stations are identically programmed to continuously send 500-byte-long MSDUs (MSDU denotes the frame payload). The consecutive MSDU transmissions are separated by at least one backoff interval and we did not use the RTS/CTS handshake or the fragmentation. The minimum contention window was $8 \cdot t_{\text{slot}}$ ($160\,\mu$s). This value does not comply with the original IEEE 802.11 standard (although, it fits with more recent implementations), but it was hardwired in the modem firmware we used in our card prototype. However, the minimum contention window value affects only the absolute value of our measurements, but not the general trends.

The nodes topology is illustrated in Figure 3. All the experimental results we show henceforth are obtained by computing the average over five replications of the same test and considering stationary conditions.

As already demonstrated in [16, 17] the AOB mechanism introduces a minimum overhead that could negatively affect the performance of the communications between two stations. However, the frame bursting is useful to reduce the protocol overheads because it permits transmitting frames with null backoff. Thus, our first set of experiments was carried out to verify the performance decrease caused by the AOB protocol in a network configuration where two stations are performing a bidirectional communication, as illustrated in Figure 4. In addition, we conducted similar tests

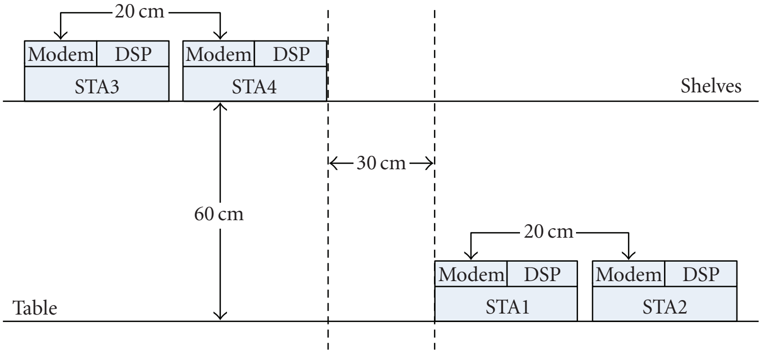


Figure 3: Node topology used in the measurements.

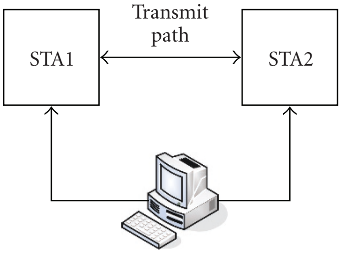


Figure 4: Bidirectional communications with two stations.

Table 1: Results for the two-station scenario.

	Average	Standard deviation	Throughput increase
Standard 802.11 MAC protocol	1546.19 kbps	108 bps	—
AOB protocol	1510.62 kbps	256 bps	−2.3%
AOB-CR protocol	1694.93 kbps	91 bps	+9.6%

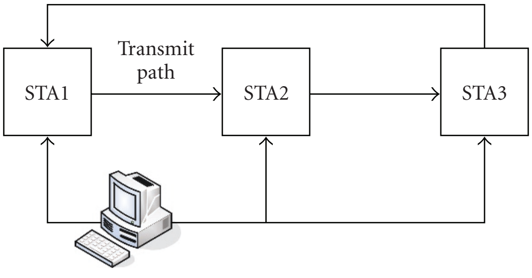


Figure 5: Three-station scenario.

to validate if the AOB-CR protocol is effective in improving the MAC protocol efficiency.

The results we obtained in this two-station configuration are reported in Table 1. In particular, the throughput at time $k \cdot T$ (where T is the sampling period equal to 100 ms) is computed as

$$TP[k \cdot T] = DT[k \cdot T] - RC[k \cdot T], \tag{8}$$

where $DT[k \cdot T]$ is the total number of frames sent to a station (either acknowledged or not acknowledged frames), while $RC[k \cdot T]$ is the number of real collisions (not acknowledged frames). The average throughput values for each station are evaluated by the DSP, internally (thanks to the implemented buffer) after 8 minutes of continuous transmission. After some computations, the throughput value is sent to a PC through the available RS-232 channel. For validating the stochastic correctness of our result, both the average and the standard deviation of throughput measures are reported in the following tables.

From the numerical results listed in Table 1, we can observe that the throughput decrease with two competing stations is less than 3% when using the AOB protocol. However, the AOB-CR mechanism is capable of improving the MAC protocol efficiency, ensuring a 10% improvement in the throughput performance.

In the second set of experiments we considered a network configuration with three stations, as depicted in Figure 5.

The experimental results we obtained in the three-station configuration are reported in Table 2. We can note that with three competing stations, the throughput decrease with the AOB protocol is almost negligible. On the other hand, it is further confirmed that the AOB-CR protocol guarantees a significant improvement with respect to the standard 802.11 MAC protocol.

Finally, the last set of experiments was carried out in the four-station scenario depicted in Figure 6, and the experimental results we measured are listed in Table 3.

These results confirm the positive trends shown in the previous experiments. In particular, with four competing stations, the AOB protocol provides a higher throughput than the standard MAC protocol. The reason is that the filtering on channel access reduces the collision probability such as that the stations can utilize more efficiently the channel resources. Furthermore, the AOB-CR protocol continues to

Table 2: Results for the three-station scenario.

	Average	Standard deviation	Throughput increase
Standard 802.11 MAC protocol	1521.32 kbps	208 bps	—
AOB protocol	1517.36 kbps	974 bps	−0.26%
AOB-CR protocol	1706.34 kbps	279 bps	+12.1%

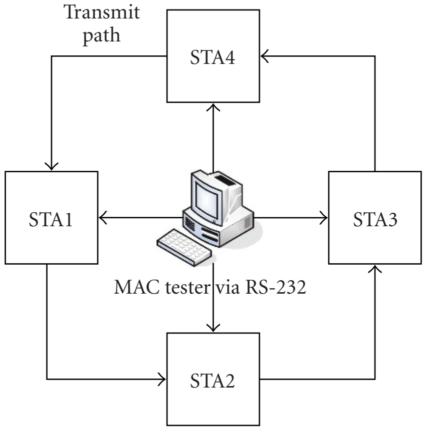


Figure 6: Four-station scenario.

Table 3: Results for the four-station scenario.

	Average	Standard deviation	Throughput increase
Standard 802.11 MAC protocol	1434.31 kbps	290 bps	—
AOB protocol	1504.0 kbps	242 bps	+4.85%
AOB-CR protocol	1681.03 kbps	451 bps	+17.5%

show better performance than the basic AOB mechanism. In the four-station scenario the throughput increase provided by the AOB-CR protocol over the standard 802.11 MAC protocol is about 17%.

The shown results clearly demonstrate that the AOB MAC protocol improves the per-station throughput as the number of stations increases, such as to approximate the maximum channel utilization. In addition, the introduction of credit-based frame-bursting capabilities permits to further increase the MAC protocol efficiency. A final remark is on the implicit capability of the AOB scheme to mitigate the negative impact of external interferences. In fact, the standard 802.11 MAC control cannot distinguish between a frame loss caused by a collision event or channel noise. Therefore, channel errors induce an increase in the backoff window as in the case of frame collisions. For this reason, when the channel is noisy, even if there are a few stations in the network, the number of retransmissions needed to successfully transmit a frame can be high. However, it is well consolidated that the standard 802.11 MAC protocol is highly inefficient when the contention level in the network is nonnegligible. On the contrary, the AOB protocol guarantees an optimal spreading of the channel access independently of the network contention level and of the number of retransmissions. The adaptability of the AOB scheme to the channel noise level explains the reason why we measured during the experiments relative improvements of per-station throughput bigger than the ones predicted by theoretical analysis. In fact, the model developed in [16] assumed ideal channel conditions and no channel errors. On the other hand, our experiments where conducted in a realistic laboratory environment where other radio sources were radiating signals in the ISM frequency band and interfering with the 802.11 frame transmissions. While the standard 802.11 MAC protocol suffered from significant throughput degradations due to this interference, our proposed credit-based extension of the AOB protocol still achieves quite good performance.

5. CONCLUSIONS

Experiments were carried out with the implementation of an enhanced IEEE 802.11 MAC card adopting the optimizations designed in [16, 17]. The card is still fully compatible with current implementations of the IEEE 802.11 technology because the radio part is compliant to the 802.11 standard. However, the presented experimental results show that the enhanced mechanism outperforms the standard 802.11 MAC protocol in real scenarios. We have also shown that the advantages of this mechanism go further than the high contention scenarios (e.g., ad hoc networks), for which it was designed, because it is also effective in lessening the negative impact of the external interferences, which traditionally decrease the performances of wireless networks in any environment.

We believe that the contributions of our work can go well beyond the implementation and testing of a specific enhanced 802.11 backoff algorithm. In fact, the NIC platform we have developed during the MobileMAN project represents a flexile and versatile hardware/software system that can be used to explore a variety of new research directions. In particular, prior work has advocated the use of cross-layering for the optimization of ad hoc network performance. It is intuitive to observe that in a cross-layered architecture the MAC layer has a fundamental role. In fact, the MAC layer could distribute “physical” information up to the higher levels, as well as it may profit from some higher layer elaborations too complex to be performed at MAC. A typical example is the interaction between MAC, routing, and transport information for congestion and network utilization purposes. If the transport is aware of the links’ status, it can distinguish between congestion due to physical failures and congestion due to the amount of traffic, such as to take the most appropriate actions to deal with these conditions. Similarly, the routing can decide different routing paths or strategies, and the MAC can modify the distribution of some information as consequence. Therefore, we are currently working

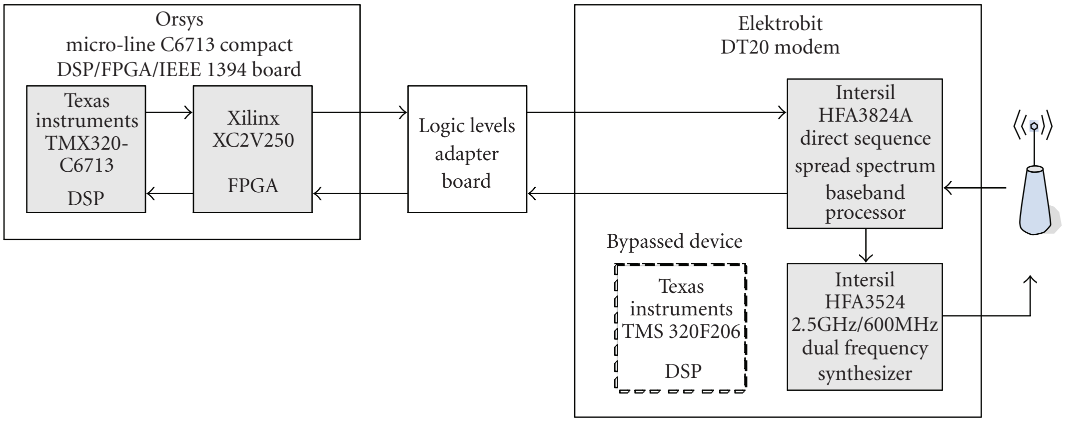


Figure 7: Overview of the enhanced 802.11 wireless network interface (PHY).

on the design of a shared memory component acting as exchange area of networking information (parameters, status, etc.) for all the layers.

APPENDIX

HARDWARE DESIGN OF THE MOBILEMAN NIC

Generally speaking, a wireless NIC has three main functional blocs: the MAC, the baseband (BB), and the radio frequency (RF). Since the main part of the conceptual work conducted in our activities is concentrated on the MAC protocol, we decided to use off-the-shelf solutions for the BB and RF parts. For these reasons, we acquired a board, called DT20 modem, produced by the Elektrobit, which implements the 802.11 PHY with the Prism I chipset produced by the Intersil. Note that at the time we started the card development, this company was the world leader manufacturer of the chipsets for wireless network interface cards.

Concerning the MAC protocol, given that our goal was to develop a new backoff algorithm over the 802.11 standard and not to entirely redesign the standard channel access mechanisms, we tried to find a flexible development platform providing an implementation of the legacy 802.11 standard. Unfortunately, the platforms provided by the major producers of wireless NICs were too expensive or with a very limited set of possible enhancements. Thus, we were forced to implement the 802.11 MAC standard from scratch. In addition, we needed a development platform ensuring a great flexibility. For these reasons, we tried to find a development platform that could fulfill the following constraints:

(i) an easy, well known, and tested development environment to speed up as much as possible the implementation of the 802.11 standard,

(ii) the possibility to develop some MAC functionalities directly in hardware to fulfill the timing constraints imposed by the 802.11 standard [1],

(iii) a processor with high performances for new and future implementations.

At the end the solution that best fitted our criteria was the Orsys Micro-line C6713Compact DSP board. The hardware overview of the enhanced wireless network interface card, integrating both the DSP board and the DT20 modem, is shown in Figure 7.

The DSP board integrates a Texas instruments TMX-320C6713 DSP and an FPGA (Xilinx XC2V250) that is very important for the implementation of the protocol functionalities characterized by stringent time constraints. Due to the fact that the DSP board and the DT20 modem board have different logic levels, 3.3 V and 5 V, respectively, a logic level adapter has been developed to allow the communication between the boards.

Implementation

The part of the 802.11 MAC protocol implemented in the C6713 DSP has been realized in standard C. On the other hand, the communication layer between the DSP and the modem has been developed on the FPGA device. Note that the FPGA module has a large computational power and it could be used in the future to accelerate other tasks (e.g., address filtering, cryptography, etc.). A more detailed overview of the interface at logic block level is presented in Figure 8.

The specific interfaces are as follows.

(i) *HFA3824A RX/TX interface*: this block operates as glue logic between the McBSP (multichannel buffered serial port) serial interface available on the DSP and the serial receive and transmit ports of the HFA3824A baseband processor.

(ii) *HFA3824A/HFA3524 control port interface*: this block is used as an interface between the DSP and the control port of the HFA3824A device. In particular, this component exploits the functionalities of the external memory interface (EMIF) found on TMS DSP devices, which normally is used to connect the DSP to different types of memory devices (SRAM, Flash RAM, DDR-RAM, etc.). In our application, the EMIF connects to the FPGA, which performs as communication interface with the modem. Through this interface,

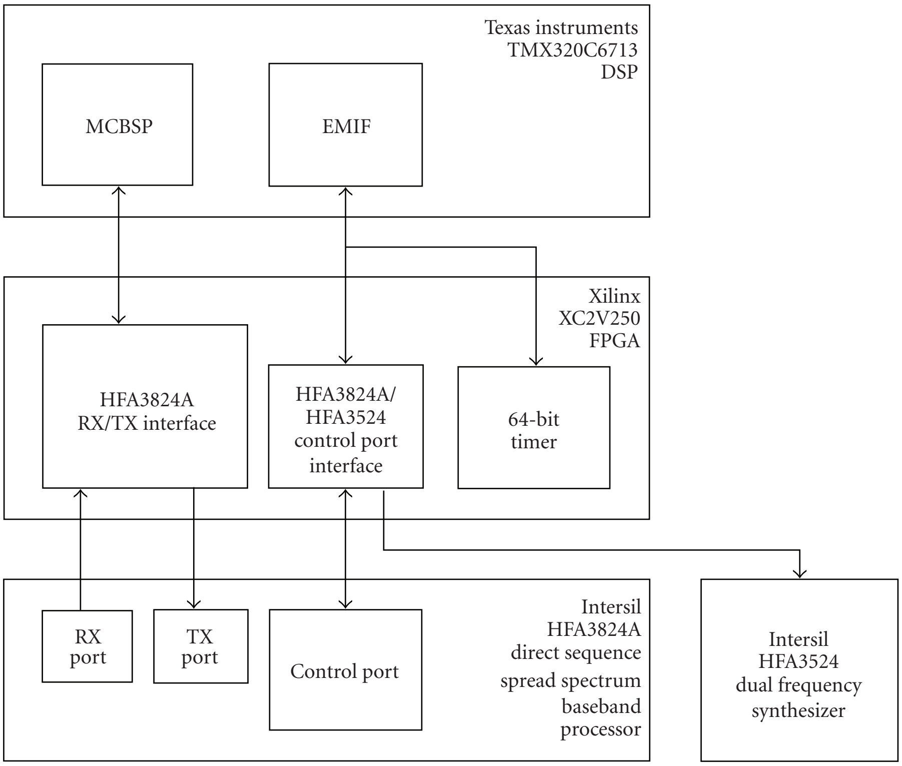


Figure 8: Logic block diagram of the MAC implementation. Note that only three functional blocks have been implemented in the FPGA.

the baseband processor and the dual frequency synthesizer can be configured.

(iii) *64-BIT TIMER*: this is a 64-bit timer that is used during the management procedures invoked at the end of 802.11 frame transmission and reception events.

The firmware was realized in such a way to maintain the maximum possible level of abstraction and to minimize the software redesign in case of change of the development platform. Thus, only few software components are specific to the C6713Compact board; among these are timing considerations, available DSP resources, configuration and control related to the specific implementation (i.e., we could not implement a general abstraction at the source code level).

The PHY firmware is subdivided into the following components.

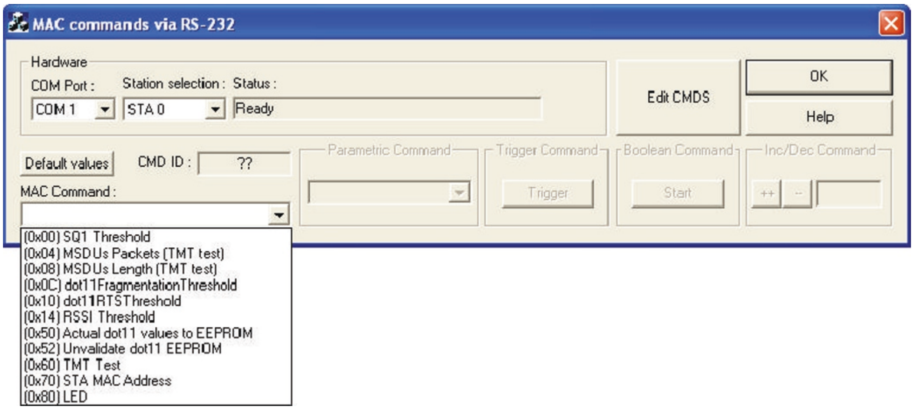
(i) *MAC firmware:* is the hard real-time software, which allows packets (fragments) to be physically transmitted and received to and from the RF interface. This part implements both the 802.11 legacy standard and the new backoff algorithm in order to allow mixed environment experiments, where enhanced systems cooperate with standard off-the-shelf components.

(ii) *Host interface firmware.* This software component is less stringent in terms of real-time requirements.

(iii) *Packet data structure.* The data structure is the communication channel between MAC firmware and host interface firmware; it is a vital part of the MobileMAN project since it allows the cross-layering functionalities between PHY/MAC and upper layers.

Nevertheless, the firmware comes without an operating system, which was not needed for the implementation of the standard 802.11 frame exchange sequence and relative tasks (fragmentation, defragmentation, fragment cache control, etc.). This is pretty a good step in direction of a better portability of the source code. On the actual system (C6713Compact board), the firmware occupies about 125 Kbytes and can reside completely in the DSP internal RAM, at run time.

The system may be used in lab environment (through the development system and the JTAG interface) during synthetic traffic tests, and it may also be used in a real environment, by using the high speed IEEE1394 bus which allows the full speed connection with a host PC. A specific PC application has been also developed to control and test the NIC when it is running as a stand-alone system (without connecting an emulator and without using the *TI code composer* as control environment). With this small and simple application, MAC parameters (e.g., station MAC address, signal quality thresholds, synthetic packets generation control) are fully accessible and can be changed by simply connecting a PC to the system with a RS-232 cable. Commands to the MAC system can be fully edited and sent with specific parameters as shown in Figures 9 and 10.



COM Port : COM 1

Station selection : STA 0

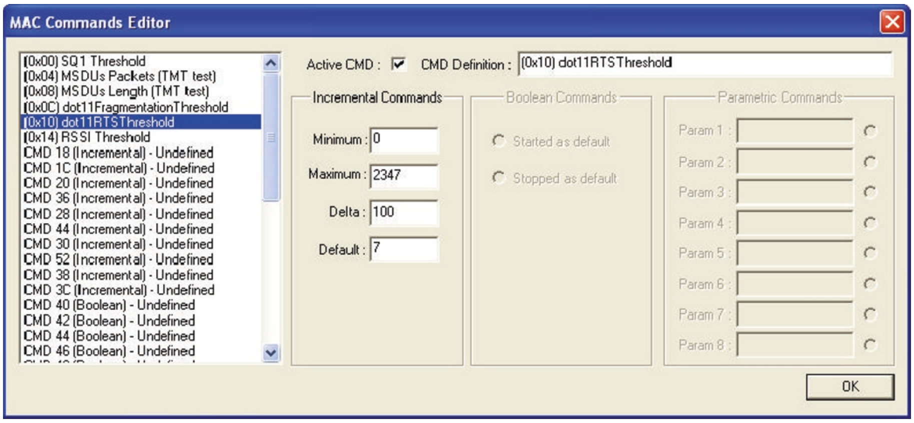
Status : Ready

CMD ID : ??

MAC Command :

(0x00) SQ1 Threshold
(0x04) MSDUs Packets (TMT test)
(0x08) MSDUs Length (TMT test)
(0x0C) dot11FragmentationThreshold
(0x10) dot11RTSThreshold
(0x14) RSSI Threshold
(0x50) Actual dot11 values to EEPROM
(0x52) Unvalidate dot11 EEPROM
(0x60) TMT Test
(0x70) STA MAC Address
(0x80) LED

Figure 9: MAC commands via RS-232.



(0x00) SQ1 Threshold
(0x04) MSDUs Packets (TMT test)
(0x08) MSDUs Length (TMT test)
(0x0C) dot11FragmentationThreshold
(0x10) dot11RTSThreshold
(0x14) RSSI Threshold
CMD 18 (Incremental) - Undefined
CMD 1C (Incremental) - Undefined
CMD 20 (Incremental) - Undefined
CMD 36 (Incremental) - Undefined
CMD 28 (Incremental) - Undefined
CMD 44 (Incremental) - Undefined
CMD 30 (Incremental) - Undefined
CMD 52 (Incremental) - Undefined
CMD 38 (Incremental) - Undefined
CMD 3C (Incremental) - Undefined
CMD 40 (Boolean) - Undefined
CMD 42 (Boolean) - Undefined
CMD 44 (Boolean) - Undefined
CMD 46 (Boolean) - Undefined

Active CMD : ☑

CMD Definition : (0x10) dot11RTSThreshold

Minimum : 0

Maximum : 2347

Delta : 100

Default : 7

Figure 10: MAC commands editor.

REFERENCES

[1] ANSI/IEEE Std. 802.11: Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) Specifications, August 1999.

[2] ANSI/IEEE Std. 802.11*e*: Wireless LAN Medium Access Control (MAC) and Physical Layer PHY) Specification/Amendment 8: Medium Access Control (MAC) Quality of Service Enhancements, September 2005.

[3] Y. Xiao, "IEEE 802.11*n*: enhancements for higher throughput in wireless LANs," *IEEE Wireless Communications*, vol. 12, no. 6, pp. 82–91, 2005.

[4] R. Karrer, A. Sabharwal, and E. Knightly, "Enabling large-scale wireless broadband: the case for TAPs," *Computer Communication Review*, vol. 34, no. 1, pp. 27–32, 2004.

[5] I. Chlamtac, M. Conti, and J. J.-N. Liu, "Mobile ad hoc networking: imperatives and challenges," *Ad Hoc Networks*, vol. 1, no. 1, pp. 13–64, 2003.

[6] G. Anastasi, M. Conti, and E. Gregori, "IEEE 802.11 ad hoc networks: protocols, performance and open issues," in *Ad Hoc Networking*, IEEE Press, John Wiley & Sons, New York, NY, USA, 2003.

[7] F. Calí, M. Conti, and E. Gregori, "Dynamic tuning of the IEEE 802.11 protocol to achieve a theoretical throughput limit," *IEEE/ACM Transactions on Networking*, vol. 8, no. 6, pp. 785–799, 2000.

[8] C. Chaudet, D. Dhoutaut, and I. Guérin Lassous, "Performance issues with IEEE 802.11 in ad hoc networking," *IEEE Communications Magazine*, vol. 43, no. 7, pp. 110–116, 2005.

[9] S. Xu and T. Saadawi, "Does the IEEE 802.11 MAC protocol work well in multihop wireless ad hoc networks?" *IEEE Communications Magazine*, vol. 39, no. 6, pp. 130–137, 2001.

[10] S. Xu and T. Saadawi, "Revealing the problems with 802.11 medium access control protocol in multi-hop wireless ad hoc networks," *Computer Networks*, vol. 38, no. 4, pp. 531–548, 2002.

[11] G. Anastasi, E. Borgia, M. Conti, and E. Gregori, "Wi-fi in ad hoc mode: a measurement study," in *Proceedings of the 2nd IEEE Annual Conference on Pervasive Computing and Communications (PerCom '04)*, pp. 145–154, Orlando, Fla, USA, March 2004.

[12] C. Chaudet, D. Dhoutaut, and I. Guérin Lassous, "Experiments of some performance issues with IEEE 802.11b in ad hoc networks," in *Proceedings of the 2nd Annual Conference on Wireless On-Demand Network Systems and Services (WONS '05)*, pp. 158–163, St. Moritz, Switzerland, January 2005.

[13] G. Bianchi, "Performance analysis of the IEEE 802.11 distributed coordination function," *IEEE Journal on Selected Areas in Communications*, vol. 18, no. 3, pp. 535–547, 2000.

[14] L. Bononi, M. Conti, and L. Donatiello, "Design and performance evaluation of a distributed contention control (DCC) mechanism for IEEE 802.11 wireless local area networks," *Journal of Parallel and Distributed Computing*, vol. 60, no. 4, pp. 407–430, 2000.

[15] R. Bruno, M. Conti, and E. Gregori, "Optimization of efficiency and energy consumption in p-persistent CSMA-based wireless LANs," *IEEE Transactions on Mobile Computing*, vol. 1, no. 1, pp. 10–31, 2002.

[16] L. Bononi, M. Conti, and E. Gregori, "Runtime optimization of IEEE 802.11 wireless LANs performance," *IEEE Transactions on Parallel and Distributed Systems*, vol. 15, no. 1, pp. 66–80, 2004.

[17] R. Bruno, M. Conti, and E. Gregori, "Distributed contention control in heterogeneous 802.11b WLANs," in *Proceedings of the 2nd Annual Conference on Wireless On-Demand Network Systems and Services (WONS '05)*, pp. 190–199, St. Moritz, Switzerland, January 2005.

[18] R. Bruno, M. Conti, and E. Gregori, "Optimal capacity of p-persistent CSMA protocols," *IEEE Communications Letters*, vol. 7, no. 3, pp. 139–141, 2003.

[19] R. Bruno, C. Chaudet, M. Conti, and E. Gregori, "A novel fair medium access control for 802.11-based multi-hop ad hoc networks," in *Proceedings of the 14th IEEE Workshop on Local and Metropolitan Area Networks (LANMAN '05)*, p. 6, Chania, Crete, Greece, September 2005.

[20] R. Bernasconi, R. Bruno, I. Defilippis, S. Giordano, and A. Puiatti, "Experiments with an enhanced MAC architecture for multi-hop wireless networks," in *Proceedings of the 1st IEEE ICPS Workshop on Multi-Hop Ad Hoc Networks: From Theory to Reality (REALMAN '05)*, Santorini, Greece, July 2005.