Bottleneck Analysis for Two-Hop IEEE 802.11e Ad Hoc Networks

Anne Remke, Boudewijn R. Haverkort, Geert Heijenk, and Lucia Cloth

Design and Analysis of Communication Systems
University of Twente
{anne,brh,heijenk,lucia}@cs.utwente.nl

Abstract. Recently, a quality-of-service (QoS) extension of the IEEE 802.11 standard (known as IEEE 802.11e) for wireless LANs has been proposed. We present a versatile and accurate performance model to study how these new QoS enhancements can be used to improve the performance of wireless nodes competing for bandwidth in a multi-hop ad hoc network. The paper presents the QoS enhancements, and shows how they can be modeled using a simple, yet effective, parameterized quasi-birth-death model. The model is developed hierarchically, in that results at packet level (e.g., as developed by Bianchi and others) are used in our flow-level model, in which a single bottleneck station interacts with a time-varying number of traffic sources. Thus, we are able to study the impact of the QoS enhancements on the flow-level performance. This has not been done before. The results of our analyses are compared with extensive simulations (using Opnet), and show excellent agreement for throughput, mean number of active sources and mean buffer occupancy at the bottleneck station. An important asset of our model is that it allows for very quick evaluations: where simulations require up to an hour per scenario, our model is solved in seconds.

1 Introduction

The availability of cheap yet powerful wireless access technology, most notably IEEE 802.11 ("wireless LAN"), has given an impulse to the development of wireless ad hoc networks. In such networks, the stations (nodes) that are in reach of each other, help each other in obtaining and maintaining connectivity. At the same time they are also competitors, as they all contend for the same resource, i.e., the shared ether as transmission medium. The medium access control of IEEE 802.11 (based on CSMA/CA) is commonly referred to as the distributed coordination function (DCF) [6,14]. Research has shown that, effectively, the DCF tends to equally share the capacity among contending stations [1,8]. Although this appears to be a nice fairness property, this fairness does lead to undesirable situations in case one of the nodes happens to function as a bridge toward either another group of nodes, or to the fixed internet, as illustrated in Figure 1.

Recently, a quality-of-service (QoS)-extension of the IEEE 802.11 standard, the so-called EDCA ("e") version has been released [5]. Roughly speaking, this

K. Al-Begain, A. Heindl, and M. Telek (Eds.): ASMTA 2008, LNCS 5055, pp. 279–294, 2008.

extension provides mechanisms to provide preferential treatment of certain traffic classes (or nodes) over others. In this paper, we study these extensions in detail, in a way not done before, as follows.

In earlier work [12], we provide a modeling framework for evaluating capacity sharing strategies, using infinite-state Markov reward models and model checking techniques; this paper was on the numerical algorithms and model checking techniques themselves, and illustrated the approach using a number of generic capacity sharing strategies, not at all related to the IEEE 802.11e standard.

In the current paper, we specialize this framework towards the IEEE 802.11e protocol, including the differentiation parameters. We embed Bianchi's model and Engelstad's extensions [1,3] into our model, to accurately describe the effective capacity, depending on the differentiation parameters in use. Both, the specialization of the modeling framework and the embedding of Bianchi's model has never been done before. Furthermore, we conduct detailed simulations that show strong evidence of the correctness of the full IEEE 802.11e model and the employed techniques.

Important to stress is that our model is a flow-level model, with makes it fundamentally different from packet-level models such as [1,3], which have been proposed to compute the share of bandwidth (radio capacity) allocated to a fixed number of sources and a bottleneck node (for various, but not all QoS enhancements). In this paper, we use the packet-level results from [1,3] (as well as extend them) in a (higher-level) flow-level model in which the number of active sources varies in time, depending on how quickly they are being served.

Earlier work on the performance of IEEE 802.11 ad hoc networks considers a variety of scenarios, see, for instance, [15] and the references therein. Also, [8] provides an extensive simulation study of the EDCA standard, investigating the impact of the various QoS mechanism on the performance for single-hop networks. However, these studies do not explicitly address the delays or throughputs in a *multi-hop* ad hoc network. The only paper we are aware of that explicitly addresses the multi-hop case (that is, the two-hop case we also address here) is [15], by using results for a generalized processor sharing model, as studied by Cohen [2]. However, that approach is limited in that it only allows for an equal sharing of transmission capacity between all active stations (including the bottleneck), hence, it does not address the new QoS-enhancements of the IEEE 802.11e standard.

In summary, the contributions of this paper are the following. The paper presents (i) an extension of the models of Bianchi and Engelstad [1,3] to deal with the $\text{TXOP}_{\text{limit}}$ QoS-enhancement; (ii) the embedding of these extended models into our modeling framework (iii) the numerical evaluation of this combined embedded model, and (iv) a comprehensive comparison with detailed (packet-level) simulations using Opnet [10], showing very favorable results. Moreover the analytical techniques are much faster than simulation.

This paper is further organized as follows. We start with a description of the IEEE 802.11 access mechanism and discuss the four QoS extensions in Section 2. A description of the generic model is given in Section 3. We then describe

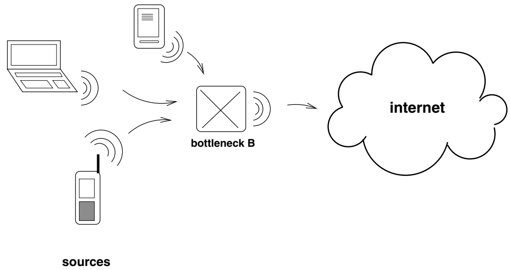

Fig. 1. Bottleneck in a two-hop ad hoc network

precisely how the four IEEE 802.11e QoS-extensions are cast into this model in Section 4. In Section 5 the detailed OPNET simulation setup is described, before we come to a careful comparison of our model's results with the simulation results in Section 6. Section 7 concludes the paper.

2 IEEE 802.11 Ad Hoc Networks

We address a wireless ad hoc network in which the individual nodes communicate with each other through the IEEE 802.11 protocol. In such a network the DCF organizes medium access through a carrier sense multiple access scheme with collision avoidance (CSMA/CA). All stations in such a network contend for the same radio capacity C (measured in packets per second). Whenever a station wants to send a packet, it first senses the medium until the medium is empty for at least a DIFS period (DIFS: DCF inter frame spacing). If the medium is initially found empty, a station that wants medium access immediately starts transmitting after sensing the medium idle for a DIFS period. If the medium is found busy, all stations that want medium access participate in the contention mechanism. After the medium has been idle for at least a DIFS period, each stations draws a random backoff time b. Each station then waits for its chosen backoff time and keeps sensing the medium. If the medium is still idle after DIFS+b time slots, the station may access the medium. As a result, the station with the smallest backoff acquires medium access, if the minimum backoff is unique.

We assume, that whenever two stations draw the same minimum backoff, a collision occurs. Note that in reality this depends on the relative signal strengths at the intended receiver. When the medium is sensed busy during the backoff period, the backoff is suspended and the station continues counting down the backoff after waiting the DIFS period from the moment the medium is sensed idle again. The random backoff is drawn uniformly from the so-called *contention window* (CW), initially set to $[0, \mathrm{CW}_{\min} - 1]$. For up to $r_{\max} - 1$ collisions, the size of the contention window doubles with every unsuccessful transmission and is reset to $\mathrm{CW}_{\min}$ after a successful transmission. Once the contention window has reached size $\mathrm{CW}_{\max} = 2^{r_{\max}} \cdot \mathrm{CW}_{\min}$ it stays unchanged until a successful transmission occurs. Note that in the standard IEEE 802.11 protocol, the values $\mathrm{CW}_{\min}$ and $\mathrm{CW}_{\max}$ are fixed.

The scenario under study, as illustrated in Figure 1, has a varying number N of active nodes, the so-called sources, which are all within reach of each other. Additionally, there is one special node B, referred to as bridge or bottleneck, reachable by all sources. B is the only node that can reach the fixed internet. Thus, all traffic originating from the sources and the traffic passing through the bridge has to share the same wireless transmission capacity. It has been shown that the DCF access mechanism effectively shares the radio capacity equally over all active nodes (a result of the fairness of the access mechanism itself [1,8]). Clearly, this situation benefits the sources as a group, as they can use a relatively large share of radio capacity to send their packets, whereas the bridge becomes a bottleneck: B gets a share as any other individual node, however, it has to support the traffic of *all* other nodes. This leads to a very high buffer occupancy in B, eventually also buffer overflow, and in any case, long delays.

The Enhanced Distributed Channel Access Function (EDCAF) of IEEE 802.11e allows multiple contention instances to be simultaneously active in a single station, each supporting a certain access category (AC). Furthermore, the standard introduces four differentiation parameters (EDCA parameters), as discussed below, which can be set individually for each access category of each individual station to enable QoS provisioning [13].

We facilitate adaptive capacity sharing between stations by letting each station have a single access category, and using the EDCA parameters for differentiating between the source stations and the bottleneck station. In principle the EDCA parameters are meant for service differentiation, while we apply it here for node differentiation. Another relevant scenario for such node level differentiation is the case of UL/DL transfer in an infrastructure-based WLAN, where the access point should get a bigger share of the resources. The considered values for the differentiation parameters per access category are introduced later on in Table 4.

In the remainder of this paper we will analyze the following four scenarios:

0. With standard IEEE 802.11, the medium needs to be idle for at least a DIFS period before stations can start to content for medium access. After winning contention a station is allowed to send exactly one packet.

In the IEEE 802.11e QoS extension, two contention-based methods are proposed to change the above procedure:

1. The initial value of the *contention window* ($\mathrm{CW}_{\min}-1$) and/or the maximum value of the contention window ($\mathrm{CW}_{\max}-1$) are set smaller for a given station, thus, this station draws its backoff from a smaller contention window, hence, has a higher probability to win contention.
2. With so-called *arbitration inter-frame spacing* (AIFS) it is possible to assign different inter-frame spacings for different service classes (or nodes) instead of the fixed DIFS. Thus, high-priority nodes can be assigned shorter AIFS, so that they can start counting off their backoff earlier, hence, have an advantage when contending for medium access.

A way to adapt the capacity sharing that does not alter the actual contention mechanism is the following:

3. The transmission opportunity limit ($TXOP_{limit}$) provides a time period during which a station may send packets after having won a contention. Thus, a station with a sufficiently high $TXOP_{limit}$ is able to send several packets and will thus be able to grab a larger share of the channel capacity than a station with a smaller $TXOP_{limit}$.

The above four parameters (CW_{min} and CW_{max}, AIFS and $TXOP_{limit}$) in the IEEE 802.11e standard can be used to reallocate the amount of radio capacity given to the sources and to the bottleneck. These three possibilities will be addressed in detail in the models we discuss next.

3 Overall Capacity Sharing Model

We model the bottleneck B, cf. Figure 1, using an infinite-state stochastic Petri net (iSPN), as given in Figure 2. The left part of this figure contains an unbounded place (double circle) *buffer* that models the (buffer of the) bottleneck of the system. Transition *input* models the total arrival stream of packets from all active sources, whereas transition *output* models the transmission of packets leaving the bottleneck B. Note that the rates of both these transitions depend on the number of active sources and the amount of radio capacity that is allocated to each source and the bottleneck node; we come back on the form this dependency takes below. We limit the maximum number of active sources to some finite number K (taken to be 10 in most evaluations). This is a reasonable restriction, as the number of active sources in an ad hoc network cannot be arbitrarily high. We do not distinguish between individual active sources, so we can model the number of active sources as shown in the right part of the iSPN in Figure 2. To obtain a memoryless behavior needed for Markovian modeling, an inactive source becomes active after a negative exponentially distributed amount of time (with mean $1/\lambda$)and immediately instantiates a flow, which has a geometrically distributed length, measured in packets. The average size of a data packet is assumed to be $E[P] = 1500$ bytes, with exponentially distributed packet length. The duration of a flow does not only depend on its size but also on the radio capacity a source can use to transmit the flow. Note that the duration of a flow implicitly gives the source departure rate, as well.

Following the parametric assumptions made in [15], the expected amount of work put forward per flow (the amount of packets comprising the flow) equals $E[F] = 500$ packets; the other values for the key system parameters are summarized in Table 1. In Table 2 we list the four state-dependent transition rates of the iSPN, where N refers to the current number of active sources (i.e., the number of tokens in place *active sources*), and B to the current number of packets queued in the bottleneck (i.e., the number of tokens in place *buffer*). Note that the transitions *input* and *output* in fact make use of the same medium, hence, they have to share the available capacity; this is exactly what the IEEE

Table 1. Values for the system parameters

parameter	
arrival rate	$\lambda \in [0.1, 0.4]\text{sec}^{-1}$
average flow size	$E[F] = 500$ packets
overall radio capacity	$C = 917$ packets/sec
maximum of active sources	$K = 10$

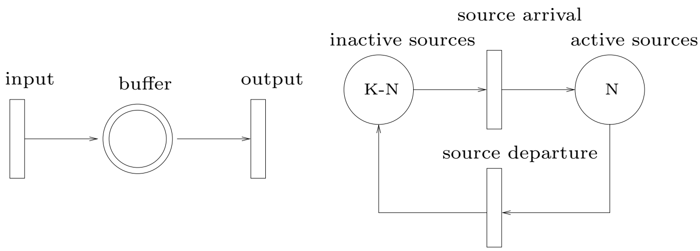

Fig. 2. High-level model as iSPN

802.11[e] access mechanism is for! The functions $S_b(\cdot)$ and $S_s(\cdot)$ (for bottleneck and source) now give the share of capacity that is allocated to the bottleneck and to all sources, respectively. Note that $S_b(\cdot)$ and $S_s(\cdot)$ depend on the number of currently active sources (N), as well as on whether or not the bottleneck has packets queued, or not ($B > 0$).

In Section 4, we will present concrete expressions for the functions $S_b(\cdot)$ and $S_s(\cdot)$; in doing so, we have achieved one generic model at the iSPN level, that can be specialized toward different QoS enhancements, by "plugging in" the appropriate bandwidth sharing functions $S_b(\cdot)$ and $S_s(\cdot)$.

Table 2. State-dependent transition rates for the iSPN

input:	**output:**
if $N = 0$ **then**	**if** $B = 0$ **then**
return 0;	**return** 0;
else	**else**
return $C \cdot S_s(\cdot)$;	**return** $C \cdot S_b(\cdot)$;
end if	**end if**

source departure:
return $C \cdot S_s(\cdot)/E[F]$

source arrival:
return $(K - N)\lambda$

Note that, in practice, the effective capacity C is not fixed, but depends on adaptive modulation, which tunes sending rates to experience link qualities.

4 Modeling the QoS Enhancements

In this section we present explicit expressions for the functions $S_s(\cdot)$ and $S_b(\cdot)$ that express the share of the wireless capacity that sources and the bottleneck receive, resp., for each of the QoS enhancements. We use the model of Bianchi [1], plus some extensions, to compute these shares.

4.1 Bianchi's Model

Bianchi [1] proposes an analytical evaluation of the saturation throughput, assuming ideal channel conditions for basic IEEE 802.11. This model allows us to accurately compute the throughput for a fixed number of stations under the assumption that they always have a packet to send, as follows. From a single station that is modeled as discrete-time Markov model, two mutually dependent stationary probabilities are obtained: the probability τ that the station transmits a packet in a generic slot time, and the probability p that a transmitted packet encounters a collision. These probabilities are expressed in the form of a system of two non-linear equations [1, Eqn. (7) and (9)].

In a generic slot time three different events can occur: the successful transmission of a packet, a collision, or just an empty slot used for counting down backoff. Thus, the length of a generic slot time depends on the event that occurs. Considering the different durations and probabilities of these events, it is possible to compute the throughput of the system.

4.2 Engelstad's Extended Model

In a recent paper [3], Engelstad extends the model of Bianchi by including the impact of the QoS enhancements on the effectively available capacity in IEEE 802.11e. Furthermore, this extended model allows to compute the throughput for stations from different access categories, also in the non-saturated case, i.e., when the stations not necessarily always have a next packet to send.

In our de-compositional analysis approach, however, we still use the saturated case. An active source sends, on average, 500 packets in a row before becoming inactive; this means that, on average, with probability $\frac{499}{500}$ there is a next packet to be sent. Due to the decomposition, inactive sources are not considered within Engelstad's model. This might be seen as an approximation, but its impact will be small, as we will see later. For our purposes, we use two different access categories, AC_i with $i = b$ for the bottleneck, and $i = s$ for the active sources, where AC_b contains zero or one station, and AC_s contains zero to ten stations.

Engelstad now proceeds to compute similar probabilities as Bianchi does, however, now for each possible access category, i.e., the following two probabilities are computed: $\tau(i)$, the probability that a station of category i transmits, and $p(i)$, the probability that a transmission of category i is successful. As in the Bianchi model, these probabilities are defined through systems of mutually dependent non-linear Equations [3, Eqn. (5) and (12)], which can easily be solved using a tool like Maple. Note that we have to solve these equations once for every combination of number of active nodes in each access category, that is, AC_b and AC_s, to obtain throughputs for every possible combination of active nodes in the high-level model. From the above probabilities $\tau(i)$ and $p(i)$, Engelstad then derives yet another three probabilities: the probability $p_{i,s}$ for a successful transmission for category i (cf. [3, Eqn. (27)]); the probability p_s for successful transmission for any category (cf. [3, Eqn. (28)]); and, finally, the probability p_b that the medium is busy during a generic time slot (cf. [3, Eqn. (11)]).

4.3 Relative throughputs

Using the probabilities $p_{i,s}$, p_s, and p_b obtained from the model of [3], we can derive the actual throughput for each access category, i.e, the throughput of the bottleneck, and the throughput of all active stations, as follows:

$$S_i = \frac{p_{i,s} \cdot \text{txop}_i \cdot t_{\text{data}}}{(1-p_b) \cdot t_{\text{slot}} + p_s \cdot T_s + (p_b - p_s) \cdot T_c}, \ i \in \{b, s\}, \tag{1}$$

where the denominator states the average duration of a generic time slot, being the sum of the times for an empty slot (t_{slot}), the time for a successful transmission (T_s), and the time for a collision (T_c), weighted by their respective probability. The nominator denotes the part of a time slot that is, on average, used for transmission of data for category i. Here, txop_i denotes the number of packets of length t_{data} sent after winning contention. Instead of modeling $\text{TXOP}_{\text{limit}}$ as a time period, we assume that for a given packet size, a station of category i can send txop_i packets during one TXOP period[1]: Note that Equation (1) provides the input for the state-dependent transition rates in the iSPN model, cf. Table 2.

To actually compute the values, Table 3 specifies the individual durations and the default values for the system parameters that are used to compute the time for a successful transmission and the time for a collision, respectively. Considering the use of Request To Send/Clear To Send (RTS/CTS), the time for a collision is given by

$$T_c = t_{\text{PHY}} + t_{\text{RTS}} + t_{\text{AIFSmin}}. \tag{2}$$

The time for the successful transmission of a packet with RTS/CTS depends on the TXOP values:

$$\begin{aligned} T_s = \ & t_{\text{PHY}} + t_{\text{RTS}} + t_{\text{SIFS}} + t_{\text{PHY}} + t_{\text{CTS}} + \\ & (t_{\text{SIFS}} + t_{\text{PHY}} + t_{\text{MAC}} + t_{\text{data}} + t_{\text{SIFS}} + t_{\text{PHY}} + t_{\text{ACK}}) \cdot \overline{\text{txop}} + t_{\text{AIFSmin}}, \end{aligned} \tag{3}$$

where $\overline{\text{txop}}$ is the average number of packets sent after a successful contention computed as weighted sum:

$$\overline{\text{txop}} = \sum_{i \in \{s,b\}} \text{txop}_i \cdot \frac{p_{i,s}}{p_s}. \tag{4}$$

[1] That is, we model a time-based mechanism by means of a count-based mechanism; a similar modeling approach has been applied successfully, for instance, by Groenendijk [4], to model time-token access mechanisms.

Table 3. Time durations & default values

parameter	value	comments	**parameter**	value	comment
t_{SIFS}	$10\mu s$		t_{slot}	$20\mu s$	
t_{PHY}	$192\mu s$	assuming long preamble	t_{RTS}	$160\mu s$	20 bytes @ 1 Mbps
t_{CTS}	$112\mu s$	14 bytes @ 1 Mbps	t_{MAC}	$25\mu s$	34 bytes @ 11 Mbps
t_{data}	$1091\mu s$	1500 bytes @ 11 Mbps	t_{ACK}	$112\mu s$	14 bytes @ 1 Mbps
t_{AIFSmin}	$50\mu s$	$2 \cdot t_{\mathrm{slot}} + t_{\mathrm{SIFS}}$			

The different values of $\mathrm{CW}_{\mathrm{min}}$, $\mathrm{CW}_{\mathrm{max}}$ are immediately taken into account, when computing $\tau(i)$ and $p(i)$, as described in [3]. When modeling access categories with different AIFS, we use the approximation as proposed in [3, Section 3.3] to compute the throughput. This implies that the minimum AIFS over all access categories is used for computing the time for a successful transmission (cf. Eqn. (3)) and the time for a collision (cf. Eqn. (2)). The remaining slots, which stations of lower categories have to wait before starting backoff countdown, are modeled as being distributed uniformly over all slots. This involves rescaling the probability that the chanel is busy, p_b. The differentiation of the AIFS is then taken into account when computing $\tau(i)$ and $p(i)$. Modeling differentiation by means of $\mathrm{TXOP}_{\mathrm{limit}}$ is also incorporated in Equation (1), which is an extension of the throughput as presented in [3].

Table 4 specifies the values for the differentiation parameters for the bottleneck and the sources that are considered in the following. The default values are marked with an asterisk.

Table 4. Considered values for the differentiation parameters per access category

parameter	value for AC_b	value for AC_s
AIFS_b (slots)	2*	2*, 7 or 12
$\mathrm{CW}_{\mathrm{min}}$ (slots)	32*	32*, 64 or 128
r_{max}	4*	4*
txop (packets)	1*, 2 or 3	1*

4.4 Measures of Interest

The bridge B is the bottleneck of the two-hop ad hoc network. We therefore study the expected number of packets in the buffer of the bottleneck (M1), as well as the throughput of the bottleneck (M2), and the expected number of active sources (M3), for all possible QoS mechanisms and for different source arrival rates. Instead of specifying the performance models of interest manually at the state level, we instead use a high-level specification mechanism. Given the generic iSPN and the expressions for the throughputs, a generation algorithm is used to automatically derive the underlying *Quasi Birth Death* (QBD) process [9,7,11].

Note, that we are not able to compute flow-related measures, as the flow time or flow throughput, as our model does not distinguish between packets of the different sources. However, we think that the throughput at the bottleneck and, possibly, the throughput at the sources is enough to capture the effects of differentiation in this scenario.

5 Simulation Model

The bottleneck scenario has also been modeled and simulated in Opnet, version 11.5. [10]. In this section we explain the simulation setup and the parameter settings for the different QoS parameters.

Opnet is organized hierarchically in levels, where every level adds more detail. At the highest level, we place ten *advanced WLAN stations*, as provided by Opnet, to model the ten sources, and two more to model the bottleneck and the sink, where all the data is sent to. We then adapt the WLAN stations in the node editor once for the sources and once for the bottleneck. For the sources, a new traffic generation process is created that is put inside each source. The traffic generation process can be either active or inactive. As soon as it becomes active it places a geometrically distributed amount of packets into its MAC layer queue. When the transmitter has been able to send all these packets to the bottleneck it gets inactive again. The packet size is set to 1500 bytes. As in the analytical model all sources are independent and becomes active with the global arrival rate λ, when currently inactive.

The WLAN station that models the bottleneck does not need a traffic generation process. Arriving packets from the MAC layer are immediately routed back to the MAC layer and are forwarded to the sink. The size of the data buffer in the MAC layer is set to the highest possible value, 10^8, to match the assumption of the infinite buffer in the analytical model as accurately as possible. Once the buffer limit is reached, data packets will be discarded until the buffer has free space to store new packets. Note that the complete access mechanism, in all its details, is being included in the Opnet simulation. No approximations or further assumptions are being done. The *wireless LAN parameters* in Opnet are set as follows:

– the data rate is set to 11 Mbps,
– regular RTS/CTS is enabled, the RTS threshold is set to 256 bytes,
– EDCA parameters are not supported for basic IEEE 802.11 and supported for IEEE 802.11e and set according to the simulation scenario under study.

The content of the MAC layer queue is sampled over time and its time average corresponds to the measure *expected buffer occupancy* (M1) in the analytical model. The number of currently active sources is also sampled and its time average corresponds to *expected number of active sources* (M3) in the analytical model. Furthermore, the throughput of the bottleneck is sampled over time and its time average is compared to the *throughput* (M3) of the bottleneck in the analytical model. For every QoS parameter we consider seven different loads:

$\lambda \in \{0.01; 0.015; 0.02; 0.025; 0.03; 0.035; 0.04\}$. Every value of λ is simulated in every scenario with ten randomly chosen seeds for two hours, leading to 70 simulation runs per curve. One simulation run takes between 20 and 50 minutes, resulting in an estimated run time per point, including confidence intervals, of 200 to 500 minutes. In the following we show the mean of the simulation results for ten seeds together with the corresponding 95% confidence interval.

6 Comparing Analytical and Simulation Results

In the following sections we compare the analytical and simulation results for the four different scenarios, followed by a discussion of the throughput that is achieved in the bottleneck, depending on the differentiation parameters.

6.1 Basic Scenario

In the basic 802.11 scenario the EDCA parameters are set to the default values, as given in Table 4. Figure 3 shows the expected buffer occupancy in IEEE 802.11 without differentiation and Figure 4 shows the expected number of active sources. In the basic scenario the results from simulation and analysis are very close to each other, the estimated expected buffer occupancy is always inside the confidence intervals. With increasing λ the variance of the simulated buffer occupancy grows as seen from the larger confidence intervals. As could be expected,

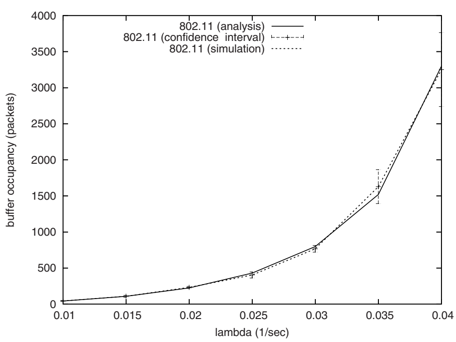

Fig. 3. Mean buffer occupancy for IEEE 802.11 at bottleneck

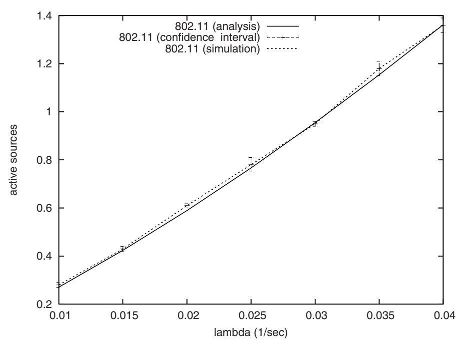

Fig. 4. Mean number of active sources for IEEE 802.11

the buffer occupancy increases exponentially with λ. For λ greater than 0.04 the system soon becomes overloaded. The steady state distribution then does not exist any more and the buffer of the bottleneck in the simulation overflows. The number of active sources grows almost linearly with increasing λ.

6.2 Differentiating with $\mathrm{CW_{min}}$

We compare the results from simulation and analysis for two different settings for the expected buffer occupancy in Figure 5 and in Figure 6 for the expected number of active sources. In this scenario, $\mathrm{CW_{min}}$ is extended in the sources to $\mathrm{CW_{min}} = 64$ and to $\mathrm{CW_{min}} = 128$, whereas the other EDCA parameters are set to the default values as specified in Table 4. Compared to basic IEEE 802.11 the mean buffer occupancy is lower when the sources operate with a larger window size. This is due to the fact that the bottleneck gets a higher capacity share. On the other hand the sources remain active longer (expected number of sources is higher than in the basic scenario).

We observe that in the parameter setting $\mathrm{CW}_{\mathrm{min},s} = 64$, simulation and analysis results are close for buffer occupancy and number of active sources, respectively. For the parameter setting $\mathrm{CW}_{\mathrm{min},s} = 128$, the analysis overestimates the capacity that is allocated to the bottleneck. Hence, the mean buffer occupancy is underestimated and the mean number of active sources is overestimated by the analysis.

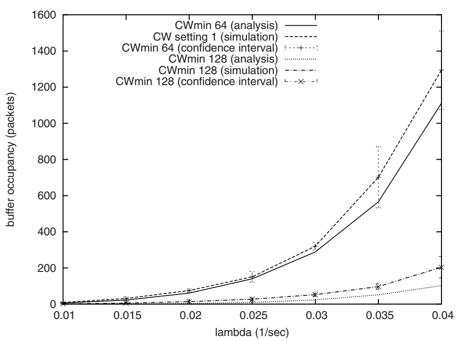

Fig. 5. Mean buffer occupancy for different $\mathrm{CW}_{\mathrm{min},s}$ and $\mathrm{CW}_{\mathrm{min},b} = 32$

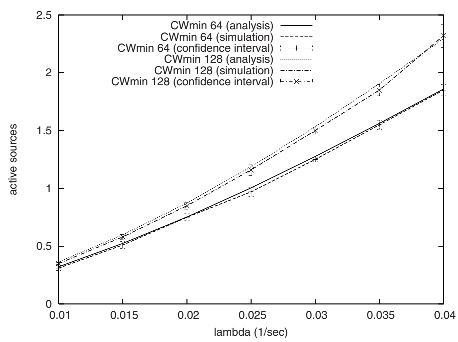

Fig. 6. Mean number of active sources for different $\mathrm{CW}_{\mathrm{min},s}$ and $\mathrm{CW}_{\mathrm{min},b} = 32$

6.3 Differentiating with AIFS

In this scenario the value of AIFS is changed first to 7 and then to 12 in the sources, while the other EDCA parameters remain set as in the basic scenario. The analytical results and the simulation results for the two different AIFS settings are compared in Figure 8 for the expected number active sources. Simulation shows that our results are highly accurate.

Figure 7 shows the expected buffer occupancy at the bottleneck on a logarithmic scale. The mean buffer occupancy is much lower than in the basic scenario. It is also lower than in the CW scenario, while the mean number of active source is comparable. This indicates that in the CW scenario more capacity is wasted due to longer backoff times.

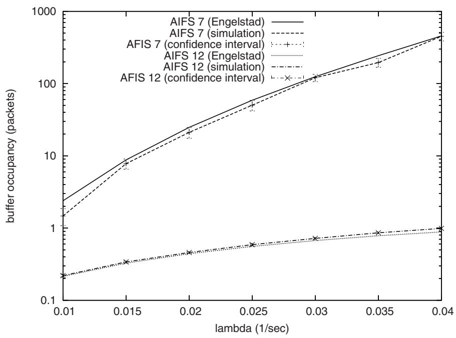

Fig. 7. Mean buffer occupancy for different AIFS_s and $\text{AIFS}_b = 2$

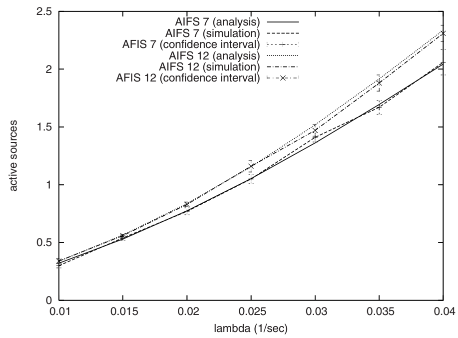

Fig. 8. Mean number of active sources for different AIFS_s and $\text{AIFS}_b = 2$

6.4 Differentiating with $\text{TXOP}_{\text{limit}}$

To study the influence of $\text{TXOP}_{\text{limit}}$ this value is set to 4000 μs, and to 6000 μs in the bottleneck only, while the other EDCA parameters remain unchanged. A $\text{TXOP}_{\text{limit}}$ of 4000 microseconds allows two data packets to be sent, whereas a $\text{TXOP}_{\text{limit}}$ of 6000 microseconds allows three data packets to be sent. The resulting curves from analysis and simulation are compared in Figure 9 for the buffer occupancy and in Figure 10 for the number of active sources, respectively. The mean buffer occupancy is much lower than in all other scenarios. The number of active sources is lower than in the other differentiated scenarios and only slightly higher than in the basic scenario. This is due to the fact that less packets have to undergo contention and thus less collisions occur with leads to a higher effective capacity.

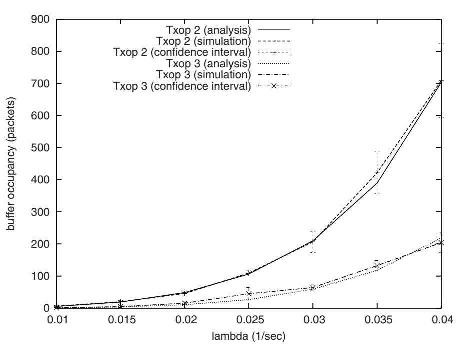

Fig. 9. Mean buffer occupancy for different TXOP_b and $\text{TXOP}_s = 1$

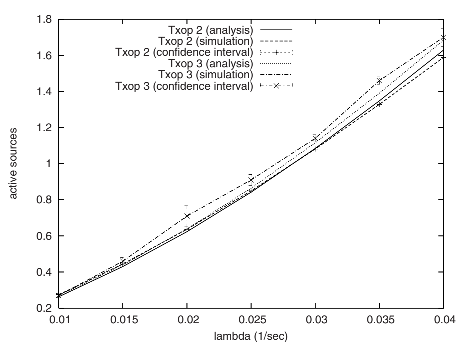

Fig. 10. Mean number of active sources for different TXOP_b and $\text{TXOP}_s = 1$

6.5 Throughput

To further analyze the impact of the differentiation parameters, we compute the throughput of the bottleneck, as a function of the parameter λ. The throughput for the basic IEEE 802.11 is compared with the throughput that is achieved in three differentiated settings: when $\mathrm{TXOP}_{\mathrm{limit}}$ of the bottleneck is set to 3, when $\mathrm{CW}_{\min,s}$ is set to 128 and when AIFS_s is set to 12. In Figure 11 we show our highly accurate analytical results for the throughput in these different settings together with the confidence intervals of the corresponding simulation runs.

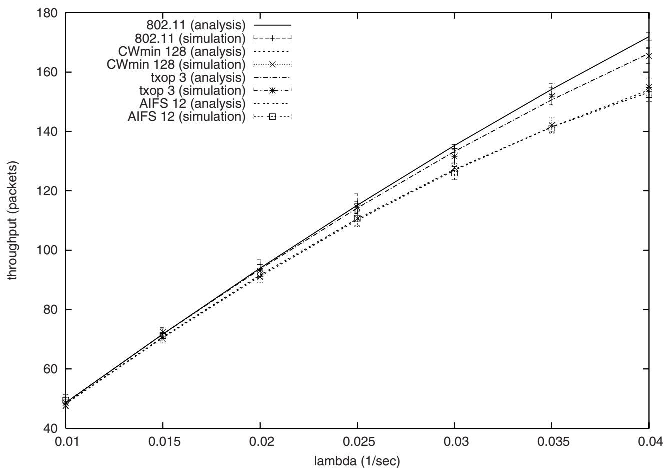

Fig. 11. Throughput of the bottleneck

For all $\lambda \in \{0.01, \ldots 0.04\}$, the largest throughput is achieved in basic IEEE 802.11. The throughput for $\mathrm{txop}_b = 3$ is just a little lower for increasing values of λ. The throughput for differentiated $\mathrm{CW}_{\min,s}$ and AIFS_s fall together and this throughput is considerably lower than in the two other cases.

This non-intuitive result is due to the fact, for given λ, the offered load depends on the average number of active sources. Recall, that the number of active sources is higher in the differentiated settings than in basic IEEE 802.11, as part of the capacity has been moved from the sources to the bottleneck. When the sources remain active for a longer time, they become active less frequently, so in total fewer packets are introduced to the bottleneck. Clearly, the bottleneck can only forward packets that have been sent to him from the sources. Since we assume infinite queue lenght, the throughput equals the offered load as long as the bottleneck queue is stable. Note that with differentiation the queue remains stable for much higher λ, compared to the standard 802.11. As a result, the *maximum* throughput using differentiation is expected to be higher than for the standard 802.11.

Finally, when CW_{min} and AIFS are increased to differentiate, this comes at the cost of a decreased effective capacity, as more time slots will pass unused. In contrast, increasing $TXOP_{limit}$ effectively increases capacity, as multiple packets are sent within $TXOP_{limit}$ after just one contention period. However, as can be seen in Figure 11 the increase in effective capacity is not enough to compensate for the slightly higher mean number of active sources for different $TXOP_b$, as shown in Figure 10.

7 Conclusions

In this paper we have presented a new model for analyzing the recently standardized quality-of-service enhancements of the IEEE 802.11e access mechanism in a two-hop ad hoc network. Our high-level model is flow-based, and uses results from packet-based models (such as those proposed by Bianchi and Engelstad [1,3]), and allows for the numerical evaluation of the buffer occupancy at the bottleneck node, the system throughput, as well as provides information on the mean number of active sources. The latter is important, as our model allows for a time-varying number of active sources, as opposed to earlier models that only allow for a fixed number of sources. The model is very easy to use (and extensible) as the basic model structure remains the same for all enhancements; just allocation functions (denotes as $S_b()$ and $S_s()$ in the paper) need to be defined. An efficient numerical solution based on the underlying quasi-birth-death structure of the model is automatically provided (using previously defined algorithms, cf. [12]).

We compare our results with extensive simulations (built using Opnet) and show that our models provide very accurate results at almost negligible cost in comparison to the simulations. No other analytical models that allow for similar evaluations have been proposed so far.

The results show that all differentiation parameters are able to allocate capacity in a more balanced way to the bottleneck and the sources, resulting in a much smaller buffer occupancy. However, the throughput of the bottleneck differs, depending on the differentiation used. In this paper, we only analyzed the throughput of the bottleneck as a function of the source arrival rate λ.

Further work is needed to compare the maximum throughput that can be achieved per differentiation parameter, for arbitrary values of λ.

Acknowledgments

We thank Patrick Goering for his help with the Opnet simulations. Anne Remke is supported through the MC=MC project (612.000.311) and Lucia Cloth is supported through the MOCS project (642.000.505), both financed by the Netherlands Organization for Scientific Research (NWO). We thank the reviewers for the detailed comments on this paper.

References

1. Bianchi, G.: Performance analysis of the IEEE 802.11 Distributed Coordination Function. IEEE Journal on Selected Areas in Communications 18, 535–547 (2000)
2. Cohen, J.W.: The multiple phase service network with Generalized Processor Sharing. Acta informatica 12, 245–284 (1979)
3. Engelstad, P.E., Osterbo, N.: Non-saturation and saturation analysis of IEEE 802.11e EDCA with starvation prediction. In: MSWiM 2005: Proceedings of the 8th ACM International Symposium on Modeling, Analysis and Aimulation of Wireless and Mobile Systems, pp. 224–233. ACM Press, New York (2005)
4. Groenendijk, W.P.: Conservation Laws in Polling Systems. PhD thesis, University of Utrecht (1990)
5. Ieee std 802.11e - 2005, 'part 11: 'Wireless LAN Medium Access Control (MAC) and Physical layer (PHY) specifications. amendment 8: Medium Access Control (MAC) Quality of Service Enhancements. amendment to IEEE 802.11 std. (2005)
6. Kurose, J.F., Ross, K.W.: Computer Networking. Addison-Wesley, Reading (2005)
7. Latouche, G., Ramaswami, V.: Introduction to Matrix Analytic Methods in Stochastic Modeling, ASA-SIAM, Philadelphia (1999)
8. Litjens, R., Roijers, R., van den Berg, J.L., Boucherie, R.J., Fleuren, M.J.: Analysis of flow transfer times in IEEE 802.11 wireless lans. Annals of Telecommunications 59, 1407–1432 (2004)
9. Neuts, M.F.: Matrix Geometric Solutions in Stochastic Models: An Algorithmic Approach. Johns Hopkins University Press (1981)
10. Opnet modeler software, `http://www.opnet.com/products/modeler`
11. Ost, A.: Performance of Communication Systems. Springer, Heidelberg (2001)
12. Remke, A., Haverkort, B.R., Cloth, L.: A versatile infinite-state Markov reward model to study bottlenecks in 2-hop ad hoc networks. In: 3rd International Conference on the Quantitative Evaluation of SysTems (QEST), pp. 63–72 (2006)
13. Roijers, F., van den Berg, J.L., Fan, X., Fleuren, M.: A performance study on service integration in IEEE 802.11E wireless LANs. Computer Communications 29, 2621–2633 (2006)
14. Schiller, J.: Mobile Communications. Addison-Wesley, Reading (2003)
15. van den Berg, J.L., Mandjes, M., Roijers, F.: Performance Modeling of a Bottleneck Node in an IEEE 802.11 Ad-hoc Network. In: Proceedings 5th International Conference on AD-HOC Networks and Wireless, pp. 321–336 (2006)