Determinist Contention Window Algorithm for IEEE 802.11

Adlen Ksentini[1], Abdelhamid. Nafaa[2], Abdelhak Gueroui[2], Mohammed Naimi[1]

[1]*LICP EA 2175, Université de Cergy-Pontoise*
3 Av. Adolph Chauvin 95302 Cergy-Pontoise - France
E-mail :{adlen.ksentini, mohamed.naimi}@dept-info.u-cergy.fr

[2]*PRiSM CNRS, Université de Versailles*
45 Av. des Etats-Unis 78035 Versailles – France
E-mail : {anaf, mogue}@prism.uvsq.fr

Abstract-With the widespread IEEE 802.11 networks use, strong needs to enhance Quality of Service (QoS) has appeared. The IEEE 802.11 Medium Access Control (MAC) protocol provides a contention-based distributed channel access mechanism that allow for wireless medium sharing. This protocol involves a significant collision rate as the network gets fairly loaded. Although the Contention Window (CW) is doubled after each collision, active stations may randomly select a backoff Timer value smaller than the preceding one. This is obviously sub-optimal since the backoff values should rather increase after each collision in order to further space between successive transmissions and thus absorbing the growing contending flows. In this paper, we propose a novel backoff mechanism, namely "Determinist Contention Window Algorithm (DCWA)", which further separates between the different backoff ranges associated to the different contention stages. Instead of just doubling the upper bound of the CW, DCWA increases both backoff range bounds (i.e., upper and lower bounds). On the other hand, after each successful transmission the backoff range is re-adjusted by taking into account current network load and past history. Simulation results show that DCWA outperforms both the Distributed Coordination Function (DCF) and the Slow Decrease (SD) scheme in terms of responsiveness to network load fluctuations, network utilization, and fairness among active stations.

Keywords: *IEEE 802.11, MAC layer, Backoff, Performances*

I. INTRODUCTION

Wireless communications are an emerging technology that became an essential feature of every day's life. Especially, the IEEE 802.11 WLAN standard [1] is being accepted for many different environments. IEEE 802.11 is now considered as a wireless version of Ethernet. Its adoption is favoured by the promises of the forthcoming high speed wireless physical layer, 802.11n [2] that is expected to provide a bandwidth ten times bigger than the classical well known IEEE 802.11b (11Mbps). The IEEE 802.11 MAC layer defines two medium access coordination mechanisms: Distributed Coordination Function (DCF), and Point Coordination Function (PCF). The DCF is the fundamental access mechanism while the PCF is used optionally. DCF uses the *Carrier Sensing Multiple Access/Collision Avoidance* (CSMA/CA) protocol. In this mode, if the medium is found idle for longer than a DIFS (Distributed InterFrame Space), then the station can transmit a packet. Otherwise, a backoff process is started and the station computes a random value called backoff time, in the range of 0 and CW (Contention Window) size. The backoff timer is periodically decremented by one for every time slot where the medium was sensed idle. As soon as the backoff timer expires, the station can access the medium. If no acknowledgment is received, the station assumes that collision has occurred, and schedules a retransmission by re-entering the backoff process. For each collision, the CW size is doubled till reaching the maximum value ($CW_{\max}$).

With the conventional backoff algorithm, the duration of a backoff period is usually selected randomly in a range delimited by zero and certain maximum time duration (CW). This interval is dynamically controlled (increased/decreased) by the backoff algorithm. Setting the length of the backoff intervals is not a trivial task. Actually, when the network load increases, we should increase the mean backoff interval (i.e., transmission differing time) to absorb the increasing number of contending flows, and hence minimizing the collision probability for those flows. Whereas, when the network load decreases, we should decrease the mean backoff interval, which reduces the spacing between successive frame transmissions; large values of backoff may indeed strongly limit the throughput of fewer backlogged flows.

At this point, it is clear that the backoff algorithm should take into account the network load to fully fill the network capacity by using the most appropriate backoff range. Particularly, it is important to not reset the CW at its initial range after each successful transmission in order to stabilize the different backoff instances (associated to different nodes) around the optimal operation point of the network. In this paper, we propose a novel backoff mechanism that provides more determinist contention resolution mechanism through the use of a sliding backoff range. In fact, we use a more tighten range at each contention stage. Thus, after each collision the station increases the backoff range's boundaries (upper and lower bound of the CW), ensuring that the next backoff interval is increased. At the same time, after a successful transmission the boundaries of the backoff range are initialised with an intermediate values based on the precedent CW boundaries' values and the current network load. This allows to adjust the backoff range to network changing conditions rather than using a static initial value (as in DCF) that often entails (*i*) monopolization of the medium by the successful stations - unfairness, (*ii*) situation where the network contention level (collision rate) oscillates with transmission cycle of stations, involving poor network exploitation and high fluctuations in throughput and delay [4],[5].

The remainder of this paper is organized as follows: the section II provides background material on the 802.11 MAC, and summarizes related work on DCF enhancements. In section III, we describe our proposed scheme DCWA and its components. Section IV, presents the performances evaluations. The concluding remarks are drawn in Section V.

II. BINARY EXPONENTIAL BACKOFF (BEB)

The IEEE 802.11 MAC protocol provides an access control that is asynchronous, time bounded, and contention-free. The basic access method in the IEEE 802.11 MAC protocol is DCF which is CSMA/CA. The main inefficiency of the DCF mechanism is the consequence of frequent collisions and the entailed wasted idle slots caused by backoff intervals associated to each contention stage. In fact, when the number of active stations increases, there are permanently too many stations backed-off with small contention windows since each successful transmission results in CW re-initialization. Therefore, the network experience excessive collisions and retransmissions, which confine the overall channel throughput. Actually, there are two major factors affecting the throughput performances in the IEEE 802.11: transmission failure (we only consider collisions) and the idle slots due to backoff during each contention period.

Many works have analyzed the performance of the IEEE 802.11 DCF mechanism with a particular focus on the throughput metric. Among them, [3] proposes an analytic model that derives the 802.11 throughput limit in saturation situations; the presented model assumes that each mobile station has always a packet to send and try to find the optimal network operation point in terms of the number of stations. The author in [3] considers that the channel is under ideal conditions (no hidden terminals). Still, the only way to achieve optimal performance (maximize the throughput) is to employ adaptive techniques to tune network parameters rather than limiting the number of stations. Particularly, both the parameters *m* and $CW_{\min}$ have an important incidence on network performances; *m* is the maximum retry limit and $CW_{\min}$ is the minimum Contention Window size. The optimal $CW_{\min}$ value depends closely on the number of contending terminals in network (1). On the one hand, low values of $CW_{\min}$ (e.g. 31) give excellent throughput performance in case of small number of contending station, while it drastically penalize the throughput in case of large number of contending stations. On the other hand, large values of $CW_{\min}$ (e.g. 1023) give reverse effects.

$$CW_{opt} = N\sqrt{2T_c} \qquad (1)$$

T_c is the time wasted by collision and N is the number of active stations

To solve this problem, one could employ a dynamic contention window adjustment algorithm. To achieve this, substantial amount of works were carried out focusing on enhancing the backoff process. In [4], the authors use a *p*-persistent protocol to study the maximum protocol capacity of 802.11. They assert that this method gives very close approximation of the 802.11 standard protocols, particularly if the average backoff interval is always the same. Unlike the standard 802.11 MAC protocol, they propose to dynamically compute the optimized contention window size that maximizes the channel utilization. However, this scheme requires knowledge of the number stations, which is difficult to achieve in concrete deployment situations. The slow decrease (SD) scheme [5] proposes to divide the previous CW over a constant "*decrease factor*" to compute the new CW after each successful transmission. Besides being appropriate only for high contentions conditions, this scheme uses a static "*decreasing factor*" without considering the network load variation. The recently proposed Fast Collision Resolution (FCR) scheme [6] aims at reducing collisions in the network by increasing the contention window when either a stations freezes the backoff timer or experiences collision. Moreover, the authors propose to exponentially decrease the backoff timer when a certain number (threshold) of successive idle slots is detected. The major deficiency of FCR is its unfairness in the sense that the deferring nodes have large backoff timer (until 2047 slots in FCR) while successful stations re-enter the contention with a small backoff timers. After a fairly long run time, this result in channel monopolization by the successful stations in the network.

Almost all of existing approaches still provide probabilistic backoff range increasing after each collision as the backoff interval is uniformly drawn from the interval [0, CW]x, where CWx is each collision. In fact, a given station can randomly select a backoff value lower than the one selected at the preceding contention stage. Thus, the number of time slots occupied by successful data transmissions decreases drastically as the collision rate increases, involving low channel utilization. This is far from the optimal point where the network is supposed to operate.

III. DCWA: Determinist Contention Window Algorithm

Currently the 802.11 DCF resolves collision through multiple levels of CWs and backoff stages (see Figure 1). In the initial backoff stage (stage 0), the value of CW has the minimal value $CW_{\min}$. After each collision, the CW will be doubled until reaching the maximum $CW_{\max}$. After each successful transmission, the backoff will resume with initial stage (0) and the CW will be reset to $CW_{\min}$ regardless the network condition or the number of competing nodes. By resetting the CW to $CW_{\min}$, DCF increases the probability of collision and frequent retransmissions remain high until the CW attains appropriate values. This is obviously no optimal since high collision rate in the network means poor network exploitation. On the other hand, the intrinsic backoff randomness makes it difficult to instantaneously absorb an increasing number of flows. The backoff process is basically intended to reduce the collision rate when using a higher contention stage.

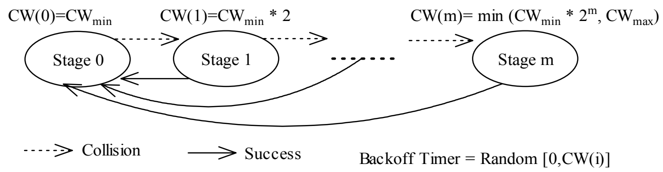

Fig 1. DCF collision resolution stage

To tackle the abovementioned issues, we first divide the overall backoff range [*0*, $CW_{\max}$] into several small ranges. Each backoff sub-range is then associated to a particular collision resolution stage. In other words, unlike DCF that increases only the upper band range's with each collision, DCWA increases both upper and lower bounds of the backoff range. As a consequence, in each contention stage, a given station drew a backoff interval from a distinct backoff range that doesn't overlap with the other backoff ranges associated to the other contention stages. Meanwhile, after each successful transmission rather than initiating the CW with $CW_{\min}$, DCWA initializes the CW range with an intermediate value that reflects the actual channel load while still considering the preceding transmission attempts information.

A. Determinist Backoff Range

In order to ensure that backoff interval is increased from a contention stage to another, DCWA splits the global range [$CW_{\min}$, $CW_{\max}$] into different range where each range corresponds to a contention's stage. Additionally, the size of the backoff range ($CW_{ub(i)}$-$CW_{lb(i)}$) depends on the stage *i* itself. In fact, DCWA increases the size at each stage, in order to minimize the collision probability between stations that are in the same contention stage. Basically, the range size is basically analytically designed to accommodate a certain number of competing stations [9]. For instance, assuming that a collision occurs at the stage *i*, it is obvious that the stage's range size does not perfectly suit the current competing stations. Therefore for the next stage $i+1$, it is necessary to increase the range's size in order to absorb the increasing network load.

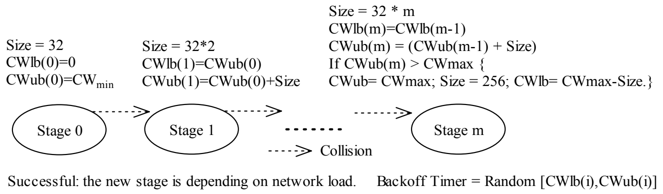

Fig 2. DCWA collision resolution stage

When reaching the last contention stage, typically in congestion conditions, the CW_{ub} is limited by the physical CW_{max}. In this situation, it is important to maintain the high backoff intervals. To do so, we maintain the size with the maximum value (256 slots). Besides selecting a backoff timer in a more determinist way, this size allows to effectively accommodate an important number of stations.

Fig 2 illustrates DCWA operation, where the backoff Timer is randomly selected from the range delimited by CW_{lb} and CW_{ub} rather than 0 and CW (see formula (2)). This way, we ensure that the backoff Timer is chosen through a determinist way and increased at each contention stage. Note that at the starting stage (t_0), the CW_{lb} and CW_{ub} are initialized with 0 and CW_{min}, respectively. After each collision the range's boundaries and the range's size are updated following (3)(4)(5), respectively. Thus for each stage we obtain a different range that does not overlap with the precedent range.

$$BackoffTimer = Random[CWlb_{(i)}, CWub_{(i)}] \tag{2}$$

$$Size_{(i)} = 32 * i \tag{3}$$

$$CWub_{(i)} = CWub_{(i-1)} * 2 \tag{4}$$

$$CWlb_{(i)} = CWub_{(i)} - size_{(i)} \tag{5}$$

B. Resetting the Backoff range

As previously stated, after each successful transmission DCWA initiates the CW range with an intermediate value in order to avoid resetting the CW range to its initial value. In this way, DCWA adjusts the CW range to current network, which permits to prevent future collisions. At the same time, the backoff range size is initiated to the minimum value (32). This ensures that the backoff is selected with a small amount of random fluctuation at the first stage of the contention.

Let B(T) be the instantaneous network load as its sensed by stations. B(T) represents the fraction of slots that the medium was observed to be busy out of the previous T slots. This includes all slots where a transmission was successfully completed, or a collision occurred. In other word B(T) symbolizes the probability that the medium is busy, so high values mean high network congestion. Otherwise, the network is in relaxed conditions. After a fairly long run, B(T) becomes inherently coordinated between stations since it is based on common network measurements (see formula (6)). Here, it is important to remove short term fluctuations due to the wireless channel's characteristic, so the network measured values B(T) is weighted in respect to past measures using EWMA (Exponential Weighted Mean Average).

$$B(T) = \alpha \times B(T)\_cur + (1-\alpha) \times B(T) \tag{6}$$

Using B(T) to re-adjust the CW after successful transmission may, in fact, favour DCWA's fairness. From formula (7), it is obvious that the new backoff's range is roughly the same after a short operation time. In addition to B(T) measurement, DCWA uses previous CW_{ub} and CW_{lb} value in order to determine the new range's boundaries. This allows taking into account the last (optimal) range that permitted successful transmission. Thus by combining B(T) measurements and the previous backoff range, DCWA considers both the preferred contention level and the current network load. From this point the new CW_{ub} and CW_{lb} can be expressed based on formulas (7) and (8). Note that the new range's boundaries are determined by affecting different weights to precedent range boundaries and the CW_{min} respectively according to current network load (B(T)).

$$CWub_{(t)} = CWub_{(t-1)} \times B(T) + CW_{\min} \times (1 - B(T)) \tag{7}$$

$$CWlb_{(t)} = CWub_{(t)} - size \tag{8}$$

Based on B(T) measurements we are able to define the CW's range according to network conditions. When the network is congested (B(T)'s value is high), it is obvious that the new CW's range should be rather close to previous CW's range. On the other hand, if the network is not heavily loaded (B(T) is low), the new CW's range should be initialised with values more close to the initial range in order to reduce the effect of wasted time slots, providing high responsiveness.

IV. PERFORMANCE EVALUATION

In order to evaluate the proposed scheme, we have constructed a simulation of the DCWA MAC protocol using ns-2 (Network Simulator). We compare DCWA with both Slow Decrease (SD) and DCF schemes. Note that SD is used with a decrease factor equal to 2; this choice provides, in fact, the best gain over DCF [5]. The simulations focus on protocols' abilities to sustain high performance while maintaining fairness among the stations as the network load increases. To better highlight DCWA's performances, we conduct two types of simulations. The first one consist in using, with each simulation run, a fixed number of stations and a fixed offered load by station (see Fig 3, Fig 4, and Fig 5). Therefore, we give the performance of DCWA in respect to an increasing number of stations, highlighting the achievable throughput gain; the given through is measured after stabilization of MAC parameters at each station. The second set of simulations is based on a more realistic network scenario where the network experience highly changing configurations with different traffic load volume over the time. In this case, we present a single 200s-lifetime simulation where the number of stations increase continuously, provoking network overload. This allows us to better capture the behaviour of each evaluated scheme in response to both increasing and decreasing in the contention level (collision rate); we particularly highlight the responsiveness of the evaluated schemes in the sense where these protocols doesn't have enough time stabilize. For completeness,

Table 1 shows the basic PHY/MAC parameters used in the simulations.

Tab 1. 802.11b Physical parameters

Parameter	CW_{min}	CW_{max}	DIFS	SIFS	SlotTime	Phy data rate	Phy basic rate
Value	31	1023	50µs	10µs	20µs	11Mbps	1 Mbps

A. Simulation Model

For the simulations, we have created a network based on several Wireless Terminals (WTs) and one access point (AP). The WTs are uniformly distributed around the AP. All WTs generates a Constant Bit Rate to WT_0 (Sink). The reason behind using CBR traffic is to put more stringent constraints on the network as well as to study the fairness between stations. In fact, multiple CBR sources would require that the network sustains the overall offered load (summation of CBR sources bit rates) throughout the simulation period, which may provoke MAC queues overflows after a fairly long run. In contrast, with multiple VBR sources, the peaks of bit rates are unlikely to occur at the same time, which allow the network to absorb the brief offered load bursts exhibited, by different traffic sources, at different time scales.

We use a value of 0.8 for α and 0.2 second as B(T)-update-period in our simulation. These two parameters leverage DCWA's responsiveness to network conditions. We believe that the chosen values provide a good balance between removing short term fluctuations impact and capturing long term trends.

As aforementioned, the second run of simulation uses a changing network configuration over the time; the simulation scenario is as follows. From time t=0 s to t=10 s, the channel is empty. Beginning at t=10 s, *ten* new flows (1500 byte with 0.02s-generation-interval) are started at *three* second interval. The network remains in this state until t=45s, where *ten* other stations are started at 3 seconds intervals; at this stage, there is an extreme contention level in the network. Thus, till t=100s, there are *twenty* WTs competing for the channel. At t=100s, five WT are stopped and restarted at t=120s. Finally at t=150s *ten* WTs are stopped. The purpose of such network dimensioning is to evaluate DCWA' abilities to maintain high network utilization and fairness among active flows in spite of fast network load changing.

B. Simulation Results

Fig 3 represents the throughput achieved by the three mechanisms when using a 1500-Bytes- packet size (600 kbps). It is clearly apparent that DCWA outperforms both SD and DCF throughout the simulation duration. This is to be expected as DCWA uses a determinist backoff range at each stage of the contention which ensures that the backoff timer is increased at each contention stage. Thus, DCWA absorbs more quickly the increasing number stations by further spacing between consecutive transmissions in each station. This, obviously, reduce the collision probability and the entailed wasted time slots. SD, on the other hand, uses a slow decrease mechanism which reduces the collision probability by reinitializing the CW to intermediate values instead of using $CW_{\min}$ as the conventional DCF do. However, both SD and DCF still draw backoff intervals with uniform distribution from the interval [0, W], W depends on the contention stage; this causes high disparity between the successively drawn backoff intervals. The mean gain of DCWA over SD and DCF, is roughly 275 kbps (5%) and 390 kbps (7%), respectively.

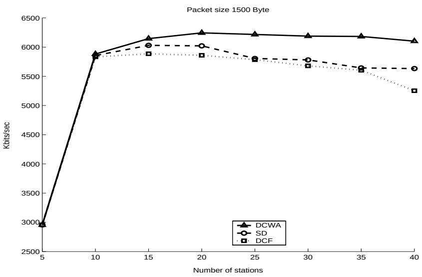

Fig 3. Throughput versus Network load (Packet Size = 1500 Byte)

In Fig 4, we illustrate also the achieved throughput gain when using 500-Bytes- packet size (200 kbps). As to be expected, there is an important drop in the total network through (for all schemes) due to the high overhead PHY/MAC overhead and the duration network collision. The higher the packet size, the higher is the number of wasted time slots by the collision. As revealed in Fig 4, DCWA still outperforms both SD and DCF with a mean throughput gain of about 100 kbps (4%) and 190 kbps (6,5%), respectively.

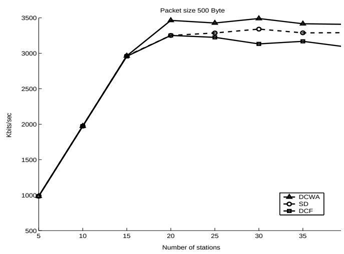

Fig 4. Throughput versus Network load (Packet size=500 Bytes)

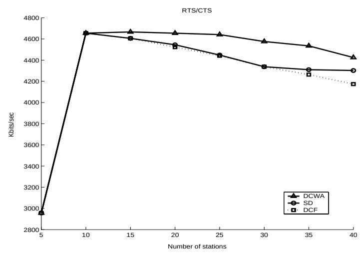

Fig 5. Throughput versus Network Load (RTS/CTS enabled)

Fig 5 shows the throughput of the three schemes when using the RTS/CTS handshaking mechanism. Here, the data packet's size is 1500 bytes (600 kbps). As for the preceding figures, DCWA outperforms the others schemes in both network configuration; the mean gain is about 117 (3%) kbps and 140 kbps (4%) compared to SD and DCF, respectively. An important observation from these results is that slow decrease mechanism is not very efficient when using RTS/CTS, particularly with high number of stations. Since the collision consumes now a fairly limited number of time slots, reducing the collision rate with globally high contention window is no longer effective. Both DCWA and SD may considerably improve their performances by taking into account the collision duration [9] when CW is slowly decreased (see formula (1)).

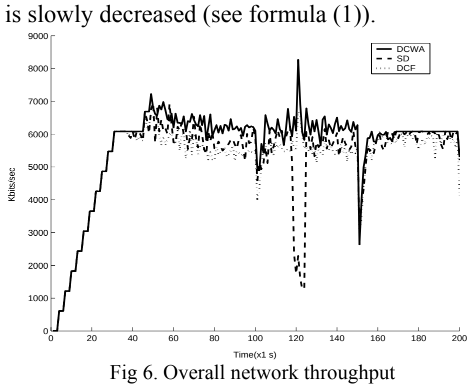

Fig 6. Overall network throughput

Now we study and illustrate the responsiveness of the simulated MAC protocols to the varying traffic load. The overall network utilization is illustrated in Fig 6 in terms of the total instantaneous network throughput throughout the 200-seconds simulation. Clearly, when the network is sufficiently relaxed (Before t=40s, and after t = 150s), there is sufficient bandwidth available and all protocols achieve similar throughputs, carrying the load as it is offered. However, under stressed conditions, DCWA clearly gains a significant advantage over DCF and SD. The goodput gain of DCWA reaches 400 kbps and 500 kbps over SD and DCF, respectively when the load is about 110 % (between t= 75s , t=100s and between t=120s , t=150s). This large gain is principally due to low collision rate achieved by DCWA compared to DCF and SD.

At t=100s (at t=150 s as well), there is a considerable drop in throughput for all protocols. This occurs after stopping five WTs. In fact when stopping these five WTs, the CW sizes are too large for the new network load, resulting in large number of idle contention slots. Here, it is important to note that DCWA adapts quickly to this situation by capturing the new channel traffic load thanks to continuous network measurements, i.e., B(T).

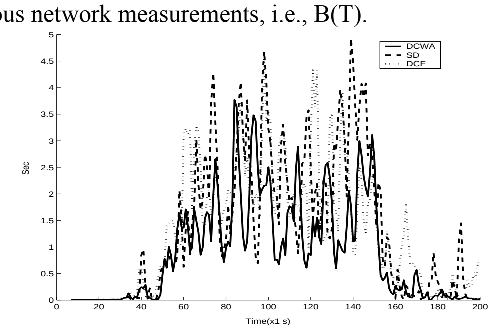

Fig 7. Average Packet Delays

Another major observation is that when suddenly stopping several flows SD performances drop severely, and for a long time, since its CW decreasing is based on dividing CW by 2 after each successful transmission, which means much longer periods to attain the perfect CW that allow fully filling the medium gap left by the stopped flows.

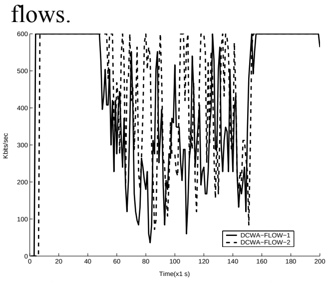

Fig 8. DCWA DataRate Fairness

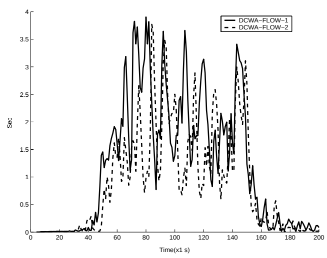

Fig 9. DCWA Delay Fairness

Fig 7 shows the average delays experienced by the three protocols with the second simulation scenario. DCWA significantly reduces the delay compared to the other schemes. The mean delays in DCWA is 0.9067s; while in SD and DCF the mean delays is 1.18 s and 1.287s, respectively. This is due the fact that DCWA decreases the collision rate, which contributes in reducing frequent packet retransmissions at MAC layer, and thus confining queuing delay at this layer. Again, the moment where SD and DCF exhibit high delays matches perfectly with throughput drops observed in Fig 6, i.e., at t=100s and t=150s. This the consequence a too high contention window sizes that translate into high inter-packet delays.

It is worth mentioning that the delay represents the amount of time made by a packet to go from the sender's application to the receiver's application, including queuing time at the MAC layer, possible retransmission procedures, and propagation time over the wireless link.

We give in Fig 8 and Fig 9 throughput and delay fairness measurements for DCWA, respectively; we show the instantaneous throughputs and delays experienced by two different flows throughout the 200s-simulation lifetime. The measurement granularity is 1 second. Globally, the fairness is acceptable level since alls DCWA's stations measure the same network condition (i.e., B(T) is coordinated among the stations), and then adjust the CW at the same contention stage when decreasing this later. The slight disparity is mostly due to the randomness of backoff interval drawing that produce different values at different at different station even if they use the same backoff range. Additionally, the CW increasing process may further enlarge the disparity since the collision event is not experienced with the same frequency at all stations.

V. CONCLUSION

In this paper, we have presented a Determinist Contention Window Algorithm (DCWA) to enhance the conventional backoff mechanism. By using a predefined CW's range at each stage of the contention, DCWA ensures that the backoff timer value is increased as the collision number increases. Furthermore, the DCWA's CW decreasing process is based on choosing an appropriate CW-range according to the current network load and the knowledge acquired from continuous network measurements. This keeps the collision rate low enough to not provoke performance collapse and to reduce, as well, the number of retransmissions. Particularly, the performed simulation shows that DCWA outperforms both SD and DCF by its remarkable responsiveness to network load fluctuations. Besides maintaining high fairness among the competing stations, DCWA also shows a high stability in the achieved throughput and delay. The measured gain is, however, limited as the collision duration decreases. Future works will focus on integrating the collision duration parameter into the DCWA's CW decreasing process.

VI. REFERENCES

[1] IEEE, Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) Specifications, IEEE Standard 802.11, 1999

[2] TGn Sync Technical Specification, http://www.tgnsync.org/techdocs/tgn-sync-proposal-technical-specification.pdf.

[3] G. Bianchi "performance analysis of the IEEE 802.11 distributed coordination function" *in IEEE Journal on selected Area In communications,* Vol. 18. No. 3, March 2000.

[4] F. Cali, M. Conti and E. Gregori, "Dynamic tuning of the IEEE 802.11 protocol to achieve a theoretical throughput limit", *in IEEE/ACM Networking Transactions*, Vol. 8, Dec. 2000.

[5] Q. Ni, I. Aad, C Barakat and T. Turletti. "Modeling and analysis of slow CW decrease for IEEE 802.11 WLAN", in Proc. of IEEE PIMRC'03, Beijing, China, Sept. 2003.

[6] Y. Kwon, Y. Fong and H. Latchman. "A Novel MAC protocol with Fast Collision resolution for wireless LAN's", in Proc. of IEEE INFOCOM'03, San Francisco, USA, March 2003.

[7] A. Ksentini, A. Gueroui and M. Naimi, "Enhancing the DCF mechanism in noisy channel", to appear in Proc. of IEEE VTC'05 Fall, Dallas, USA, Sept 2005.

[8] Network Simulator, ns-2, http://www.isi.edu/nsnam/ns.

[9] A. Nafaa, A. Ksentini, and A. Mehaoua., "SCW: Sliding Contention Window for Efficient Service Differentiation in IEEE 802.11 Networks", in Proc. of IEEE WCNC'05, New Orleans, LA, 13-17 March 2005.