Universiteit Antwerpen

This item is the archived peer-reviewed author-version of:

Bounds selection: dynamic reset protocol for wireless ad hoc LANs

Reference:

Bounds Selection - Dynamic Reset Protocol for Wireless Ad Hoc LANs

Sylwia Romaszko
University of Antwerp - IBBT
Middelheimlaan 1
B-2020 Antwerpen, Belgium
Email: sylwia.romaszko@ua.ac.be

Chris Blondia
University of Antwerp - IBBT
Middelheimlaan 1
B-2020 Antwerpen, Belgium
Email: chris.blondia@ua.ac.be

Abstract— We propose the "Bounds selection - Dynamic Reset Algorithm" (sB-DRA) protocol for wireless ad hoc LANs. The sB-DRA consists of the selection/adjustment of the Contention Window (CW) interval (selection Bounds algorithm - *sB* from our previous work [14]) and the resetting of the Contention Window value dynamically after a dropped packet or successful transmission (*enhanced* Dynamic Resetting Algorithm - *DRA*). The *sB* algorithm allows to adjust the lower and upper bounds of the CW interval taking into account the number of 1-hop active neighbours and the number of transmission attempts during the recovery mechanism. The *DRA* considers the number of 1-hop neighbours and their change (fast/slow increase/decrease) during recovery process. The *enhanced* DRA algorithm can be tuned dynamically depending on the channel conditions. The sB-DRA protocol achieves better throughput and significantly reduces the number of collisions and packet loss compared to the IEEE 802.11 DCF standard [1]. We have also shown that the sB algorithm outperforms the *basic* (without *resetting the Backoff ranges* mechanism) "Deterministic Contention Window Algorithm" (DCWA) [17] in mobile ad hoc networks.

I. INTRODUCTION

IEEE 802.11 [1] has become the standard for Wireless Local Area Networks. It specifies the contention-based MAC mechanism, Distributed Coordination Function (DCF) for ad hoc and infrastructure network configurations. The DCF is based on CSMA/CA -Carrier Sense Multiple Access with Collision Avoidance. However, the nature of the infrastructure environment is far from the real ad hoc conditions. Deploying the 802.11 DCF in wireless multi-hop networks leads to inefficient utilization of the bandwidth and energy due to many unnecessary collisions. The IEEE 802.11 specifies the *Binary Exponential Backoff (BEB)* algorithm, where the duration of a backoff period is selected randomly in a range limited by zero and a certain *CW* value. After each collision this value is doubled ($CW_{NEW} = 2 * (CW_{OLD} + 1)$). This solution is not only unfair but also inefficient. When the number of active neighbours increases, the number of collisions increases as well. Although the *CW* size is doubled after each collision, too many stations can back-off with small contention windows, because they can still pick up a slot randomly from the *[0,CW]* interval. On the other hand, the *CW* is set to the minimum upon a successful transmission in the DCF mechanism. However, receiving a packet successfully does not say anything about the contention level, only about picking (randomly) a

convinient CW value luckilly.

Therefore, in this work, first, we focus on a better selection of a lower and upper bound of the Contention Window interval. The aim of this selection is to decrease the probability of collisions and prolonging the lifetime of the network. Secondly, to improve the performance upon successful transmission and dropped packet cases we combine this selection Bounds algorithm with the *enhanced* Dynamic Resetting (Contention Window) Algorithm.

In order to avoid the hidden and exposed problems in the wireless medium, the CSMA/CA protocol is extended with a virtual carrier sensing mechanism, namely Request-To-Send (RTS) and Clear-To-Send (CTS) control packets. The nodes send these packets in order to reserve the medium before the data transmission. However, in [10] has been indicated that phenomena occuring at the physical layer makes the effectiveness of the RTS/CTS exchange arguable, because the hidden station phenomenon rarely occurs. In [5], [6] the usefulness of the current RTS/CTS mechanism is under discussion in IEEE 802.11 ad hoc networks. Therefore, we have executed simulations with both the Basic access scheme and RTS/CTS scheme.

The remainder of the paper is organized as follows. In Section II the related work is presented. In Section III the sB-DRA protocol is described. In Section IV we provide simulation results and analysis. Finally, concluding remarks are formulated in the last section.

II. RELATED WORK

Many approaches have been proposed to reduce the number of collisions by substituting the binary exponential backoff algorithm of the IEEE 802.11 by novel backoff approaches or selecting an intermediate value instead of resetting the CW value to its initial (minimum) value. In [7], [8], [9] feedback-based mechanisms have been shown for adapting the backoff algorithm and maximizing channel utilization. All these works are based on analytic models of the IEEE 802.11 network, providing the optimal setting of the backoff parameters for achieving the best channel utilization. The Fast Collision Resolution MAC protocol has been described in [19]. The protocol allows the latest successful nodes to use a smaller *CW* and some nodes can reduce their backoff

timer exponentially when they continuously meet idle time slots. In [3], the authors have shown that the number of contending nodes strongly affects the optimal *CW* size and they have proposed the Multiplicative Increase and Linear Decrease algorithm. However, the throughput degradation is observed when the number of nodes is large and the number of active nodes changes sharply from high to low. In [18], a similar scheme has been designed, called Linear/Multiplicative Increase and Linear Decrease. It uses an additional piece of information, namely, the overheard collisions. In [12] the Asymptotically Optimal Backoff (AOB), dynamically adapting the backoff window size to the current network contention level has been proposed. There are still many other interesting works in this field.

Most of previous and recent works focus on new backoff schemes or resetting algorithms. Only several works, mentioned below, abandon this approach and interest with changing a lower and upper bound of CW interval or substituting the picking slot uniformly at random by other solutions. The first work on this issue has started in the first drafts of the IEEE 802.11 Enhanced DCF [2] mechanism. The EDCF is designed to provide differentiated, distributed channel access for frames with 8 different priorities by enhancing the DCF. The EDCF specifies, inter alia, the $CW_{min}[AC]$, and $CW_{max}[AC]$, where the AC value represents 4 different access categories (best effort, video probe, video and voice). Depending on the particular access category the values of the $CW_{min}[AC]$ (or $CW_{max}[AC]$) are different. Thus, particular terminals choose various values of minimum and maximum CW interval taking into account the type and priority of their traffic. However, in the EDCF, the lower bound of the contention window always remains 0. In [16] the Sift protocol has been proposed for event-driven wireless sensor networks, where a fixed-size CW is used with a non-uniform, geometrically-increasing probability distribution for picking a transmission slot in the contention window. In [15] the Predictable Random Backoff (PRB) algorithm has been proposed to mitigate the selfish MAC misbehaviour. In this work a node selects a lower bound of CW (! = 0) for the next CW_{i+1} upon a successful data transmission if the CW_i and $\alpha * CW_i$ are less than the CW_{thresh} (see details in [15]). Differently from [15] we distinguish different bounds only during the recovery mechanism (upon unsuccessful transmissions) in our proposed protocol. The most relevant work to the first part of our protocol is [17]. In [17] the Deterministic Contention Window Algorithm (*DCWA*) increases upper (CW_{ub}) and lower bounds (CW_{lb}) instead of just doubling the upper bound of the CW. The backoff timer is randomly selected from the range delimited by CW_{lb} and CW_{ub}, where:
(Stage 0) $CW_{lb}(0) = 0$ *and* $CW_{ub} = CW_{min}$ *and Size = 32;*
(Stage n) $CW_{lb}(n) = CW_{ub}(n-1)$ *and*
$CW_{ub}(n) = CW_{ub}(n-1) + Size$ *and* $Size = 32 * n;$
If $CW_{ub}(n) > CW_{max}$ *then* $CW_{ub} = CW_{max}$, $Size = 256$
and $CW_{lb} = CW_{lb} = CW_{max} - Size;$
Thus, in each contention stage, a station draws a backoff interval from a distinct backoff range that does not overlap with the other backoff ranges associated to the other contention stages. After all, in the second part of the DCWA protocol, the backoff range is re-adjusted upon each successful transmission by taking into account the current network load and history (*resetting the Backoff ranges* mechanism -see details in III.B section [17]). Thus, after each successful transmission the CW value is not reset to its initial value, only to an intermediate value, where at the same time, the backoff range size is initialized to the minimum value (32).

III. PROPOSED PROTOCOL

In the IEEE 802.11 DCF mechanism a *CW* is used by a node in order to control the backoff window. Each node picks randomly a contention slot from the [0, *CW*] interval, where the *CW* size depends on the number of retransmission attempts (usually caused by collisions). After each unsuccessful transmission retry the *CW* is doubled. Although the CW is doubled, there is always a probability that contending nodes choose the same contention slots randomly, especially when the number of active nodes increases. On the other hand, upon a successful transmission or upon a maximum transmission attempts, the *CW* is reset to the minimum value according to the 802.11 standard. However, receiving a packet successfully does not mean that the contention level is dropped. In case of a dropped packet this assumption is even more doubtful.

To cope with both issues we have designed the "Bounds selection and Dynamic Resetting Algorithm" (*sB-DRA*) protocol which takes care of the CW issues in both cases. Our protocol consists of two parts. The first part, the Bounds selection (*sB*) algorithm, considers an adjustment of the CW range and the CW backoff size after each transmission. The second part, the *enhanced* Dynamic Reset Algorithm (*DRA*), takes care of the CW resetting upon a successful transmission and dropped packet (when the number of retransmission attempts reaches the limit; notice that in [17] the *resetting the Backoff ranges* process is only considered after each successful transmission).

A. CW selection Bounds algorithm

In [14] we have proposed the three algorithms that modify the distribution of the CW size (lower bound, upper bound and both) based on the number of 1-hop neighbours and the number of transmission attempts. We have observed that the algorithm adjusting both bounds (we called it the *sB* algorithm in this work) outperforms the 802.11 DCF standard and the other two schemes. In the Bounds selection (sB) algorithm, first we let a node double the CW size, if the *BEB* algorithm is used (when a retransmission occurs), as in [1], and then the CW is adjusted. Afterwards, the backoff timer is randomly selected from the range delimited by the lower (*lB*) and upper bound (*uB*): $backoff\ timer = random[lB_i, uB_i]$.

The *sB* algorithm allows each node to change both a lower and upper bound of the *CW* interval as depicted in Fig. 1 (the numbers (*1, 2, 3, 4*) in this figure refer to the different possible
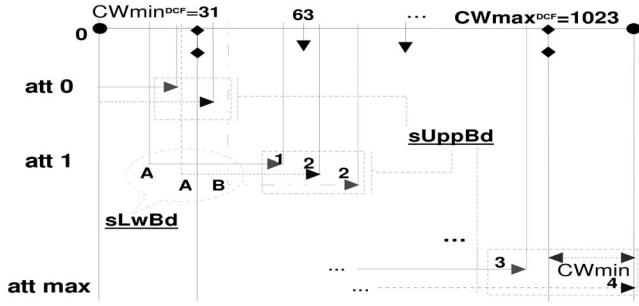
Fig. 1. *sB* algorithm

cases of un *upper bound*, and letters (*A, B*) in the figure, to a *lower bound*: larger (*B*) or lower (*A*) than the DCF upper bound).

First we search the best value of the *lower bound* (lB_i) for the *CW* size. Depending on the number of transmission attempts (nr_{ATT}) and number of 1-hop active neighbours [1] (Nr_{NEIGH}) the lower bound of CW will vary. Thus, the size of the CW interval changes dynamically depending on the Nr_{NEIGH} and the nr_{ATT}. Notice that in [17], the CW range sizes are already known and depend on the number of $stage_i$: $Size_i = 32 * i$. In the *sB* algorithm, a node with a lower nr_{ATT} can always access a set of earlier contention slots that cannot be accessed by a node with a larger nr_{ATT}.

The algorithm of the *sB* scheme is as follows:
If $Nr_{ATTEMPTS} == 0$ then   *// no retransmissions*

$$lB_i = 0; \quad uB_i = (uB_i)*log_{10}(Nr_{NEIGH}+\gamma); \quad (1)$$

If $Nr_{ATTEMPTS} > 0$ then   *// retransmissions*

$$lB_i = (\frac{uB_{i-1}}{2} + Nr_{NEIGH} + nr_{ATT}) * log_{10}(nr_{ATT} + \gamma); \quad (2)$$

where γ is a constant and equals 3.5.

$$uB_i = (uB_i)*log_{10}(Nr_{NEIGH}+nr_{ATT}+\gamma); \quad (3)$$

where γ is a constant and equals 3.5, if $Nr_{NEIGH} < 2$, and 0 otherwise. If $uB > CW_{max} + CW_{min}$ then $uB = CW_{max} + CW_{min}$. If a node resides in a dense network with many active nodes, this will reflect in a larger value of the lB_i, apart from the current nr_{ATT}. If a node has only a few active neighbours, the lB_i value will be small. Based on our simulations we have noticed that, first, the lB_i should not be too large when the nr_{ATT} equals 1, and secondly, the lB_i should not increase too fast (see details in [14]). The second, logarithmic part of our formula (1), takes care of this.

We also let each node shrink or extend the *upper bound* (uB_i). The uB_i is logarithmically dependent on the Nr_{NEIGH} and nr_{ATT}. This way we receive only a slight change (an increase or decrease) of the uB_i compared to the uB_i achieved by [1].

[1]Each node can estimate how many neighbours it has in its 1-hop neighbourhood, based on successfully detected signals or using the table that is built by the routing mechanism. In [12] the utilization rate of the slots (*Slot Utilization*) observed on the channel by each station is used for a simple, effective and low-cost load estimate of the channel congestion level.

An upper bound of the *CW* interval should not increase too fast, because of unnecessary deferring of contending nodes. We also let a node exceed the CW_{max} value, but not more than the number of CW_{min} slots.

Differently from [17], a drawn *backoff interval_i* from *backoff stage_i* by a given station may overlap with a drawn *backoff interval_{i-1}* from *backoff stage_{i-1}* in the *sB* algorithm. This way the algorithm is less prone to unnecessary loss of free slots both in sparse and dense networks (many neighbours can be occasionally active).

B. Enhanced Dynamic Resetting Algorithm (DRA)

The *basic* Dynamic Resetting Algorithm (*DRA*) has been designed in our previous work [13] to avoid unnecessary collisions upon a successful transmission and dropped packet. In the *basic* DRA algorithm the *CW* value is reset based on the number of 1-hop neighbours, their change during the recovery mechanism and the number of retransmission attempts. The *basic* DRA defines the variable NrN_{THR}, which is the established maximum number of 1-hop active neighbours and is assumed to be fixed value. In this work we modify our *basic* resetting algorithm by letting the NrN_{THR} value to be adjusted depending on the channel conditions (and therefore we called it *enhanced* DRA algorithm). It can be modified taking into account instantaneous channel conditions (e.g. based on Signal-to-Interference-plus-Noise-Ratio (SINR)) or higher data rates. The NrN_{THR} can be adjusted according a simple threshold based technique, when taking into account instantaneous channel conditions, thus by estimating the $SINR$. The NrN_{THR} can be chosen by comparing the channel quality estimate against a series of thresholds representing the desired performance bounds of the link quality (excellent, good, fair, poor and no link). Let NrN_{thr_1}, NrN_{thr_2}, ..., NrN_{thr_N} represent the set of possible values of the NrN_{THR} variable and ξ_1, ..., ξ_N the SINR thresholds. The threshold of the number of neighbours in 1-hop neighbourhood can be chosen as shown below:

NrN_{thr_1} $if \; SINR < \; \xi_1$.

NrN_{thr_i} $if \; \xi_i < SINR < \; \xi_{i+1}, \; where \; i = 1, ..., N-1$.

NrN_{thr_N} $otherwise$.

We do not show certain values of the *SINR*, because depending on the modulation scheme different threshold values are valid. For example, for Bit-Error-Rates (*BERs*) less than or equal to $1E^{-5}$, the minimum required SINR values corresponding to each data rate have been shown in Table I. The NrN_{THR} value can be chosen from the $[4-8]$ interval. An *initial* value of the NrN_{THR} can be set manually. The *initial* value of the NrN_{THR} corresponds to the fair link quality. In our simulation the NrN_{THR} value is set to 6, to make sure it can be modified from the $[5-7]$ interval, when taking instantaneous channel conditions into consideration.

In [13] we have also defined a concept of the fast/slow increase/decrease of the number of active 1-hop neighbours. We assume that our threshold of the fast/slow increase/decrease,

Fig. 2.   Resetting algorithm

TABLE II

SIMULATIONS PARAMETERS

Parameter	Values
Number of active nodes	25, 50
Simulations area (m)	$<= 1500x1500$
Topology	Random
txPower	100m
Initial Energy (J)	5.0
Capture Threshold (dB)	10
Radio Propagation Model	Shadowing
Traffic model	CBR/UDP
Payload size (bytes)	512
Simulation time (s)	100
Nr of simulation scenarios	40
Bandwidth	11Mbps
Routing	DSR
Movement	random and constant
Maximal speed (m/s)	20
DCF scheme	Basic access and RTS/CTS exchange
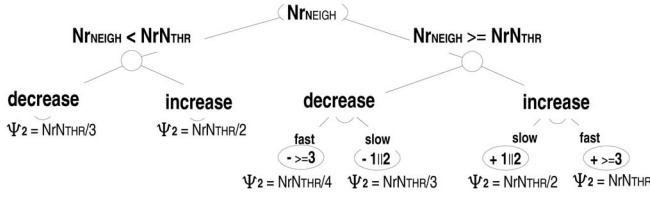
FS_thresh, equals $\frac{NrN_{THR}}{2}$. If $Nr_{NEIGH} => NrN_{ATT1} + FS\_thresh$ (or $Nr_{NEIGH} <= NrN_{ATT1} - FS\_thresh$) the fast increase (or decrease) occurs, and the slow otherwise. The NrN_{ATT1} represents the number of neighbours when the first retransmission has occured.

When resetting the Contention Window value upon a successful transmission or dropped packet, first the *Contention Window* value is reset to the minimum, then an extra ψ value is added, thus $CW_{min} = CW_{min} + \psi$. The ψ value consists of ψ_1 and ψ_2. The ψ_1 represents the influence of the number of retransmissions attempts and ψ_2, the influence of the number of 1-hop neighbours and their change (increase/decrease) during the recovery process.
The ψ_1 is estimated according formula (4).

$$\psi_1 = Nr_{NEIGH} * (1 - CW_{min}/CW_{old}) * \chi \qquad (4)$$

where CW_{old} is the last value of the *CW* after which a packet is received. We consider following cases during selection of the χ value:

1) When a packet has been received successfully and the $nr_{ATT} == 0$ then $\chi = 0$, thus $\psi_1 = 0$.
2) When a packet has been received successfully and $nr_{ATT} > 0$ then $\chi = \frac{nr_{ATT}+1}{10}$.
3) Finally, when the packet has been dropped $(nr_{ATT} > limit)$ then $\chi = 1$.

The ψ_2 value is reached as depicted in Fig. 2. First the current Nr_{NEIGH} is compared with the NrN_{THR}. We have assumed that the NrN_{THR} equals 6 in this figure. Secondly, if the Nr_{NEIGH} is smaller than the threshold, we check if an increase/decrease of the Nr_{NEIGH} has occurred comparing the Nr_{NEIGH} with the NrN_{ATT1}. If the Nr_{NEIGH} is larger or equal to the threshold, we check if the fast/slow increase/decrease has occurred.
Summarizing, the ψ value depends on:

TABLE I

MINIMUM REQUIRED SINR VALUES FOR $BERs <= 1E^{-5}$

Modulation	Coding Rate	Rates (Mbps)	SINR (dB)
BPSK	1/2	6	6.02
BPSK	3/4	9	7.78
QPSK	1/2	12	9.03
QPSK	3/4	18	10.79
16-QAM	1/2	24	17.04
16-QAM	3/4	36	18.80
64-QAM	2/3	48	24.05
64-QAM	3/4	54	24.56

1) the number of retransmission attempts (nr_{ATT})
   a) If *(no retransmissions)* only the change of the numbers of 1-hop neighbours is taken into account
   b) If*(retransmissions <= max)* both ψ_1 and ψ_2 are important
   c) If *(pkt dropped)* then $\psi = (\psi_1 + \psi_2)$ and $\chi = 1$
2) the increase/decrease (fast/slow) of the number of neighbours ψ_2.
   If *(no increase/decrease)* occurs then $\psi_2 = 0$;

IV. PERFORMANCE EVALUATION

The proposed sB-DRA protocol has been implemented in the ns-2.29 network simulator [22]. The following performance metrics are used:

- total number of packets received (*PRcvd*)
- number of collisions (*Coll*)
- PL - packets loss (*PL*)
- average aggregate delay(*Delay*).

We have run simulations comparing the IEEE 802.11 DCF standard (STD) against the DCF standard and the added *DRA* mechanism (STD-DRA) against the *sB* and against the *sB* and added *DRA*, thus *sB-DRA* protocol. In Table II we present the simulation parameters. Fig. 3 and Fig. 4 depict the *total number of packets received* versus the number of simulation scenario in the indoor and outdoor environments, respectively. We can easily notice that *sB*, *STD-DRA* and *sB-DRA* are more advantageous in the outdoor environment, where conditions could be worse than in the indoor environment, because of more frequent movement, loosing links, higher distances etc. Applying the *DRA* algorithm to the standard improves it by 12.91% in terms of the total number of packets received. The *sB* algorithm in these simulations achieves 14.46% gain, and the *sB-DRA* mechanism outperforms all schemes improving the performance of the standard by 24.97%. As we see in Fig. 3 in all of the scenarios *sB-DRA* receives more packets than the other schemes, where in the worst case (scenario 4) it still receives 52 packets more than the standard and 12 more than the second best performing scheme in this scenario. We can
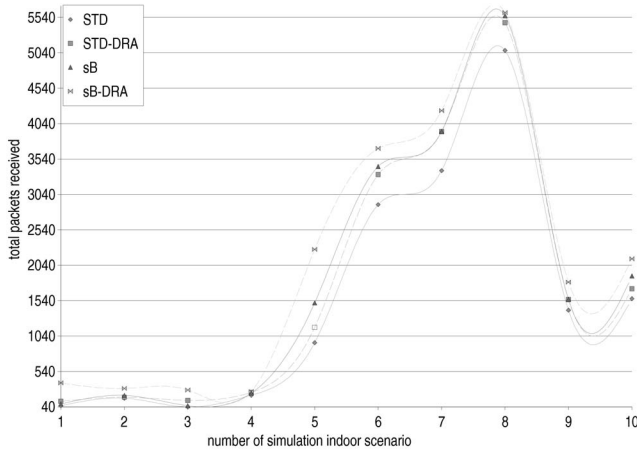
Fig. 3.   Simulation results (50 nodes, mobile ad hoc networks, indoor environment)

Fig. 5.   Simulation results (25 nodes, mobile ad hoc networks, outdoor environment)
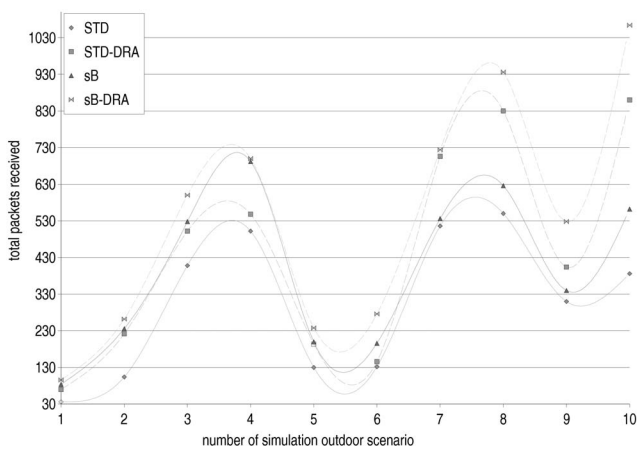

Fig. 4.   Simulation results (50 nodes, mobile ad hoc networks, outdoor environment)

also notice that the performance of the *sB* and *STD-DRA* is comparable in terms of total number of packets received. We do not want to compare these two schemes, because they are different and a goal of this work is to use both mechanisms but we want to show that applying them separately to the *STD* also improves the performance of the *basic* standard (*STD*). However, an interesting observation can be found in the outdoor environment, where in the first 6 scenarios the *sB* scheme outperforms the *STD-DRA* mechanism and in other scenarios we found the opposite situation. However, we should wait with drawing any conclusions till executing more simulations with different topologies and other settings. However, this is not the scope of this work. The *sB-DRA* mechanism achieves the best performance over all schemes. We can easily notice that it outperforms other protocols in all of the scenarios where in some of scenarios it receives much more packets than others. We have also executed simulations with a lower number of active nodes, namely, 25 and 15 nodes (simulations parameters as in Table II). We have observed that both the *sB* and *DRA* approaches, used together or separately,
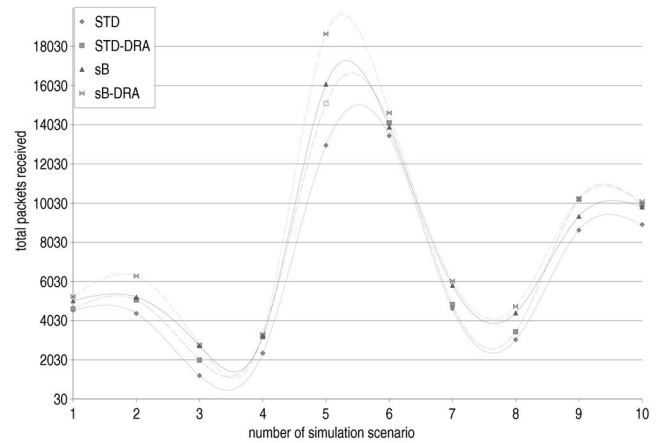
improve significantly the performance of the standard with 25 nodes. Fig. 5 shows *total number of packets received* versus number of simulation scenario in the outdoor environment in mobile ad hoc networks. The *sB-DRA* outperforms all schemes, where in the two scenarios the 2nd and the 5th, it performs much better than other approaches.

Table III shows the gain in percentage (simulations with 50 and 25 nodes -40 scenarios) of all the proposed approaches over the DCF 802.11 standard for all the metrics. As we can see in this table, all schemes outperform the standard in terms of total number of packets received, the number of collisions and packet loss and insignificantly the average delay. The *sB-DRA* outperforms all considered approaches, receiving much more packets, decreasing significantly the number of collisions and packet losses.

In the network with 15 nodes, the gain is insignificant, but using the algorithms in the sparse network does not degrade the performance. It is natural, that the *sB-DRA* will be the most advantageous in dense mobile networks, since it takes into consideration the number of neighbours in 1-hop neighbourhood and the change (fast/slow increase/decrease) of the number of neighbours during recovery mechanism. It can cope with different environments because it can be tuned taking into account the channel conditions.

Finally, we have performed preliminary simulations with DCWA algorithm [17], where we compared the *sB* to the DCWA but without *resetting the Backoff ranges* mechanism in the DCWA (we call it the *basic* DCWA). In the *basic* DCWA we have reset CW ranges to its initial value upon a successful transmission. The changed simulation parameters used in these

TABLE III
GENERAL SIMULATIONS RESULTS

	PRcvd	PL	Coll	Delay
STD-DRA	12.54	8.23	10.16	3.13
sB	15.34	9.94	21.36	2.89
sB-DRA	28.86	19.38	29.87	4.12

TABLE IV

SIMULATIONS PARAMETERS (*basic* DCWA VERSUS SB)

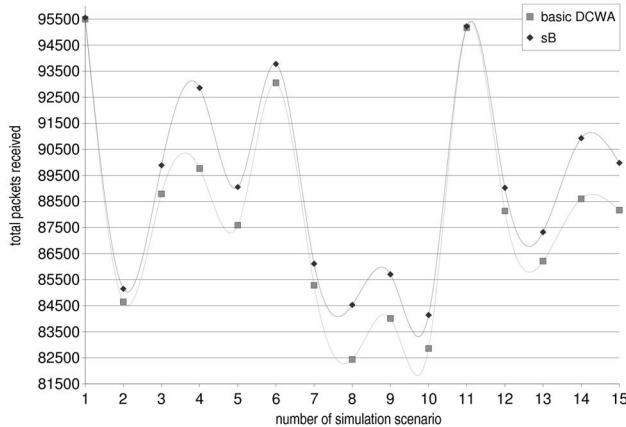| Parameter	Values
Number of active nodes	25
Simulations area (m)	$<= 500x500$
Radio Propagation Model	Shadowing Outdoor
Nr of simulation topologies	15
Payload size (bytes)	768
Routing	DSR, AODV, OLSR
DCF scheme	RTS/CTS exchange

Fig. 6. The sB versus the basic DCWA (25 nodes, mobile ad hoc network)

simulations have been listed in Table IV (other parameters remain as in the previous simulations). Because lack of the space we could not show all of the results in this work. Fig. 6 represents the total number of packets received by the *sB* and the *basic* DCWA versus the number of topology. We can observe in the figure that the *sB* algorithm clearly outperforms the *basic* DCWA. In the two worst case scenarios, namely the 1st and the 11th, the *sB* algorithm receives 62 and 50 packets more respectively, than the *basic* DCWA.

In our future work we will compare the *sB-DRA* algorithm with the *full* DCWA algorithm.

ACKNOWLEDGMENT

V. CONCLUSION

In this paper we have presented the contention window "selection Bounds - Dynamic Reset Algorithm" protocol (sB-DRA) for wireless ad hoc LANs. The CW selection Bounds algorithm adjusts a distribution of both a lower and upper bound of the *CW* interval. The *enhanced* Dynamic Resetting Algorithm resets the CW value upon a successful transmission and dropped packet considering the number of currently active neighbours and the change of 1-hop neighbours during recovery mechanism. We have compared the *sB-DRA* protocol against the *DCF 802.11 DCF* standard against the standard

with applied *DRA* mechanism and against the *sB* algorithm. The *sB-DRA* outperforms significantly the IEEE DCF 802.11 in dense mobile networks and insignificantly in sparse mobile networks. We have also shown that the *sB* mechanism outperforms the *basic* DCWA scheme in mobile ad hoc networks.

REFERENCES

[1] The Institute of Electrical and Electronic Engineers, *IEEE Computer Society LAN MAN Standards Committee: Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) Specifications. ANSI/IEEE Std. 802.11"*, ANSI/IEEE Std. 802.11, 1999 Edition, 1999.

[2] The Institute of Electrical and Electronic Engineers, *IEEE Standard, Part 11: Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) specifications: Amendment 8: Medium Access Control (MAC) Quality of Service Enhancements*, IEEE Std. 802.11e, 2005.

[3] V. Bharghavan, A. Demers, S. Shenker, and L. Zhang, *MACAW: a media access protocol for wireless LANs*, Conference on Communication Archtectures, Protocols and Applications, 215-225, ACM Press, April, 1994.

[4] G. Bianchi, L. Fatta, and M. Oliveri, *Performance evalutation and enhancement of the CSMA/CA MAC protocol for 802.11 wireless LANs*, IEEE PIMRC96, Taiwan, October, 1996.

[5] S. Xu, and T. Saadawi, *Does IEEE 802.11 MAC protocol work well in multi-hop wireless ad hoc networks?*, IEEE Communication Magazine, June, 2001.

[6] S. Xu, and T. Saadawi, *Revealing the problems with 802.11 MAC protocol in multi-hop wireless ad hoc networks*, Journal of Computer Networks, Vol. 38. No. 4, March, 2002.

[7] G. Bianchi, *Performance analysis of the IEEE 802.11 Distributed Coordination Function*, IEEE Journal on selected areas in communications, Vol. 18. No. 3, March, 2000.

[8] F. Cali, M. Conti, 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, December, 2000.

[9] F. Cali, M. Conti, E. Gregori, *Dynamic IEEE 802.11: design, modeling and performance evaluation*, IEEE Journal on selected areas in communications, Vol. 18. No. 9, pp. 1774-1786, September, 2000.

[10] M. Conti, *Body, personal, and local ad hoc wireless networks*, Handbook of ad hoc wireless networks, CRC Press, 2002.

[11] Y. Yang, J. Wang, R. Kravets, *Distributed optimal contention window control for elastic traffic in wireless LANs*, INFOCOM, 2005.

[12] L. Bononi, M. Conti, E. Gregori, *Runtime optimalization of IEEE 802.11 wireless LANs performance*, IEEE Transactions on Parallel and Distributed Systems, Vol. 15, No. 1, 66-80, January, 2004.

[13] S. Romaszko and C. Blondia, *Neighbour -aware, collision avoidance MAC protocol (NCMac) for mobile ad hoc networks*, ISWCS'06, Spain: Valencia, September, 2006.

[14] S. Romaszko and C. Blondia, *Dynamic distributed contention window control in wireless ad hoc LANs*, ATNAC'06, Australia: Melbourne, December, 2006.

[15] L. Guang, Ch. Assi *Mitigating smart selfish MAC layer misbehaviour in ad hoc networks*, WiMob06, Canada: Montreal, June, 2006.

[16] K. Jamieson, H. Balakrishnan, Y.C. Tay *Sift: A MAC protocol for event-driven wireless sensor networks*, Third European Workshop, EWSN, Switzerland: Zurich, February, 2006.

[17] A. Ksentini, A. Nafaa, A. Gueroui, M. Naimi, *Determinist Contention Window Algorithm for IEEE 802.11*, PIMRC 05, 16th IEEE Personal, Indoor and Mobile Radio Communications, Germany:Berlin, 2005.

[18] J. Deng, P.K. Varshney, and Z.J. Haas, *A new backoff algorithm for the IEEE 802.11 Distributed Coordination Function*, CNDS 04, 215-225, San Diego, January, 2004.

[19] Y. Kwon,Y. Fang, and H. Latchman, *A novel MAC protocol with fast collision resolution for wireless LANs*, IEEE INFOCOM, April, 2003.

[20] "M. Taifour, F. Nait-Abdesselam, D. Simplot-Ryl, *Neighbourhood back-off algorithm for optimizing bandwidth in single hop wireless ad-hoc networks*, MobiWac05, June, 2005.

[21] J. Hu, C.D. Raymond, *A statistics based design of MAC protocols with distributed collision resolution for Ad Hoc networks*, MobiWac05, June, 2005.

[22] The Network Simulator NS: http://www.isi.edu/nsnam/ns.