

# Dimensioning the Contention Channel of DOCSIS Cable Modem Networks

J. Lambert, B. Van Houdt, and C. Blondia

University of Antwerp, Department of Mathematics and Computer Science,  
Performance Analysis of Telecommunication Systems Research Group,  
Middelheimlaan, 1, B-2020 Antwerp - Belgium  
{joke.lambert,benny.vanhoudt,chris.blondia}@ua.ac.be

**Abstract.** This paper presents an open queueing model with blocking that enables us to determine the optimal fraction  $c^*$  of the uplink channel capacity that should be dedicated to the contention channel in order to minimize the mean response time in a DOCSIS cable network. To assess the performance of this queueing network we make use of a decomposition technique. One of the key components of the model exists in capturing the behavior of the conflict resolution algorithm by means of a single processor sharing queue. The model is validated in three successive steps by means of several simulation programs. We also explore the impact of a variety of systems parameters, e.g., the number of cable modems, the initial backoff window size, etc. , on the optimum  $c^*$ .

**Keywords:** DOCSIS Cable Networks, contention channel, performance evaluation, queueing networks, decomposition algorithm.

## 1 Introduction

Recently, the rapid growth of the number of residential Internet users and the increased bandwidth requirements of multimedia applications have necessitated the introduction of an access network that can support the demand of such services. The Data Over Cable Service Interface Specifications (DOCSIS) [3] are the dominant specifications for carrying data over Cable TV Distribution (CATV) networks and have been developed by CableLabs and MCNS (Multimedia Cable Networks Systems), which is a group of major cable companies, to support IP flows over Hybrid Fiber Coaxial (HFC) networks. DOCSIS defines modulation and protocols for high speed bi-directional data transmissions over cable systems. It has been accepted by most major vendors and is now a widely used specification to provide high-speed residential access. DOCSIS specifies a set of interface protocols between the cable modem customer side and the termination network side.

The Media Access Control (MAC) protocol defined in the DOCSIS RFIv1.1 is based on Time Division Multiple Access (TDMA). It uses MAC management messages, referred to as MAP messages, to describe the usage of the uplink channel. A given MAP message indicates the upstream bandwidth allocation over the next MAP time, termed the *MAP length*. The MAP assigns some uplink minislots to particular cable modems (CMs) to transmit data, other slots are available as contention slots to request bandwidth. This is one of the critical components of the DOCSIS MAC layer and the DOCSIS specification purposely does not specify these bandwidth allocation algorithms so that vendors are able to develop their own solutions. In this paper, we develop an open queueing network with blocking, whose performance determines the optimal ratio between contention slots and reservation slots in a single MAP length.

The paper is structured as follows. Section 2 presents a general description of the cable network considered. The contention resolution algorithm specified by the DOCSIS standard is discussed in Section 3. The queueing network model together with the decomposition techniques used to assess its performance is introduced in Section 4, whereas Section 5 validates this model in three successive steps. Finally, we numerically explore the influence of several system parameters in Section 6.

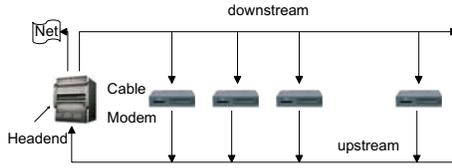


Fig. 1. Logical topology of a cable modem network

## 2 The DOCSIS Cable Network

The DOCSIS cable modem network considered in this paper is shown in Figure 1 and consists of a single Cable Modem Termination System (CMTS), located in the head-end of a cable operator or service provider and a number of Cable Modems (CMs) that are installed at the end-users. At initialization time each CM registers itself with the CMTS and at least 2 service flows are created for each CM: one in the downstream and one in the upstream direction. Since the head-end is the only transmitter in the downstream channel, no downstream media access control (MAC) mechanism is needed. The upstream channel, on the other hand, is shared by a number of CMs and transports signals from the CMs to the head-end. The available bandwidth is divided into fixed length allocation units, called minislots, and the Data Over Cable Service Interface Specifications (DOCSIS) specify a reservation-based, centralized approach for distributing these minislots among the CMs.

Periodically, the CMTS sends a bandwidth allocation map (MAP) message over the downstream channel. A MAP message contains a number of data grants, such a grant indicates when a particular CM is allowed to transmit data on the uplink channel. Moreover, the MAP also identifies the minislots part of the contention channel. The total time scheduled in a single MAP message is referred to as the MAP length and the time interval assigned to the contention channel is called the contention region, whereas the remaining part of the MAP length is termed the reservation region. Figure 2 illustrates the upstream mapping.

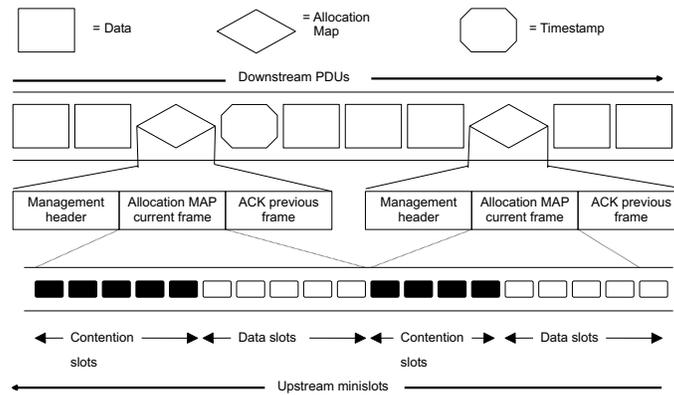


Fig. 2. Upstream bandwidth allocation

Any CM that wants to transmit data must request bandwidth to the CMTS by means of a request message that contains a count of the number of minislots needed. The request can be transmitted in 2 ways: (1) On the one hand the CM can send the request in the contention region. In this interval collisions may occur, meaning the request might get lost. The contention resolution algorithm (CRA) specified by the DOCSIS standard and operating on the contention channel, will be discussed in Section 3. (2) On the other hand, once a CM has received a grant, it has the opportunity to piggyback the new request in its reserved upstream allocation window, avoiding

any collisions. Thus, piggybacking allows stations to request new bandwidth when transmitting data packets, without reentering the contention based request process. After processing a request, the CMTS generates a data grant and transmits it in the next MAP if capacity is available.

It is the objective of this study to determine how to distribute the minislots part of the MAP length between the contention and the reservation region in an optimal way, and to examine the influence of various system parameters, e.g., the number of CMs, on the location of this optimum. The obtained results apply to any DOCSIS cable network whose bandwidth allocation algorithm fairly distributes the upstream bandwidth between the requesting CMs, and given the mean packet length, are also independent of the data packet length distribution at hand.

### 3 The Contention Resolution Algorithm

This section discusses the collision resolution algorithm, being a truncated binary exponential backoff algorithm (BEB), specified by the DOCSIS standard. The aim of the BEB is to minimize the collision probability between the request packets transmitted in the contention region. Therefore, each CM has to postpone every transmission attempt by a random time interval. The length of this time interval, called the backoff interval, indicates the number of contention slots<sup>1</sup> that the CM must let pass before it transmits its own request and grows as the number of consecutive unsuccessful transmissions increases. That is, at each attempt to transmit a request packet, the length of the backoff interval is uniformly chosen in the range  $[0, w - 1]$ , where  $w$  is called the current contention window (CW). The value of the parameter  $w$  depends on the number of transmission failures that occurred so far for the particular request packet. At the first transmission attempt, the CM initializes its backoff counter to 0 and  $w$  is set equal to  $CW_{\min}$ , called the minimal CW. After transmitting the request packet with a backoff counter equal to  $i - 1$ , for  $i > 0$ , the CM waits for an ACK. The contention resolution phase ends if the request is successfully received by the CMTS, otherwise the CM increases his backoff counter (to  $i$ ) and the contention window is set equal to  $W_i = 2^{\min(i, m')} CW_{\min}$ , where  $CW_{\max} = 2^{m'} CW_{\min}$  is the maximum length of the CW. Thus, as soon as the maximum value  $CW_{\max}$  is attained, the CW remains equal to its maximum value until there is either a successful transmission or until the maximum backoff counter value  $m + 1$  is reached. When the backoff counter is increased to  $m + 1$ , the CM drops the request packet. The DOCSIS standard specifies that the maximum number of retries  $m$  should be set to 15. The values of  $CW_{\min}$  and  $CW_{\max}$  are not specified by the standard and are controlled by the CMTS.

In order to construct a queueing model to dimension the contention channel in a DOCSIS cable network, it is vital to capture the behavior of the BEB algorithm accurately. Although the BEB is considered as hard to model mathematically, Bianchi [2] managed to develop a fairly simple mathematical framework that provides accurate results for the BEB in an 802.11 setting. This model was further extended by a number of authors, including [8] and [6]. Although we are not considering the BEB in a 802.11 setting, we will demonstrate that nearly the same approach can be taken to accurately model the BEB in a DOCSIS world. A fundamental role in developing a model for the BEB is played by the saturation throughput  $S(n)$ .  $S(n)$  is defined as the limit reached by the throughput of a system with  $n$  stations as the offered load increases and represents the maximum load that a system with  $n$  stations can carry under stable conditions. The saturation throughput values  $S(n)$  are used in Section 4.1 to develop a processor sharing (PS) type of queue to model the contention channel. The computation of  $S(n)$  is analogue to the approach taken by Bianchi [2] and Wu [8] and is therefore presented in Appendix A.

### 4 A Queueing Model for a DOCSIS Cable Network

To enable a mathematical analysis of the performance of DOCSIS cable networks, we will develop an open queueing network with blocking composed of  $N + 2$  queues, where  $N$  represents the number

---

<sup>1</sup> The size of a contention slot, expressed in minislots, depends on the modulation scheme and equals the time needed to transmit a single request packet.

of CMs connected to the access network. The transmission buffer of each CM is represented by a single queue, whereas the remaining two queues represent the contention and the reservation channel, respectively. Details on the service disciplines of the contention and reservation queues are given after the general model description.

A data packet that needs to be transmitted from CM  $i$  toward the network is placed in the transmission buffer of CM  $i$ . If the packet finds the buffer empty upon arrival, CM  $i$  needs to generate a request for this data packet and has to send it to the CMTS via the contention channel, before the actual data transmission can take place. On the other hand, if some of the earlier data is still stored within the transmission buffer, we can use a piggybacking strategy to transmit the request collision free. In our queueing model, we represent each data packet by a customer. Depending on the progress of the packet transmission, its corresponding customer  $C$  is in one of the following 3 queues:

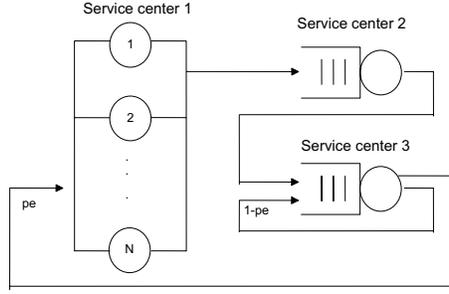
1. Customer  $C$  is waiting in queue CM  $i$ , whenever there are still other (older) packets that need to be (partially) transmitted first by CM  $i$ .
2. Customer  $C$  is part of the contention queue, if the data packet found the transmission buffer empty upon arrival and the CM is trying to transmit a request via the contention channel.
3. Customer  $C$  is part of the reservation queue, if all other (older) packets have been transmitted and a request for this packet was either piggybacked or successfully transmitted using the contention channel.

A new packet arrival at CM  $i$  corresponds to a new customer arrival in queue CM  $i$ , whereas customers who complete service in the reservation queue will leave the queueing network. Finally, a customer leaving the contention queue, will enter the reservation queue. Notice, the contention and reservation queues hold at most one customer per CM at a time. Thus, a customer at the head of line in a CM queue is blocked until the previous customer generated by the same CM has left the queueing system.

The reservation channel is modeled as a processor sharing (PS) queue, reflecting the DOCSIS MAC design principle of distributing the transmission capacity fairly among the active stations. The rate  $\nu$  of the server is determined by (i) the rate of the uplink channel  $r$  (expressed in bits/msec), (ii) the mean length of a data packet  $L$  (including overhead) and (iii) the fraction  $1 - c$  of the MAP length dedicated to the reservation region. That is,  $\nu = (1 - c)r/L$ . The contention channel is also modeled as a PS queue, but the service rate  $\mu(n)$  depends upon the number of customers  $n$  present in the contention queue. The rate  $\mu(n)$  is chosen as  $cS(n)r/Q$ , where  $c$  denotes the fraction of the MAP length associated with the contention region,  $Q$  the request size expressed in bits and  $S(n)$  the saturation throughput. This is a fairly logical choice as  $1/S(n)$  is also the mean time between two successful transmissions in a saturated system with  $n$  stations. The accuracy of this model for the contention channel is validated in Section 5.2. Using the saturation throughput as the service rate of a PS queue when modeling a contention channel is a proven technique (within the 802.11 setting), e.g., [4, 6]. Finally, we assume that the packets arrive at the CMs according to a Poisson process with rate  $\lambda$  (expressed in packets/msec). In practice, the traffic at the CMs is likely to be more bursty and correlated in comparison with the Poisson arrivals, however, it should be noted that when dimensioning the contention channel, assuming Poisson arrivals will result in a pessimistic prediction for the amount of contention channel needed, because the piggybacking mechanism is more effective in dealing with bursty arrivals. Hence, we can use the optimal fraction  $c^*$  obtained for the Poisson arrivals as an upperbound for real world traffic scenarios. The current model can, without too much effort, also be extended to more bursty arrival processes like the batch Poisson process or a Markovian arrival process (MAP).

In general, queueing networks with blocking typically do not have a closed-form solution and therefore it is very difficult to analyze the system as a whole. In order to solve this queueing model, we make use of a decomposition scheme similar to [7]. The general approach in decomposition is to decompose the entire system into smaller subsystems, analyze these subsystems individually and then take into account the interactions between the various subsystems in putting them back together. We will make use of two subsystems, described in the next subsections, where the analysis of each subsystem requires information that can be obtained by solving the other subsystem.

## 4.1 The First Subsystem: A Closed Queueing Network



**Fig. 3.** The first subsystem

This subsystem is obtained by removing the waiting rooms in front of each of the  $N$  CM queues and by closing the network (see Figure 3). The resulting network is of the BCMP type (see [1]) with 3 service centers and  $N$  identical customers. Service center 2 (top right) represents the contention channel, whereas service center 3 (bottom right) represents the reservation channel. Service center 1 (left) has an infinite server (IS) queueing discipline, i.e., the number of servers in the service center is greater than or equal to the number of customers  $N$  in the network. A customer arriving at service center 1 always finds a free server and never queues for service. The service rate of service center 1 is equal to the arrival rate  $\lambda$  of packets at each CM, meaning a customer will on average stay within service center 1 for  $1/\lambda$  msec (this corresponds to the mean time needed for a new packet to enter an empty transmission buffer). Service centers 2 and 3 are identical to the Processor Sharing (PS) disciplines of the original queueing network. Notice, the packet length distribution is irrelevant (apart from its mean) as the performance measures of such a BCMP network are insensitive to the service time distribution of its PS service centers.

The routing matrix of this closed queueing network is equal to

$$\begin{bmatrix} 0 & 1 & 0 \\ 0 & 0 & 1 \\ p_e & 0 & 1 - p_e \end{bmatrix}, \quad (1)$$

where  $p_e$  represents the probability that a packet finds the transmission buffer empty upon arrival. The value of  $p_e$  is determined by solving the second subsystem. Notice,  $p_e$  can be regarded as the probability that an arbitrary data packet leaves an empty transmission buffer behind, therefore, the CM buffer becomes empty for some time (with mean  $1/\lambda$ ). To calculate the mean response times  $E[R_{\text{cont}}]$  and  $E[R_{\text{res}}]$  of service center 2 and 3 of this closed queueing network, one may use either one of the following algorithms [5]: the Convolution, the Mean Value Analysis (MVA) or the Local Balance algorithm. These mean response times are used as input values in the second subsystem.

## 4.2 The second subsystem

The second subsystem consists of  $N$  independent finite single server queues, one for each CM. The packets arrive at server  $i$  according to a Poisson process with rate  $\lambda$ , expressed in packets/msec, and the service discipline is FCFS. The service time at each server accounts for the time spent on the reservation channel (to transmit the data packet) and the possible time needed to visit the contention queue (to send the request), therefore its rate  $\mu_{CM}$  is chosen as  $\mu_{CM} = \frac{1}{E[R_{\text{cont}}]p_e + E[R_{\text{res}}]}$ , where the mean response time of the contention channel, respectively the reservation channel, was denoted by  $E[R_{\text{cont}}]$ , respectively  $E[R_{\text{res}}]$ , and these values are obtained by solving the first subsystem.

The queue is finite and has a capacity of  $B$  customers, i.e.,  $B - 1$  in the waiting room and one in the server. New arrivals are lost whenever the queue contains  $B$  customers. Hence, queue  $i$  can be solved as an  $M/M/1/B$  queue. For this subsystem we can easily determine the steady state probabilities, from which we compute the probability  $p_e$  that the server is found empty by a new arrival. This probability is used as the new input value of  $p_e$  in the first subsystem.

The decomposition algorithm executes successive iterations during which both subsystems are solved one after the other, until the probability  $p_e$  has converged. That is, as soon as the absolute difference of  $p_e$  between two successive iterations is less than  $\epsilon = 10^{-14}$ . After the algorithm has converged, the probability  $p_e$  is used to calculate the total mean response time, which equals  $E[R_{\text{cont}}]p_e + E[R_{\text{res}}]$ , and the loss probability  $p_{\text{loss}}$ .

## 5 Model Validation

Our model is validated in three consecutive steps. First, we investigate the accuracy of the saturation throughputs  $S(n)$  in a DOCSIS setting. This is done by comparing the analytical results obtained in Appendix A with a simulation of the BEB under saturated conditions. Second, we compare the performance of the contention queue with the performance of the BEB under Poisson arrivals and a finite population of size  $N$ . Finally, we simulate the original open queueing network with blocking to demonstrate the precision of the decomposition method.

### 5.1 The Saturation Throughputs $S(n)$

In Appendix A we presented an analytic evaluation of the saturation throughput. Although, [2] and [8] have validated the analytical saturation throughputs for the basic access and RTS/CTS access mechanisms in an 802.11 environment, we validate this model for a DOCSIS setting by comparing its results with those obtained by a simulator. We feel that this is necessary as the packet lengths, empty slot size and the collision lengths differ significantly in both network types. We let  $n$ , the number of CMs, vary between 10 and 150. The size of the contention window depends on the number of transmission failures  $i$  for the packet and was equal to:

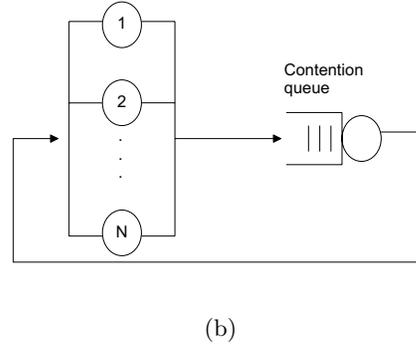
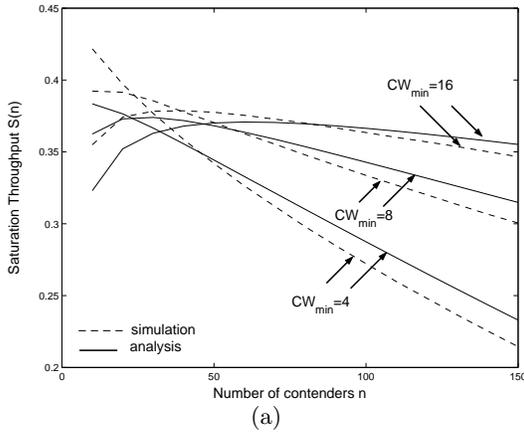
$$\begin{aligned} W_i &= 2^i CW_{\min} & i \leq m', \\ W_i &= 2^{m'} CW_{\min} & m' \leq i \leq m. \end{aligned} \quad (2)$$

The minimum contention window  $CW_{\min}$  is increased by a factor of 2, starting from 4 to 16, the maximum number of retransmissions  $m$  equals 15 and the maximum value of the contention window  $CW_{\max}$  varies between 128 and 512, i.e.,  $m' = 5$ . The simulation is written in the C++ programming language and the duration of the simulation is  $10^7$  contention slots. Figure 4(a) shows that the analytical model is fairly accurate and both the analytical model and the simulation show the same tendencies. The largest deviations are observed for very small or for large populations. It can be noted that the analytical model performs better as the minimum contention window  $CW_{\min}$  grows. Figure 4(a) also depicts that the saturation throughput strongly depends on the number of stations in the network. In particular, the throughput degrades, in most cases, when the network size increases; caused by an increased collision probability. As expected, this decreasing tendency is more pronounced for small values of the minimum contention window  $CW_{\min}$ . It can also be observed that the analytical saturation throughputs  $S(n)$  are optimistic when  $n$  is large (say,  $n \geq 50$ ), especially when  $CW_{\min}$  is small.

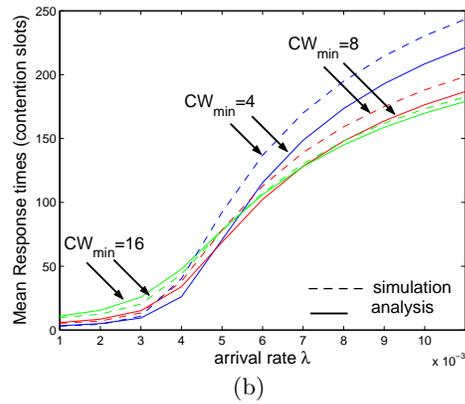
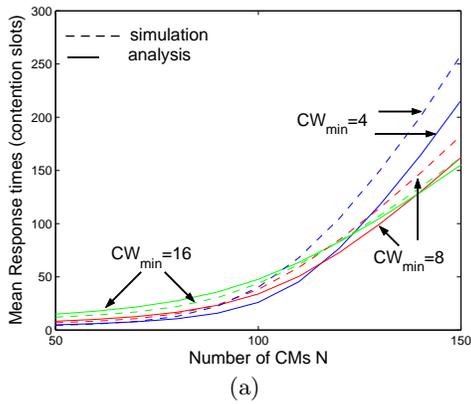
### 5.2 The contention channel

In this section we compare the performance of the contention queue with the performance results obtained by simulating the BEB under Poisson arrivals and a finite population of size  $N$  (see Figure 4(b)). The simulation results were gathered after  $10^7$  contention slots.

Figure 4(b) shows the analytical model containing the contention queue. It is a closed system consisting of 2 service centers and  $N$  identical customers. Service center 1 has an infinite server (IS)



**Fig. 4.** (a) Saturation Throughput: analysis versus simulation, (b) Model used to validate the contention queue



**Fig. 5.** Validating the contention queue by comparing the mean response time for various parameter settings

queueing discipline, while service center 2 is our contention queue. This queueing network is also of the BCMP type and hence, we can rely on the same algorithms as mentioned in subsection 4.1, to compute the mean response time of the contention queue and compare it with the simulation results. We let  $N$ , the number of CMs, vary between 50 and 150 and the arrival rate  $\lambda$  is chosen small (between 0.001 and 0.011). The choice of  $\lambda$  was determined by studying the average number of simultaneously contending CMs for different values of  $N$  and  $CW_{\min}$ . According to the Cisco document “Cisco - Understanding Data Throughput in a DOCSIS World”, a realistic value for this average number is 10. Finally, we set  $m$  and  $m'$  equal to 15 and 6, respectively.

Figure 5(a) shows the mean response times as a function of  $N$  when  $\lambda$  equals 0.004, whereas in Figure 5(b)  $\lambda$  varies and  $N$  is fixed at 100. Both figures indicate that there is a good agreement between simulation results and the analytical model. Similar to what we observed in the previous section, the deviation is somewhat larger for lower values of the minimum contention window  $CW_{\min}$ . This is understandable as the saturation throughputs  $S(n)$  are used as the service rates of the contention queue. The differences between the analytical and simulation results are mainly caused by the inaccuracy of the (optimistic) saturation throughput values  $S(n)$ .

Parameters	Symbol	Value
Upstream channel capacity	$r$	$10^4$ bits/msec
request	$Q$	16 bytes
data packet	$L$	438 bytes
arrival rate	$\lambda$	0.01 p/msec
number of stations	$N$	100
size of queue	$B$	10
min. contention window	$CW_{\min}$	4
max. value	$m'$	6
max. backoff stage	$m$	15

**Table 1.** Default System Parameters

### 5.3 The Decomposition Scheme

This subsection compares the analytical results obtained by means of the decomposition scheme with a detailed event-driven simulation of the original open queueing network with blocking. The parameter values used in this Section are presented in Table 1. Most of these parameters are chosen as the default values proposed in the Cisco specification “Cisco - Understanding Data Throughput in a DOCSIS World”. We assume an average packet length of 438 bytes, this includes an 8% forward error correction (FEC) overhead, the 6 byte overhead of the DOCSIS header, etc. The 10 Mbit channel supports both the contention channel and the reservation channel. The fraction of contention channel, resp. reservation channel, is represented by  $c$ , resp.  $1 - c$ , thus the mean service rate of the reservation channel,  $\nu$ , and the mean service rate of the contention channel,  $\mu(n)$ , equal:

$$\nu = (1 - c) \frac{10^4}{438 * 8} \quad (3)$$

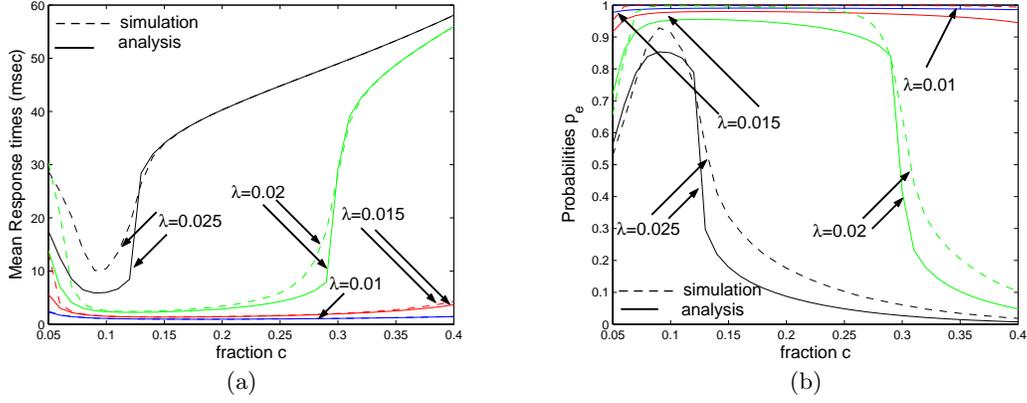
$$\mu(n) = cS(n) \frac{10^4}{16 * 8}. \quad (4)$$

Remark that the unit of these rates is  $1/msec$  (this is in correspondence with the fact that the arrival rate  $\lambda$  is expressed in packets/msec). The data load of the system can be computed as  $\rho_{\text{data}} = \lambda N \frac{438 * 8}{10^4}$ , while the effective load on the reservation channel is obviously higher and equals  $\rho_{\text{res}} = \frac{\rho_{\text{data}}}{1 - c}$ . A comparison of the mean response time and the probability that a new packet finds the transmission buffer empty upon arrival  $p_e$ , for different values of the arrival rate  $\lambda$ , is made in Figure 6(a) and Figure 6(b), respectively. Although there is no perfect agreement between the analytical and simulation results, especially for larger arrival rates  $\lambda$ . We observe that the location of the optimal  $c$  value, that is, the optimal fraction dedicated to the contention channel, does agree to a high level of accuracy. The sudden increase in the response times are caused by the fact that the effective load on the reservation channel  $\rho_{\text{res}}$  becomes larger than 1. Similar results were obtained for different  $N$  values (e.g.,  $N = 50, 200$ ).

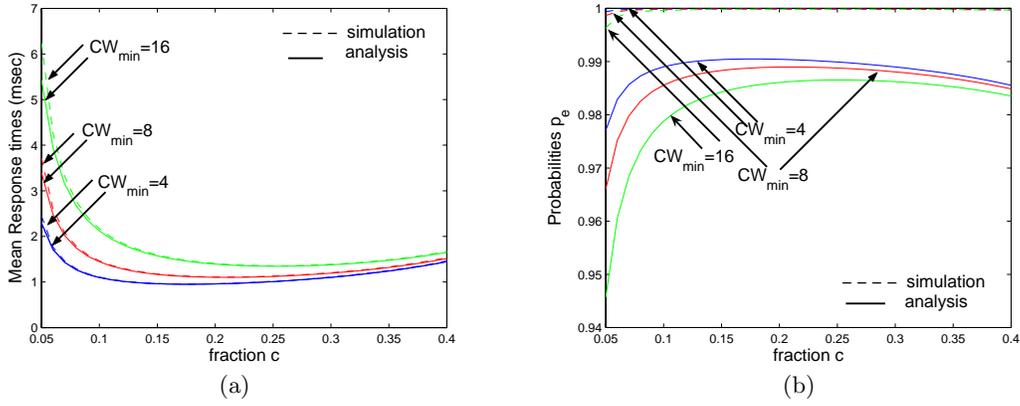
The influence of the minimum contention window  $CW_{\min}$  is depicted in Figures 7(a) and 7(b). We see that the analytical model is quite accurate: the analytical results for the mean response time nearly coincide with the simulation results. The analytical results for the probability  $p_e$  do deviate a little from the simulation results.

## 6 Numerical Results

In this section we study the optimal fraction  $c^*$  of the MAP length that should be dedicated to the contention channel such that the mean response time is minimal. Besides the mean response time, we also evaluate the loss probability  $p_{\text{loss}}$  and the probability that a new packet finds the transmission buffer empty upon arrival. Unless otherwise stated the protocol parameters are set as indicated in Table 1.



**Fig. 6.** (a) Validating the decomposition technique: Mean response times, (b) Validating the decomposition technique: The probability of arriving in an empty transmission buffer  $p_e$

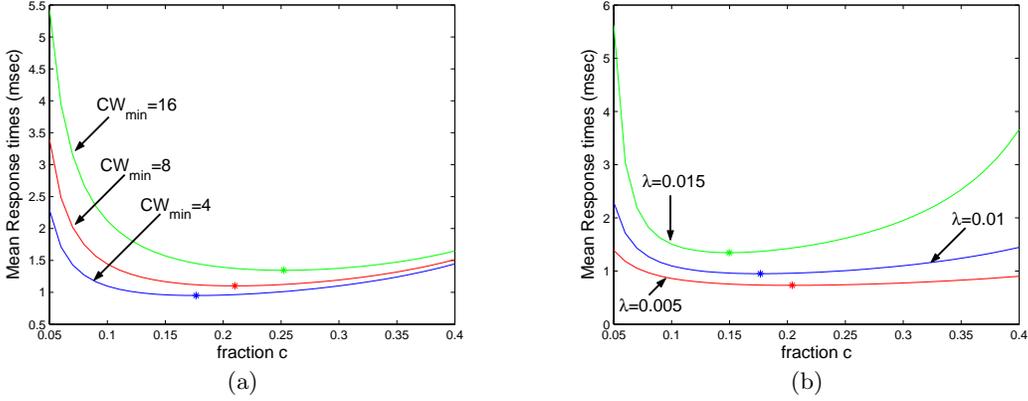


**Fig. 7.** (a) Validating the decomposition technique: the influence of the minimum contention window  $CW_{\min}$  on the mean response time, (b) Validating the decomposition technique: the influence of the minimum contention window  $CW_{\min}$  on  $p_e$

We start by investigating the impact of the minimum contention window  $CW_{\min}$  on the optimum  $c^*$ . When plotting the influence of the contention fraction  $c$  we add a “\*” to identify the optimum  $c^*$ . Figure 8(a) shows that the optimum  $c^*$ , as well as the minimum mean response time, increases as a function of  $CW_{\min}$ . Such an increase reduces the collision probability, because, on average, a station defers its transmission by a greater number of contention slots. Thus, a CM needs less attempts to transmit a request, but the time between two attempts is substantially larger, causing higher delays on the contention channel and augmenting the need for more contention slots.

Experiments not included here, have shown that the transmission buffer size  $B$  has no significant influence on the mean response time or on the probability  $p_e$  that a packet finds the transmission queue empty (unless it is chosen extremely small), the loss probability obviously does increase for smaller buffers. We have also altered the maximum contention window  $CW_{\max}$  from 128 to 512, meaning that  $m'$  varies between 5 and 7, but this had no significant impact on the performance of the system either.

Let us now focus on the arrival rate  $\lambda$ . Increasing  $\lambda$  corresponds to a higher data load  $\rho_{\text{data}}$ . Consequently, the probability that a new packet finds an empty transmission buffer diminishes. Thus, more request packets can make use of the piggybacking scheme and the contention channel



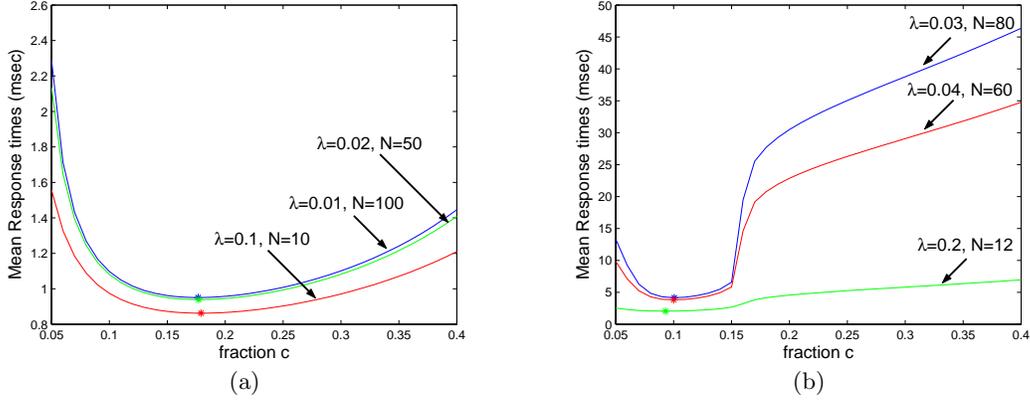
**Fig. 8.** (a) Mean response time when varying  $CW_{\min}$ , (b) Mean response time when the arrival rate  $\lambda$  varies

becomes less needed. Figure 8(b) confirms that the optimal fraction  $c^*$  of the contention channel is lower. Clearly, the minimum mean response time and the loss probability increase as a function of the load  $\rho_{\text{data}}$ . It is worth noticing that the optimum is quite broad, meaning that even if the fraction  $c$  is not exactly equal to  $c^*$  we still have a nearly minimum mean response time. The optimum does become somewhat less broad for higher arrival rates  $\lambda$ , which is intuitively clear as the choice of  $c$  becomes more critical as more bandwidth is being consumed. As such we recommend to dimension the contention channel for high load traffic scenarios as this will also guarantee a close to optimal performance for the low load cases as well. Similar results are observed when the number of CMs  $N$  is varied and  $\lambda$  is held constant.

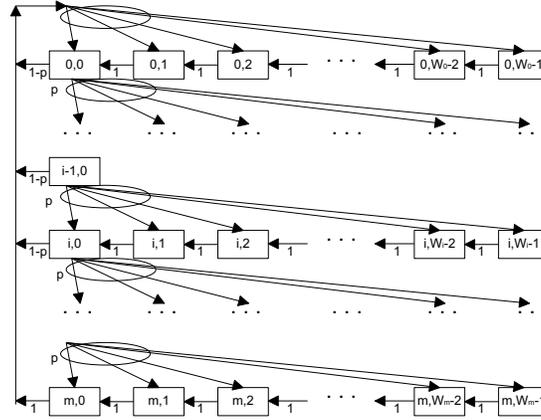
Finally, we examine the influence of having a fixed load  $\rho_{\text{data}}$ , but a varying number of CMs  $N$  (i.e.,  $\lambda N$  is kept constant). When the number of CMs  $N$  grows, which corresponds to a decrease in the arrival rate  $\lambda$ , the traffic on the data channel becomes less regular, creating a greater need for a larger reservation channel. On the other hand, the probability  $p_e$  that a new packet arrives in an empty transmission buffer is substantially higher when the number of CMs  $N$  increases (and  $\rho_{\text{data}}$  is fixed), leading to a greater demand for a larger contention channel as well. The tradeoff between these two opposing forces will decide whether the optimum  $c^*$  de- or increases with  $N$ . Figure 9(a) shows some combinations for  $N$  and  $\lambda$  that result in a data load of 0.35, whereas Figure 9(b) corresponds with a data load of 0.84. Numerical results not included here, have shown that the probability  $p_e$  increases much more severely as a function of  $N$  for the  $\rho_{\text{data}} = 0.84$  scenario. Therefore, we may conclude that for higher loads the need for more contention channel capacity, has a stronger impact, which results in a higher optimal fraction  $c^*$ , whereas for lower loads the burstiness of the traffic gets the upper hand, causing a reduction of the fraction  $c^*$  assigned to the contention channel.

## 7 Conclusion

Based on the results presented in section 6 we recommend to dedicate 10 to 15% of the minislots part of the MAP length to the contention channel as this will result in a near optimal mean response time for all arrival rates  $\lambda$ . Indeed, for high data loads the optimum is small and close to 10%, whereas for lower data loads the optimal fraction is larger, but broader as well, meaning that the a fraction between 10 to 15% still provides near optimal results. The exact optimal fraction  $c^*$  depends on several system parameters, e.g., the minimum contention window  $CW_{\min}$ , the number of CMs, etc. In view of the pessimistic nature of the Poisson arrival process when dimensioning the contention channel, assigning 10% of the available uplink channel to the contention channel should be sufficient for real world traffic scenarios.



**Fig. 9.** (a) Mean response time for a fixed load  $\rho_{\text{data}} = 0.35$ , (b) Mean response time for a fixed load  $\rho_{\text{data}} = 0.84$



**Fig. 10.** Markov chain model of new backoff window scheme

## Acknowledgment

B. Van Houdt is a post-doctoral Fellow of the FWO-Flanders.

## Appendix A: Modeling the Binary Exponential backoff algorithm

Assume a fixed number of contending stations  $n$ , each always having a packet available for transmission. In part one of the analysis the behaviour of a single station is modeled by a Markovian model. The objective of this part is to obtain the stationary probability  $\tau$  that a station transmits a packet in a randomly chosen slot. The second part expresses the throughput  $S(n)$  as a function of  $\tau$ .

*Part 1* The key approximation made by Bianchi [2] is that, at each transmission attempt and regardless of the number of retransmissions suffered, each packet collides with a constant and independent probability  $p$ , called the conditional collision probability. Once the independence is assumed, two stochastic variables can be used to model the protocol behavior (see in Figure 10). Let  $b(t)$  be the random variable representing the remaining backoff interval size for the given station at slot time  $t$  and let  $s(t)$  represent the backoff counter value ( $0, \dots, m$ ) of the tagged station at time  $t$ . The bi-dimensional process  $\{s(t), b(t)\}$  is then a discrete-time Markov Chain

(MC). The stationary distribution of this MC, denoted by  $b_{i,k}$ , for  $0 \leq i \leq m$  and  $0 \leq k \leq W_i - 1$ , is given by Eqn. (5), see [8]. The probability  $\tau$  depends on the conditional collision probability  $p$

$$\begin{aligned}
b_{0,0} &= \begin{cases} \frac{2(1-2p)(1-p)}{CW_{\min}(1-(2p)^{m+1})(1-p)+(1-2p)(1-p^{m+1})} & m \leq m' \\ \frac{2(1-2p)(1-p)}{CW_{\min}(1-(2p)^{m'+1})(1-p)+(1-2p)(1-p^{m-m'})+CW_{\min}2^{m'}p^{m'+1}(1-2p)(1-p^{m-m'})} & m > m' \end{cases} \\
b_{i,0} &= p^i b_{0,0} & \text{for } 0 \leq i \leq m \\
b_{i,k} &= \frac{W_i - k}{W_i} b_{i,0}, & \text{for } 0 \leq i \leq m
\end{aligned} \tag{5}$$

and can be expressed as

$$\begin{cases} \tau = \frac{1-p^{m+1}}{1-p} b_{0,0} \\ p = 1 - (1-\tau)^{n-1} \end{cases} \tag{6}$$

This nonlinear system of equations in the unknowns  $\tau$  en  $p$  can be solved easily by numerical techniques.

*Part 2* In the second part the events that can occur within a randomly chosen contention slot are studied in order to get a formula for the system throughput  $S(n)$ , defined as the probability that a successful transmission occurs in an arbitrary contention slot when  $n$  stations contend on the channel. Since  $n$  stations contend on the channel and each transmits with probability  $\tau$ , we can define the probability  $P_{tr}$  that there is at least one transmission in the considered slot time as  $P_{tr} = 1 - (1-\tau)^n$ , and the probability  $P_s$  that a transmission occurring on the channel is successful is given by  $P_s = \frac{n\tau(1-\tau)^{n-1}}{P_{tr}}$ . We can conclude that the saturation throughput  $S(n)$ , which equals the probability that a successful transmission occurs in an arbitrary slot, equals  $P_s P_{tr}$ , thus  $S(n)$  becomes  $S(n) = n\tau(1-\tau)^{n-1}$ . Notice, this formula is a simplified version of Bianchi's as the request packets, empty slots and collision periods all have a fixed length equal to one contention slot.

## References

1. F. Baskett, K. M. Chandy, R. R. Muntz, and F. G. Palacios. Open, closed, and mixed networks of queues with different classes of customers. *Journal of the Association for Computing Machinery*, 22(2):248–260, Apr 1975.
2. G. Bianchi. Performance analysis of the IEEE 802.11 distributed coordination function. *IEEE Journal on Selected Area in Comm.*, 18(3), Mar 2000.
3. CableLabs. *Data-Over-Cable Service Interface Specifications, Radio Frequency Interface Specification*. Available at <http://www.cablemodem.com/specifications>.
4. C. H. Foh and M. Zukerman. Performance analysis of the IEEE 802.11 MAC protocol. In *European Wireless 2002*, Florence, Feb 2002.
5. S.S Lavenberg. *Computer Performance Modeling Handbook*. Academic Press, 1983.
6. R. Litjens, F. Roijers, J.L. van den Berg, R.J. Boucherie, and M.J. Fleuren. Performance analysis of wireless LANs: an integrated packet/flow level approach. In *Proceedings of ITC-18*, Berlin, Sep 2003.
7. S. Ramesh and H. G. Perros. A multilayer client-server queueing network model with synchronous and asynchronous messages. *IEEE transactions on software engineering.*, 26(11), Nov 2000.
8. H. Wu, Y. Peng, K. Long, S. Cheng, and J. Ma. Performance of reliable transport protocol over IEEE 802.11 wireless LAN: Analysis and enhancement. In *IEEE Infocom'2002*, New York, Jun 2002.