



**HAL**  
open science

## Queueing Delay Analysis of IEEE 802.11e EDCA

Paal E. Engelstad, Olav N. Østerbo

► **To cite this version:**

Paal E. Engelstad, Olav N. Østerbo. Queueing Delay Analysis of IEEE 802.11e EDCA. WONS 2006 : Third Annual Conference on Wireless On-demand Network Systems and Services, INRIA, INSA Lyon, Alcatel, IFIP, Jan 2006, Les Ménuires (France), pp.123-133. inria-00001016

**HAL Id: inria-00001016**

**<https://inria.hal.science/inria-00001016>**

Submitted on 30 Jan 2006

**HAL** is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

# Queueing Delay Analysis of IEEE 802.11e EDCA

Paal E. Engelstad and Olav N. Østerbø

**Abstract**— The majority of analytical work on the performance of IEEE 802.11 [1] focuses on predicting the throughput and the mean delay of only the medium access, although higher layer applications and protocols are interested in the total performance of the MAC layer. Seen in this perspective, surprisingly little focus has been on predicting the queueing delay. The main contribution of this paper opposed to other works is that it presents the full delay distribution through the z-transform. As a result, the mean medium access delay is found by the first order moment of the transform, and the mean queueing delay by the second order moment. Together this gives the average total delay associated with the MAC layer. The z-transform is derived from an analytical model that works in the whole range from a lightly loaded, non-saturated channel to a heavily congested, saturated medium. The model describes the priority schemes of the Enhanced Distributed Channel Access (EDCA) mechanism of the IEEE 802.11e standard [2]. EDCA provides class-based differentiated Quality of Service (QoS) to IEEE 802.11 WLANs, and distinguishes between four different traffic classes – called Access Categories (AC). By setting the number of ACs to one, and by using an appropriate parameter setting, the results presented are also applicable to the legacy 802.11 Distributed Coordination Function (DCF) [1]. The model predictions are calculated numerically and validated against simulation results. A good match between the analytical predictions and simulations was observed.

**Index Terms**—802.11e, Queueing Delay, Performance Analysis, EDCA, Z-transform of the Delay, Virtual Collision, Non-Saturation, AIFS, Starvation.

## I. INTRODUCTION

**D**URING recent years the IEEE 802.11 WLAN standard [1] has been widely deployed as the most preferred wireless access technology in office environments, in public hot-spots and in the homes. Due to the inherent capacity limitations of wireless technologies, the 802.11 WLAN easily becomes a

bottleneck for communication. In these cases, the Quality of Service (QoS) features of the IEEE 802.11e standard [2] will be beneficial to prioritize for example voice and video traffic over more elastic data traffic.

The 802.11e amendment works as an extension to the 802.11 standard, and the Hybrid Coordination Function (HCF) is used for medium access control. HCF comprises the contention-based Enhanced Distributed Channel Access (EDCA) and the centrally controlled Hybrid Coordinated Channel Access (HCCA). EDCA has received most attention recently, and it seems that this is the WLAN QoS mechanism that will be promoted by the majority of vendors. EDCA is therefore the area of interest of this paper, and HCCA will not be discussed any further here.

EDCA allows for four different access categories (ACs) at each station and a transmission queue associated with each AC. Each AC at a station has a conceptual module responsible for channel access for each AC, and in this paper the module is referred to as a "backoff instance".

The majority of analytical work on the performance of 802.11e EDCA focuses on predicting the throughput, the frame dropping probabilities and the mean delay of the medium access. Surprisingly little focus has been on predicting also the queueing delay.

The importance of the queueing delay is evident. In realistic network scenarios, most of the MAC frames will carry a higher-layer packet, such as a TCP/IP or a RTP/UDP/IP packet, in the payload. A higher layer protocol or application will normally not interfere with the inner workings of the MAC layer. It might observe that it is subject to network delay (which is the case for TCP and many applications running on top of RTP), but it will normally not be able to distinguish between the types of delay. Thus, in most cases it is the total delay that counts. For analytical predictions of the delay of 802.11e EDCA to be useful, *both the queueing delay and the medium access delay should be considered.*

With little generated traffic (or with rate limiting e.g. in order to satisfy the Differentiated Services Expedited Forwarding Per-Hop-Behaviour) the mean queue length can be less than a packet, and the medium access delay is dominant. However, this case is not of the highest interest. First, the medium access delay then is typically less than a couple of milli-seconds (ms), and can normally be neglected compared to the comparably higher total end-to-end delay experienced for common Internet communication. Second, QoS analysis is not

Manuscript received September 19, 2005. This work was supported in part by the Open Broadband Access Network (OBAN) STREP project of the 6<sup>th</sup> Framework Program of the European Commission.

Paal E. Engelstad is with Telenor R&D, 1331 Fornebu, Norway (phone: +47 41633776; fax: +47 67891812; e-mail: paal.engelstad@telenor.com). He is also associated with "Universitetsstudiene på Kjeller" (UniK University Graduate Center).

Olav N. Østerbø is also with Telenor R&D, 1331 Fornebu, Norway (phone: +47 48212596; fax: +47 67891812; e-mail: olav-norvald.osterbo@telenor.com).

of very high interest with abundant channel resources. It is at the point when the channel becomes saturated that the differentiating features of 802.11e EDCA comes into play.

When the inter-arrival time of the generated traffic approaches the medium access delay, the queue begins to grow. It is observed that the medium access delay is still comparably low in this situation, while the queueing delay easily becomes dramatically higher. (The benefit of keeping the queue finite as a counter-measure is normally restricted, since a higher layer protocol is indifferent to whether the delay of the packet exceeds the limit for being useful or whether the packet is dropped in the queue.)

This paper presents a prediction of the mean queueing delay in addition to the mean medium access delay also predicted in earlier works. The z-transform of the delay is first found. This can provide all higher order moments of the delay. With the second order moment at hand, the mean queueing delay is easily derived. However, in order to derive the z-transform of the delay, an analytical model that also covers non-saturated channel conditions is first needed.

Most of the recent analytical work on the performance of 802.11e EDCA stems from the simple and fairly accurate model proposed by Bianchi [3] to calculate saturation throughput of 802.11 DCF. Later, Ziouva and Antonakopoulos [4] improved the model to find saturation delays, however, still of the undifferentiated DCF. They also improved the model by stopping the backoff counter during busy slots, which is more consistent with the IEEE 802.11 standard. Based on this work, Xiao [5] extended the model to the prioritized schemes provided by 802.11e by introducing multiple ACs with distinct parameter settings, such as the minimum and maximum contention window. Furthermore, this model also introduced finite retry limits. These additional differentiation parameters lead to more accurate results than previous models. (A list of references for other relevant efforts and model improvements of DCF can also be found in [5].)

We use a version of Xiao’s model, however, extended as follow:

- The presented model predicts the performance not only in the saturated case, but in the whole range from a non-saturated medium to a fully saturated channel. (Some works, such as [6] and [7] have explored non-saturated conditions, however, only of the one-class 802.11. They are also primarily focussing on the non-saturation part instead of finding a good descriptive solution for the whole range.)
- In the non-saturation situation, our model accounts for “post-backoff” of an AC, although the queue is empty, according to the IEEE 802.11 standard. If the packet arrives in the queue after the “post-backoff” is completed, the listen-before-talk (or CSMA) feature of 802.11 is also incorporated in the model.
- Our model describes the use of AIFSN as a differentiating parameter, in addition to the other differentiation

parameters encompassed by Xiao’s efforts and other works.

- Virtual Collisions between the different transmission queues internally on a node are incorporated in the model.

The remaining part of the paper is organized as follows: The next section summarizes the differentiation parameters of 802.11e and provides the basis for understanding the analytical model. Section III presents the analytical model with AIFS differentiation and starvation prediction. Expressions for the throughput are first presented in Section IV (to give a complete presentation of the model), although it is the delay expressions presented in Section V that are the main contributions of this paper. The z-transform of the delay is first found. Then, the medium access delay is found as the first order moment of the transform. Finally, the queueing delay is found by means of the second order moment. In Section IV, the throughput expressions of the model are first validated against simulations to illustrate the accuracy of the model. Then, the mean access delay is validated, mainly because the prediction of the traffic intensity at which the queues grows to infinity, depends on it. By the end of the validation section, the queueing delay expression is validated. Our findings are finally summarized in the conclusions.

## II. DIFFERENTIATION PARAMETERS OF 802.11E

### A. Selecting Contention Windows (CWs)

The traffic class differentiation of EDCA is based on assigning different access parameters to different ACs. First and foremost, a high-priority AC,  $i$ , is assigned a minimum contention window,  $CW_{i,\min} + 1$ , and maximum contention window,  $CW_{i,\max} + 1$ , that are lower than (or at worst equal to) that of a lower-priority AC.

For each AC,  $i(i = 0, \dots, 3)$ , let  $W_{i,j}$  denote the contention window size in the  $j$ -th backoff stage i.e. after the  $j$ -th unsuccessful transmission; hence  $W_{i,0} = CW_{i,\min} + 1$ . Let also  $j = m_i$  denote the  $j$ -th backoff stage where the contention window has reached  $CW_{i,\max} + 1$ . Finally, let  $L_i$  denote the retry limit of the retry counter. Then:

$$W_{i,j} = \begin{cases} 2^j W_{i,0} & j = 0, 1, \dots, m_i - 1 \\ 2^{m_i} W_{i,0} = CW_{i,\max} + 1 & j = m_i, \dots, L_i \end{cases} \quad (1)$$

### B. Arbitration Inter-Frame Spaces (AIFSs)

Another important parameter setting is the Arbitration Inter-Frame Space (AIFS) value. When a backoff instance senses that the channel is idle after a packet transmission, it normally waits a guard time, AIFS, during which it is not allowed to transmit packets or do backoff countdown. Each AC[i] of 802.11e uses an Arbitration Inter-Frame Space (AIFS[i]) that consists of a SIFS and an AIFSN[i] number of additional time slots. In this paper  $A_i$  is defined as:

$$A_i = AIFSN[i] - AIFSN[N-1] , \quad (2)$$

where  $N$  is the number of different ACs (i.e. normally four), and  $AIFSN[N-1]$  is the AIFSN value of the highest priority AC, i.e. the lowest possible value. The 802.11e standard mandates that  $AIFSN[i] \geq 2$ , where the minimum limit of 2 slots corresponds to the Distributed Interframe Space (DIFS) interval of legacy 802.11.

### C. Transmission Opportunities (TXOPs)

Due to space limitations, priority based on differentiated Transmission Opportunity (TXOP) limits is not treated explicitly in this paper. Calculating the model with respect to different packet lengths and adjusting it to also cover contention-free bursting (CFB) is not difficult.

## III. ANALYTICAL MODEL

### A. The Markov Model

Figure 1 illustrates the Markov chain for the transmission process of a backoff instance of priority class  $i$ .

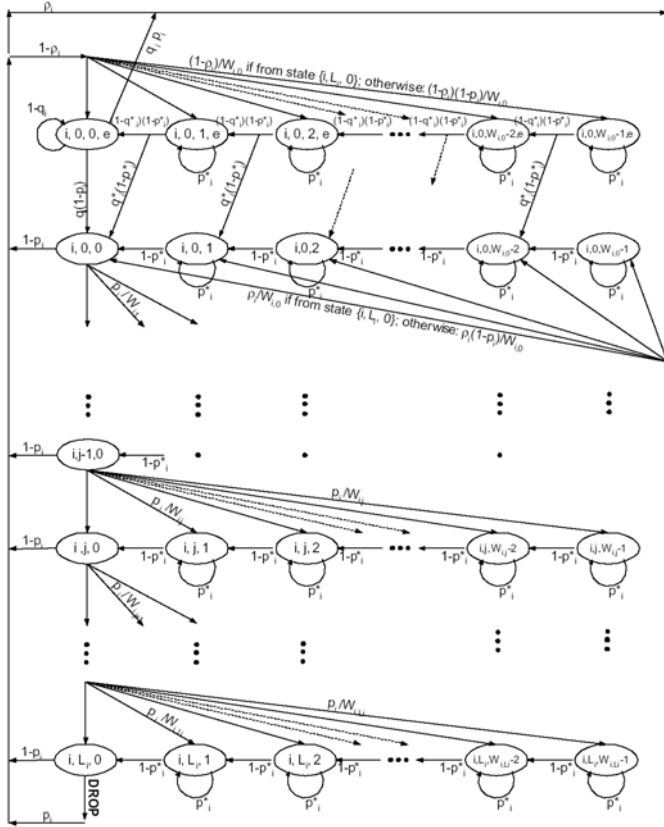


Figure 1. Markov Chain (both saturation and non-saturation)

In the Markov chain, the utilization factor,  $\rho_i$ , represents the probability that there is a packet waiting in the transmission queue of the backoff instance of AC  $i$  at the time a transmission is completed (or a packet dropped). Now, the backoff selects a backoff interval  $k$  at random and goes into

post-backoff. If the queue is empty, at a probability  $1 - \rho_i$ , the post-backoff is started by entering the state  $(i, 0, k, e)$ . If the queue on the other hand is non-empty, the post-backoff is started by entering the state  $(i, 0, k)$ . Hence,  $\rho_i$  balances the fully non-saturated situation with the fully saturated situation, and therefore plays a role to model the behaviour of the intermediate semi-saturated situation. When  $\rho_i \rightarrow 1$  the Markov chain behaviour approaches that of the saturation case similar to the one presented by Xiao [5].

On the other hand, when  $\rho_i \rightarrow 0$  the Markov chain models a stochastic process with a channel that is non-saturated. Then the backoff instance will always go into “post-backoff” after a transmission without a new packet ready to be sent.

While in the “post-backoff” states  $(i, 0, k, e)$  where  $k > 0$ , the probability that a backoff instance of AC  $i$  is sensing the channel busy and is thus unable to count down the backoff slot from one timeslot to the other is denoted with the probability  $p_i^*$ . If it has received a packet while in the previous state at a probability  $q_i^*$ , it moves to a corresponding state in the second row with a packet waiting for transmission. Otherwise, it remains in the first row with no packets waiting for transmission.

While in the state  $(i, 0, 0, e)$ , the backoff instance has completed post-backoff and is only waiting for a packet to arrive in the queue. If it receives a packet during a timeslot at a probability  $q_i$ , it does a “listen-before-talk” channel sensing and moves to a new state in the second row, since a packet is now ready to be sent. If the backoff instance senses the channel busy, at a probability  $p_i$ , it performs a new backoff. Otherwise, it moves to state  $(i, 0, 0)$  to do a transmission attempt. The transmission succeeds at a probability  $1 - p_i$ . Otherwise, it doubles the contention window and goes into another backoff.

For each unsuccessful transmission attempt, the backoff instance moves to a state in a row below at a probability  $p_i$ . If the packet has not been successfully transmitted after  $L_i + 1$  attempts, the packet is dropped.

Let  $b_{i,0,k,e}$  and  $b_{i,j,k}$  denote the state distributions of the Markov chain. Since, the probability that transmission attempts enter stage  $j$  (where  $j = 0, 1, \dots, L_i$ ) is  $p_i^j$ , chain regularities yield:

$$b_{i,j,0} = p_i^j b_{i,0,0} ; j = 0, 1, \dots, L_i . \quad (3)$$

Furthermore, a backoff instance transmits when it is in any of the states  $(i, j, 0)$  where  $j = 0, 1, \dots, L_i$ . Hence, if  $\tau_i$  denotes the transmission probability (i.e. the probability that a backoff instance in priority class  $i$  transmits during a generic slot time, independent on whether the transmission results in a collision or not), it gives:

$$\tau_i = \sum_{j=0}^{L_i} b_{i,j,0} = b_{i,0,0} \frac{1-p_i^{L_i+1}}{1-p_i}. \quad (4)$$

Ways to express  $b_{i,j,0}$  and  $p_i$  in terms of  $\tau_i$  are presented in the following. Hence, a complete description of the system can be found by solving the above set of equations (one equation per AC  $i$ ).

From chain regularities, and by working recursively through the chain from right to left in the upper row, it is seen that:

$$b_{i,0,k,e} = \frac{(1-\rho_i)b_{i,0,0}}{W_{i,0}(1-p_i^*)} \frac{1-(1-q_i^*)^{W_{i,0}-k}}{q_i^*}; \quad k=1,2,\dots,W_{i,0}-1. \quad (5)$$

Furthermore:

$$b_{i,0,0,e} = \frac{(1-\rho_i)b_{i,0,0}}{W_{i,0}q_i} \frac{1-(1-q_i^*)^{W_{i,0}}}{q_i^*} \quad (6)$$

and also:

$$b_{i,0,k} = \frac{W_{i,k}-k}{W_{i,0}(1-p_i^*)} (b_{i,0,0} + q_i p_i b_{i,0,k,e}) - b_{i,0,k,e} \quad (7)$$

for  $k=1,2,\dots,W_{i,0}-1$ .

The same analysis for the rest of the chain, gives:

$$b_{i,j,k} = \frac{W_{i,j}-k}{W_{i,j}(1-p_i^*)} p_i^j b_{i,0,0}; \quad j=1,\dots,L_i, k=1,\dots,W_{i,0}-1. \quad (8)$$

Finally, normalization yields:

$$\frac{1}{b_{i,0,0}} = \sum_{j=0}^{L_i} \left[ 1 + \frac{1}{1-p_i^*} \sum_{k=0}^{W_{i,j}-k} \frac{W_{i,j}-k}{W_{i,j}} \right] p_i^j + \frac{1-\rho_i}{q_i} \frac{1-(1-q_i^*)^{W_{i,0}}}{W_{i,0}q_i^*} \left( 1 + \frac{(W_{i,0}-1)q_i p_i}{2(1-p_i)} \right). \quad (9)$$

The first sum in Eq. (9) represents the saturation-part, while the second part is the dominant term under non-saturation. Hence, the expression provides a unified model encompassing all channel loads from a lightly loaded non-saturated channel, to a highly congested, saturated medium. This full-scale model will be validated in Section VI

By performing the summations in Eq. (9) above and by assuming  $m_i \leq L_i$ , Eq. (4) might be expressed as:

$$\frac{1}{\tau_i} = \frac{(1-2p_i^*)}{2(1-p_i^*)} + \frac{W_{i,0} \left( (1-p_i)(1-(2p_i)^{m_i}) + (1-2p_i)(2p_i)^{m_i} (1-p_i^{L_i-m_i+1}) \right)}{2(1-p_i^*)(1-2p_i)(1-p_i^{L_i+1})} + \left( \frac{1-p_i}{1-p_i^{L_i+1}} \right) \frac{1-\rho_i}{q_i} \frac{1-(1-q_i^*)^{W_{i,0}}}{W_{i,0}q_i^*} \left( 1 + \frac{(W_{i,0}-1)q_i p_i}{2(1-p_i)} \right) \quad (10)$$

Eq. (10) is the key result in the analysis of the model. It represents a set of  $N$  equations that must be solved. They are normally inter-dependent in such a way that they must be

solved numerically. However, there are cases, such as the one presented in [8], where a closed form solution can be found.

#### A. Estimating $p_i$ without Virtual Collision Handling

The probability of unsuccessful transmission,  $p_i$ , from one specific backoff instance is given when at least one of the other backoff instances does transmit in the same slot. Thus,

$$p_i = 1 - \prod_{c=0, c \neq i}^{N-1} (1-\tau_c)^{n_c} = 1 - \frac{1-p_b}{1-\tau_i}, \quad [\text{without VC}], \quad (11)$$

where  $p_b$  denotes the probability that the channel is busy (i.e. at least one backoff instance transmits during a slot time):

$$p_b = 1 - \prod_{i=0}^{N-1} (1-\tau_i)^{n_i}. \quad (12)$$

Furthermore,  $n_i$  denotes the number of backoff instances contending for channel access in each priority class  $i$ , and  $N$  denotes the total number of classes.

Eq. (11) is valid if each QSTA is transmitting traffic of only one AC and there are therefore totally  $\sum_{i=0}^{N-1} n_i$  number of QSTAs transmitting traffic. Hence, no virtual collisions (VCs) will occur between different transmission queues on one QSTA.

#### B. Estimating $p_i$ with Virtual Collision Handling

If each station is transmitting traffic of more than one AC, on the other hand, there *will* be virtual collision handling between the queues. Upon a virtual collision (VC) the higher priority AC will be attempted for transmission while the colliding lower priority traffic goes into backoff.

To illustrate this, consider that each QSTA is transmitting traffic of all  $N$  possible ACs, AC[ $N-1$ ],...,AC[0]. In this paper AC[ $N-1$ ] is by definition of the highest priority (normally equal to the ‘‘AC\_VO’’ of 802.11e) and AC[0] of the lowest (normally equal to the ‘‘AC\_BK’’ of 802.11e). The virtual collision handling implies that a backoff instance can transmit packets if other backoff instances don't transmit, *except* the backoff instances of the lower priority ACs *on the same QSTA*. Hence, instead of Eq. (11),  $p_i$  is now found by:

$$p_i = 1 - \frac{1-p_b}{\prod_{c=0}^{N-1} (1-\tau_c)}, \quad [\text{with VC}] \quad (13)$$

where  $p_b$  is calculated as before [i.e. as in Eq. (12)].

#### B. Estimating $p_i^*$ with Starvation Prediction

The reason for the distinction between  $p_i$  and  $p_i^*$  in the model is that AIFS-differentiation can be modelled with pretty good accuracy by adjusting the countdown blocking probability,  $p_i^*$ .

Lower priority backoff instances of class  $i$  have to suspend additional  $A_i$  slots after each backoff countdown. By assuming these are being smeared out randomly and distributed uniformly over all slots, it is possible to "scale down" the probability of detecting an empty slot, as illustrated in Figure 2.

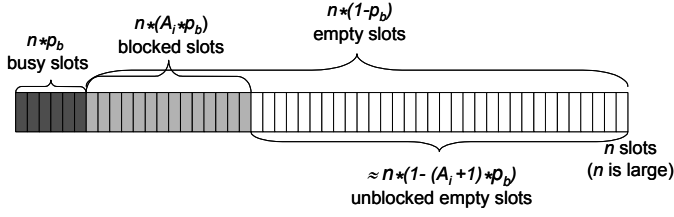


Figure 2. Simplified illustration of the principle of AIFS differentiation.

With this assumption,  $p_i^*$  can be approximated as:

$$p_i^* = \min(1, p_i + \frac{A_i p_b}{1 - \tau_i}) . \quad (14)$$

(The resemblance with Eq. (11) stems from the fact that the countdown blocking is not directly affected by the virtual collisions handling.)

Thus, starvation for AC  $i$  can be roughly predicted to occur when  $p_i^* = 1$  or  $p_b \geq \frac{1}{1 + A_i}$  by Eq (14).

### C. Estimating $\rho_i$

For a G/G/1 queue, the probability that the queue is non-empty,  $\rho$  is given by  $\rho = \lambda \bar{D}$ , where  $\lambda$  represents the traffic rate in terms of packets per second and  $\bar{D}$  is the average service time. In this context  $\bar{D}$  is the frame transmission delay from the time a packet has reached the front of the transmission queue and is the first packet to be transmitted until the packet is successfully transmitted or dropped.

For simplicity, here it is assumed that the traffic rate faced by all backoff instances of a class is the same on all stations, and use  $\lambda_i$  to denote the traffic rate (in terms of packets per seconds) of traffic class  $i$  on one station. Then;

$$\min(1, \lambda_i \bar{D}_i^{NON-SAT}) \leq \rho_i \leq \min(1, \lambda_i \bar{D}_i^{SAT}) , \quad (15)$$

where  $\bar{D}_i^{SAT}$  and  $\bar{D}_i^{NON-SAT}$  are the delay with or without taking into account the post-backoff, respectively. It is correct to include the postback-off under full saturation and in this region  $\bar{D}_i^{SAT}$  provides the best description of the delay. Under perfect non-saturation conditions, on the contrary, it is correct to omit the effects of post-backoff, and here  $\bar{D}_i^{NON-SAT}$  provides the best delay description. The minimum bounds in Eq. (15) ensure that under saturation conditions, when the queue is always full of packets ready to be transmitted, the utilization,  $\rho_i$ , never exceeds 1. It is possible

to use arguments to determine  $\rho_i$  with higher accuracy. Due to space limitations, this is beyond the scope of this paper.

Expressions for  $\bar{D}_i^{SAT}$  and  $\bar{D}_i^{NON-SAT}$  delays will be provided in Section IV.

### D. Estimating $q_i$ and $q_i^*$

To estimate  $q_i$  of the non-saturation model it is assumed that the traffic arriving in the transmission queue is Poisson distributed, i.e. that the system is of the M/G/1 type.  $q_i$  is the probability that at least one packet will arrive in the transmission queue during the following generic time slot under the condition that the queue is empty at the beginning of the slot.

Thus,  $q_i$  is calculated as:

$$q_i = 1 - (p_s e^{-\lambda_i T_e}) + (1 - p_b) e^{-\lambda_i T_e} + (p_b - p_s) e^{-\lambda_i T_e} . \quad (16)$$

We tested a number of different expressions for  $q_i^*$ , and observed that setting  $q_i^*$  equal to  $q_i$  for simplicity worked as a good approximation in all the scenarios explored.

## IV. THROUGHPUT

Although the main focus of this paper is on the delay, the throughput predictions of the model is first presented. The reason is that it introduces some important probability definitions used also for the delay predictions later. Moreover, later in this paper the model is validated in terms of not only the delays, but also the throughput to give a more complete description of the accuracy of the model.

Let  $p_{i,s}$  denote the probability that a packet from any of the  $n_i$  backoff instances of class  $i$  is transmitted successfully (at probability  $\tau_i(1 - p_i)$ ) in a time slot:

$$p_{i,s} = n_i \tau_i (1 - p_i) . \quad (17)$$

where  $p_i$  is determined from Eq. (13) if there are virtual collisions [or Eq. (11) otherwise].

Let also  $p_s$  denote the probability that a packet from any class  $i$  is transmitted successfully in a time slot:

$$p_s = \sum_{i=0}^{N-1} p_{i,s} . \quad (18)$$

Then, the throughput of class  $i$ ,  $S_i$  can be written as the average real-time duration of successfully transmitted packets by the average real-time duration of a contention slot that follows the special time scale of our model:

$$S_i = \frac{p_{i,s} T_{i,MSDU} B}{(1 - p_b) T_e + p_s T_s + (p_b - p_s) T_c} , \quad (19)$$

where  $T_e$ ,  $T_s$  and  $T_c$  denote the real-time duration of an empty slot, of a slot containing a successfully transmitted packet and of a slot containing two or more colliding packets, respectively. The length of the longest colliding packet on the channel determines  $T_c$ . If all packets are of the same length, which will be considered in this paper, then  $T_c = T_s$ . (Otherwise refer to [3] to calculate  $T_c$  based on the average duration of the longest colliding data packet on the channel.) Finally,  $B$  denotes the nominal data bit-rate (e.g. 11 Mbps for 802.11b [9]), and  $T_{i,MSDU}$  denotes the average real-time required transmitting the MSDU part of a data packet at this rate.

## V. DELAY

### A. Z-transform of the Medium Access Delay

We first deal with the delay associated with counting down backoff slots for the packets to be transmitted. While being blocked during countdown, the weighted average delay is

$\frac{P_s}{p_b} T_s + (1 - \frac{P_s}{p_b}) T_c$ , and the corresponding z-transform is:

$$D(z) = \frac{P_s}{p_b} z^{T_s} + (1 - \frac{P_s}{p_b}) z^{T_c}. \quad (20)$$

While the backoff instance is counting down, the probability of facing an empty slot is  $1 - p_i^*$  while the probability of being blocked is  $p_i^*$ . Hence, the z-transform of this blocking delay is:

$$D_{bi}^i(z) = \frac{1 - p_i^*}{1 - p_i^* D(z)}. \quad (21)$$

When it is not blocked anymore, the system will spend an empty time-slot,  $T_e$ , when moving to the next countdown state. Hence, the z-transform of the total delay associated with one countdown state is:

$$D_{state}^i(z) = z^{T_e} D_{bi}^i(z). \quad (22)$$

The total delay in a backoff stage is derived by a geometric sum over the probabilities associated with each countdown state:

$$D_{stage,j}^i(z) = \frac{1}{W_{ij}} \frac{1 - (D_{state}^i(z))^{W_{ij}}}{1 - D_{state}^i(z)}, \quad (23)$$

where the factor  $1/W_{ij}$  reflects the uniform distribution of the selection of the number of backoff slots at each stage.

For simplicity the term  $D_{level,j,s}^i(z)$  is introduced as:

$$D_{level,j,s}^i(z) = \prod_{l=s}^j D_{stage,l}^i(z) \quad (24)$$

Here,  $s$  is set to 0 under saturation conditions, because the post-backoff is undertaken before the transmission of each

packet. Then the transform for the saturation delay may be written as:

$$D_{Sat}^i(z) = (1 - p_i) \sum_{j=0}^{L_i} p_i^j z^{T_s + jT_c^*} D_{level,j,0}^i(z) + p_i^{L_i+1} z^{(L_i+1)T_c^*} D_{level,L_i,0}^i(z). \quad (25)$$

Under extreme non-saturation conditions, on the contrary, the post-backoff is always completed before a new packet arrives in the transmission queue. Thus, under these conditions the post-backoff will not add to the transmission delay, as it did when the saturation delays were calculated above, and  $s$  is now set to 1 in Eq. (24). Then, the transform of the non-saturation delay can be found as:

$$D_{Non-Sat}^i(z) = (1 - p_i) \sum_{j=0}^{L_i} p_i^j z^{T_s + jT_c^*} D_{level,j,1}^i(z) + p_i^{L_i+1} z^{(L_i+1)T_c^*} D_{level,L_i,1}^i(z), \quad (26)$$

where  $D_{level,0,1}^i(z) = 1$  has been defined for convenience.

The first part of Eq. (25) and of Eq. (26) represent the delay associated with packets that are eventually transmitted successfully on the channel, where  $p_i$  is the probability of colliding after each  $j$ -th stage, adding an extra delay of  $T_c^*$  (thus the factor  $z^{T_c^*}$  per stage).  $(1 - p_i)$  is the probability of finally transmitting the packet after a stage, which adds an extra delay of  $T_s$  (thus the factor  $z^{T_s}$ ). The last part of Eq. (25) and of Eq. (26) represent the delay of packets that go through all  $0, \dots, L_i$  stages without being transmitted successfully, and are eventually dropped.

### B. Mean Medium Access Delay

Finally, the mean medium access delay when the post-backoff delay is taken into account,  $\bar{D}_i^{SAT}$ , is found directly from the transform in Eq. (25):

$$\bar{D}_i^{SAT} = D_{Sat}^{i(1)}(1) = (1 - p_i^{L_i+1})(T_s + T_c^* \frac{p_i}{1 - p_i}) + \frac{\bar{D}_i^{state}}{2} R_i^i \quad (27)$$

where  $\bar{D}_i^{state}$  is defined as the mean delay associated by a countdown state:

$$\bar{D}_i^{state} = D_{state}^{i(1)}(1) = T_e + \left[ \frac{P_s}{p_b} T_s + (1 - \frac{P_s}{p_b}) T_c \right] \frac{p_i^*}{(1 - p_i^*)} \quad (28)$$

and the sum  $R_i^i$  is given by:

$$R_i^i = \sum_{j=0}^{L_i} p_i^j (W_{ij} - 1). \quad (29)$$

By performing the summation above in Eq (25) for the case  $m_i \leq L_i$  the following explicit expression for  $R_i^i$  is obtained:

$$R_i^i = W_{i0} \left( \frac{1 - (2p_i)^{m_i+1}}{1 - 2p_i} + 2^{m_i} \frac{p_i^{m_i+1} - p_i^{L_i+1}}{1 - p_i} \right) - \frac{1 - p_i^{L_i+1}}{1 - p_i} . \quad (30)$$

The mean medium access delay when the post-backoff delay is not taken into account,  $\overline{D}_i^{NON-SAT}$ , can be calculated similarly using Eq. (26), or it may alternatively be found by:

$$\begin{aligned} \overline{D}_i^{NON-SAT} &= \left[ \frac{D_{Sat}^i}{D_{Stage,0}^i} \right]^{(1)} (z=1) \\ &= \frac{D_{Sat}^i (1) D_{Stage,0}^i (1) - D_{Sat}^i (1) D_{Stage,0}^i (1)}{D_{Stage,0}^i (1)} \\ &= D_{Sat}^i (1) - D_{Stage,0}^i (1) = \overline{D}_i^{SAT} - \overline{D}_i^{state} \frac{W_{i0} - 1}{2} , \end{aligned} \quad (31)$$

where  $\overline{D}_i^{state}$  is given in Eq.(28). The resolution of  $D_{stage,j}^i (1)$ , shown by the last equality, is found by simple derivation of Eq. (23) and subsequent application of L'Hôpital's rule three times.

### C. Mean Queueing Delay

By considering the medium access delay as the “service time” for a packet in an single server queue we may obtain the mean queueing delay by applying the corresponding formula for the M/G/1 queueing model,  $\overline{\Delta}_i$ ; given through the second order moment of the delay [10]:

$$\overline{\Delta}_i = \frac{\lambda_i \overline{D}_i^2}{2(1 - \rho_i)} . \quad (32)$$

To apply an M/G/1 model for the queueing delay we must also assume that the medium access times are independent stochastic variables. This will not be an exact assumption, however, it is believed that the dependencies will be weak, so that Eq. (32) will provide an accurate approximation.

We will first consider the queueing delay when effects of the post-backoff delay are taken into account, which gives the best description close to saturation conditions. Later in this section we deal with the opposite case, which better describes the non-saturation situation. The second order moment of the delay is found by derivation of the z-transform [10]:

$$\begin{aligned} \overline{D}_i^{SAT} &= \overline{D}_i^{SAT} + D_{Sat}^i (2) (1) = (1 - p_i) \sum_{j=0}^{L_i} (T_s + jT_c)^2 p_i^j + \\ &2(1 - p_i) \sum_{j=0}^{L_i} (T_s + jT_c) p_i^j \overline{D}_{level,j,0}^i + (1 - p_i) \sum_{j=0}^{L_i} p_i^j \overline{D}_{level,j,0}^i{}^2 + \\ &((L_i + 1)T_c)^2 p_i^{L_i+1} + 2(L_i + 1)T_c p_i^{L_i+1} \overline{D}_{level,L_i,0}^i + p_i^{L_i+1} \overline{D}_{level,L_i,0}^i{}^2 \end{aligned} \quad (33)$$

where

$$\overline{D}_{level,j,0}^i = D_{level,j,0}^i (1) = \overline{D}_i^{state} \sum_{l=0}^j \frac{W_{il} - 1}{2} \quad (34)$$

and

$$\begin{aligned} \overline{D}_{level,j,0}^i{}^2 &= D_{level,j,0}^i (2) (1) + D_{level,j,0}^i (1) (1) = \\ &(\overline{D}_i^{state})^2 \left( \left( \sum_{l=0}^j \frac{W_{il} - 1}{2} \right)^2 - \sum_{l=0}^j \left( \frac{W_{il} - 1}{2} \right)^2 + \sum_{l=0}^j \frac{(W_{il} - 1)(W_{il} - 2)}{3} \right) + \\ &\overline{D}_i^{state^2} \sum_{l=0}^j \frac{W_{il} - 1}{2} . \end{aligned} \quad (35)$$

Furthermore,  $\overline{D}_i^{state}$  is given by Eq.(28) and

$$\begin{aligned} \overline{D}_i^{state^2} &= D_{state}^i (2) (1) + D_{state}^i (1) (1) = \\ T_e^2 &+ \left[ \frac{P_s}{P_b} T_s (2T_e + T_s) + (1 - \frac{P_s}{P_b}) T_c (2T_e + T_c) \right] \frac{P_i^*}{1 - P_i^*} + \\ &2 \left[ \frac{P_s}{P_b} T_s + (1 - \frac{P_s}{P_b}) T_c \right]^2 \left( \frac{P_i^*}{1 - P_i^*} \right)^2 . \end{aligned} \quad (36)$$

By performing the summations in the expressions above we obtain:

$$\begin{aligned} \overline{D}_i^{SAT} &= (1 - p_i^{L_i+1}) \left( T_s^2 + T_c^2 \frac{P_i}{1 - p_i} \right) + \\ &2T_c^* (1 - (L_i + 1)p_i^{L_i} + L_i p_i^{L_i+1}) \left( T_s + T_c \frac{P_i}{1 - p_i} \right) \frac{P_i}{1 - p_i} + \\ &\overline{D}_i^{state} \left[ \left( T_s + T_c \frac{P_i}{1 - p_i} \right) (R_1^i - p_i^{L_i+1} R_2^i) + T_c^* R_3^i \right] + \\ &(\overline{D}_i^{state})^2 \left( \frac{R_4^i}{3} + \frac{R_5^i - R_3^i}{2} \right) + \overline{D}_i^{state^2} \frac{R_1^i}{2} , \end{aligned} \quad (37)$$

where the sum  $R_1^i$  is defined by Eq. (28) and the other sums  $R_2^i, \dots, R_5^i$  are defined by:

$$R_2^i = \sum_{j=0}^{L_i} (W_{ij} - 1) , \quad (38)$$

$$R_3^i = \sum_{j=1}^{L_i} j P_i^j (W_{ij} - 1) , \quad (39)$$

$$R_4^i = \sum_{j=0}^{L_i} p_i^j (W_{ij} - 1)(W_{ij} - 2) , \quad (40)$$

$$R_5^i = \sum_{j=1}^{L_i} p_i^j (W_{ij} - 1) \sum_{s=0}^{j-1} W_{is} . \quad (41)$$

Explicit expressions for the sums  $R_2^i, \dots, R_5^i$  in Eq. (38) – Eq. (41) can be found by performing the summation for the case  $m_i \leq L_i$ . These explicit expressions are found in the Appendix of this paper.

One may make the same type conversion between  $\overline{D}_i^{SAT}$  and  $\overline{D}_i^{NON-SAT}$  using a similar approach as in Eq. (31):

$$\overline{D}_i^{NON-SAT} = \overline{D}_i^{SAT} - \overline{D}_{stage,0}^i{}^2 - 2\overline{D}_i^{NON-SAT} \overline{D}_{stage,0}^i . \quad (42)$$

where

$$\overline{D}_{stage,0}^i = D_{stage,0}^i (1) = \overline{D}_i^{state} \frac{W_{i0} - 1}{2} , \quad (43)$$



and

$$\overline{D_{stage,0}^i}^2 = D_{stage,0}^{i(2)} + D_{stage,0}^{i(1)} = \left(\overline{D_i^{state}}\right)^2 \frac{(W_{i,0}-1)(W_{i,0}-2)}{3} + \overline{D_i^{state}^2} \frac{(W_{i,0}-1)}{2}, \quad (44)$$

and where  $\overline{D_i^{state}}$  and  $\overline{D_i^{state}^2}$  are given by Eq.(28) and Eq.(36). Furthermore,  $\overline{D_i^{NON-SAT}}$  and  $\overline{D_i^{SAT}}$  are given by Eq.(31) and Eq.(37).

## VI. VALIDATIONS

### A. Simulation Setup

We compared numerical computations in *Mathematica* with ns-2 simulations, using the TKN implementation of 802.11e [11] for the ns-2 simulator.

The scenario selected for validations is 802.11b with long preamble and without the RTS/CTS-mechanism. The parameter settings for 802.11b are found in [9]. Based on these, the model parameters  $T_e = 20\mu s$ ,  $T_{i,MSDU} = T_{1024} = 520\mu s$  and  $T_S = T_c = 1321\mu s$  were estimated. Finally, setting the time a colliding station has to wait when experiencing collision,  $T_c^*$ , equal to the time a non-colliding station has to wait when observing a collision on the channel,  $T_c$ , corresponds with the ns-2 implementation used for validations.

Parameters such as CWmin and CWmax are overridden by the use of 802.11e [2]. For the validations, the default 802.11e values, also shown in Table 1 in [8], were used.

The node topology of the simulation uses five different stations, QSTAs, contending for channel access. Each QSTA uses all four ACs, and virtual collisions therefore occur. Poisson distributed traffic consisting of 1024-bytes packets was generated at equal amounts to each AC.

The throughput values of our ns-2 simulations were measured over 3 minutes of simulation time. The simulations were started with a 100 seconds transition period to let the system stabilize before the measurements were started.

### B. Validation of the Throughput Predictions

Although the main focus of this paper is on the delay, the throughput predictions of the model is first validated, in order to give a more complete impression of the accuracy of the model that is being used.

Figure 3 compares numerical throughput calculations of the analytical model with the actual simulation results. It is

observed that the model corresponds relatively well with the outcome of the simulations. However, there are some differences that exceed the 95% confidence interval of the simulations. (Since the intervals are so small they have only been shown for 3000 Kbps and 5000 Kbps in Figure 3).

We also see that the starvation of AC[0] and AC[1], experienced with simulations, is described with relatively good accuracy by the analytical model. However, the starvation expression in Eq. (14) seems to be a little too coarse-grained to model the exact throughput behavior when these ACs face starvation. In the semi-saturation-part (middle part) of the figure it is also observed some inaccuracies in the numerical calculations of model. *Mathematica* have difficulties in converging in this region, for example when the traffic generated per AC is around 2500 Kbps.

### C. Validation of the Medium Access Delay Predictions

Even in all the cases where the queueing delay is significantly higher than the medium access delay, the latter is not unimportant. It is the medium access delay that determines whether the service rate of the MAC is able to match the traffic rate that enters the queue. For this reason, the medium access predictions are validated first.

Figure 4 compares numerical mean delay calculations of the analytical model with the actual simulation results. The solid marked with triangles show the numerical results for the mean saturation delay,  $\overline{D_i^{SAT}}$ , i.e. the delay that includes the post-backoff. The dotted curves marked with triangles show the mean non-saturation delay,  $\overline{D_i^{NON-SAT}}$ , i.e. the delay that does not take into account the effects of the post-backoff. Ideally, the mean delay,  $\overline{D_i}$ , of each AC  $i$  (represented by dashed curves marked with 'X's in Figure 4) should lie between the two numerically calculated curves for  $\overline{D_i^{NON-SAT}}$  and  $\overline{D_i^{SAT}}$ :

$$\overline{D_i^{NON-SAT}} \leq \overline{D_i} \leq \overline{D_i^{SAT}}. \quad (45)$$

We observe that this is the case in most parts of the figure. However, the model predicts a delay for the second highest priority AC, AC[2], that is slightly lower than experienced by the simulations around 3000 Kbps. The 95% confidence interval for AC[2] - drawn at 3000 Kbps in Figure 4 - shows that this discrepancy cannot be explained by simple statistical variations. (The 95% confidence intervals are also shown for 5000 Kbps.)

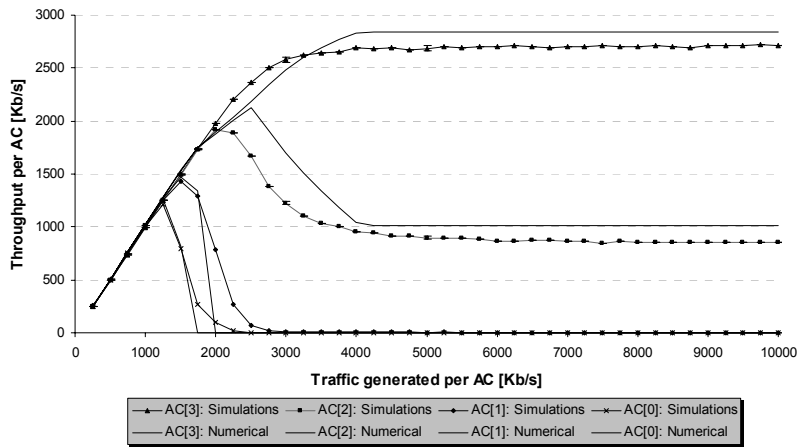


Figure 3. Throughput comparison between analytical (numerical) and simulation results with four ACs per station and varying traffic per AC.

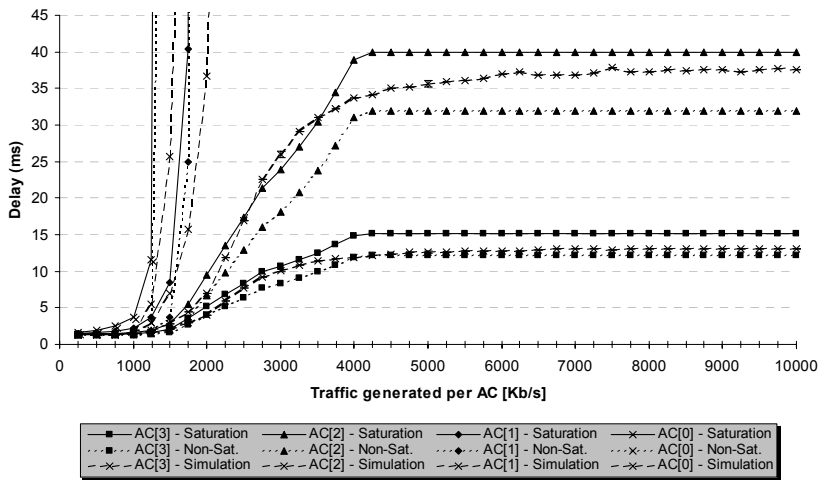


Figure 4. Mean Medium Access Delay comparison between analytical (numerical) and simulation results with four ACs per station and varying traffic per AC. (The “Saturation” curves refer to the delay calculated when the effects of the post-backoff delay are taken into account, while for the “Non-Sat.” curves, these effects are not considered.)

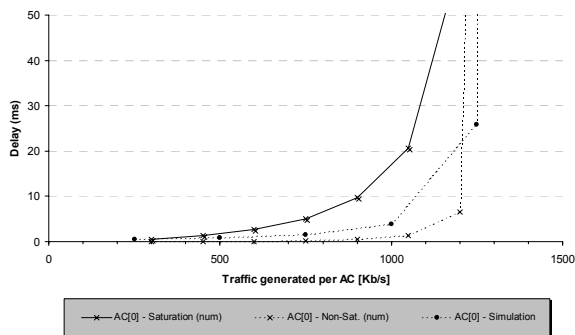


Figure 5. Mean Queueing Delay comparison of AC[0] between analytical (numerical) and simulation results.

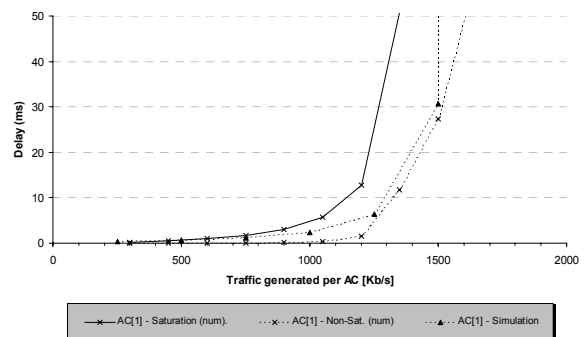


Figure 6. Mean Queueing Delay comparison of AC[1] between analytical (numerical) and simulation results.

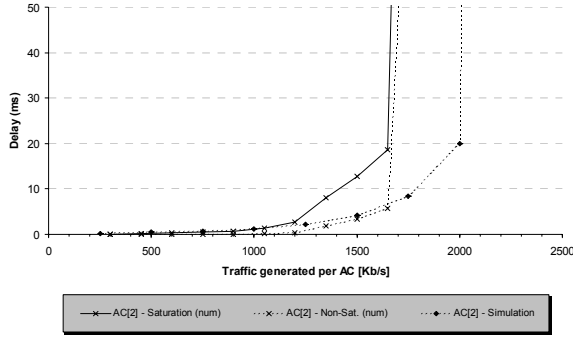


Figure 7. Mean Queueing Delay comparison of AC[2] between analytical (numerical) and simulation results.

Furthermore, the model predicts that the delay for the lowest priority ACs, AC[0] and AC[1], increases to infinity a little faster than observed in the simulations. This result corresponds well with the inaccuracies seen for the throughput in the corresponding region in Figure 3.

#### D. Validation of the Queueing Delay Predictions

The predicted mean queueing delay (numerically calculated) is compared with simulation results for the same 5-station scenario above. As argued for earlier, the mean queueing delay,  $\bar{\Delta}_i$ , should lie between the two numerical predictions,  $\bar{\Delta}_i^{SAT}$  and  $\bar{\Delta}_i^{NON-SAT}$ , depending on whether the effects of the post-backoff delay are taken into account or not:

$$\bar{\Delta}_i^{NON-SAT} \leq \bar{\Delta}_i \leq \bar{\Delta}_i^{SAT}. \quad (46)$$

Figure 5, Figure 6, Figure 7 and Figure 8 show the queueing delay comparisons for AC[0], AC[1], AC[2] and AC[3], respectively. It is observed that the simulation results are largely within the prediction range given by Eq. (46).

However, close to the saturation singularity where queues grow infinitely, it seems that the model is less accurate, and the simulation results are outside this range (except for AC[0] in Figure 6). The most important reason is probably that inaccuracies in the mean delay directly affect the exact location on the abscissa axis (x-axis) where this singularity occurs. This is seen directly from Eq. (32), since in the nominator  $\rho_i$  is determined by  $\rho_i = \text{Max}[1, \lambda_i \bar{D}_i]$ . Hence, the prediction of whether the system has reached the saturation requirement,  $\rho_i = 1$ , or not at a given traffic intensity,  $\lambda_i$ , is fully dependent on the size of  $\bar{D}_i$ . Small inaccuracies in the prediction of  $\bar{D}_i$  can translate into large inaccuracies in the prediction of the exact traffic intensity where the singularity will occur.

## VII. CONCLUSIONS

This paper demonstrates the importance of the queueing delay, and shows how analytical models can predict it. Using a saturation model makes no sense, since the queueing delay is infinite under saturation conditions. Instead, a model that has been extended to cover the full range from a non-saturated to a fully saturated channel is used. Furthermore, a simple way to

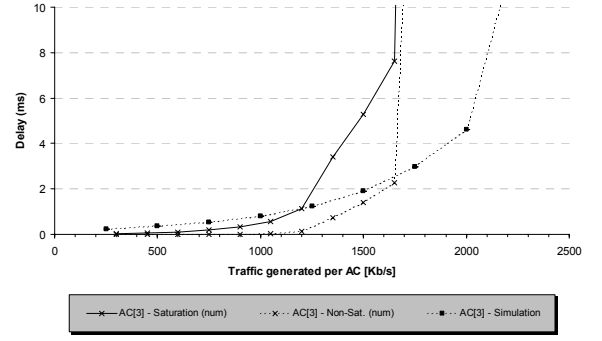


Figure 8. Mean Queueing Delay comparison of AC[3] between analytical (numerical) and simulation results.

introduce AIFS differentiation into the model is proposed. Thus, the default medium access parameters recommended by the 802.11e specification, which uses this kind of differentiation, can be studied.

Earlier works (such as [5]) have mostly focused on mean values for the medium access delay. Here, however, all higher-order moments of the delay are found, through the explicit z-transform of the delay. The average queueing delay can then be predicted by means of the second order moment of the delay transform, as a direct consequence of basic queueing theory. The mean medium access delay and the mean queueing delay together, constitute the average total delay of the MAC, as seen from an upper layer protocol or application.

The mean queueing delay predictions of the model are calculated numerically and validated against simulations. The mean access delay was also validated, since it has a direct impact on when saturation occurs and when queues as a result grow to infinity. To make the analysis complete, validations of the throughput were also presented.

It is observed that the predictions of the mean queueing delay give a relatively good match with simulations. The mean access delay and the throughput were also relatively well predicted by the model.

It is finally pointed out that expressions for the mean queueing delay and mean access delay were found under two extremities of the model, namely whether or not the effects of the post-backoff queuing delay were taken into account (depending on whether one wants to find the delay close to saturation or under non-saturation conditions). In fact, the z-transform was given in terms of these two limits. Thus, the delay predictions presented here say that the real mean delay values must lie somewhere between these limits. The presented model, however, contains parameters (such as  $q_i^*$  shown in Figure 1) that should make it feasible to derive a more exact expression. As a first order approximation, the following could be attempted:

$$\begin{aligned} \bar{D}_i &\approx (1 - \rho_i) \bar{D}_i^{NON-SAT} + \rho_i \bar{D}_i^{SAT}; \\ \bar{\Delta}_i &\approx (1 - \rho_i) \bar{\Delta}_i^{NON-SAT} + \rho_i \bar{\Delta}_i^{SAT}. \end{aligned} \quad (47)$$

These and more exact expressions will be explored in a follow-up paper.

This paper presents the medium access delay distribution through the z-transform. In addition to finding the moments of the delay, the z-transform can be inverted numerically with a configurable error bound. By assuming an M/G/1 queueing model it is possible to obtain a complete delay description, containing the distributions both of the MAC delay, the queueing delay and the total delay. All desirable delay percentiles follow. This follow-up work will be published and presented soon [12].

#### APPENDIX

The explicit expressions for the sums  $R_2^i, \dots, R_5^i$  of Eq. (38) – Eq. (41) can be found by performing the summation for the case  $m_i \leq L_i$ :

$$R_2^i = W_{i0} (2^m (2 + L_i - m_i) - 1) - (L_i + 1) , \quad (48)$$

$$R_3^i = W_{i0} \left( \frac{2p_i(1-(m_i+1)(2p_i)^{m_i} + m_i(2p_i)^{m_i+1})}{(1-2p_i)^2} + \right. \\ \left. 2^m \frac{p_i(m_i+1)p_i^{m_i} - m_i p_i^{m_i+1} - (L_i+1)p_i^{L_i} + L_i p_i^{L_i+1}}{(1-p_i)^2} \right) + \\ \frac{p_i(1-(L_i+1)p_i^{L_i} + L_i p_i^{L_i+1})}{(1-p_i)^2} , \quad (49)$$

$$R_4^i = W_{i0}^2 \left( \frac{1-(4p_i)^{m_i+1}}{1-4p_i} + 4^m \frac{p_i^{m_i+1} - p_i^{L_i+1}}{1-p_i} \right) - \\ 3W_{i0} \left( \frac{1-(2p_i)^{m_i+1}}{1-2p_i} + 2^m \frac{p_i^{m_i+1} - p_i^{L_i+1}}{1-p_i} \right) + 2 \frac{1-p_i^{L_i+1}}{1-p_i} , \quad (50)$$

$$R_5^i = W_{i0}^2 \left( \frac{1-(4p_i)^{m_i+1}}{1-4p_i} - \frac{1-(2p_i)^{m_i+1}}{1-2p_i} \right) \\ - W_{i0} \left( \frac{1-(2p_i)^{m_i+1}}{1-2p_i} - \frac{1-p_i^{m_i+1}}{1-p_i} \right) + \\ W_{i0} (2^m W_{i0} - 1) \left( \frac{2^m p_i (p_i^{m_i} - (L_i - m_i + 1) p_i^{L_i} + (L_i - m_i) p_i^{L_i+1})}{(1-p_i)^2} + \right. \\ \left. (2^m - 1) \frac{p_i^{m_i+1} - p_i^{L_i+1}}{1-p_i} \right) . \quad (51)$$

#### ACKNOWLEDGMENT

We would like to thank Bjørn Selvig for help with the development of the simulation tool used for the validations.

#### REFERENCES

- [1] IEEE 802.11 WG, "Part 11: Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) specification", IEEE 1999.
- [2] IEEE 802.11 WG, "Draft Supplement to Part 11: Wireless Medium Access Control (MAC) and physical layer (PHY) specifications: Medium Access Control (MAC) Enhancements for Quality of Service (QoS)", IEEE 802.11e/D13.0, Jan. 2005.
- [3] Bianchi, G., "Performance Analysis of the IEEE 802.11 Distributed Coordination Function", IEEE J-SAC Vol. 18 N. 3, Mar. 2000, pp. 535-547.
- [4] Ziouva, E. and Antonakopoulos, T., "CSMA/CA performance under high traffic conditions: throughput and delay analysis", Computer Communications, vol. 25, pp. 313-321, Feb. 2002.
- [5] Xiao, Y., "Performance analysis of IEEE 802.11e EDCF under saturation conditions", Proceedings of ICC, Paris, France, June 2004.
- [6] Malone, D.W., Duffy, K. and Leith, D.J., "Modelling the 802.11 Distributed Coordination Function with Heterogeneous Load", Proceedings of Rawnet 2005, Riva Del Garda, Italy, April 2005.
- [7] Barkowski, Y., Biaz, S. and Agrawal P., "Towards the Performance Analysis of IEEE 802.11 in multihop ad hoc networks", Proceedings of MobiCom 2004, Philadelphia, PA, USA, Sept.-Oct. 2004.
- [8] Engelstad, P.E., Østerbø O.N., "Differentiation of the Downlink 802.11e Traffic in the Virtual Collision Handler", Proceedings of the Fifth International IEEE Workshop on Wireless Local Networks (WLN '05), Sydney, Australia, Nov. 15-17, 2005.
- [9] IEEE 802.11b WG, "Part 11: Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) specification: High-speed Physical Layer Extension in the 2.4 GHz Band, Supplement to IEEE 802.11 Standard", IEEE, Sep. 1999.
- [10] Kleinrock, L., "Queueing Systems, Vol. 1", John Wiley, 1975.
- [11] Wietholter, S. and Hoene, C., "Design and verification of an IEEE 802.11e EDCF simulation model in ns-2.26", Technische Universität Berlin, Tech. Rep. TKN-03-019, November 2003.
- [12] Engelstad, P.E., Østerbø O.N., "The Delay Distribution of IEEE 802.11e EDCA ", (Currently under review). The paper will be available at <http://www.unik.no/~paalee/research.htm>.