Hindawi Publishing Corporation
EURASIP Journal on Wireless Communications and Networking
Volume 2010, Article ID 323598, 13 pages
doi:10.1155/2010/323598

Research Article

Introducing an Adaptive Method to Tune Initial Backoff Window (CW_{min}-ATM) in IEEE 802.11 Wireless Networks

Navid Tadayon and Saadan Zokaei

Electrical Engineering Department, Khajeh Nasir University of Technology, Tehran, Iran

Correspondence should be addressed to Navid Tadayon, navid.tadayon@gmail.com

Received 30 September 2009; Revised 29 December 2009; Accepted 14 March 2010

Academic Editor: Ping Wang

IEEE802.11 access protocol uses CSMA/CA in its Medium Access control layer as the main access function, which carries several deficiencies. In these networks, as the number of active stations increases, delay and throughput degrade severely. As far as throughput and service delay are vital elements in Quality of Service (QoS) determination, such degradation could lead to intolerable situations and reduce the efficiency of WLANs. Networks (WLANs). Studies proved this problem arises due to constant initial backoff windows size (CW_{min}), which is an important parameter in determination of network behavior. In this paper, we introduce a new method to tune this parameter adaptively according to changes in channel load. In this method, we do tune this parameter after every transmission using a feedback from transmission channel. Later it will be proven that applying this method in MAC layer enhances network stability; delay trend in all traffic classes exhibits a considerable reduction when compared with simple Enhanced Distributed Coordination Access (EDCA) scenarios. Also throughput exhibits a salient improvement in level. In other word, QoS improves. Especially, with the aid of this method, delay variations in all decrease considerably and more smoothen delay trends are achieved.

1. Introduction

In recent years, desires to utilize Local Area Network (LAN) for communication increased dramatically. Undoubtedly, one of the most important classes of these access networks is IEEE 802.11 that was innovated in 1999 [1]. IEEE 802.11 networks work based on a contention-based access mechanism namely Carrier Sense Multiple Access supported with collision avoidance capability (CSMA/CA). This was the subject of investigation for many researchers during these years [2–4]. As time went by and new, delay sensitive services emerged with real-time requirements, attentions were attracted toward applying diff-serve model on IEEE 802.11 Medium Access control (MAC) layer. Henceforth, many literatures focused on this subject [5–13].

Arrival of IEEE 802.11e standard into scene was a clear response to these efforts that have not been stopped yet. In this paper, we follow the idea behind [14–16] which is improving the 802.11 MAC performance using a channel adaptive backoff scheme. We investigate the merits and shortcomings of Distributed Coordination Function (DCF) and Enhanced Distributed Coordination Access (EDCA) and verify their dependency on network parameters. It would be claimed that using the legacy exponential backoff technique in DCF and EDCA leads to destructive dependence of network performance on initial backoff windows size, number of stations, and network load. This is apparently a drawback from network viewpoint. By applying this adaptive method, each station could periodically estimate network load (by continuously hearing to channel activities). That helps us directly in tuning of CW_{min}. Simulation results confirm the suitability of this method.

We apply our proposed method on EDCA of IEEE 802.11e to probe its effect on different traffic classes. As DCF is a specific case of EDCA, the totality of argument is reserved.

2. Legacy IEEE 802.11

In this section, we do summarize DCF performance and then limitations of this protocol on supporting Quality of service (QoS) will be discussed.

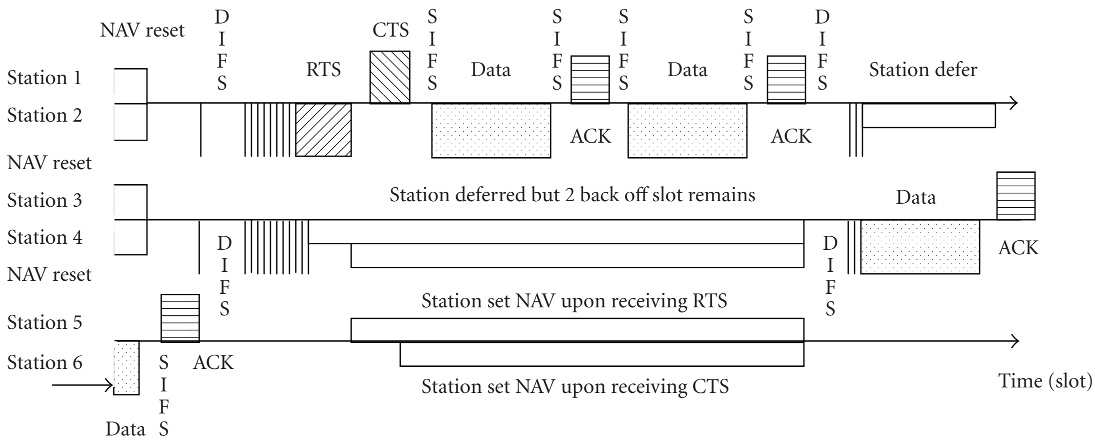

Figure 1: Interactions between six stations. In this example, station 6 is hidden from station 2 but can perceive station 1's CTS.

2.1. DCF. The IEEE 802.11 MAC layer protocol is a distributed coordination function and works based on carrier sense multiple access technique. In this technique, each station transmits its MAC service data units (MSDUs) after sensing the channel and ensuring that no transmission is in progress. In case two or more stations find channel idle and hence transmit simultaneously, the collision occurrence is inevitable. Therefore, IEEE 802.11 working group devised a mechanism, namely, collision avoidance, to reduce the collision probability. In this mechanism, stations start a backoff procedure before transmission; to that end, they should keep silent for a random amount of time immediately after channel remains idle for DCF Inter frame Space (DIFS). The DIFS value considered to be around $34\,\mu$s in IEEE 802.11a standard. Upon the expiration of this random time, stations are allowed to transmit. The length of this random time should be multiple of slot length. In fact, each station carries a parameter, namely, contention window from which this random time is to be extracted.

Upon the correct reception of each data frame, recipient terminal transmits an acknowledgement packet back to sender confirming the correct reception of previous data frame. In case a collision occurs, the contention window size is multiplied by a persistent factor (PF) mentioned in standard. This mechanism is named exponential backoff. It would be titled truncated exponential backoff scheme in case there is an upper bound on contention windows size.

All frames that would not be acknowledged during a predefined amount of time (ACK-Timeout) should be scheduled for retransmission; but with a doubled contention window size. This procedure definitely lessens the collision probability when several stations are attempting to access the channel.

Stations that deferred their channel access due to medium busyness, do not initiate a new random backoff time; instead, they continue to count down their most recent frozen values as soon as channel remains idle for at least DIFS. Finally, after each successfully transmitted MSDU, a new random backoff procedure needs to be initiated regardless of the fact that whether the transmitter queue is empty or containing an MSDU (ready to send). This routine is entitled post backoff because of its initiation after each transmission not before that.

Essentially, there is one case in which no backoff procedure is required to be performed, that is, when, the last post backoff has already been finished while the queue is still empty. Thus, an arrived MSDU from higher layer would be immediately transmitted without need to perform a new backoff routine. All other MSDUs coming after last one should be transmitted after this backoff procedure.

In order to reduce the collision length in long frames, the standard suggested fragmentation scheme. In this scheme, long MSDUs should be fragmented to a series of smaller units to be transmitted sequentially one by one and acknowledged as well. The principal benefit of fragmentation is that, in case of collision occurrence, errors are identified in a swifter manner. Apparently, fragmentation's intrinsic drawback is the huge overhead it imposes on network.

In order to confront with hidden terminal problem, which is one of the most prevalent difficulties in CSMA/CA, Request to Send/Clear to Send (RTS/CTS) mechanism is devised. In this mechanism, sender station transmits a short control frame, namely, RTS prior to sending its data frame. Then, RTS recipient replies with another control frame, namely, CTS. Both RTS and CTS frames contain information about the length of following data frame. Following to reception of RTS by terminals in proximity of sender and reception of CTS by hidden terminals in proximity of receiver, all terminals should refrain from sending another frame in order to avoid collision occurrence. In fact, this mechanism helps in protecting system against sending long collided frames, especially in situations where hidden terminal problem is probable. Using Fragmentation, several smaller frames would be transmitted in series whereas using RTS/CTS method, a long frame would be transmitted but with less overhead and in a faster way.

In between each of RTS, CTS, Acknowledgment (ACK), and data transmissions, there should be idle separation intervals, namely, short inter frame space (SIFS) that is lower than DIFS, in length. Short Inter Frame Space (SIFS) value is typically around $16\,\mu$s (according to IEEE 802.11a specification). Figure 1 illustrates the performance of DCF. As

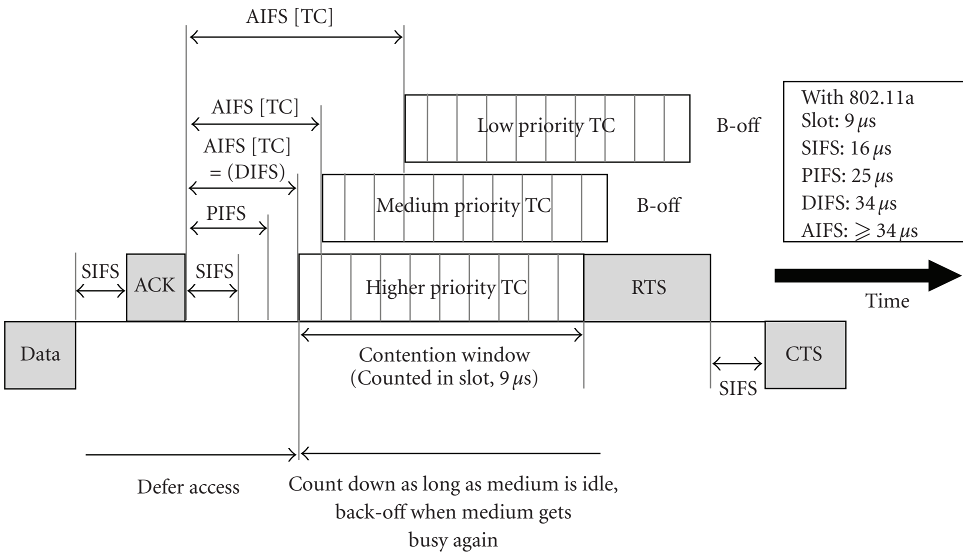

Figure 2: Comparison of different backoff classes with different priorities.

SIFS' length is always smaller than DIFS' length, subsequent frames coming after SIFS intervals are logically furnished with relatively higher priority rather than frames coming after DIFS interval. Such an invention helps in providing ACK, RTS, and CTS with higher priority compared to Data frames.

3. Quality of Service Provisioning Mechanism in 802.11e Access Protocol

In order to support QoS in 802.11 wireless LANs, many proposals have been presented up to now and a lot more are under studying. Recently, IEEE 802.11 task group *E* approved a new standard by adding few enhancements to MAC layer of IEEE 802.11. The result is a new enhanced distribution coordination function, namely, EDCA. IEEE 802.11e constituted from two distinct access phases, namely, contention period (CP) phase and contention free period (CFP) phase. These two phases alternate steadily in a superframe framework over time. Like DCF, EDCA is a contention-based protocol that is utilized in CP phase.

3.1. EDCA. Now we commence with a definition of access category (AC). AC is the classification of different traffic classes in order to serve them with different requirements. To each AC in a station, a distinct EDCA is dedicated. They perform backoff procedure and act independently from each other. Here, backoff procedure starts immediately after channel stays idle for AIFS duration. Depending on physical characteristic of each AC, Arbitration Inter Frame Space (AIFS) might extend from DIFS, which is the bottom value, to larger amounts. Immediately after waiting for AIFS, each backoff entity sets its counter to an integer value extracted uniformly from $[1, \mathrm{CW}+1]$ interval where Contention Window (CW) in each AC varies from a minimum value, namely, $\mathrm{CW}_{\min}$, up to the bound $\mathrm{CW}_{\max}$.

For traffic category *i*, let $\mathrm{CW}_{i,j}$ denote contention window size at j_{th} retransmission stage and let $\mathrm{CW}_{i,\max}$ denote the maximum contention window size. Also, $L_{i,\text{retry}}$, σ_i, and m_i are, respectively, retry limit, persistent factor and number of retransmissions after which CW_i stays constant. (All in class *i*.) Below equation encompasses all these parameters and shows their interaction with each other.

$$\mathrm{CW}_{i,j} = \begin{cases} {\sigma_i}^{j} \cdot \mathrm{CW}_{i,0} \\ {\sigma_i}^{m_i} \cdot \mathrm{CW}_{i,0} \\ \sigma_i^{j} \cdot \mathrm{CW}_{i,0} \end{cases}$$
$$= \mathrm{CW}_{i,\max}\begin{cases} 0 \le j \le m_i - 1, & \text{if } L_{i,\text{retry}} > m_i, \\ m_i \le j \le L_{i,\text{retry}}, & \text{if } L_{i,\text{retry}} > m_i, \\ 0 \le j \le L_{i,\text{retry}} - 1, & \text{if } L_{i,\text{retry}} \le m_i. \end{cases} \tag{1}$$

Meanwhile, m_i value is equal to

$$m_i = \log_{\sigma_i}\left(\frac{\mathrm{CW}_{i,\max}}{\mathrm{CW}_{i,\min}}\right). \tag{2}$$

Figure 2 illustrates EDCA contention in different classes with different priorities.

Figure 3 is the Two-Dimensional Markov chain of backoff mechanism in IEEE 802.11e networks.

Here, it just suffices to mention common equations of 802.11 and 802.11e networks without any proof. These equations are simply derivable from the discrete time Two-Dimensional Makov chain devised by Bianchi. We refer interested readers to [17, 18] for more details. Fundamentally, the transmission probability in class *i*, is the probability that a station belonging to this class transmits at a given time slot. Furthermore, the collision probability in this class is the probability that a transmitted frame from a station belonging to this class collides with a transmission from another station. Finally, the busyness probability is an indication of the chance of having busy channel at a given time slot (successful transmission or collision). We express transmission probability by τ_i, collision probability by p_i, and busyness probability by p_b. Below i, j, k, respectively represent traffic category of that station (or queue), retransmission stage, and value of backoff counter in each retransmission stage. Therefore,

$$b_{i,j,0} = p_i^j \cdot b_{i,0,0}, \quad 0 \le j \le L_{i,\text{retry}},$$
$$b_{i,j,k} = \frac{\text{CW}_{i,j} - k}{\text{CW}_{i,j}} \cdot \frac{1}{1 - p_i} \cdot b_{i,j,0}, \tag{3}$$
$$0 \le j \le L_{i,\text{retry}}, \quad 0 \le k \le \text{CW}_{i,j} - 1, \quad 1 \le i \le M,$$

$$\sum_{j=0}^{L_{i,\text{retry}}} \sum_{k=0}^{\text{CW}_{i,j}-1} b_{i,j,k}$$
$$= 1 \xrightarrow{\text{Substitute from (3)}} \sum_{j=0}^{L_{i,\text{retry}}} \sum_{k=0}^{\text{CW}_{i,j}-1} \frac{\text{CW}_{i,j} - k}{\text{CW}_{i,j}}$$
$$\cdot \frac{1}{1 - p_i} \cdot p_i^j \cdot b_{i,0,0} \tag{4}$$
$$= 1 \Longrightarrow$$
$$b_{i,0,0} = \frac{1 - p_i}{\sum_{j=0}^{L_{i,\text{retry}}} \sum_{k=0}^{\text{CW}_{i,j}-1} \left(\left(\text{CW}_{i,j} - k\right)/\text{CW}_{i,j}\right)}$$
$$= \frac{1 - p_i}{\sum_{j=0}^{L_{i,\text{retry}}} \left(1 + \sum_{k=1}^{\text{CW}_{i,j}-1} \left(\left(\text{CW}_{i,j} - k\right)/\text{CW}_{i,j}\right)\right)},$$

$$\tau_i = \sum_{j=0}^{L_{i,\text{retry}}} b_{i,j,0} \xrightarrow{\text{From (3), (4)}} \tau_i$$
$$= b_{i,0,0} \cdot \frac{1 - p_i^{L_{i,\text{retry}}}}{1 - p_i}$$
$$= \frac{1 - p_i^{L_{i,\text{retry}}+1}}{\sum_{j=0}^{L_{i,\text{retry}}} \left(1 + \sum_{k=1}^{\text{CW}_{i,j}-1} \left(\left(\text{CW}_{i,j} - k\right)/\text{CW}_{i,j}\right)\right) \cdot p_i^j}$$
$$1 \le i \le M, \tag{5}$$

$$p_b = 1 - \prod_{h=0}^{M-1} (1 - \tau_h)^{n_h},$$
$$p_i = 1 - \left(\prod_{h=0}^{i-1} (1 - \tau_h)^{n_h}\right) \cdot (1 - \tau_i)^{n_i - 1}$$
$$\cdot \left(\prod_{h=i+1}^{M-1} (1 - \tau_h)^{n_h}\right) \quad 1 \le i \le M.$$

If we define the successful transmission probability as the situation in which at a given time slot all stations refrain from transmission except one, then below equation would be apparent for this parameter:

$$p_{s,i} = n_i \cdot \tau_i \cdot (1 - \tau_i)^{n_i - 1}$$
$$\cdot \prod_{h=0, h \ne i}^{M-1} (1 - \tau_h)^{n_h} \quad 1 \le i \le M. \tag{6}$$

In above set of equations, M corresponds to the number of traffic categories exist in network and n_i to the number of active stations surviving at class i_{th}. The absolute success probability at a given station belonging to class i (instead of conditional one computed above) is achieved after removing the condition in (6) dividing it by p_b. Hence,

$$p_{\text{success},i} = \frac{p_{s,i} = n_i \cdot \tau_i \cdot (1 - \tau_i)^{n_i - 1} \cdot \prod_{h=0, h \ne i}^{M-1} (1 - \tau_h)^{n_h}}{p_b = 1 - \prod_{h=0}^{M-1} (1 - \tau_h)^{n_h}},$$
$$1 \le i \le M. \tag{7}$$

Now we are ready to introduce two important quantities that will be exploited in our next section treatments; the first one is the mean number of station retries to transmit its packet successfully; the second one is the mean number of consecutive idle slots in an idle burst. These two quantities play important roles in our treatments (referring to [18]):

$$N_{s,i} = \frac{1}{P_{\text{success},i}} - 1,$$
$$N_{\text{Idle}} = \frac{1}{p_b} - 1. \tag{8}$$

Up to now, we have presented formulas for EDCA. Hereafter, to the end of this section (and for the reason of simplification), we get back to DCF and extract a simplified equation for transmission probability based on absolute collision probability, p. According to [17], the transmission probability in DCF mode would be equal to

$$\tau = \frac{2 \cdot (1 - 2 \cdot p)}{(1 - 2 \cdot p) \cdot (\text{CW}_{\min} + 1) + p \cdot \text{CW}_{\min} \cdot \left(1 - (2 \cdot p)^m\right)}. \tag{9}$$

Assuming $(2 \cdot p)^m \underset{m \to \infty}{\to} 0$ (9) gets more simplified for τ versus CW and p,

$$\tau = \frac{2}{\text{CW}_{\min} \cdot ((1 - P)/(1 - 2 \cdot P)) + 1}. \tag{10}$$

Figure 4 demonstrates that for larger m and smaller p, the difference between two curves is trivial and the approximation is well.

The larger the value of m (the lower the value of p), the better two curves fit each other.

A delicate difference between DCF and EDCA is regarding the exact time that frozen counters start to decrease. In DCF, decrement occurs at the edge of first slot coming

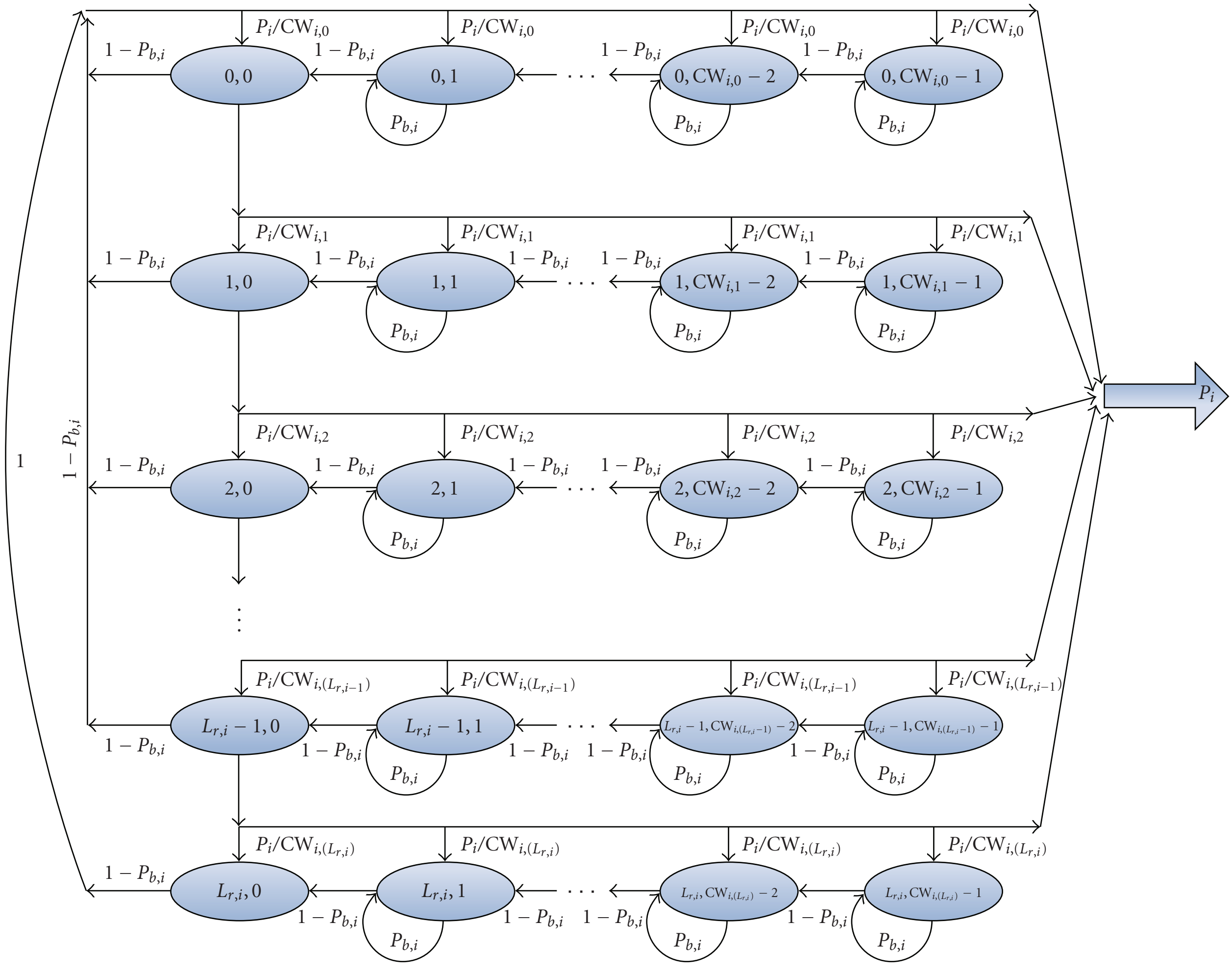

Figure 3: Two-Dimensional Markov chain of backoff mechanism in IEEE 802.11e networks.

immediately after DIFS idle time, whereas in EDCA the counter reduction is accomplished at the first edge of last AIFS idle slot. Although this little difference may not lead to a tangible influence on performance, it makes the analysis more convenient in EDCA case.

In EDCA mechanism, each station may have up to eight internal queues, each one representing a virtual station working independent of each other. In addition, each queue has its dedicated QoS parameters. When, two or more ACs (queue) inside a station attempt to transmit simultaneously, the virtual collision handler is activated. In this mechanism, among those ACs engaged in collision, the one with higher priority transmits right away and others refrain from transmission. This is in contrast with real collision situations in which two real concurrent transmissions lead to real collisions. Figure 5 illustrates this situation.

3.2. EDCA Performance Evaluation. In order to gain deeper understanding of our dynamic tuning scheme, we establish a set of simulations to study EDCA basic performance. The aim of this section is to prove that differentiation mechanism is only a tradeoff between different classes and cannot improve the overall level of QoS in network. In other word, using differentiation methods, when one of QoS metrics (Throughput, Delay, Utilization...) improves in some classes, we should certainly expect to see degradation in other classes (on the same metric).

We utilized predefined model of IEEE 802.11e existing in Opnet modeler.14 [19]. In this set of simulations, we apply constant traffic load to a group of stations. Traffic is generated based on exponential interarrival rate and constant payload size, the condition that happens in many real situations. We set Mean interarrival time to 0.005 second and mean packet size to 1500 byte, what envisaged being a high load condition. In order to set up a stable working condition, an offset time should elapse after simulation beginning and before traffic generation in stations. To evaluate EDCA performance, three traffic categories have been defined; first, interactive multimedia class; second, interactive voice class; third, Best effort class.

The exploited topology is a ring of radius 20 meter over which 6 stations placed in equal distances. Each two stations belong to the same class standing face to face. No channel fading or noise effects are considered in these scenarios and

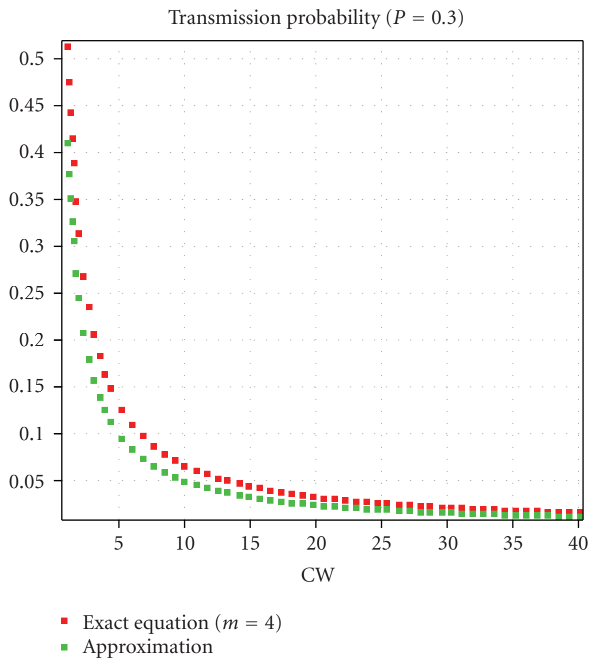

Figure 4: Comparison of Exact and approximate Tau-CW plots. Evidently, the larger the value of m is (the lower the value of p), the better two curves fit each other.

Table 1: Physical Layer Parameters.

Physical characteristic	DSSS
Data Rate	5.5 Mbit/s
Transmit Power (W)	0.005
Packet Reception Power Threshold (dBm)	−95
RTS Threshold (byte)	None
Fragmentation Threshold (byte)	None
CTS to Self Option	Enabled
Short Retry Limit	7
Long Retry Limit	4
Max Receiver Lifetime (s)	0.5
Buffer Size (bits)	256,000
Roaming Capability	Disabled

the only source of error is collision. Also hidden terminal possibility is avoided by putting all stations in hearing range of each other. For the sake of comparison, all three ACs benefit from the same traffic load condition. Other physical layer parameters are cited at Table 1.

It is important to note that, no access point functionality is considered in this article and all stations work in a distributed arrangement as illustrated in Figure 6.

Note that stations in each class only interact with each other, not with stations of other classes; that is to say voice by voice and video by video. At this section, we establish two sets of simulations; firstly, we evaluate network behavior by changing application load and secondly, we evaluate performance by changing MAC layer parameters like CW_{min},

Table 2: MAC Layer Parameters.

AC	CW_{min} (Slot)	CW_{max} (Slot)	AIFS (Slot)	TXOP (μs)
VO	7	255	2	3264
VI	31	511	2	3264
BE	63	2047	2	3264

Transmission Opportunity (TXOP), and AIFS. In the first set, we apply MAC parameters as mentioned in Table 2.

Figure 7 depicts load level and load variation for each of the tree scenarios. In each scenario, we have 100% load increment than last scenario. Figure 8 depicts total delay for these three different load conditions. This plot emphasizes that as stations' load grows larger, delays increase boundlessly.

Figure 9 shows delay trends of voice class for three different load situations (with 100% growth rate at each situation). As other two classes exhibited the same trend as the voice class, we avoid showing them in this place.

Figures 10 and 11 show, respectively, Delay-Time trend and Delay-CDF (Cumulative Distribution Function) trend in all three classes under highest load condition (0.005 second). The comparisons simply reveal that higher priority class has greater chance to access the channel.

Now it is time to observe other parameters' effects (like CW_{min}, TXOP, and AIFS) on EDCA performance. Let us start with CW_{min}. Firstly, we change CW_{min} in both voice classes simultaneously and equally and observe its changing effects on delay value of all classes. As it is obvious, this changing should have direct influence on voice class delay as well as other class delays. However, the interesting fact is something else. The totals mean delay never changes in these scenarios; but what happens is just a simple tradeoff between classes meaning that higher priority class grasps more chance to transmit while the lower priority class looses it, and hence it starves; but the total mean stays always intact. Figures 12 and 13 are good illustrations to envisage the situation. All parameter is the same as Table 2 but the voice class CW_{min} that takes three values: 7, 31, and 63.

Because the best effort class showed the same trend as video class, we avoid mentioning it here. By aggregating delays in all classes, we reach to below plot that discloses an important fact. The differentiation mechanism has no impact on overall delay improvement but instead is just a simple interclass tradeoff. This is evident from Figure 14 that mean values are equal but depending on many factors we may decrease or increase delay variation (Jitter).

Now we have enough tools in hand to follow our aim, which is our novel adaptive tuning method.

4. Adaptive Method of Tuning Initial Backoff Windows Size

As a starting point in this section, we go over a simple generic definition for throughout; this universal definition is the

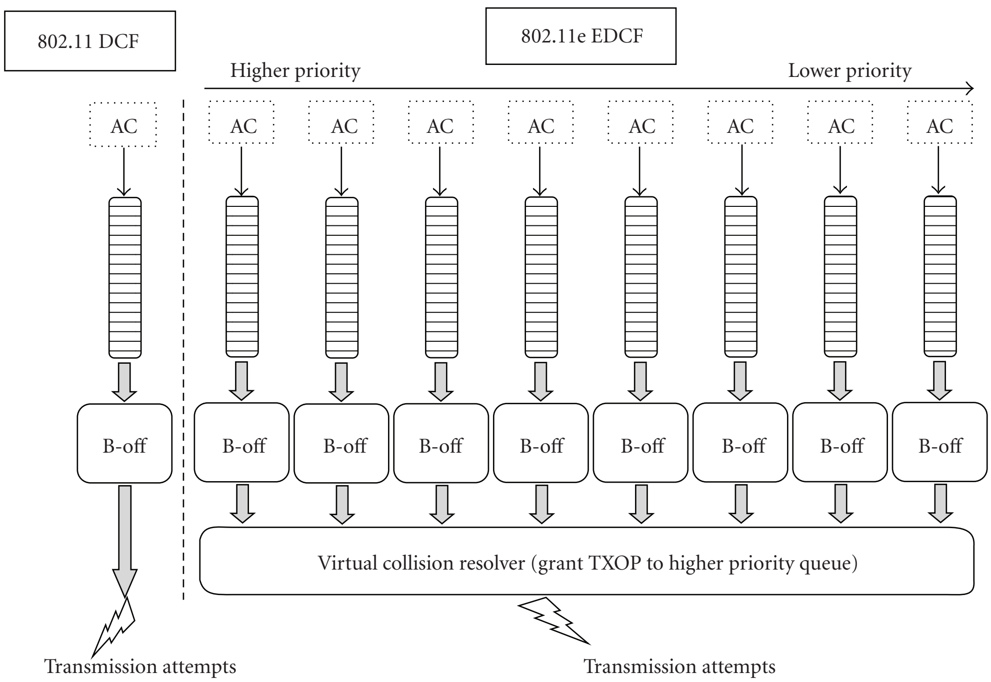

FIGURE 5: ACs in each real station. There are eight traffic categories with different parameters in each station: (1) Left figure shown DCF with AIFS = 34 μs, CW_{min} = 15, PF = 2; (2) Right figure shown EDCA with AIFS [AC] ≥ 34 μs, CW_{min} [AC] = 0–255.

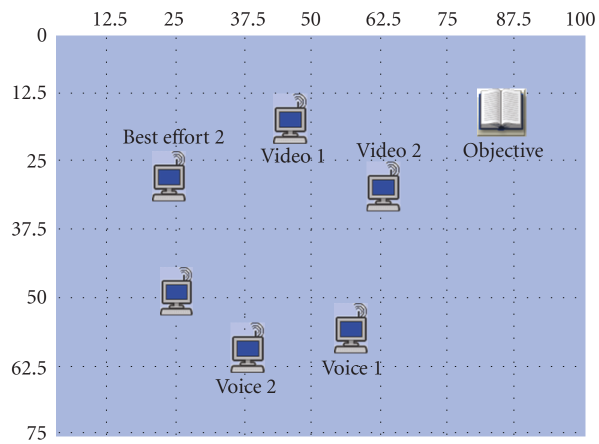

FIGURE 6: Utilized topology. Six stations placed in equal distances in a Ring topology.

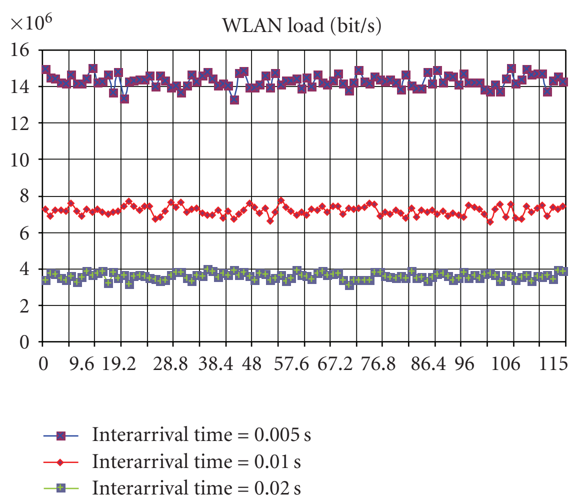

FIGURE 7: Load parameter depicted above for three scenarios. In each scenario, we have 100% load decrement than last scenario.

amount of successfully transmitted payload in a transmission period. Therefore,

$$\text{Throu} = \frac{T_{\text{payload}} \cdot p_{\text{success}}}{(T_s + N_{\text{Idle}}) \cdot p_b}. \tag{11}$$

This equation is logical and need no more explanation. In order to find an optimized point at which delay is minimized (and consequently throughput maximized), it is necessary to find the maximal point of above equation by taking first derivative and equating it to zero. Therefore,

$$\frac{d(\text{Throu})}{d(\text{CW})} = 0 \longrightarrow \frac{d(\text{Throu})}{d(\tau)} \cdot \frac{d(\tau)}{d(\text{CW})} = 0,$$

$$\frac{d(\tau)}{d(\text{CW})} \neq 0 \quad \text{because } \tau = \frac{2}{\text{CW}_{\min} \cdot ((1-P)/(1-2 \cdot P)) + 1}. \tag{12}$$

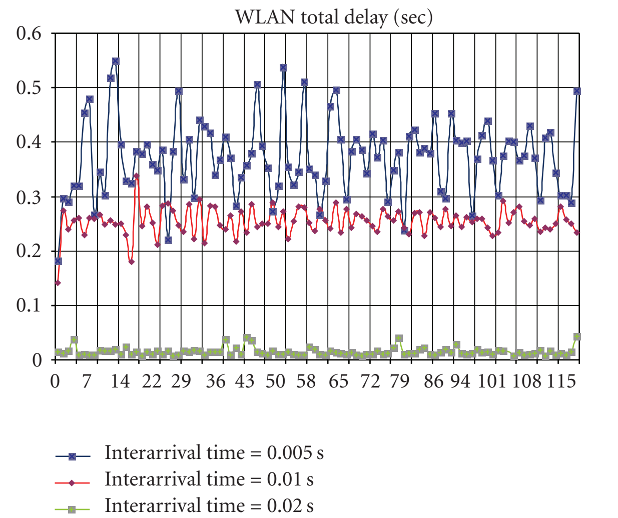

FIGURE 8: Total delay for each load condition. As load increases, the total delay grows correspondingly.

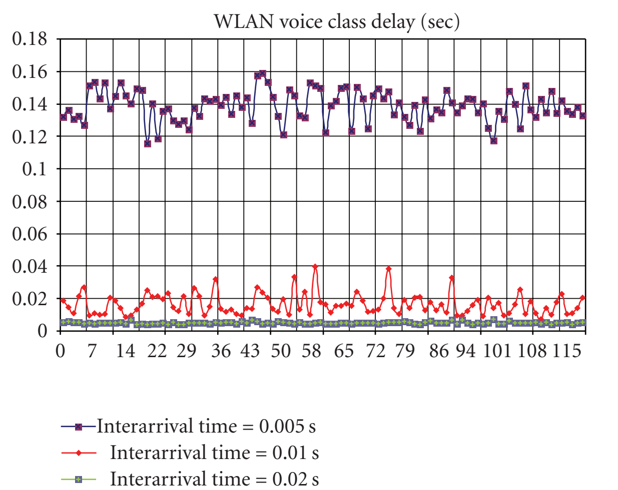

FIGURE 9: Voice class delay comparison for three load situations. As load increases, this quantity grows correspondingly.

This could be possible only if $d(\text{Throu})/d(\tau) = 0$. After derivation and algebraic simplification:

$$(1-\tau)^{n-2} \cdot (1 - n \cdot \tau) \cdot (T_s - (T_s - 1) \cdot (1-\tau)^n) - n \cdot (T_s - 1) \cdot \tau \cdot (1-\tau)^{2 \cdot n - 2} = 0 \longrightarrow \frac{T_s}{T_s - 1} \cdot (1 - n \cdot \tau) = (1-\tau)^n. \tag{13}$$

The intersection point at Figure 15 represents the value of τ at which Throughput maximized.

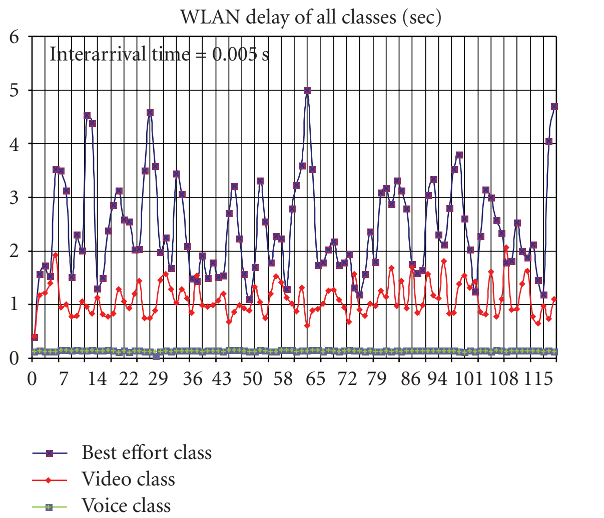

FIGURE 10: Delay Time-Trend for all traffic classes (0.005-second interarrival time).

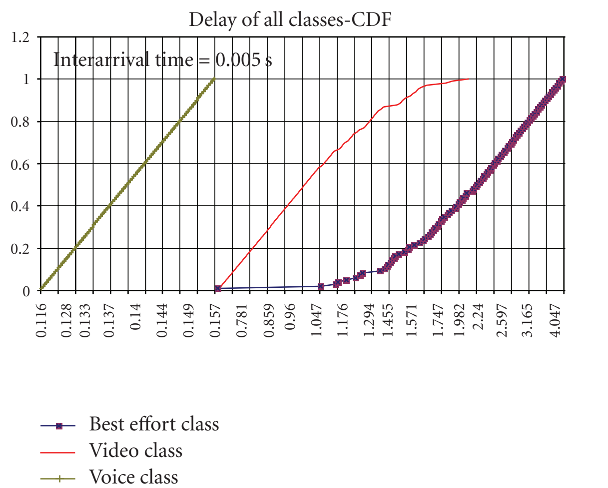

FIGURE 11: Delay CDF (Cumulative Distribution Function) Trend for all traffic classes (0.005-second interarrival time).

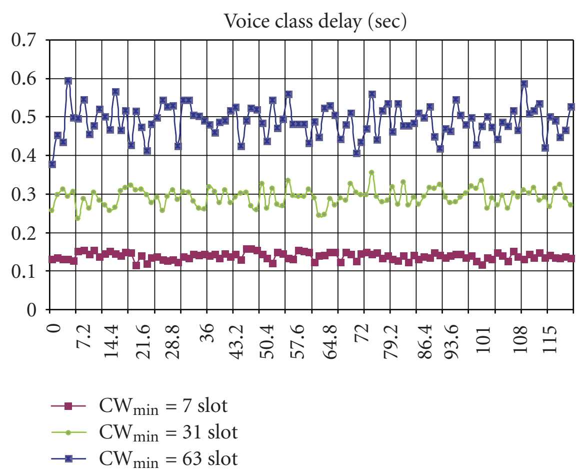

FIGURE 12: Voice class delay for three different CW_{min} value.

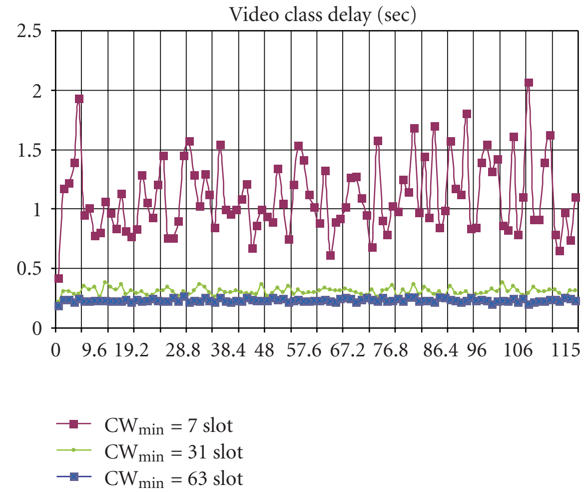

FIGURE 13: Video class delay decrease due to increment of voice class CW_{min}.

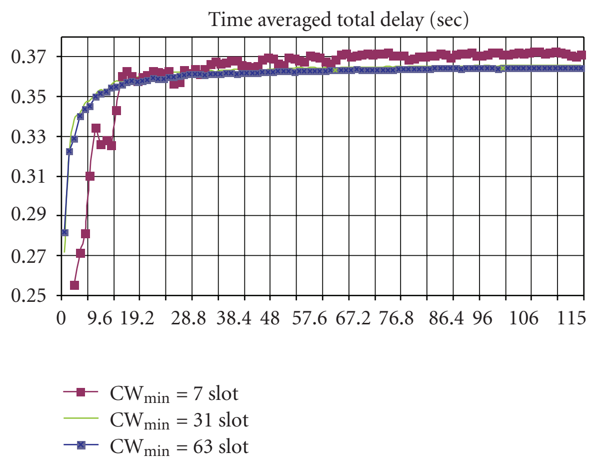

FIGURE 14: Altering voice class CW_{min} has no sensible impact on Overall Delay although each class delay is affected differently.

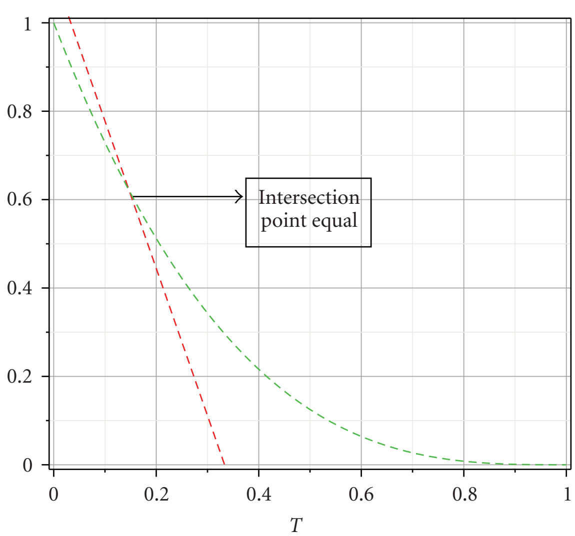

FIGURE 15: Optimum Tau value at which Throughput maximized (Intersection Point).

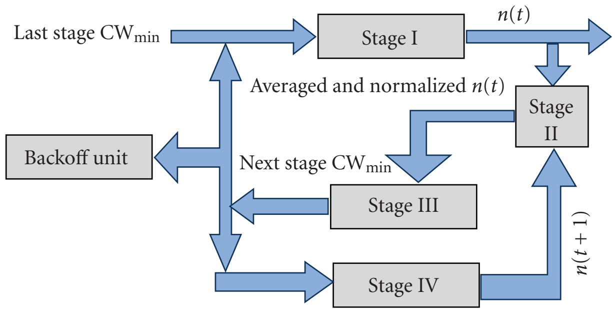

FIGURE 16: Our proposed schematic flowchart diagram.

By further simplification in right hand side of equation, we reach to

$$
\begin{aligned}
\tau &= -\frac{T_s/(T_s-1)-1}{n-1} - \sqrt{\left(\frac{T_s/(T_s-1)-1}{n-1}\right)^2 + \frac{2}{n}\cdot\frac{T_s/(T_s-1)-1}{n-1}} \\
&= \frac{T_s/(T_s-1)-1}{n-1} \cdot \left(-1+\sqrt{1+\frac{2}{n}\cdot\left(\frac{n-1}{T_s/(T_s-1)-1}\right)}\right) \\
&\xrightarrow{n\approx n-1} \left(\frac{T_s/(T_s-1)-1}{n-1}\right)\cdot\left(\sqrt{2\cdot T_s-1}-1\right) \\
&= \frac{2}{\left(\sqrt{2\cdot T_s-1}+1\right)\cdot(n-1)}.
\end{aligned}
\tag{14}
$$

Now we merge (14) with (10):

$$
\begin{aligned}
&\frac{2}{\left(\sqrt{2\cdot T_s-1}+1\right)\cdot(n-1)} \\
&\quad = \frac{2}{CW_{min}\cdot((1-P)/(1-2\cdot P))+1} \\
&\quad \longrightarrow CW_{min} = \frac{(n-1)\cdot\sqrt{2\cdot T_S-1}}{(1-P)/(1-2\cdot P)}.
\end{aligned}
\tag{15}
$$

Using this equation we could adaptively compute CW_{min} as a function of n (each terminal estimation of number of contending stations is not the real number of stations!), p, and Ts. This equation constitutes the heart of our recycling flowchart in third stage (Figure 16).

4.1. Schematic Diagram. To accomplish our task, we have drawn a schematic cyclic flowchart. Let us proceed through its all stages to derive appropriate practical equations. Up to now, we have derived a suitable equation for third stage (14). Now, its first stage's turn!

4.1.1. Stage I. Let us suppose S_B to be the number of busy slots observed by an observing station during a period of length B. Therefore, it is logical to infer that the channel busyness probability from this station's perspective would be equal to

$$\frac{S_B}{B} = 1-(1-\tau)^{n-1} \longrightarrow S_B = B\cdot\left(1-(1-\tau)^{n-1}\right). \tag{16}$$

By the second order approximation of $(1-\tau)^{n-1}$ term using Newton's equation, it turns possible to simplify above equation as

$$S_B = B\cdot\left(1-(1-\tau)^{n-1}\right) \Longrightarrow S_B = B\cdot\left(1-\left(1-(n-1)\cdot\tau + \frac{(n-1)\cdot(n-2)}{2}\cdot\tau^2\right)\right). \tag{17}$$

By further simplification of this equation and arranging n as a function of $\mathrm{CW}_{\min}$ and p and Coupling it with (10) below equation is obtained:

$$\begin{aligned} n-1 &= \frac{1}{2}+\frac{1}{\tau}-\sqrt{\left(\frac{1}{2}+\frac{1}{\tau}\right)^2-\frac{2}{\tau^2}\cdot\frac{S_B}{B}} \\ \tau &= \frac{2}{\mathrm{CW}_{\min}\cdot((1-P)/(1-2\cdot P))+1} \end{aligned} \Longrightarrow$$

$$\begin{aligned} n &= 1+\frac{((1-P)/(1-2\cdot P))\cdot \mathrm{CW}_{\min}+2}{2} \cdot\left(1-\sqrt{1-2\cdot\frac{S_B}{B}}\right) \\ &= 1+\frac{((1-P)/(1-2\cdot P))\cdot \mathrm{CW}_{\min}+2}{2}\cdot p_b. \end{aligned} \tag{18}$$

This is a simple, linear, and effective equation and permits each station to estimate the number of contending terminal every time required. In cases of small busyness probability, (light contention) $\sqrt[m]{1-x} \approx 1-x/m$ is a suitable and tight approximation for $x \ll 1$; thus, a linear equation with least difficulty in computations is obtained. As p_b and S_B/B are closely equal, we could apply them interchangeably.

4.1.2. Stage II. In order to avoid rapid and undesirable changes in $n(t)$, which arise from incorrect estimation of p_b or S_B/B by stations, we devised an effective solution with less error that could furnish network with larger stability and persistency. To that end, we utilize a buffer to save several consecutive $n(t)$ values. Then, by applying a weighted normalization routine to these consecutive values, an implicit stabilizer is formed:

$$n(t+1) = \alpha\cdot n(t) + (1-\alpha) \cdot\frac{\sum_{j=1}^{q} n(t-j)}{q}, \quad 0<\alpha<1. \tag{19}$$

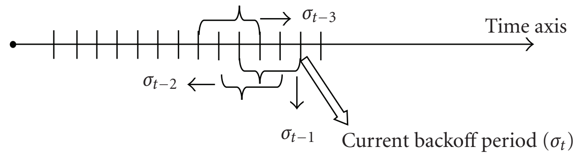

FIGURE 17: A time axis for Dynamic $\alpha(t)$ Tuning.

Clearly, α and q are both constant values that are tunable depending on requirements. The larger α results in more variable and instantaneous $n(t+1)$ and hence an unreliable behavior with greater variations, but with swifter acting accomplished. Despite that, the smaller α value leads to stable $n(t+1)$ behavior with little variation and slow movements.

4.1.3. Stage III. At this stage we apply (15) to adaptively compute $\mathrm{CW}_{\min}$ as a function of $n(t)$ (which is an estimation of channel traffic). Finally, this initial window value will be exploited by stations for next transmissions.

4.1.4. Stage IV. As long as α value stays constant during backoff periods, we may expect trivial performance improvement when working in adaptive tuning mode. To achieve better improvements, it is better to tune α at each stage according to an appropriate logic. We have verified the fitness of below set of equations (20) and a comparison done with a scenario in which α stayed static. The comparison proved a sensible improvement in all quantities when dynamically tuning α. For clarification, imagine below axis (Figure 17) that divided into separate divisions by straight lines. Each part is a backoff period during which a packet will be transmitted successfully after several trials. In addition, σ_{t-j} shows the mean variation of $\mathrm{CW}_{\min}$ during q last periods. We are interested to find $\alpha(t)$ for next period based on most recent α (namely, $\alpha(t-1)$). Thus, we follow these equations:

$$\begin{aligned} \alpha(t) &= \alpha(t-1)\cdot\sqrt{\frac{\sigma_{t-1}^2}{(1/A)\cdot\sum_{j=1}^{A}\sigma_{t-j}^2}}, \\ \sigma_{t-j}^2 &= \sum_{i=j}^{q+j}\frac{\mathrm{CW}_{\min}(t-i)}{q} - \left(\frac{\sum_{i=j}^{q+j}\mathrm{CW}_{\min}(t-i)}{q}\right)^2. \end{aligned} \tag{20}$$

As it is evident, our goal from exploiting several summations in this set of equations is to have a smoother $\alpha(t)$ as far as possible. The principal philosophy behind using the right hand side equation is that, upon an abrupt change in $\mathrm{CW}_{\min}$ of previous stage, the $\alpha(t)$ increases automatically to provide more dependency on exactly former $n(t)$ rather than farther ones (refer to equation... $n(t+1) = \cdots$).

Finally, when there was no variation in last stage of $\mathrm{CW}_{\min}$, $\alpha(t)$ remains almost unchanged. We release further explanation at this point.

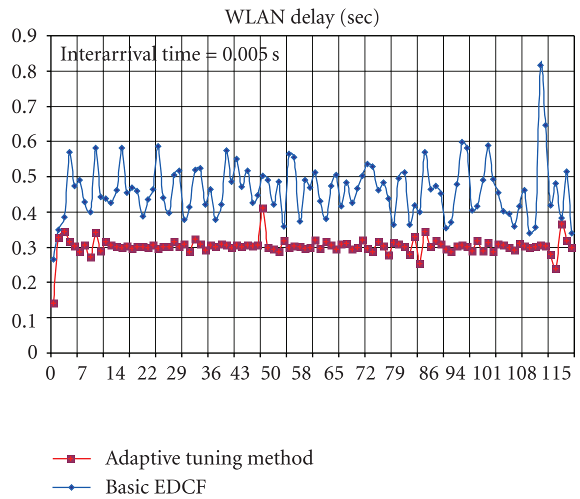

Figure 18: Total delay compared for both scenarios. Total delay decreased in CW_{min}-ATM in comparison with EDCA.

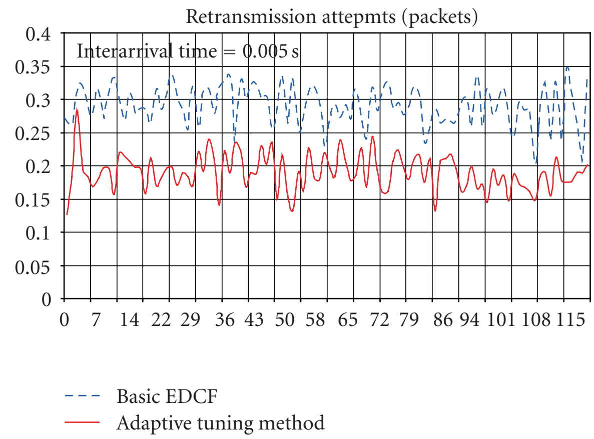

Figure 20: Comparison of total retransmission attempts for CW_{min}-ATM and basic EDCA method. Also this parameter lessened considerably because collision probability is minor in CW_{min}-ATM.

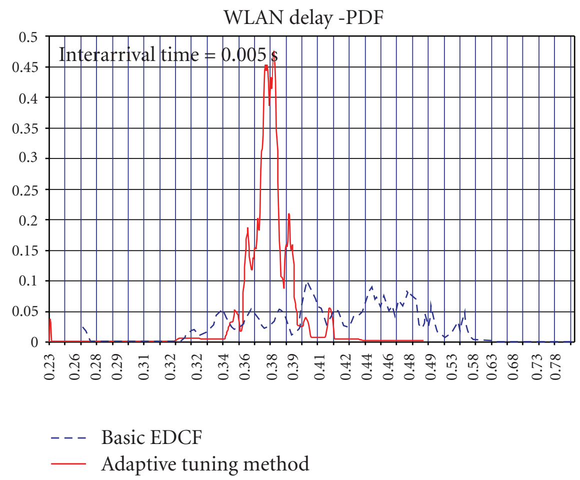

Figure 19: Probability Density Function (PDF) of total delay for both scenarios. Jitter and Delay both decreased (Improved) in CW_{min}-ATM.

5. Simulation Results

In this section, we present simulation results of adaptive tuning method proposed in last section. Again, all results are achieved by applying network parameters mentioned in Tables 1 and 2. In addition, all traffic generation parameters are the same as what mentioned in Section 3.2 except interarrival time that is set to be 0.005 s in these set of simulations.

We evaluated several performance measures like Delay (Media access delay), Jitter, Throughput, Retransmission Attempts, and utilization to support our claims as strong as possible. We compare our proposed scheme with common EDCA in IEEE 802.11e. First, let us go over delay. Figure 18 shows the superiority of adaptive scheme against EDCA for total delay parameter.

Since Jitter (Delay Variation) is premier parameter in telecommunication than absolute delay, Figure 19 demonstrates how in addition to gaining smaller delay, even smaller jitter is achieved in adaptive method. In fact, the narrower picked PDF is the evident to this claim.

Another important parameter is total retransmission attempt. It is the total number of retransmission attempts until either a packet is successfully transmitted or it is discarded (because of reaching short or long retry limit). Figure 20 shows this parameter lessened considerably because collision probability is minor in adaptive method.

Utilization is the measure of the consumption to date of available channel bandwidth. This quantity ranges between 0–100 where a value of 100.0 would indicate full usage. Figure 21 presents utilization improvement for three classes.

At last, Throughput results presented in Figure 22, which is the total data traffic in bits/sec are successfully received and forwarded to the higher layer by each access category of the WLAN MAC. Here, we only present total throughput because we are interested to know how CW_{min}-ATM affect overall system performance. This figure shows that total throughput increases in adaptive method. The underneath reason is the decreased collision probability that have direct influence on decreasing the number of packets dropped due to retry threshold exceeding and ones dropped due to overflow. Hence, throughput increases.

6. Conclusion

In this paper, we presented a novel adaptive tuning method. We proved that applying this method on MAC layer of 802.11 family networks improves network performance and stability in a sensible manner. Early in this paper, we have established a simple set of simulations on 802.11e networks to give readers a wider perspective of what followed latter. Then simulation results of our adaptive method are presented and compared with simple legacy EDCA results.

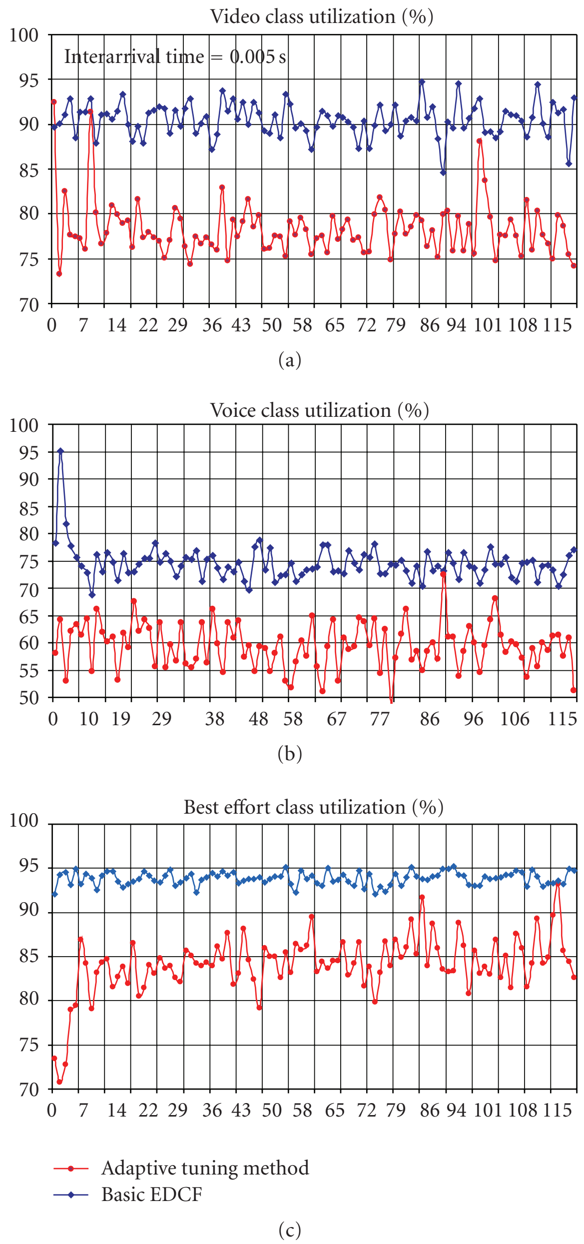

Figure 21: Utilization of Voice, Video, Best Effort classes for both scheme (EDCA and CW_{min}-ATM). Video and best effort class utilization improved (Lessened) but voice class utilization slightly increased.

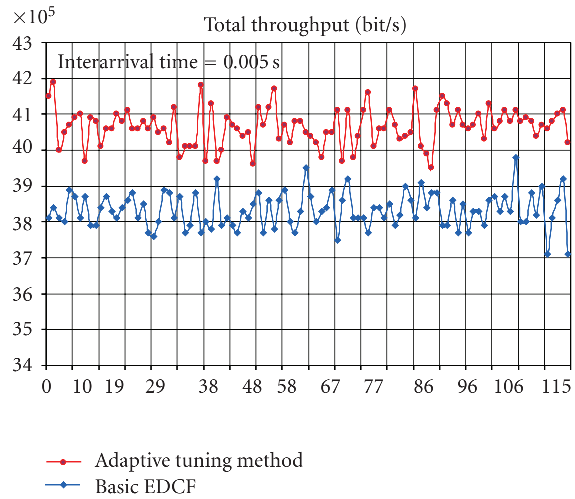

Figure 22: Total throughput comparison for basic and adaptive methods. This figure shows that total throughput increases in adaptive method.

The comparisons demonstrated that delay, throughput, retransmission attempt, and utilization were all positively impacted when applying it in backoff procedure of EDCA. These are all network quantities that could be considered. The differences between results were far enough to decisively vote to suitability of this method.

In the coming paper, we proposed a new QoS improving method based on prioritization. We showed that it is possible to combine CW_{min}-ATT and differentiation methods in order to achieve better QoS in 802.11e networks.

References

[1] "Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) Specifications," IEEE standards 802.11, January 1997.

[2] M. S. Chhaya and S. Gupta, "Performance modeling of asynchronous data transfer methods of IEEE 802.11 MAC protocol," *Wireless Networks*, vol. 3, no. 3, pp. 217–234, 1997.

[3] A. Kumar, E. Altman, D. Miorandi, and M. Goyal, "New insights from a fixed-point analysis of single cell IEEE 802.11 WLANs," *IEEE/ACM Transactions on Networking*, vol. 15, no. 3, pp. 588–601, 2007.

[4] Y. C. Tay and K. C. Chua, "A capacity analysis for the IEEE 802.11 MAC protocol," *Wireless Networks*, vol. 7, no. 2, pp. 159–171, 2001.

[5] J. Deng and R.-S. Chang, "A priority scheme for IEEE 802.11 DGF access method," *IEICE Transactions on Communications*, vol. E82-B, no. 1, pp. 96–102, 1999.

[6] Y. Xiao, "A simple and effective priority scheme for IEEE 802.11," *IEEE Communications Letters*, vol. 7, no. 2, pp. 70–72, 2003.

[7] A. Veres, A. T. Campbell, M. Barry, and L.-H. Sun, "Supporting service differentiation in wireless packet networks using distributed control," *IEEE Journal on Selected Areas in Communications*, vol. 19, no. 10, pp. 2081–2093, 2001.

[8] I. Aad and C. Castelluccia, "Differentiation mechanisms for IEEE 802.11," in *Proceedings of the 20th Annual Joint Conference on the IEEE Computer and Communications Societies (INFOCOM '01)*, vol. 1, pp. 209–218, Anchorage, Alaska, USA, April 2001.

[9] X. Pallot and L. E. Miller, "Implementing message priority policies over an 802.11 Based mobile ad hoc network," in *Proceedings of Military Communications Conference (Milcom '01)*, vol. 2, pp. 860–864, McLean, Va, USA, October 2001.

[10] IEEE 802.11e WG, "Medium Access Control (MAC) Enhancements for Quality of Service," IEEE 802.11e/D2.0, November 2001.

[11] S. Mangold, S. Choi, P. May, O. Kein, G. Hiertz, and L. Stibor, "IEEE 802.11e Wireless LAN for quality of service," in *Proceedings of European Wireless*, pp. 32–39, Florence, Italy, 2002.

[12] Y. Xiao, "IEEE 802.11 E: QoS provisioning at the MAC layer," *IEEE Wireless Communications*, vol. 11, no. 3, pp. 72–79, 2004.

[13] Y. Xiao, H. Li, and S. Choi, "Protection and guarantee for voice and video traffic in IEEE 802.11e wireless LANs," in *Proceedings of the 23rd Annual Joint Conference of the IEEE Computer and Communications Societies (INFOCOM '04)*, vol. 3, pp. 2152–2162, Hong Kong, March 2004.

[14] F. Cali, M. Conti, and E. Gregori, "IEEE 802.11 protocol: design and performance evaluation of an adaptive backoff mechanism," *IEEE Journal on Selected Areas in Communications*, vol. 18, no. 9, pp. 1774–1786, 2000.

[15] J. Weinmiller, H. Woesner, J.-P. Ebert, and A. Wolisz, "Modified backoff algorithms for DFWMAC's distributed coordination function," in *Proceedings of the 2nd ITG Fachtagung Mobile Kommunikation*, pp. 363–370, Neu-Ulm, Germany, September 1995.

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

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

[18] E. Ziouva and T. Antonakopoulos, "CSMA/CA performance under high traffic conditions: throughput and delay analysis," *Computer Communications*, vol. 25, no. 3, pp. 313–321, 2002.

[19] "Opnet Modeler," http://www.opnet.com/.