DYNAMIC TUNING OF THE CONTENTION WINDOW MINIMUM (CW_{MIN}) FOR ENHANCED SERVICE DIFFERENTIATION IN IEEE 802.11 WIRELESS AD-HOC NETWORKS

Lassaâd Gannoune and Stephan Robert
Ecole d'Ingénieurs du Canton de Vaud (EIVD), Rue de Galilée 15, CH-1400 Yverdon-les-Bains, Switzerland
{lassaad.gannoune, stephan.robert}@eivd.ch

Abstract- The proposed IEEE 802.11e draft standard defines new MAC protocols for QoS in wireless networks, mainly HCF and EDCF. EDCF is a contention-based channel access scheme and is part of HCF for infrastructure networks and may be used as a separate coordination function for wireless ad-hoc networks. In this paper we propose to extend EDCF with a dynamic adaptation algorithm of the minimum contention window (CW_{min}) that enables each station to tune the size of the CW_{min} used in its back-off algorithm at run time. The purpose of our scheme is to reduce delay and jitter and increase the efficiency of the transmission channel. Priorities between access categories are provisionned by updating the size of the CWmin according to application requirements and channel conditions. The performances of the IEEE 802.11e EDCF, enhanced with our adaptation algorithm, are extensively investigated by simulations. Results obtained indicate that CWmin adaptation scheme outperforms the 802.11e EDCF standard in terms of channel utilization, throughput, and packet delay. Indeed, the delay for high priority access category decreases by up to 30%, total throughput increases by up to 18%, and channel capacity is 15% higher.

Keywords: IEEE 802.11e, QoS, EDCF, back-off algorithm, contention window minimum (CW_{min}).

I. INTRODUCTION

Multimedia applications, including voice, require a certain quality of service (QoS) support such as guaranteed bandwidth, delay, jitter and error rate. Guaranteeing those QoS requirements is a challenging task with regard to 802.11 WLAN [9],[10],[13], protocols and Medium Access Control (MAC) functions.

In order to support QoS in 802.11 WLAN, several priority schemes has been developed [1],[3],[6],[8]. Currently, there are some priority schemes under discussion [10],[2]. The IEEE 802.11 Task Group E is currently defining enhancements to the 802.11 MAC access methods to support QoS, providing the classes of service, enhanced security and authentication mechanism. These enhancements are defined in 802.11e draft [2] which introduces two main access methods, the Hybrid Coordination Function (HCF) and the Enhanced Distributed Coordination Function (EDCF), renamed in latest 802.11e draft [2] to EDCA (enhanced distributed channel access).

EDCF is a contention-based channel access scheme [4], [2] and [5]. It is part of HCF for infrastructure networks and may be used as a separate coordination function for wireless ad-hoc networks. EDCF provides differentiated service, distributed access to the wireless medium for 4 delivery priorities or access categories [2]. A Traffic Categories TC in 802.11e is defined as the application traffic related to a special user priority UP specified in IEEE 802.1D [12]. The mapping between traffic categories TCs and access categories ACs is presented in 802.11e draft [2].

EDCF access channel on each QoS Station (QSTA) uses at most 4 prioritized output queues, one for each delivery priority, called Access Categories (ACs). Figure 2 illustrates the different queues for different priorities. As for a station, a QoS-supporting Access Point (QAP) should support at least 4 Access Categories (ACs). In EDCF, relative priorities are provisioned by configuring the time to access the channel [6], [4] once it is sensed idle defined as arbitrary interframe space (AIFS) as shown in figure 1. Differentiation is also provided by changing the size of the contention window (CW). EDCF uses the contention window to assign priority to each access category. Indeed, assigning a short contention window to a high priority AC ensures that in most cases, high priority AC is able to transmit ahead of low priority one. Thus, the CWmin and CWmax parameters can be set differently for different access categories, such as, a high priority AC with small values of CWmin and CWmax.

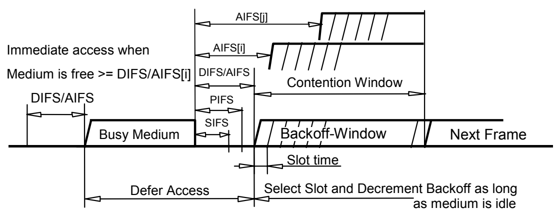

Fig. 1. Some IFS Relationship

After any unsuccessful transmission a new contention window is calculated with the help of the persistence factor $PF[AC_i]$ and another uniformly distributed backoff counter out of this new, enlarged CW is drawn, to reduce the probability of a new collision.

$$newCW\left[AC_i\right] = ((oldCW\left[AC_i\right]+1)\times PF[AC_i])-1$$

Whereas in legacy 802.11 [9], CW is always doubled after any unsuccessful transmission (i.e., PF[AC_i]=2). EDCF, uses the PF to increase the CW different for each access category. In the latest 802.11e draft [2] PFs differentiation per access category are no longer considered, i.e., PFs equals to 2 for all access categories.

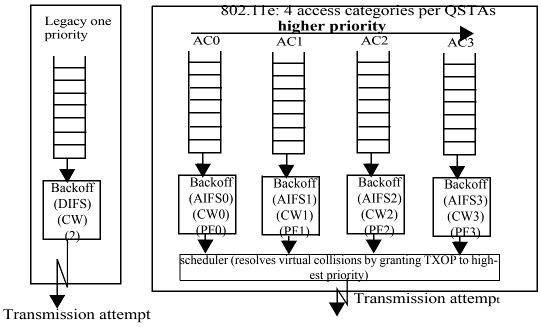

Fig. 2. Queue based EDCF vs legacy DCF

In addition, the CW never exceeds the parameter CWmax[AC_i], which is the maximum possible value for contention windows associated with each access categories. In this paper we focus on the dynamic tuning of the minimum contention window (CW_{min}) with a special designed scheme. We compare the performance of the proposed scheme with the basic EDCF which does only consider a static CW_{min} value.

In the following sections we present the CWmin adaptation scheme that we propose, we describe its implementation, simulation and we discuss its performance results.

II. CW_{MIN} ADAPTATION

Both in the legacy DCF [11] and EDCF [2], the backoff algorithm reduces the contention window size to CWmin when there is a successful transmission. The problem is that a such reduction of the CW could lead to more collisions when the transmission channel is loaded or in congested state. The main idea of the dynamic adaptation of CWmin is to adapt periodically the CWmin[*i*] value for a certain access category *i* to the traffic load and channel conditions. We assume that AC_i is the *ith* access category, with *i* varies between 0 and 3 and that the high priority level is 0 and low priority is 3.

The problem is that when setting CWmin[*i*] to a small value, this could lead to a more collisions if the station has already experienced one or more consecutive collisions. Contributing to more collisions, results in an increase in delay and jitter for traffic categories associated with that access category *i* and may be for other traffic categories attempting to access the medium at the same time. On the other hand setting CWmin[*i*] to a high value leads to higher delay and jitter especially in a low loaded environment. So, we believe that adapting the CWmin[*i*] parameter according to the traffic load will lead to reduce the overall number of collisions and reduce the delay and jitter for the TCs in the different access categories.

The purpose of the proposed scheme is to dynamically adapt the CWmin for each access category *i* by setting a higher value of CWmin[i] when the channel is estimated to be loaded and a small value (closest to its static CWmin[*i*] value of EDCF) when the channel load is estimated to be low.

A. Scheme description

In the basic EDCF scheme [4],[2], after each successful transmission the contention window is reduced to CWmin[*i*]. So, we propose that after each successful transmission of a frame (i.e., MAC Packet Data Unit) from an access category *i*, we compute an adaptive value of the minimum contention window, i.e., DCWmin[*i*] and we reduce the contention window to that dynamic value. We note that we use both an adaptive mechanism for computing the value of DCWmin[*i*] (dynamic contention window minimum) according both to traffic load and the static value of CWmin[*i*]. We also differentiate between access categories while updating DCWmin[*i*] for access categories related to different priority levels.

The next sections detail how the contention window of each access category is set after each collision and after each successful transmission and the method used to estimate the collision rate at each wireless (or QoS station) station *p*.

B. Setting CW after each successful transmission

In our scheme we propose that each access category updates each CWmin[i] parameter in an adaptive way using the estimated collision rate at regular update period T_{update} expressed in time slots. In the proposed scheme we re-use the same method defined in [5] to estimate the average collision rate as seen by a station *p*.

Instantaneous collision rate f^j_{avg} at the j^{th} update period T_{update} is calculated using the number of collisions and the number of packets sent during that period. Collision rate is given by (1):

$$f^j_{curr} = \frac{E(collisions_j[p])}{E(data\_sent_j[p])} \quad (1)$$

Where $E(collisions_j[p])$ is the number of collisions at station *p* during the period (update period or step) *j* and $E(data\_sent_j[p])$ is the number of packets sent during the update period T_{update}. In order to get an estimation of the collision rate that minimizes random fluctuations, an Exponentially Weighted Moving Average (EWMA) is used to smooth the series of collision rates (i.e., f^j_{curr}). Equation (2) gives the corresponding value of collision rate at step *j*.

$$f^{j}_{avg} = (1-\alpha)\times f^{j}_{curr} + \alpha \times f^{j-1}_{avg} \quad (2)$$

Where α is a smoothing factor in the range [0, 1], j refers to the j^{th} update period T_{update}, f^{j}_{curr} stands for the instantaneous collision rate. Using the estimated average of collision rate f^{j}_{avg} at step j the numerical expression of the proposed scheme for CWmin[i] adaptation is presented in equation (3).

$$DCW_{min}[i] = \left(1-f^{j}_{avg}\right)\times CW_{min}[i] + f^{j}_{avg}\times(CW_{max}[i]-CW_{min}[i])\times 2^{i-2} \quad (3)$$

Where $DCW_{min}[i]$ stands for the adaptive value of contention window minimum for an access category i, CWmin[i] is the minimum contention window (according to EDCF) assigned for the same access category i and f^{j}_{avg} represents the estimated collision rate at step j. We propose here to perform a *slow adaptation* for high priority access category and a *fast adaptation* for low priority access category. This leads to a fast increase in CWmin for lower priority ACs and a slow increase in CWmin for higher priority ACs. Both slow adaptation and fast adaptation of CWmin can be achieved by introducing the level of priority i for an access category in the formula of DCWmin.

The dynamic contention window minimum for AC i obtained in equation (3) varies between a lower bound of CWmin[i], when the collision rate eqauls to zero, and an upper bound of $(CWmax[i] - CWmin[i])*2^{i-2}$, when the collision rate equals to 1. So, this upper bound depends on the priority level of the access category and limits the increase of the *DCWmin[i]* which results in a slow adaptation of higher priority traffic. Indeed, this upper bound of DCWmin[i] is lower for high priority traffic and greater otherwise.

In order to ensure that the adaptive contention window maximum has an upper bound, the derived formulas (in equations 3 and 4) use the static value of CWmax according to EDCF along with the following formula:

$$DCW_{min}[i] = min(DCW_{min}[i], CW_{max}[i]) \quad (4)$$

We note that this formula (4) is not useful for our specific simulation scenario, since we have only used three access categories and the upper bound of $DCW_{min}[3]$ is ($CW_{max}[3]$-$CW_{min}[3]$) which is obviously less than $CW_{max}[3]$.

C. Setting CW after each collision

After each unsuccessful transmission the contention window for an access category i is set as the following. In our scheme we are using a PF which equals to 2 for all access categories, so that, the contention window is doubled while remaining less than the maximum contention window, i.e, $CW_{max}[i]$. In this case, we do not modify the basic EDCF scheme.

$$CW_{new}[i] = min(CW_{max}[i], 2\times CW_{old}[i]) \quad (5)$$

III. SIMULATION PARAMETERS AND TOPOLOGY

We implemented the proposed scheme in ns-2 simulator [7] using EDCF semi-package to support QoS enhancement feature from Atheros [7]. This section presents the generic simulation topology used in order to evaluate the performance of the dynamic CWmin adaptation scheme as well as a detailed analysis of the results.

A. Generic simulation topology

We use a generic topology (circular routing scenario) shown in figure 3, which consists of n stations indexed from *1* to n. Each station generates three type of UDP data streams, labelled with high, medium and low according to their priorities. These data streams belong to the three traffic categories (TCs), respectively, audio (high), video (medium) and background traffic (low). Station n sends packets to station *1* and station i sends packets to station $i+1$. The highest priority queue in each station generates packets at sending rate of 64Kbps (PCM audio flow) which corresponds to a packet size of 160 bytes and an inter-packet arrival of 20ms. The medium priority traffic queue, generates packets at sending rate of 1024Kbps which corresponds to a packet size of 1280 bytes and inter-packet interval of 10ms. The low priority queue sending rate is 128Kbps which represents a packet size of 200 bytes and an inter-arrival packet of 12.5 ms.

RTS/CTS mode is not used. In addition, all nodes are within the same Independent Basic Service Set (IBSS), such that, each station can detect the transmission from any other station. The different nodes are uniformly spread out of 500X500 m^2 dimensions in 2D space.

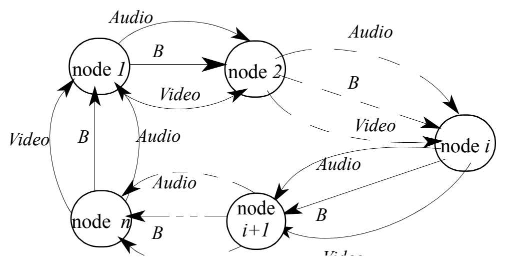

Fig. 3. Simulation topology

Table 1 shows the different MAC parameters for the three access categories used for all QoS STA and in the different simulation scenarios.

Table 1. MAC parameters for the three ACs

Parameters	High	Medium	Low
CW_{min}	7	15	31
CW_{max}	200	500	1023
AIFS(μs)	34	43	52
PF	2	2	2
Packet size (bytes)	160	1280	200
Packet Interval (ms)	20	10	12.5
Sending rate (Kbps)	64	1024	128

In the following simulations, we assume that each QSTA operates at IEEE 802.11a PHY mode 6 [11] (i.e., modulation 16-QAM, coding rate of 3/4, data rate of 36 Mbps). Table 2 presents the different PHY/MAC parameters used in simulations.

Table 2. 802.11a PHY/MAC parameters used in simulations

SIFS	16μs
DIFS	34μs
ACK size	14 bytes
Data rate	36 Mbps
Slot time	9μs
CCA time	3μs
MAC header	28 bytes
Modulation	16-QAM
Preamble Length	20μs
RxTxTurnaround time	1μs
PLCP header length	4μs

B. Impact of the smoothing factor and the update period

As described earlier the proposed scheme uses two main parameters, a smoothing factor α and an update period T_{update}. In order to select proper parameters, we have done a several simulation experiments. First, to deal with the effect of the smoothing factor we vary the value of α in the range of [0, 1] and we set the update period to T_{update} = 8000 time slots. Simulations are run for a fixed number of stations, i.e., 20 stations. Results are averaged over 20 simulations. Two performance criteria are used, total throughput (or goodput) and the mean audio delay. Goodput is defined as the total application layer received bytes divided by total simulation time. Figure 4 and 5 show resp. the effect of smoothing factor on total goodput and mean audio delay. It can be seen that, a value of α in the range [0.6, 0.9] achieves a lower delay and a lowest audio delay is for α=0.9.

From figure 5, we can see that, a value of α in the range [0.55, 0.7] achieves a higher goodput with a maximum of goodput for α=0.6. In addition, we have higher goodput for values of α in the range of [0, 0.2].

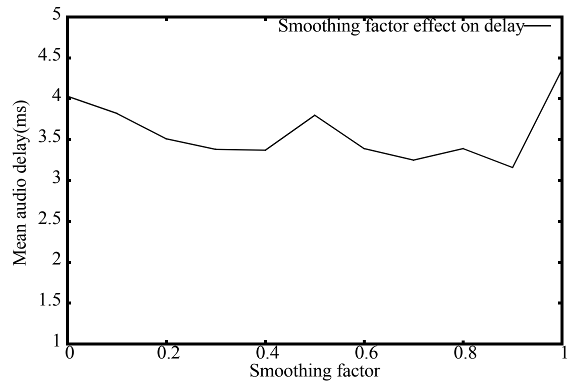

Fig. 4. Impact of smoothing factor on audio delay

Since, small α values could contribute to random fluctuations we consider only values in the range of [0.55, 0.7]. Therefore, we can note that values of α in the range of [0.6, 0.7] achieves a best trade-off between higher total goodput and low mean audio delay. So, in the following simulations we set α to 0.6.

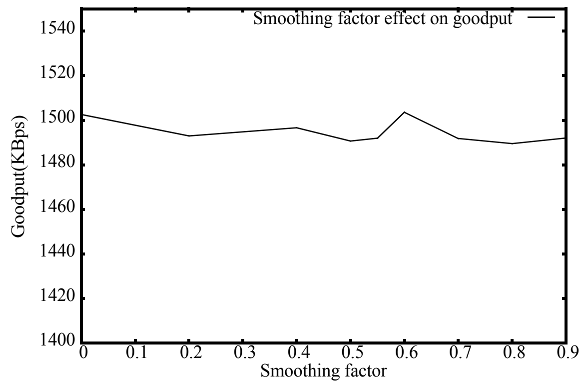

Fig. 5. impact of smoothing factor on goodput

Figure 6 and 7 show the variations of resp. total goodput and mean audio delay as a function of the update period expressed in time-slots. The choice of the value of update period, *Tupdate,* should take into account that higher values make adaptations less useful and smaller values could hurt the adaptation scheme since high frequent updates of CW_{min} could be influenced by channel fluctuations. As illustrated by figure 6, lower audio delay is achieved with values of *Tup-*

date in the range of [500, 10000] time-slots with a lower audio delay at Tupdate=4000 time-slots.

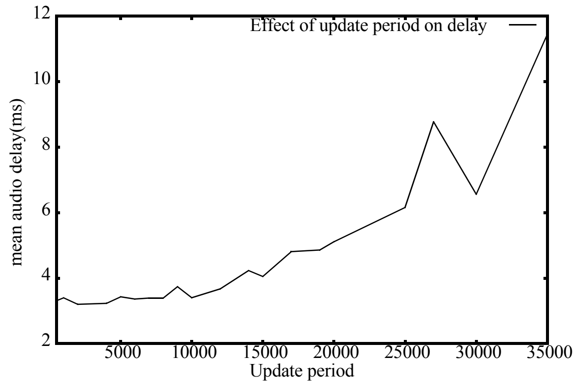

Fig. 6. Impact of update period on audio delay

Figure 7 shows the total goodput as a function of update period and it can be seen that we have a higher goodput for update periods in the range of [500, 6000] and also for *Tupdate* value of 500 time-slots. So, a value of *Tupdate*=4000 time-slots achieves a tradeoff between a higher goodput and a lower delay.

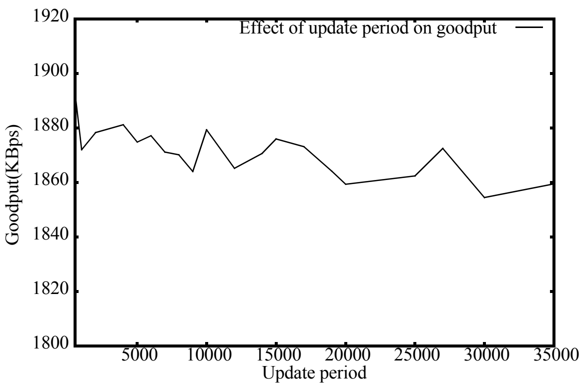

Fig. 7. impact of update period on goodput

In the following simulations we set α and *Tupdate resp.* to 0.6 and 4000 time-slots.

IV. SIMULATION RESULTS

In order to evaluate the performance of the dynamic CW_{min} adaptation scheme, we investigate in this section the impact of traffic load and compare it to the basic EDCF scheme. The different access categories ACs used for simulations are described in Table 1. We simulate various loads of the system by instantiating the simulation topology in figure 3 for a special number of stations. All the stations are located within the same Independent Basic Service Set (IBSS), so that, every station can detect the transmission from any other station. The following QoS metrics are used to evaluate the performance of the different simulations:

- Gain of goodput: stands for the gain (in %) on the average goodput (AG) of the proposed scheme (DCWmax) compared with basic EDCF:

$$Gain\_of\_goodput = \frac{AG_{DCWmax} - AG_{EDCF}}{AG_{EDCF}}$$

- Mean delay: stands for the average of all flows that have the same priority in the different stations. This metric is used to evaluate how well the scheme can accommodate real-time flows.
- Latency distribution: allows to trace the percentage of packets of the same priority access category that have latency less than the maximum delay required by an application (or traffic category).
- Collision rate: represents the average number of collisons that occurs per second.
- Medium utilization (M_u): the medium utilization represents the percentage of time used for the transmission of data frames and it is given by :

$$M_u = \frac{Totaltime - Collisiontime - Idletime}{Totaltime} \times 100\%$$

For the different scenarios used in this section, all the traffic categories (associated with the access categories) are launched at around 3.0 seconds with small individual offsets to have accurate CDFs (Cumulative Distribution function) of the latency. The simulation duration is 18 seconds. In order to have confidence in results obtained by simulations, we run 15 simulations and results are averaged on theses simulations.

A. Channel utilization comparisions

Figure 8 shows the collision rate for CWmin adaptation and EDCF. The collison rate is the same for CWmin adaptation and EDCF for a very low traffic load, i.e., 5 stations.

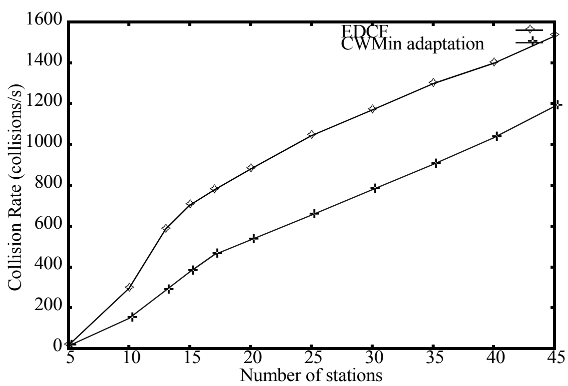

Fig. 8. Collision rate for CWmin adaptation and EDCF

As the traffic increases, the collision rate in CWmin adaptation maintains a lower increase (the gap between the two curves in figure 8 increases) than in EDCF starting from a system load of 10 stations. It can be seen that, for 25 stations, the collision rate in CWmin adaptation is 40% lower than in EDCF. We believe that the dynamic adaptation of CWmin has contributed to reduce the number of collisions in the IBSS. As shown in figure 9, under most system loads, CW_{min} adaptation scheme has much better channel utilization than EDCF. This is because, CWmin adaptation adjusts the size of CWmin[i] upon a successful transmission according to the network condition. However, EDCF just blindly resets CW[i] to a static CWmin[i] and as a result contribute to more collisions when the system load is not very low (as shown by figure 8).

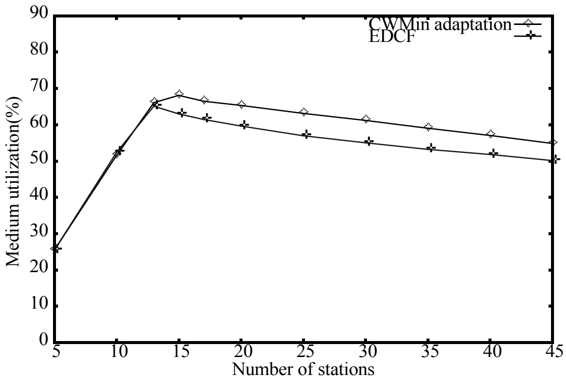

Fig. 9. Medium utilization for CWmin adaptation and EDCF

Indeed, the capacity in CWmin adaptation is higher than in EDCF (maximum channel utilization in EDCF is reached for 13 stations while in CWmin adaptation corresponds to15 stations). So, the channel capacity is 15% higher than in EDCF. Furthermore, the maximum channel utization reached in EDCF (65.11%), corresponds to a channel capacity of 13 stations and 20 stations for CWmin adaptation which leads to a gain of up to 54% of channel capacity.

B. Throughput comparison

Figure 10 shows the gain of goodput for CW_{min} adaptation scheme over EDCF.

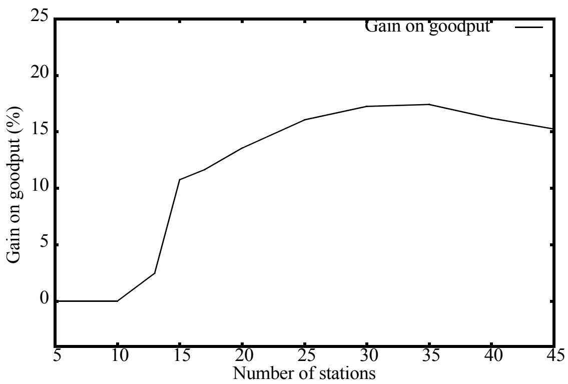

Fig. 10. Gain of goodput for CWmin adaptation over EDCF

The goodput improves in CW_{min} adaptation and the gain of goodput for CW_{min} adaptation over EDCF is up to 18%. Furthermore, the gain increases when traffic load is greater than 10 stations as shown in figure 10. So, according to system throughput CWmin adaptation outperforms EDCF. This throughput improvement is due to the increase in channel utilization because of the CWmin adaptation algorithm.

C. Packet delay comparisons

In this subsection, we compare the average packet delay under EDCF and CWmin adaptation scheme. Figure 11 shows the mean audio (high priority traffic) as a function of traffic load for both CWmin adaptation scheme and EDCF. The mean audio delay improves significantly in CWmin compared to EDCF. Indeed, CWmin scheme maintains a lower audio delay than EDCF even in low traffic load, i.e., for a number of stations less than 10. As the traffic load increases, CWmin is able to maintain a lower delay than EDCF.

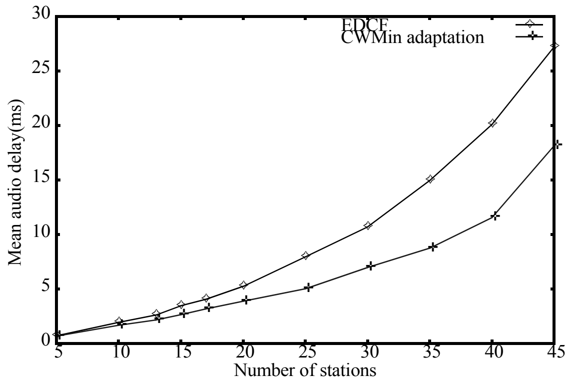

Fig. 11. Mean audio delay for CWmin adaptation and EDCF

The audio delay in CWmin adaptation scheme is 34% lower than in EDCF for a traffic load of 30 stations and results in lower delay and jitter for high priority access categories. As it can be seen in figure 12, there is a substantial improvement of the mean video delay (medium priority traffic) in CWmin adaptation compared to EDCF.

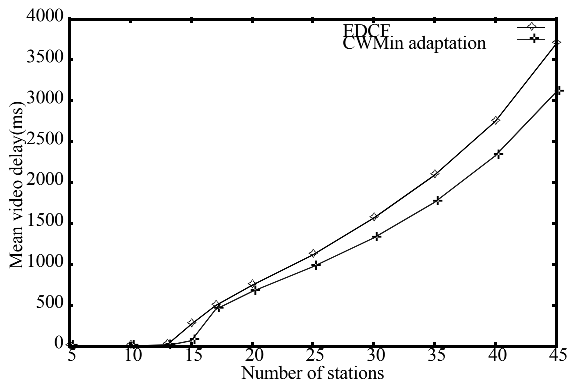

Fig. 12. Mean video delay for CWmin adaptation

Both EDCF and CWmin adaptation have the same mean video delay when the traffic load is low, i.e., less than 13 stations. However, the delay improves in CWmin adaptation as the traffic load increases. The video delay is 75% lower (65.08ms in our scheme and 267.44ms in EDCF) in CWmin adaptation scheme than in EDCF for a system load of 15 stations (channel capacity). This can be explained by the adaptation algorithm used to ajust the size of CWmin[i] that performs better than a static CWmin in medium and loaded channel system.

According to these QoS metrics, we can conclude that CWmin adaptation scheme outperforms EDCF in light, medium and high system load before and after the staturation of the channel system. Moreover, the channel capacity (or efficiency) improves and is 15% higher than in EDCF.

V. CONCLUSIONS AND FUTURE WORK

In this paper we have proposed a new dynamic approach for the adaptation of the minimum contention window (CW_{min}) in order to enhance the service differentiation for 802.11e WLANs. We have extended the basic EDCF scheme by a distributed algorithm that enables each station to tune the size of the CW_{max} used in its back-off algorithm at run time. The performances of the proposed adaptation scheme investigated by simulations have indicated that our scheme outperform EDCF and improves delay and jitter for all access categories. The mean audio delay in CW_{min} adaptation scheme is up to 34% lower than in EDCF. Also, the throughput is improved by up to 18% and the overall channel capacity is 15% higher than in EDCF.

In a future work, we will compare the performance of the proposed CWmin adapation scheme with related work in the context of CW adapation. Furthermore, based on this work, we are going to investigate how the proposed scheme can be adapted for infrastructure networks. We intend also to design and implement a hybrid adaptation approach of CWmin and CWmax parameters and study its performance.

ACKNOWLEDGEMENTS

The work reported in this paper is supported by the switzerland HES-SO R&D fund with grant number QoS-1505.

REFERENCES

[1] D-J. Deng and R-S. Chang. A priority scheme for IEEE 802.11 DCF access method. IEICE Transactions on Communications, E82-B(1), January 1999.

[2] IEEE WG, Draft Supplement to Standard for Telecommunications and Information Exchange Between Systems-LAN/MAN specific requirements -Part 11: Wireless Medium Access Control (MAC) and Physical Layer specifications: Medium Access Control (MAC) Enhancements for Quality of Service (QoS), IEEE 802.11e /Draft D5.0, july 2003.

[3] N. H. Vaidya, P. Bahl, and S. Gupta. Distributed fair scheduling in wireless LAN. In Sixth Annual International Conference on Mobile Computing and Networking, Boston, August 2000.

[4] Mangold, S., Choi, S., May, P., Klein, O., Hiertz, G. and Stibor, L. "IEEE 802.11e Wireless LAN for Quality of Service", in Proc. European Wireless '02, Florence, Italy, February, 2002.

[5] Romdhani, L., Ni, Q., and Turletti, T., "Adaptive EDCF: Enhanced Service Differentiation for IEEE 802.11 Wireless Ad-Hoc Networks ", in Proceedings of the IEEE WCNC'03.

[6] Imad Aad and Claude Castellucia. Differentiation mechanisms for IEEE 802.11. In Proceedings of IEEE Infocom 2001, Anchorage, Alaska, April 2001.

[7] Ns-2 network simulator home page. Available at: http://www.isi.edu/nsnam/ns/

[8] A. Veres, A.T. Campbell, M. Barry, and L-H. Sun. Supporting service differentiation in wireless packet networks using distributed control. IEEE Journal of Selected Areas in Communications (JSAC), Special Issue on Mobility and Resource Management in Next-Generation Wireless Systems, vol. 19, No. 10, pp. 2094-2104, October 2001.

[9] IEEE WG, Part 11: Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) Specifications, IEEE 802.11 Standard, 1999.

[10] IEEE WG home page: http://grouper.ieee.org/groups/802/11

[11] IEEE WG, "802.11a, Part 11: Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) Specifications, IEEE 802.11 Standard", Sept. 1999.

[12] ANSI/IEEE Standard 802.1D, 1998 ed., ìIEEE standard for information technology-Telecommunications and information exchange between systems-Local and metropolitan area networks-Common specifications. Part 3: media access control (MAC) bridges" (Institute of Electrical and Electronics Engineers, 1998).

[13] F. Cali, M. Conti and E. Gregori. IEEE 802.11 Protocol: Design and Performance Evaluation of an Adaptive Backoff Mechanism. In IEEE JSAC, vol. 18, no. 9, september 2000.